



Applied Systems Engineering, Inc.

Technical Note #21

DNP Master Mode Requests

About Master Requests

In Master mode the GPT supports the following logical device requests. These requests were chosen to represent the type of requests found in many different RTUs. Requests are logical in the sense that each protocol component is free to map these requests into any of the operations supported by the protocol. The following table defines the logical device commands supported by the GPT master.

Attribute	Description
Command	Type of command: <ul style="list-style-type: none"> • Initialize. Used to initialize a remote device. • Counter Control. Counter freeze, reset, or freeze and reset. • Poll. Background poll or interrogation. Used to read analog, binary and counter input data from a remote device. • Exception Poll. Foreground poll. Used to read events from a remote device. • Clock Sync. Time synchronization. • Read. Read specified variables. • Write. Write specified variables. • Protocol-specific. Specific to protocol.

If we look at a specific protocol like DNP we can see the relationship between the logical device requests and DNP3 ASDUs. This mapping is provided in the following table.

GPT Logical Device Command	Equivalent IEC ASDU or data link request
Initialize	DNP Clear device restart.
Counter Control	DNP freeze, reset, and poll counters.
Poll	DNP class 0, 1, 2, and 3 data scan
Exception Poll	DNP class 1, 2, and 3 data scan.
Clock Sync	DNP write time.
Read	DNP read static objects.
Write	DNP write static objects.

Request processing proceeds as follows:

- The user application formats a device command. Any required arguments are provided in the request. The user assigns a unique ID or identification to the request.
- The protocol specific PT translates the logical request into one or more protocol specific operations on a device. For example a control request (logical Write command) may require that the protocol component build separate select and operate messages that are transmitted to the device.

Technical Note 21: DNP3 Master Mode Requests

- The protocol component traces the progress of a request. When a user request completes either successfully or with error, the PT writes the request back to user. The PT communicates to the user application the final result of the request using the unique identification that the user assigns to the request. Most requests complete synchronously. The user issues the request, the request is sent to the remote device, and the result of the operation is returned to the user application. Some protocols like IEC use activation. One a request is activated in the remote device it may run asynchronously with other commands that are being sent by the PT. For these protocols the PT hangs onto the request until the remote device terminates the activation. When the operation completes the final result is then returned to the user application.

The following table defines the fields of the DAC REQUESTINPUT structure

Function	Description
DAC_REQ_INITIALIZE	Initialize a remote station. The initialization is protocol specific.
DAC_REQ_POLL	Poll (Interrogate) a remote station's static data
DAC_REQ_POLL_EVENTS	Poll a remote station for static and event data
DAC_REQ_READ_VARIABLES	Read variables at a remote station and write their values into variables defined at the local station
DAC_REQ_WRITE_VARIABLES	Read variables at the local station and write their values into variables at a remote station.
DAC_REQ_CLOCK_SYNC	Synchronize the clock at a remote station
DAC_REQ_COUNTER_CONTROL	Freeze/Reset/Acquire counter variables at a remote station.
DAC_REQ_SPECIAL	Used to request a protocol specific feature. Special requests are document in the protocol's profile document.
DAC_REQ_RESET	Reset/restart remote device
DAC_REQ_CONNECT	Connect a device to a circuit.
DAC_REQ_DISCONNECT	Disconnect a device from a circuit.
DAC_REQ_FILE	Transfer a file between the local stations and a remote station.
DAC_REQ_TEST	Test the operation of the remote station. Only supported in protocols that provide a test service at the data link or application layer.

Function modifiers may be specific to a single function are supported for all functions. The following table shows the supported modifier codes:

Request Function	Modifier	Description
DAC_REQ_COUNTER_CONTROL	DAC_REQ_MODIFIER_FREEZE	Freeze Counters
	DAC_REQ_MODIFIER_POLL	Acquire Counters
	DAC_REQ_MODIFIER_RESET	Reset Counters
	DAC_REQ_MODIFIER_FROZEN	Acquire Frozen Counters
DAC_REQ_CLOCK_SYNC	DAC_REQ_MODIFIER_CHAN_DELAY	Compute channel delay
DAC_REQ_WRITE_VARIABLES	DAC_REQ_MODIFIER_SELECT	Select variable

Technical Note 21: DNP3 Master Mode Requests

	DAC_REQ_MODIFIER_EXECUTE	Execute control
	DAC_REQ_MODIFIER_SELECT_EXECUTE	Select and Execute
	DAC_REQ_MODIFIER_IMMEDIATE	Immediate control no select required
DAC_REQ_RESTART	DAC_REQ_MODIFIER_COLD_START	Cold start remote device
	DAC_REQ_MODIFIER_WARM_START	Warm start remote device
DAC_REQ_FILE	DAC_REQ_MODIFIER_FILE_READ	Read a file
	DAC_REQ_MODIFIER_FILE_WRITE	Write a file
	DAC_REQ_MODIFIER_FILE_OPEN_READ	Open a file for read
	DAC_REQ_MODIFIER_FILE_OPEN_WRITE	Open a file for writing
	DAC_REQ_MODIFIER_RECORD_READ	Read a record
	DAC_REQ_MODIFIER_RECORD_WRITE	Write a record
	DAC_REQ_MODIFIER_FILE_CLOSE	Close a file
	DAC_REQ_MODIFIER_DIRECTORY_READ	Read a directory file
	DAC_REQ_MODIFIER_DIRECTORY_WRITE	Write a directory file
Any Function	DAC_REQ_LINK_SERVICE	If protocol supports function at both data link and application layer use data link service
	DAC_REQ_APPLICATION_SERVICE	If protocol supports function at both the data link and application layer use data link service

DNP Requests

The DNPPT uses the above structure to map DAC requests to DNP 3.0 services. The following table shows the DNPPT maps request function codes to DNP requests.

Function	Description
DAC_REQ_INITIALIZE	Clear Restart bit
DAC_REQ_POLL	Class 0 Poll
DAC_REQ_POLL_EVENTS	Class 1/2/3/0 Poll
DAC_REQ_READ_VARIABLES	Read DNP 3.0 object
DAC_REQ_WRITE_VARIABLES	Write DNP 3.0 Object
DAC_REQ_CLOCK_SYNC	Write time
DAC_REQ_COUNTER_CONTROL	Freeze/Reset/Acquire counter variables at a remote station.
DAC_REQ_SPECIAL	Assign Class Enable/Disable umode
DAC_REQ_RESET	Warm/Cold start device

The following table shows how the request fields are used by the DNPPT.

Field	Description
-------	-------------

Technical Note 21: DNP3 Master Mode Requests

Global	If true, the request applies to all devices on the line.
Device	If the request is not a global request, this field specifies the address of the remote station to which the request is sent.
ObjId	Dnp 3.0 Object number
Group	DNP 3.0 variable class 0x00 – All Classes 0x01 – Static Class 0x02 – Class 1 0x04 – Class 2 0x08 – Class 3
Modifier	DAC request modifier.
Qualifier	The range of variables in the source object that the request operates on: --DAC_REQ_QUALIFIER_ALL. All variables in source object --DAC_REQ_QUALIFIER_ID_COUNT. The variables starting at a specified Id and count. --DAC_REQ_QUALIFIER_START_STOP. The variables starting at a specified start and stop ids
Variation	If 0 the remote device is instructed to use the default variation in building its response, otherwise the specific DNP variation is requested.
Start	Starting Id for the DAC_REQ_QUALIFIER_ID_COUNT
Count	Variable count for the DAC_REQ_QUALIFIER_ID_COUNT, or ending id for DAC_REQ_QUALIFIER_START_STOP
Args	Pointer to user supplied arguments used when writing variables in a remote station.
Retry	The number of times to retry a request that is not successfully sent to a remote station.
Protocol	For special requests the protocol field contains the a DNP function code.

The following table shows which fields described above are used for the various DNP requests. Note: the global, device, and modifier fields are used for all requests.

DAC Request	Request Fields	Notes
DAC_REQ_INITIALIZE		Modifier DAC_REQ_MODIFIER_APPLICATION_CONFIRM DAC_REQ_MODIFIER_LINK_CONFIRM
DAC_REQ_POLL	Group	Modifier DAC_REQ_MODIFIER_APPLICATION_CONFIRM DAC_REQ_MODIFIER_LINK_CONFIRM
DAC_REQ_POLL_EVENTS	Group	Modifier DAC_REQ_MODIFIER_APPLICATION_CONFIRM DAC_REQ_MODIFIER_LINK_CONFIRM
DAC_REQ_READ_VARIABLES	ObjId Start Count Qualifier Variation	Modifier DAC_REQ_MODIFIER_APPLICATION_CONFIRM DAC_REQ_MODIFIER_LINK_CONFIRM

Technical Note 21: DNP3 Master Mode Requests

DAC_REQ_WRITE_VARIABLES	ObjId Start Count Qualifier Args	Modifier DAC_REQ_MODIFIER_APPLICATION_CONFIRM DAC_REQ_MODIFIER_LINK_CONFIRM DAC_REQ_MODIFIER_SELECT DAC_REQ_MODIFIER_SELECT_EXECUTE DAC_REQ_MODIFIER_IMMEDIATE
DAC_REQ_SPECIAL	Start Count Group Qualifier	Modifier DAC_REQ_MODIFIER_APPLICATION_CONFIRM DAC_REQ_MODIFIER_LINK_CONFIRM Protocol Assign Class Disable/Enable Umode
DAC_REQ_RESET		Modifier DAC_REQ_MODIFIER_APPLICATION_CONFIRM DAC_REQ_MODIFIER_LINK_CONFIRM DAC_REQ_MODIFIER_COLD_START DAC_REQ_MODIFIER_WARM_START
DAC_REQ_COUNTER_CONTROL	ObjId Qualifier Variation Start Count	Modifier DAC_REQ_MODIFIER_APPLICATION_CONFIRM DAC_REQ_MODIFIER_LINK_CONFIRM DAC_REQ_MODIFIER_FREEZE DAC_REQ_MODIFIER_RESET DAC_REQ_MODIFIER_POLL
DAC_REQ_CLOCK_SYNC		Modifier DAC_REQ_MODIFIER_CHAN_DELAY

Writing request Data

After a request is transmitted to a remote station the remote station may respond with variable data. This data is written to the user application by the DNPPT using the DAC WriteEvents, Find, and Write APIs. If the data returned from the remote station is static data the DNPPT calls the DAC find service to locate the offset of the starting id returned by the remote. If the find call succeeds the data is written to the user application using the DAC Write API. For example:

DNPPT send a request to read 5 analog variables starting at id 1. The remote station responds with the 5 analog values. The DNPPT does the following:

- Calls the DAC Find API on the analog input object with an id of 1, the first id returned by the remote station. If the find succeeds the user application returns the offset in the static analog input array of this variable.
- Calls the DAC Write API using the starting offset returned by the find procedure and an ending offset of start + 4.

Determining the source of the data write

Data written to the user application can be the result of either a request created by the user application or an umode transmission from the remote device. To determine the reason for the write the user application can do the following:

Technical Note 21: DNP3 Master Mode Requests

```
DacAnalogInputWrite( LPDACENV ep,...)
```

```
If ( ep->Filter & DAC_FILTER_REQUEST ) Then  
    Ep->Request = to the request id created by the user  
Else  
    Umode data sent by remote device.
```

Variable Addressing

The generic DAC variable data structures include group and ID fields for identifying the target variable. The generic DAC environment structure provides line and device identification (see *ASE Protocol Translator Interface Specification* for more information on the DAC environment structure).

The DNP 3.0 protocol does not require the group field. The protocol addresses variables by variable offset, starting at 0 for the first variable associated with a DNP 3.0 object. The DNPPT and the user application use the DAC ID field of the DAC variable data structures for variable offset.