NORTEK MANUALS

Integrator's Guide

Generation 2 2D Horizontal Profiler

NORTEK

Table of Contents

1 Introduction

This document includes an overview of data formats and commands relevant for the 2D Horizontal Profiler. You can see if this includes your instrument by looking at its serial number. The 2D Horizontal Profilers are characterized by a six-digit serial number preceded by a four-letter combination. For the 400 kHz 2D profiler, the letter combination is **S4SH**, while for the 1 MHz instrument, the letter combination is either **S1VH** or **S1SH**. Note that instruments with different frequency have slightly different functionality, and not all information given in this manual is necessarily relevant for your instrument. For data formats relevant for the previous generation of 2D instruments, AWAC 2D and AquaPro 2D, please refer to Integrators Guide - Classic.

Nortek online
At our website, www.nortekgroup.com, you will find technical support, user manuals, FAQs and the latest software and firmware. General information, technical notes, and user experience can also be found here. Your feedback is appreciated. If you find errors, omissions, or sections poorly explained, please do not hesitate to contact us. We appreciate your comments and your fellow users will as well.

Contact Information
If you need more information, support or other assistance, you are always welcome to contact us or any of our subsidiaries by email or phone. We recommend first contacting your local sales representative before the Nortek main office. We strive to answer all requests to our support email within one working day. Please include as much information as possible in you initial request, such as instrument serial number, data files, configuration, and information about the set up.

Email: support@nortekgroup.com (for technical support), or inquiry@nortekgroup.com (for other general inquiries)
Phone: +47 67 17 45 00

You can also write us at:
Nortek AS
Vangkroken 2
1351 RUD
Norway

Version/revision	**Date**	**Comments**
Version 2024.1	14.10.24	First release
Version 2026.1	05.03.26	Interfaces chapter included. Updates to Data Formats and Commands chapter. Chapter describing GETALL response included.
Version 2026.2	06.05.26	Links updated

2 Basic interface concepts

This chapter covers the operational modes and how to switch between them. These modes are frequently used during instrument communication, so understanding their role is important before proceeding to interfaces and commands.

2.1 Modes

The instrument operates in several distinct modes, each associated with specific commands for controlling its behavior. For details on which commands are available in each mode, refer to the List of Commands section. Most commands are issued while the instrument is in Command mode.

The available operating modes are:

Command - Used for general commands and control
Data Retrieval - Used to temporarily stop a deployment and downloading data from the recorder
Measurement - Active data collection
Confirmation - Used to confirm transition between modes

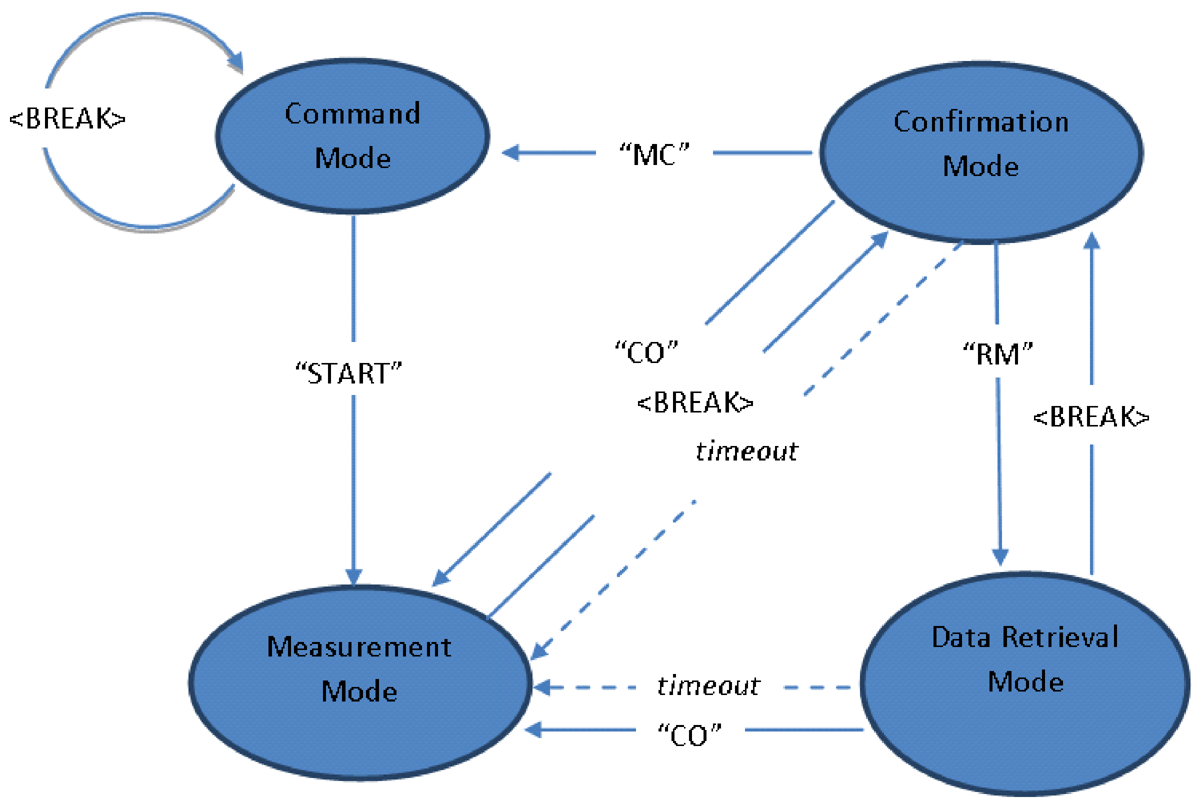

Figure: Instrument modes of operation

Communication with the instrument is initialized by sending a **<BREAK>** signal, as defined in the next section. Depending on the current mode, the **<BREAK>** either places the instrument into Confirmation Mode or restarts Command Mode. The available options for switching modes depend on the instrument's current mode (see the diagram above for details). If no commands are received within a given mode, a timeout occurs, prompting the instrument to resume operation automatically.

Default timeout durations are as follows:

- Confirmation Mode: 60 seconds
- Data Retrieval Mode: 60 seconds
- Command Mode (Serial interface only): 5 minutes

In Command Mode when operating over the serial interface, if no commands are received for 5 minutes, the processor must be reawakened. This can be done by sending either a **<BREAK>** or a sequence of @@@@@@.

All the time out durations listed above can be altered using the **SETINST** command.

2.2 Break

<BREAK> over the serial RS232/RS422 interface is defined as:

@@@@@@ <delay 150 milliseconds> **K1W%!Q** <delay 400 milliseconds> **K1W%!Q**

The @@@@@@ are used to wake up the processor when it is in sleep mode since the instrument will only be able to monitor activity on the serial line when it sleeps. The second sequence of the actual break characters is there to ensure that a break is detected even when the instrument is waking up due to some other cause (e.g. alarm from the real time clock). This ensures that the processor will interpret the following command correctly.

The figure and the table below show the specified timing of the **<BREAK>** sequence:

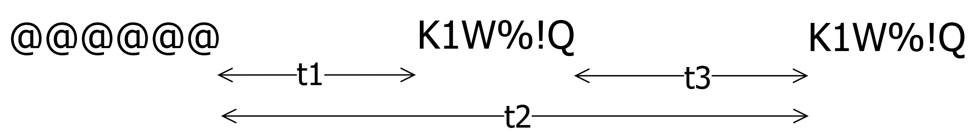

Figure: Break timing

Symbol	Parameter	Min.	Typical	Max.	Unit
t1	Time from end of @-sequence to start of ***first*** K1W%!Q-sequence.	100	150		ms
t2	Time from end of @-sequence to start of ***second*** K1W%!Q-sequence.	500	1000	2000	ms
t3	Time between ***first*** and ***second*** K1W%!Q-sequence.	300	400		ms

2.3 Dual Processor

The 400 kHz profiler uses a two-processor (DSP) design; one dedicated to Doppler processing (BBP) and the other to Interface (SEC). The primary interface is Ethernet, so the Interface processor is only powered when external power is applied. Note that powering through the Ethernet cable will also power the rest of the electronics. As the primary interface, the Ethernet cable takes priority, so if it is powered while the Serial interface is also powered, you will only be able to use Ethernet for communications and data output.

3 Interfaces

This chapter provides an overview of the available communication methods for the 2D Horizontal Profilers. The 1 MHz instrument communicate exclusively via a serial interface, which is accessed through the 8-pin connector. The 400 kHz instrument supports both Ethernet and serial communication. Ethernet communication is handled through the 6-pin connector and serial communication through the 8-pin connector.

> Note: The Ethernet cable must be powered for the interface to function. When powered, the Ethernet connection takes priority over the serial connection, meaning the serial interface will be unavailable while Ethernet is active.

The command structure and data output formats are consistent across both Ethernet and serial communication. In most cases, when the option "Serial output" is enabled, the instrument transmits data over both interfaces simultaneously. You can use either interface to receive the data and issue commands, there is no need to explicitly configure the preferred interface.

3.1 Serial Operation

The 2D Horizontal Profiler (both the 1 MHz and the 400 kHz versions) uses a serial communication configuration of eight data bits, no parity, and one stop bit (8N1). The two frequency variants are described in more detail below.

1 MHz

The instrument supports both RS232 and RS422 protocols. The serial cable provided with the instrument is compatible with both protocols, but a protocol-specific converter is required. We recommend always using converters supplied by Nortek.

By default, the 1 MHz 2D Horizontal Profiler is delivered with RS422 as the communication protocol. If you need to change the communication protocol to RS232, this can be done in the Nortek Deployment software. Make sure to use the latest version of the software, navigate to the Connection tab and follow the step-by-step guide found in the Help section.

400 kHz

Serial communication is handled through the 8-pin connector. The instrument supports both RS232 and RS422 protocols. The serial cable provided with the instrument is compatible with both protocols, but a protocol-specific converter is required. You can switch between RS232 and RS422 using the SETINST command. To change between RS232 and RS422, you must connect via Ethernet. For step-by-step instructions, refer to the FAQ: How can I change the communication protocol from RS232 to RS422 on my AWAC 2 by commands?

The Nortek Deployment software supports connections via both serial and Ethernet. You can also use any other terminal program that supports standard serial communication.

3.2 Ethernet Operation

Communication through Ethernet is only available for the 400 kHz instrument.

The instrument uses TCP (Transmission Control Protocol) for both command processing and data transmission. In TCP/IP networks, each communication channel is uniquely identified by a combination of the IP address and port number. On the Signature, different ports are used for different types of communication:

- **Port 9000** is a Telnet-style ASCII interface (require username / password authentication)
- **Port 9001** is a raw, binary interface (requires username / password authentication)
- **Port 9002** is a data-only channel (no input accepted)
- **Port 9004** is an ASCII data-only channel (no input accepted).

The different ports are described in more detail in the following chapters.

If password protection is disabled via the web interface, the system will accept any input, including an empty password.

When the instrument is in Measurement Mode, any command sent should be preceded by the BBPWAKEUP command. This ensures that the BBP (Baseband Processor) is active and ready to process incoming instructions. If more than 2 seconds have passed since the previous command, another BBPWAKEUP must be issued. If you're unsure of the current instrument mode, it's good practice to send BBPWAKEUP before commands like GETSTATE or INQ.

3.2.1 Telnet Connection (Port 9000)

The Telnet interface, accessible via TCP/IP port 9000, is used for direct user interaction with the instrument. This dedicated port accepts ASCII-based commands and returns human-readable responses. The list of supported commands is provided in the Commands section.

To connect using the Windows Telnet client, enter the following command in the terminal:

```
telnet <ip_address> 9000
```

You will be prompted for a username and password, these are by default:

```
Username: nortek
Password: (leave blank - password protection is disabled by default)
```

Password protection can be enabled through the instrument’s web interface.

When logging in you will see:

```
Nortek Signature Command Interface
```

This interface is very similar to the direct serial interface (RS232/RS422), with some additions to simplify interaction. Most notably, you can send a <BREAK> command to the Doppler DSP by pressing Ctrl-C

(ASCII 0x03). The internal application handles waking up the Doppler processor and ensures proper timing of the break signal.

The Telnet server does not echo typed characters by default. If you wish to see or edit commands before sending them, you must enable local echo and local line editing in your Telnet client. A client such as PuTTY supports these features.

Port 9000 is dedicated solely to ASCII communication. Do not use a Telnet client to connect to other instrument ports that transmit binary data (see next section for details).

To terminate the Telnet session, press Ctrl-X (ASCII 0x18).

3.2.2 Raw Connections (Ports 9001, 9002, and 9004)

A port can be understood as a address point between two communicating parts. When first connecting to a data listening port, the following identification string is sent by the instrument:

```
\r\nNortek <name> Data Interface\r\n
```

Here, <name> is replaced by the instrument's host name, allowing clients to verify which device responded.

The instrument uses the following TCP ports for data streaming and control:

Port 9001 - Binary data and command interface

- Used for machine-driven communication and control
- Requires username/password authentication
- Provides access to binary data generated in Measurement Mode
- Translates serial port data directly into TCP/IP over Ethernet, and the commands and data formats are identical to those used over the serial port.
- The instrument handles correct internal timing when breaks are sent. Break signal can be sent in two ways:
  - Sending the string **K1W%!Q<CR><LF>** as described in the Break chapter
  - Send **Ctrl-C** (ASCII 0x03) as a standalone character (not embedded in a command)
- Standard streaming record delineation techniques must be used to ensure the received data is correctly synchronized for decoding

Port 9002 - Data Output Channel

- Data-only channel with no command input
- Can be used by real-time display applications while configuration is handled by a separate system.

Port 9004 - ASCII Data Output Channel

- Outputs ASCII-formatted telemetry data with no command input
- Only active when the instrument is configured to generate telemetry output in ASCII format

A Telnet client should **not** be used to access ports 9001, 9002, or 9004. Telnet introduces its own control sequences and binary interpretation rules, which are not compatible with the raw data these ports provide. Using a Telnet client on these ports will result in unintended control characters being sent and binary characters being misinterpreted.

3.2.3 FTP

The instrument's internal data recorder can be accessed over Ethernet using any standard FTP (File Transfer Protocol) client. When combined with telemetry options, FTP offers a convenient method for downloading measured data at regular intervals, especially in applications where true real-time data access is not required. Only the telemetry file (`telemetryfile.bin`) can be deleted via FTP. Other data files are read-only.

When an FTP connection is active, the internal state of the instrument is changed so that commands are no longer processed. Any commands sent during an active FTP session will return an error. If a break is sent during an active FTP transfer, the FTP connection will be forcibly terminated.

If an FTP connection is done while the instrument is in Measurement Mode (see Modes diagram), the FTP connection is made through Data Retrieval Mode. Upon terminating the FTP session, the instrument returns to its previous mode. An FTP session will terminate automatically if no FTP commands or data transfers occur for 120 seconds.

The telemetry option implemented in the Doppler processor enables system integrators to regularly offload subsets of the data by using FTP. When the network processor receives an incoming FTP request, it will interrupt the Doppler processor by entering Data Retrieval Mode and mount the file system of the recorder. The telemetry file can be deleted after it has been downloaded, which is particularly suitable for event driven data downloads. If the instrument was started using the DEPLOY command, it will resynchronize to its measurement time base once the FTP session ends.

3.2.4 HTTP

The instrument includes a built-in HTTP interface for downloading individual data files. This feature is accessible from the instrument's web page by clicking the "Data Download (HTTP)" link on the main page. HTTP download is useful in environments where firewall or security settings restrict the use of FTP.

3.2.5 UDP

The instrument also supports UDP (User Datagram Protocol) for real-time data transmission. Unlike TCP, UDP does not require a connection to be established; the data collection software simply waits for incoming data from the instrument.

UDP is useful in scenarios where instrument power may be intermittently interrupted, as the data stream resumes automatically after a power cycle. However, UDP does not guarantee delivery—on unreliable or noisy networks, some datagrams may be lost. If reliable delivery is critical, TCP is recommended.

To enable UDP in a power-resilient configuration use the instrument's web interface to configure the IP address and port of the data collection software. The IP address specifies the target client; the port (range: 9000–9500) can be used to distinguish instruments. The same port may be reused across instruments if the software uses the source IP address to identify them. Once configured, the instrument's Ethernet processor will begin automatically transmitting real-time data to the specified address/port. After a power cycle, an instrument originally in Measurement Mode re-enters Measurement Mode and resumes transmission without needing to re-establish communication.

3.2.6 PTP/NTP

The instrument supports time synchronization via PTP (Precision Time Protocol) and NTP (Network Time Protocol), both of which can be enabled through the instrument's web interface.

Precision Time Protocol (PTP – IEEE 1588)

PTP enables distribution of high-resolution absolute time across an Ethernet network. The Signature can be configured to operate as a PTP slave, synchronized to a customer-supplied PTP master clock located on the same LAN. The instrument contains a high-resolution clock which is synchronized and conditioned using PTP when enabled. The timestamps contained within the data records are then generated from this clock. When synchronized, these timestamps are typically aligned to within ~10 microseconds. The PTP master clock must use UDP (layer three) and be configured for two-step operation with an end-to-end delay mode in order to be compatible with the Signature series. Using PTP does not affect the choice of UDP or TCP for the transport of data.

Network Time Proyocol (NTP)
NTP provides time synchronization accuracy of approximately ±1 ms on a local area network (LAN) and ±10 ms across a wide area network (WAN).

3.3 Software

The primary way to communicate with your 2D Horizontal Profiler is through the instrument software called Nortek Deployment. This software is free of charge and can be accessed via our software web page or downloaded directly from the Microsoft Store.

The Nortek Deployment software offers the quickest and easiest way to perform tasks such as:

1. Configuring and starting a new deployment
2. Stopping a deployment and downloading data from the recorder
3. Performing a function test of the instrument
4. Checking for and updating the instrument firmware

Most standard deployment setups can be completed using the Plan and Deploy feature in the software. For more advanced configurations, beyond what the software allows, you can customize your setup using the commands described in the Commands chapter. We recommend using the software whenever possible, as it covers most common cases. By activating the Advanced mode in the software you will get the option to add custom commands in the last step of the Plan and Deploy process. The .deploy file which is generated by the software and saved to the instrument is command-based and can be read directly into a command interface.

3.4 Web Browser

When connected to the 400 kHz instrument over Ethernet, you can access the instrument web page through your web browser. The site can be accessed directly by typing in the IP address into the web browser.

The instrument web page allows you to for example

1. Check for updates and update firmware
2. Set your desired Network Configuration, for example enabling UDP data transmission
3. Retrieve support file
4. Enable password protection
5. Download data

3.5 Command interface

The command interface enables direct communication with the instrument via either a serial or Ethernet connection. Commands are described in the Commands chapter and can be used both to interact with the instrument through a terminal and to customize the configuration. The .deploy file is command-based and can be read directly into the command interface.

Each command accepts a defined set of arguments, which are detailed in the relevant chapters. Some commands require at least one argument in order to execute properly.

The command interface is ASCII-based and line-oriented. Each command must be terminated with a carriage return and line feed (CR/LF). For enhanced data integrity, commands may also be used in NMEA format, which includes a prefix and checksum. When using NMEA-style commands, the instrument will return argument names in the response.

3.5.1 Error messages and validation

Comprehensive validation and error handling are built into the command interface. When you send the **SAVE** command, the current configuration is immediately verified. If you skip **SAVE**, the deployment plan will instead be validated when issuing the **START** or **DEPLOY** command.

If the deployment plan contains invalid elements, such as parameters with values outside their permitted range, an **ERROR** will be returned. To diagnose the issue, use the **GETERROR** command. This will return a detailed error message, including the parameter and its valid range. Each validation error must be corrected before proceeding, and additional errors may be revealed as each one is resolved. Below is an example of how to use the **GETERROR** command.

```
% Setting the measurement interval to 21600 seconds on a 2D Horizontal Profiler 400 kHz using the SETPLAN command and receiving an OK
SETPLAN,MIAVG=21600
$PNOR,OK*2B

% Saving the configuration and receiving an ERROR
SAVE,CONFIG
$PNOR,ERROR*77

% Following up with the GETERROR command to investigate
GETERROR
$PNOR,GETERROR,NUM=227,STR="Invalid setting: Plan Profile Interval",LIM="GETPLANLIM,MIAVG=([1;7200])"*56
```

The response to the **GETERROR** command indicates that the measurement interval must be within 1:7200 seconds and you have to go back and reconfigure this before you can proceed. Note that the limits will change depending on the rest of your configuration

3.5.2 Limits

The valid range for the various arguments can be retrieved by using the **GETxxxLIM** commands where **xxx** is the command you want the limits for. For example, if you need the limits for all arguments associated with the **SETAVG** command, you send the **GETAVGLIM** command. You can also retrieve the limits for one specific argument by including this when sending the **GETxxxLIM** command. As in the example above when you are configuring the measurement interval for the average measurements, you can get the valid range beforehand by sending the command **GETPLANLIM,MIAVG**. The same goes for arguments associated with all types of applications, the valid ranges for burst measurements can be retrieved by the command **GETBURSTLIM** etc**.**

3.5.3 Starting the measurements

There are two commands that can be used to start the instrument: **DEPLOY** and **START**. While both initiate measurement, they differ significantly in how the instrument behaves after transitions between modes.

When the instrument is started using the **START** command, it will begin measuring immediately each time it enters Measurement Mode—such as after applying power or following a timeout from Data Retrieval Mode.

In contrast, the **DEPLOY** command synchronizes the instrument's measurements to a fixed time schedule. Even if the instrument has been out of Measurement Mode, it will align its operation with the original timing. This means that if a scheduled measurement time has already passed when the instrument powers on, it will wait until the next scheduled interval before beginning to ping.

The **START** command is useful in integration scenarios where immediate measurement is required. However, it may result in inconsistent time stamps in the recorded data. Use **DEPLOY** if synchronized, regular sampling intervals are critical.

3.6 Telemetry

The term telemetry refers to both the transfer of data, via serial or Ethernet connection, and the storage of selected data subsets. When telemetry is enabled, you can choose to activate file output, serial output, or both.

Telemetry data can be formatted in various ways, including NMEA and binary formats. For a complete list of supported formats, refer to the Telemetry Data Formats chapter. Note that different data types use distinct telemetry data formats.Telemetry for each data type must be configured individually using its specific command. For example does SETTMAVG configure telemetry of average current data while SETTMWAVE configure telemetry of processed wave data.

The instrument stores each individual ping to the raw data file on its internal recorder. Telemetry output (file or serial) can optionally include internally averaged velocity data, which provides a lower-bandwidth, processed data product suitable for real-time or near-real-time applications. Whenever averaging is enabled, the data undergoes an internal quality control prior to the averaging. For more details about the quality control, please refer to the Telemetry Quality Control chapter. After deployment, the raw data can be retrieved and post-processed as needed.

File output

When file output is enabled, a subset of the data is saved to the recorder in the configured data format. The file is always named `telemetryfile.bin`, and all telemetry data is written to this file. The file output option is typically used when real-time, continuous output is not feasible, and data must be stored for later retrieval. The file can be downloaded either in chunks or as a complete file. A checksum or CRC can be applied during download to ensure data integrity. System integrators can offload the telemetry file regularly using FTP or the command interface, and the file can be deleted after retrieval. The recorder is accessible in both Command Mode and Data Retrieval Mode (see Modes diagram). Downloading in Data Retrieval Mode allows the same deployment to continue when returning to Measurement Mode. Note that the instrument does not perform measurements while in Data Retrieval Mode.

Serial output

When serial output is enabled, the instrument streams telemetry data directly over the active communication interface, which may be either serial or Ethernet. The data transmitted matches exactly what is stored in the telemetry file.

4 Commands

This chapter covers the commands that can be used to control an 2D Horizontal Profiler instrument. Please note that not every command is relevant for every instrument. The commands relevant for your specific instrument depends on the instrument type and licenses.

Some pointers:

- A configuration of the instrument should always start with setting the default configuration, e.g.:
```
SETDEFAULT,CONFIG
OK
```

- All command parameters should be set explicitly, e.g.:
```
SETAVG,NC=10,BD=0.7
OK
```

- Sometimes you may get an ERROR response after tying to save the configuration or start/deploy the instrument. This doesn't necessarily mean that something is wrong with the instrument, but is most often a sign that the configuration isn't going to work. Any ERROR response can be interrogated with **GETERROR**, e.g.:
```
SAVE,CONFIG
ERROR
GETERROR
GETERROR,NUM=56,STR="Invalid setting: Avg Average Interval too low for the configured number of pings and profiling distance",LIM="GETAVG1LIM,AI=([360;1800])"
OK
```
  Here, the instrument is reporting that we have set the average interval to be too short, and it provides the limits for the AI that are allowed if we are going to keep the same number of pings and profiling distance. You could increase the average interval, decrease the number of pings, or decrease the profiling distance (i.e. number/size of cells) to fix the error.

Data Limit Formats

The limits for the various arguments are returned as a list of valid values, and/or ranges, enclosed in parenthesis **()**. An empty list, **()**, is used for arguments that are unused/not yet implemented. Square brackets **[]** signify a range of valid values that includes the listed values. String arguments are encapsulated with **“”**, like for normal parameter handling. A semicolon, **;**, is used as separator between limits and values.

The argument format can also be inferred from the limits, integer values are shown without a decimal point, floating point values are shown with a decimal point and strings are either shown with the string specifier, **“”**, or as a range of characters using **‘’** for specifying a character.

Examples:

[1;128] - Integer value, valid from 1 to 128

([1300.00;1700.00];0.0) - Floating point value, valid values are 0.0 and the range from 1300.00 to 1700.00.

(['0';'9'];['a';'z'];['A';'Z'];'.') - String argument with valid characters being . and the character ranges a-z, A-Z, 0-9 .

("BEAM") - String argument with BEAM being the only valid string.

(0;1) - Integer value with two valid values, 0 and 1.

NMEA interface example:

```
$PNOR,GETAVGLIM*22
$PNOR,GETAVGLIM,NC=([1;128]),CS=([0.25;2.00]),BD=([0.10;45.00]),CY=
("BEAM"),PL=([-40.0;0.0];-
100.0),AI=([1;300]),VP=([0.000;0.100]),VR=([1.25;5.00]),DF=([0;3]),
NPING=([1;4])*46
$PNOR,OK*2B
```

Regular interface example:

```
GETPLANLIM
([1;3600]),(0;1),(),([0;2]),(),([0.0;50.0]),(0;1),([10;21600]),(),
([1300.00;1700.00];0.0),(['0';'9'];['a';'z'];['A';'Z'];'.'),(0;1)
OK
```

4.1 List of Commands

Below is a list of all available commands with a short description and information about which mode they can be used in. For more information about each command see the following chapters. The arguments that can be used with each command are described in the respective chapter. Note that some of the commands requires at least one argument to be used.

Command	Description	Mode
BREAK	Wakes up the processor when it is in sleep mode	ALL
SETINST	Set instrument main settings	COMMAND
GETINST	Get instrument main settings	COMMAND
GETINSTLIM	Set instrument main setting limits	COMMAND
SETCLOCK	Set instrument clock	COMMAND RETRIEVAL
GETCLOCK	Get instrument clock	COMMAND RETRIEVAL
SETCLOCKSTR	Set instrument clock as string	COMMAND RETRIEVAL
GETCLOCKSTR	Get instrument clock as string	COMMAND RETRIEVAL
SETSWMETA	Set software metadata (software version).	COMMAND
GETSWMETA	Get software metadata (software version).	COMMAND
GETSWMETALIM	Get software metadata limits	COMMAND
SETPLAN	Set deployment plan parameters	COMMAND
GETPLAN	Get deployment plan parameters	COMMAND
GETPLANLIM	Get deployment plan parameter limits	COMMAND
SETAVG	Set instrument average mode settings	COMMAND
GETAVG	Get instrument average mode settings	COMMAND
GETAVGLIM	Get instrument average mode limits	COMMAND
GETMEM	Get recorder data memory usage	COMMAND
GETPWR	Returns the power consumption in mW for the various measurements enabled as well as the overall value	COMMAND
READABSPRESSURE	Returns the absolute pressure reading from the instrument	COMMAND
GETPRECISION	Returns the precision for primary plan	COMMAND
GETPRECISION1	Returns the precision for secondary plan	COMMAND
SETUSER	Set instrument user settings	COMMAND
GETUSER	Get instrument user settings	COMMAND

GETUSERLIM	Get instrument calibration parameter limits	COMMAND
ID	Get instrument Id	COMMAND
SETDEFAULT	Reload default settings	COMMAND
SAVE	Save settings for next measurement	COMMAND
DEPLOY	Deploy the instrument	COMMAND
START	Start the instrument	COMMAND
MC	Go into command mode	CONFIRMATION
RM	Go into data retrieval mode	CONFIRMATION
CO	Go into measurement mode	CONFIRMATION RETRIEVAL
FWRITE	Write tag or string to file	COMMAND CONFIRMATION RETRIEVAL
POWERDOWN	Set instrument in sleep mode	COMMAND
ERASE	Erase all files on the recorder	COMMAND
FORMAT	Format the recorder	COMMAND
SECREBOOT	Reboots the SEC	COMMAND
LISTFILES	Lists the files on instrument	COMMAND RETRIEVAL
DOWNLOAD	Read file data	COMMAND RETRIEVAL
READCFG	Read current configuration	COMMAND
INQ	Inquires the instrument state	COMMAND CONFIRMATION RETRIEVAL MEASUREMENT
GETSTATE	Returns information about the current operational state of the instrument	COMMAND CONFIRMATION MEASUREMENT RETRIEVAL
GETERROR	Returns a full description of the last error condition to occur	COMMAND CONFIRMATION MEASUREMENT RETRIEVAL
GETALL	Retrieves all relevant configuration information for the instrument	COMMAND
RECSTAT	Return recorder state	COMMAND RETRIEVAL

GETMISCLIM	This command returns configuration limits that cannot be returned through the relevant commands	COMMAND
GETXFAVG	Returns the "Beam to XYZ" transfer matrix for average measurements	COMMAND
GETXFBURST	Returns the "Beam to XYZ" transfer matrix for burst measurements	COMMAND
SETTMAVG	Set averaging mode telemetry settings	COMMAND
GETTMAVG	Get averaging mode telemetry settings	COMMAND
GETTMAVGLIM	Get instrument averaging mode limits	COMMAND
DOWNLOADTM	Read telemetry file data	COMMAND RETRIEVAL
STOREHEADERTM	Stores the GETALL to the telemetry file	COMMAND
ERASETM	Erase the telemetry file	COMMAND RETRIEVAL
TAG	Write a Tag to output file and data output	COMMAND CONFIRMATION MEASUREMENT RETRIEVAL
PTPSET	Set precision time protocol parameters	COMMAND
PTPGET	Get precision time protocol parameters	COMMAND
BBPWAKEUP	Wakes up the Doppler processor (ethernet interface only)	COMMAND CONFIRMATION MEASUREMENT RETRIEVAL
GETHW	Returns hardware specifications	COMMAND CONFIRMATION MEASUREMENT RETRIEVAL
ADDLICENSE	Add license key	COMMAND
DELETELICENSE	Delete license key	COMMAND
LISTLICENSE	Lists all license keys in instrument	COMMAND

4.2 Wake processor from sleep mode

Command: BREAK
Command type: ACTION
Mode: ALL

Used to wake up the processor when it is in sleep mode.

The @@@@@@ are used to wake up the processor when it is in sleep mode since the instrument will only be able to monitor activity on the serial line when it sleeps. The second sequence of the actual break characters is there to ensure that a break is detected even when the instrument is waking up due to some other cause (e.g. alarm from the real time clock). This ensures that the processor will interpret the following command correctly.

See more info on Break

Example:

```
@@@@@@ <delay 100milliseconds> K1W%!Q <delay 300milliseconds> K1W
%!Q
```

4.3 Instrument main settings

Commands: SETINST, GETINST, GETINSTLIM,
Command type: CONFIGURATION
Mode: COMMAND

Instrument main settings

Argument	Description
BR	Baud Rate 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400, 460800, 625000, 1250000 Unit: [bit/s]
RS	Serial protocol 232, 422
LED	Enable/disable LED blink in head. When set to "ON24H" the LED will illuminate the first 24 hours of the measurement. "OFF": Turn LED Off "ON": Turn LED On "ON24H": Turn LED On for 24 hours
ORIENT	Sets the instrument orientation. Not used for DVL. "AUTOXUPDOWN": Auto X up/down "AUTOYUPDOWN": Auto Y up/down "AUTOZUPDOWN": Auto Z up/down "ZUP": Z up "ZDOWN": Z down "XUP": X up "XDOWN": X down "YUP": Y up "YDOWN": Y down "AHRS3D": The instrument will detect and change the orientation as the instrument is moved. Used if the instrument has an AHRS installed and does not have a constant defined UP direction

CMTOUT	Command mode timeout Unit: [s]
DMTOUT	Data retrieval mode timeout Unit: [s]
CFMTOUT	Confirmation mode timeout Unit: [s]

SETINST

Set instrument main settings

Example:
```
SETINST,LED="ON"
```

GETINST

Get instrument main settings

Example:
```
GETINST,BR=460800,RS=232,LED="ON",ORIENT="AUTOZUPDOWN",CMTOUT=300,D
MTOUT=60,CFMTOUT=60
```

GETINSTLIM

Set instrument main setting limits

4.4 Clock settings

Commands: SETCLOCK, GETCLOCK,
Command type: CONFIGURATION
Mode: COMMAND, RETRIEVAL

Instrument Real Time Clock specified in date parts

Argument	Description
YEAR	The year, e.g. 2020 Values: [0; 9999]
MONTH	The number of month 1-12 (Jan = 1) Values: [1; 12]
DAY	The number day of month 1-31 Values: [1; 31]

HOUR	The hour of day 0-23 Values: [0; 23]
MINUTE	The minute of hour 0-59 Values: [0; 59]
SECOND	The second of minute 0-59 Values: [0; 59]

Note: Note that all parameters must be set when using the set command

SETCLOCK

Set instrument clock

Example:

```
SETCLOCK,YEAR=2020,MONTH=11,DAY=28,HOUR=13,MINUTE:15,SECOND=45
```

GETCLOCK

Get instrument clock

Example:

```
GETCLOCK
2022,12,13,15,24,33
OK
```

4.5 Clock settings as strings

Commands: SETCLOCKSTR, GETCLOCKSTR,
Command type: CONFIGURATION
Mode: COMMAND, RETRIEVAL

Set or retrieve the Real Time Clock using a string. Must use the format as shown: yyyy-MM-dd HH:mm:ss

Argument	Description
TIME	Text string with this format yyyy-MM-dd HH:mm:ss (use UTC)
OFFSET	Offset from the standard time
TZ	Time zone. Valid values are UNKNOWN, LOCAL and a string that complies to "+/-HH:MM"

SETCLOCKSTR

Set instrument clock as string

Example:
```
SETCLOCKSTR,TIME="2020-11-12 14:27:42"
```

GETCLOCKSTR

Get instrument clock as string

Example:
```
GETCLOCKSTR, TIME
GETCLOCKSTR, TIME = "2014-11-12 14:27:42"
```

4.6 Software metadata

Commands: SETSWMETA, GETSWMETA, GETSWMETALIM,
Command type: CONFIGURATION
Mode: COMMAND

Set or retrieve software version string using SWVER (e.g. 1.23.45(e)).

Argument	Description
SWVER	Arbitrary version tag, e.g. 1.23.45(e)

SETSWMETA

Set software metadata (software version).

Example:
```
SETSWMETA,SWVER="1.23.45(e)"
```

GETSWMETA

Get software metadata (software version).

Example:
```
GETSWMETA,SWVER
GETSWMETA,SWVER="1.23.45(e)"
```

GETSWMETALIM

Get software metadata limits

4.7 Deployment plan parameters

Commands: SETPLAN, GETPLAN, GETPLANLIM,
Command type: CONFIGURATION
Mode: COMMAND

Set and get high-level plan configuration settings.

Argument	Description
MIAVG	Averaging measurements interval Unit: [s]
AVG	Enable(1)/disable(0) averaging measurements.
SA	Salinity. Unit: [ppt]
BURST	Enable(1)/disable(0) burst measurements.
MIBURST	Burst measurements interval. Unit: [s]
SV	Sound velocity. SV = 0 will set sensor to use measured sound velocity Unit: [m/s]
FN	Filename of the raw data file where all the measured binary data will be stored. If FN="", no data is stored on the recorder. FN must be changed if a file FN with a different configuration already exists on the recorder. Values: ['a'; 'z']; ['A'; 'Z']; ['0'; '9']; '_'; '.' | Max Length: 30
SO	Enable(1)/disable(0) serial output With this option enabled, the instrument will stream all collected raw binary data live, either over serial or ethernet connection.
FREQ	Transmit frequency. This is normally the instruments (one) frequency. In the case of the Signature 55 we may choose between 55Hz or 75Hz. Unit: [khz]
NSTT	Number slot time table. Set to 0 giving the default number of slots

Note: The valid range for the various arguments should be verified using the GETPLANLIM command, as the values listed here may change with firmware versions and instrument frequencies.

SETPLAN

Set deployment plan parameters

Example:
```
SETPLAN,FREQ=75
```

GETPLAN

Get deployment plan parameters

GETPLANLIM

Get deployment plan parameter limits

4.8 Average mode settings

Commands: SETAVG, GETAVG, GETAVGLIM,
Command type: CONFIGURATION
Mode: COMMAND

The instrument averaging mode settings and relevant limits.

Argument	Description
NC	Number of cells
CS	Cell size Unit: [m]
BD	Blanking distance Unit: [m]
CY	Co-ordinate system BEAM, XYZ, ENU
PL	Power level (range -20.0 dB to 0.0 dB, -100 dB to switch off transmit). Unit: [dB] | Values: [-20; 0]; -100
AI	Average interval Unit: [s]
VR	Velocity range along beam Unit: [m/s]
NPING	Number of pings
NB	Number of beams
CH	Beam selection. E.g. CH=123 will enable beams 1, 2 and 3.
MUX	Enable(1)/disable(0) multiplexing. 0: ping all beams in parallel 1: ping beams in sequence
BW	Bandwidth selection.

	NARROW, BROAD
ALTI	Enable(1)/disable(0) altimeter measurements
BT	Enable(1)/disable(0) bottom tracking
ICE	Enable(1)/disable(0) ice tracking
ALTISTART	Altimeter start Unit: [m]
ALTIEND	Altimeter start Unit: [m]
RAWALTI	Raw altimeter recording interval
DF	Data format 7: DF7 Binary v7
ZCELL	Enable(1)/disable(0) Z cell measurements

Note: The actual valid range for the various parameters for your specific instrument and firmware can be found by using the GETAVGLIM command. This command has the same arguments as the SETAVG/GETAVG commands shown in the list above. The output format for limits is described in Data Limit Formats

SETAVG

Set instrument average mode settings

Example:
```
SETAVG,BD=1
```

GETAVG

Get instrument average mode settings

GETAVGLIM

Get instrument average mode limits

4.9 Get memory usage

Command: GETMEM
Command type: INFO
Mode: COMMAND

Returns the amount of memory that will be stored on the recorder per hour for the burst and average data as well as the combined plan value. Alternate mode values is also supported.

Argument	Description
PLAN	Combined burst and average memory usage for primary plan Unit: [MB/h]
BURST	Burst memory usage for primary plan Unit: [MB/h]
AVG	Average data memory usage for primary plan Unit: [MB/h]
PLAN1	Combined burst and average memory usage for secondary plan Unit: [MB/h]
BURST1	Burst memory usage for secondary plan Unit: [MB/h]
AVG1	Average data memory usage for secondary plan Unit: [MB/h]
TOTAL	Total memory usage Unit: [MB/h]

Example:
`GETMEM,TOTAL`

4.10 Get power use

Command: GETPWR
Command type: INFO
Mode: COMMAND

Returns the power consumption in milliWatts for the various measurements enabled as well as the overall value. The plan values include the sleep mode power consumption in addition to the sum of average and burst mode values. The reported values are adjusted according to the input voltage to the system when the command is executed.

Argument	Description
PLAN	Combined power use on both burst and average sampling for secondary plan Unit: [mW]
BURST	Power use on burst data sampling for secondary plan Unit: [mW]
AVG	Power use on average data sampling for secondary plan Unit: [mW]
TOTAL	Total power use Unit: [mW]

Example:
```
GETPWR,TOTAL
```

4.11 Read absolute pressure

Command: READABSPRESSURE
Command type: INFO
Mode: COMMAND

Returns the current absolute pressure measurement from the instrument in dBar.

Argument	Description
VAL	Absolute pressure value returned by the instrument. Unit: [dBar]

Example:
```
READABSPRESSURE
10.812
OK
```

4.12 Get measured precision

Commands: GETPRECISION, GETPRECISION1,
Command type: INFO
Mode: COMMAND

Returns the precision in the horizontal plane and along the beam in cm/s for the various measurement modes.

Argument	Description
AVGHORZ	Precision in the horizontal plane in average mode Unit: [cm/s]
BURSTHORZ	Precision in the horizontal plane in burst mode Unit: [cm/s]
BEAM5	Precision in the vertical plane (beam 5) in burst mode Unit: [cm/s]
AVGBEAM	Precision along beam in average mode Unit: [cm/s]
BURSTBEAM	Precision along beam in burst mode Unit: [cm/s]

GETPRECISION

Returns the precision for primary plan

Example:
```
GETPRECISION,AVGHORZ
```

GETPRECISION1

Returns the precision for secondary plan

Example:
```
GETPRECISION1,AVGHORZ
```

4.13 Instrument user settings

Commands: SETUSER, GETUSER, GETUSERLIM,
Command type: CONFIGURATION
Mode: COMMAND

The SAVE,USER command must be sent to save changes in USER parameters.

Argument	Description
POFF	Set the offset value of the pressure sensor. Unit: [dbar] | Values: [0; 11]
DECL	Magnetic declination. The value will be added to the compass value Unit: [deg]
HX	Hard iron x-component. Unit: [Gauss]
HY	Hard iron y-component. Unit: [Gauss]
HZ	Hard iron z-component. Unit: [Gauss]

SETUSER

Set instrument user settings

Example:
```
SETUSER,POFF=4.5,DECL=1.4,HX=100,HY=100,HZ=100
OK
SAVEUSER
OK
```

GETUSER

Get instrument user settings

GETUSERLIM

Get instrument calibration parameter limits

4.14 Get instrument ID

Command: ID
Command type: INFO
Mode: COMMAND

Commands for accessing instrument name and serial number

Argument	Description
STR	Instrument name Values: '_'; '.' | Max Length: 64
SN	Serial number Values: [0; 2147483647]

Example:

```
ID
"Signature1000",900002
ID,STR
"Signature1000"
```

4.15 Reload default settings

Command: SETDEFAULT
Command type: ACTION
Mode: COMMAND

Reload default settings.

Argument	Description
CONFIG	Restore all settings below except USER and INST to default values. Legacy argument ALL acts as CONFIG.
AVG	Restore average mode settings to default.
INST	Restore instrument main settings to default.
TMAVG	Restore telemetry average settings to default.
PLAN	Restore plan settings to default.

BURST	Restore burst settings to default.
PTP	Restore precision time protocol to default.
BT	Restore bottom track settings to default.
USER	Restore user calibration to default.
TMBURST	Restore telemetry burst to default.
TMALTI	Restore altimeter telemetry settings to default.
DVL	Restore DVL settings to default.

Example:
`SETDEFAULT,CONFIG`

4.16 Save settings

Command: SAVE
Command type: ACTION
Mode: COMMAND

Save current settings for next measurement. At least one argument must be specified for the SAVE command.

Argument	Description
CONFIG	Save all settings except INST and USER settings. Legacy argument ALL acts as CONFIG.
AVG	Save AVG settings.
INST	Save INST settings.
TMAVG	Save telemetry average settings.
PLAN	Save plan settings.
BURST	Save burst settings.
PTP	Save precision time protocol settings.
TMBT	Save telemetry bottom track settings.
USER	Save user instrument settings.
TMBURST	Save telemetry burst settings.
TMALTI	Save altimeter telemetry settings.
DVL	Save DVL settings.

Example:
`SAVE,CONFIG`

4.17 Deploy instrument

Command: DEPLOY
Command type: ACTION
Mode: COMMAND

Deploy the instrument. Start the measurement at the time specified in the string argument. The format must be exactly as shown. If no time value is passed, the deployment will start immediately.

The number of seconds to the specified deployment time is returned.

The DEPLOY command will save the configuration as well as deploying the instrument, as if a SAVE,CONFIG command were sent.

Argument	Description
TIME	yyyy-MM-dd HH:mm:ss

Note: The difference between DEPLOY and START is that the latter will immediately start a measurement any time the instrument state returns to Measurement mode such as by applying power or timeout from Data Retrieval Mode. If DEPLOY is used, be aware that if the deployment time has passed when the battery is connected, the instrument will resynchronize its data sampling according to the deployment time and the instrument configuration. This means you may have to wait for one average measurement interval or one burst measurement interval before the instrument starts to ping.

Example:
```
DEPLOY,TIME="2020-12-18 15:30:00"
592
OK
```

4.18 Start instrument

Command: START
Command type: ACTION
Mode: COMMAND

Start the instrument (go in measurement mode). Specify the order of the alternating plans by using the PLAN argument.

Note that the START command will save the configuration as well as starting the measurement, as if a SAVE,CONFIG command were sent.

Argument	Description
PLAN	Select plan to start 0: Plan 1: Plan1

Example:
`START,PLAN=0`

4.19 Enter command mode

Command: MC
Command type: ACTION
Mode: CONFIRMATION

Sets instrument in command mode from confirmation mode.

Example:
`MC`

4.20 Data retrieval mode

Command: RM
Command type: ACTION
Mode: CONFIRMATION

Go into data retrieval mode from confirmation mode and access the instruments recorder.

Example:
`RM`

4.21 Enter measurement mode

Command: CO
Command type: ACTION
Mode: CONFIRMATION, RETRIEVAL

Continue in measurement mode from confirmation mode or data retrieval mode. Instrument returns to collecting data according to the current configuration.

Example:
`CO`

4.22 Write to file

Command: FWRITE
Command type: ACTION
Mode: COMMAND, CONFIRMATION, RETRIEVAL

Write tag or string to file

Argument	Description
FNUM	File identifier for telling which file the info is written to. Default is 0. 0: File defined in the PLAN command 1: Telemetry file Default: 0
ID	Identifier. Tell the parser how to interpret the string. Default value: 0 0: Comment 1: Dive location 2: Surface location Default: 0
STR	String. Max Length: 200

Note: Fwrite STR and B64STR cannot be set together.

Example:
`FWRITE,FNUM=1,STR="ABCDEF"`

4.23 Power down

Command: POWERDOWN
Command type: ACTION
Mode: COMMAND

Power down the instrument to set it in sleep mode.

Example:
`POWERDOWN`

4.24 Erase files on recorder

Command: ERASE
Command type: ACTION
Mode: COMMAND

Erase all files on the recorder

Argument	Description
CODE	Code should be 9999 9999

Example:
`ERASE,CODE=9999`

4.25 Format recorder

Command: FORMAT
Command type: ACTION
Mode: COMMAND

Format the recorder. Note that this can take minutes depending on the recorder size.

Argument	Description
CODE	Code should be 9999 9999

Example:
```
FORMAT,CODE=9999
```

4.26 SEC reboot

Command: SECREBOOT
Command type: ACTION
Mode: COMMAND

This command is only relevant for the 400 kHz instrument

Reboots the SEC (Interface processor). This command makes it possible to reboot the SEC without going via the web interface.

Argument	Description

Example:
```
SECREBOOT,9999
```

4.27 List files

Command: LISTFILES
Command type: INFO
Mode: COMMAND, RETRIEVAL

Lists the files stored on the instrument recorder.

Argument	Description
OPT	Amount of file information to list. "normal": Normal file info. "la": Lists extended information.

Example:

```
LISTFILES,OPT="NORMAL"
```

4.28 Download

Command: DOWNLOAD
Command type: ACTION
Mode: COMMAND, RETRIEVAL

This command reads a file from the recorder to the terminal window.

Argument	Description
FN	Filename Values: ['a'; 'z']; ['A'; 'Z']; ['0'; '9']; '_'; '.' | Max Length: 30
SA	Start address (offset) of the first byte to be returned
LEN	Number of bytes to be downloaded
CRC	Use Cyclic redundancy check
CKS	Use Checksum

Note: If no parameters other than the file name are sent with the DOWNLOAD command, the complete file is directly returned, without the number of bytes to follow. The end of the file can then be detected by parsing the OK<CR><LF>.

The parameters can be used to download the file in several pieces. The number of bytes to follow will then be returned in ASCII format and terminated with <CR><LF> before the data is output. The end of the file stream is terminated with OK<CR><LF>. A cyclic redundancy check or a checksum will then be added to verify data integrity during download. The complete file can also be downloaded in this way by specifying SA=0 and a large value for LEN. The actual file size is then returned before the data follows.

Example:

```
DOWNLOAD,FN="TestFile.ad2cp",SA=0,LEN=4096,CRC=1,CKS=0
4096
<binary or ASCII data>
23432
OK
```

4.29 Read configuration

Command: READCFG
Command type: CONFIGURATION
Mode: COMMAND

Read current configuration. If a filename parameter is given the current configuration is stored to a file. If no parameter is given the current configuration is written to console.

The output of this command can be used to configure the instrument to a known configuration.

Argument	Description
FN	Write the output to file. Values: ['a'; 'z']; ['A'; 'Z']; ['0'; '9']; '_'; '.' | Max Length: 30

4.30 Inquire state

Command: INQ
Command type: INFO
Mode: COMMAND, CONFIRMATION, RETRIEVAL, MEASUREMENT

The INQ command queries the instrument state. Note that when operating over RS232 or RS422 serial lines, it should be preceded with @@@@@@ <delay 400ms> and a flush of the input buffer in case the instrument is in power down or in a low power mode taking measurements.

Consult section "Modes" for a description of the Instrument modes.

Note: Parameters used to represent instrument mode:

0000: Bootloader/Firmware upgrade

0001: Measurement

0002: Command

0004: Data Retrieval

0005: Confirmation

0006: FTP-mode

Example:
(In command mode)
```
INQ
0002
```
(In measurement mode)
```
INQ
0001
```
(In confirmation mode)
```
INQ
0005
```
(In data retrieval mode)
```
INQ
0004
```
(In firmware upgrade mode)
```
INQ
0000
```

4.31 Get instrument state

Command: GETSTATE
Command type: INFO
Mode: COMMAND, CONFIRMATION, MEASUREMENT, RETRIEVAL

Returns information about the current operational state of the instrument

Argument	Description
MODE	Current instrument state. 1: Measurement (START command received). 2: In command mode. 3: DEPLOY command received and deployment running. 4: Data retrieval. 5: Confirmation. 6: Network FTP. 8: DEPLOY command received, but deployment has not, yet started. 9: Confirmation in measurement mode. 10: Confirmation in deploy mode. 11: Confirmation in pre-deploy mode. 12: internal processing in progress.
DEPTIME	Seconds since deployment: 0 – DEPLOY command has not been received. < 0 – Number of seconds until deployment starts. > 0 – Number of seconds that deployment has been running. Unit: [s]
MEASTIME	Seconds with measurements: 0 – START command has not been received. > 0 – Number of seconds that measurement has been running. Unit: [s]
CURRTIME	The current instrument time. Time format is "YYYY-MM-DD HH:MM:SS" Unit: [Time]
WAKEUP	Reason for instrument wakeup. 0: Last startup/reboot caused by loss/low voltage. 1: Last startup/reboot caused by power on. 2: Last startup/reboot caused by BREAK command. 3: Last startup/reboot caused by Wakeup by Real time clock. 4: Last startup/reboot caused by WatchDog
INTPROC	Internal processing Active

Example:
```
GETSTATE,WAKEUP,CURRTIME
```

4.32 Get error

Command: GETERROR
Command type: INFO
Mode: COMMAND, CONFIRMATION, MEASUREMENT, RETRIEVAL

GETERROR retrieves a full description of the last error condition to occur. The error number is returned first followed by a string with the text description of the last error condition. A second string is also returned which contains information on the valid range of the failing argument.

Argument	Description
NUM	Integer error value
STR	Text description
LIM	Valid limits as text

Example:

```
SETAVG,CS=2.5
OK
SAVE,CONFIG
ERROR
GETERROR
40,"Invalid setting: Avg Cell Size","GETAVGLIM,CS=([0.20;2.00])"
OK
```

4.33 Get all

Command: GETALL
Command type: INFO
Mode: COMMAND

GETALL retrieves all relevant configuration information for the instrument. This information can either be displayed on the command line or saved to a data file on the instrument. For the Nortek post-processing software to read a valid Nortek file it must contain both the Header and Data Record. The Header information can be obtained by using the command GETALL.

Argument	Description
FN	Write the output to this file saved on instrument

Example:

```
See GETALL chapter.
```

4.34 Get recorder state

Command: RECSTAT
Command type: INFO
Mode: COMMAND, RETRIEVAL

Returns recorder state.

Argument	Description
SS	Number of bytes in a sector. Unit: [bytes]
CS	Number of bytes in one cluster. Unit: [bytes]
FC	Number of bytes in free clusters. Unit: [bytes]
TC	Total number of bytes in clusters. Unit: [bytes]
VS	Volume Size in bytes. Unit: [bytes]
FF	Number of free files.
TF	Total number of files.
PNM	SD card type and firmware

Example:
```
RECSTAT,VS
```

4.35 Get configuration limits

Command: GETMISCLIM
Command type: INFO
Mode: COMMAND

Returns configuration limits.

Argument	Description
AVGPR	Returns the total profiling range for averaged measurements (SETAVG) Unit: [m]
BURSTPR	Returns the total profiling range for burst profile measurements (SETBURST) Unit: [m]
BURSTHRPR	Returns the total profiling range for burst HR profile measurements (SETBURSTHR) Unit: [m]

Note: The output format for limits is described in Data Limit Formats

Example:

GETMISCLIM,BURSTPR

4.36 Get transfer matrices

Commands: GETXFAVG, GETXFBURST,
Command type: INFO
Mode: COMMAND

Returns the "Beam to XYZ" transfer matrix for the current setup. If the number of beams is 1 or 2, only ROWS and COLS are returned.

Argument	Description
ROWS	Number of rows
COLS	Number of rows
M11	
M12	
M13	
M14	
M21	
M22	
M23	
M24	
M31	
M32	
M33	
M34	
M41	
M42	
M43	
M44	

Note: Return is ROWS, COLS, M11, M12 ... M43, M44.

Cell indexes are: first number for row and second index for column.

GETXFAVG

Returns the "Beam to XYZ" transfer matrix for average measurements

Example:
GETXFAVG

GETXFBURST

Returns the "Beam to XYZ" transfer matrix for burst measurements

Example:

```
GETXFBURST
GETXFBURST,ROWS=4,COLS=4,M11=1.183,M12=0.000,M13=-
1.183,M14=0.000,M21=0.000,M22=1.183,M23=0.000,M24=-
1.183,M31=0.552,M32=0.000,M33=0.552,M34=0.000,M41=0.000,M42=0.552,M
43=0.000,M44=0.552
GETXFAVG,ROWS=3,COLS=3,M11=1.183,M12=0.000,M13=-
1.183,M21=1.183,M22=-2.366,M23=1.183,M31=0.552,M32=0.000,M33=0.552
```

4.37 Average mode telemetry settings

Commands: SETTMAVG, GETTMAVG, GETTMAVGLIM,
Command type: CONFIGURATION
Mode: COMMAND

The instrument averaging mode telemetry settings and relevant limits.

Argument	Description
EN	Enable averaging mode telemetry.
CD	Cells divisor. Define the interval at which cells should be outputted from your profile. If CD=1 every cell will be included, if CD=2 every second cell will be outputted etc.
PD	Profile divisor. Define how often the collected profile should be outputted. If PD=1 the profile from each ping will be outputted, if PD=2 every second profile will be outputted etc. Note that PD>1 is not available when averaging is enabled.
AVG	Number of seconds included in the telemetry average. If 0, no averaging is done and every ping will be outputted Unit: [s]
TV	Enable(1)/disable(0) velocity output.
TA	Enable(1)/disable(0) amplitude output
TC	Enable(1)/disable(0) correlation output.
CY	Coordinate system. BEAM, XYZ, ENU
FO	Enable(1)/disable(0) file output. The telemetry data will be written to the file named telemetryfile.bin on the recorder

SO	Enable(1)/disable(0) serial output. The telemetry data will be outputted live, either over serial or ethernet connection
DF	Telemetry data format. For examples, see Telemetry Data Format chapter.
DISTILT	Disable tilt.
TPG	Enable(1)/disable(0) output of the Percentage Good value.
MAPBINS	Enable vertical bin mapping.
CORRTH	Correlation threshold

Note: It is recommended to enable TPG when averaging (AVG>0) is also enabled.

The actual valid range for the various parameters for your specific instrument and firmware can be found by using the GETTMAVGLIM command. This command has the same arguments as the SETTMAVG/GETTMAVG commands shown in the list above. The output format for limits is described in Data Limit Formats.

SETTMAVG

Set averaging mode telemetry settings

Example:
```
SETTMAVG,EN=1,AVG=30
```

GETTMAVG

Get averaging mode telemetry settings

GETTMAVGLIM

Get instrument averaging mode limits

4.38 Download telemetry

Command: DOWNLOADTM
Command type: ACTION
Mode: COMMAND, RETRIEVAL

This command enables reading the telemetry file which can be created during measurement by using the appropriate SETTMxxx commands. The formats are described in the section detailing Data Formats.

Argument	Description
SA	Start address (offset) of the first byte to be returned.

LEN	Number of bytes to be downloaded.
CRC	Use Cyclic redundancy check (CRC-16). CRC=1 enables CRC. CRC cannot be enabled when CKS=1.
CKS	Use Checksum. CKS=1 enables checksum. CKS cannot be enabled when CRC=1.

Note: If no parameters are sent with the DOWNLOADTM command the complete file is directly returned, without the number of bytes to follow. The end of the file can then be detected by parsing the OK<CR><LF>.

The parameters can be used to download the telemetry file in several pieces. The number of bytes to follow will then be returned in ASCII format and terminated with <CR><LF> before the data is output. The end of telemetry stream is terminated with OK<CR><LF>. A cyclic redundancy check or a checksum will then be added to be able to verify data integrity during download. The complete file can also be downloaded in this way by specifying SA=0 and a large value for LEN. The actual file size is then returned before the data follows. See also TMSTAT for retrieving file information.

Example:

```
DOWNLOADTM,SA=0,LEN=4096,CRC=1,CKS=0
4096 (number of bytes that follow)
<binary or ASCII data>
23432 (checksum/crc value)
OK
```

4.39 Save configuration to file

Command: STOREHEADERTM
Command type: ACTION
Mode: COMMAND

This command stores the GETALL (complete configuration) to the telemetry file.

Example:

```
STOREHEADERTM
```

4.40 Erase telemetry file

Command: ERASETM
Command type: ACTION
Mode: COMMAND, RETRIEVAL

Erase the telemetry file.

Argument	Description
CODE	Code should be 9999

	9999

Note: The telemetry file can also be erased over FTP.

Example:
```
ERASETM,CODE=9999
```

4.41 Write tag output

Command: TAG
Command type: ACTION
Mode: COMMAND, CONFIRMATION, MEASUREMENT, RETRIEVAL

Write a Tag to output file and data output.

The TAG command adds a tag to the both the output file and the output data, if enabled. The output is a String Record as defined in the FWRITE command. The ID of the String Record Data packet is 19dec.

Argument	Description
STR	Tag string. Maximum 200 bytes.
CLK	Add clock in tag.

Note: Remember to use the BBPWAKEUP command when sending commands to an instrument in Measurement mode, when using Ethernet.

Example:
```
TAG,STR="This is a test tag.",CLK=1
a5 0a a0 10 2f 00 42 8c 42 5d 13 32 // Binary header
30 31 37 2d 30 31 2d 32 34 20 30 38 // String Record ID = 19dec
3a 34 32 3a 35 37 2e 34 34 39 20 2d // "2017-01-24 08:42"
20 54 68 69 73 20 69 73 20 61 20 74 // ":57.449 - This i"
65 73 74 20 74 61 67 2e 00 // "s is a test tag."
OK
```

4.42 Precision time protocol

Commands: PTPSET, PTPGET,
Command type: CONFIGURATION
Mode: COMMAND

This command is only relevant for the 400 kHz instrument

Configure the precision time protocol parameters.

Argument	Description

EN	Choose a time protocol. 0: Time protocol disabled. 1: Use precision time protocol (PTP). 2: Use network time protocol (NTP).
OFFSET	Offset Unit: [µs]
CL	Time value representing the number of seconds elapsed since 00:00 hours, Jan 1, 1970 UTC Unit: [s]
IP	The numeric IP address of the NTP server to use for syncing

Note: For the case of NTP, the instrument is not capable of DNS name resolution. In order to retrieve the IP address associated with a name, use the "nslookup" tool from the computer command line

PTPSET

Set precision time protocol parameters

Example:
```
PTPSET,EN=0,OFFSET=0,CL=1609193402
OK
PTPSET,EN=2,IP="129.240.3.3"
OK
```

PTPGET

Get precision time protocol parameters

Example:
```
PTPGET,EN,OFFSET,CL
```

4.43 Wake Doppler processor

Command: BBPWAKEUP
Command type: ACTION
Mode: COMMAND, CONFIRMATION, MEASUREMENT, RETRIEVAL

This command is only relevant for the 400 kHz instrument

Commands available in measurement mode should be preceded by the command BBPWAKEUP. This ensures that the BBP is ready to process the command when it is received (see Checking instrument state over Ethernet). In measurement mode, another BBPWAKEUP must be sent when more than 2 seconds has elapsed since the previous command.

If uncertain of the active mode it is good practice to send BBPWAKEUP before sending GETSTATE or INQ.

Example:
`BBPWAKEUP`

4.44 Get hardware specifications

Command: GETHW
Command type: INFO
Mode: COMMAND, CONFIRMATION, MEASUREMENT, RETRIEVAL

Returns Firmware versions and Board revisions.

Argument	Description
FW	Running DSP FW version
FWMINOR	Running DSP FW version (minor part)
FPGA	Running FPGA FW version
BOOT	DSP bootloader FW Version
DIGITAL	Board revision. Example: C-0
INTERFACE	Board revision. Example: C-0
ANALOG	Board revision. Example: C-0
SENSOR	Board revision. Example: C-0

Example:
`GETHW,FW,MINOR`

4.45 Add license

Command: ADDLICENSE
Command type: PRODUCTION
Access: User
Mode: COMMAND

Reads a license key and checks it against the instrument serial number before adding it to the license key flash.

Argument	Description
KEY	The license key to add

Example:
```
ADDLICENSE,KEY="9H3F5PE47HUUB"
OK
```

4.46 Delete license

Command: DELETELICENSE
Command type: PRODUCTION
Access: User
Mode: COMMAND

Deletes a license key from the license key flash.

Argument	Description
KEY	The license key to delete

Example:
```
DELETELICENSE,KEY="9H3F5PE47HUUB"
OK
```

4.47 Lists license keys

Command: LISTLICENSE
Command type: INFO
Access: User
Mode: COMMAND

Lists all license keys contained in the license key flash along with a description of the functionality enabled by the key and the variant number of the license.

Argument	Description
KEY	License key
DESC	Information about license type
TYPE	License id

Example:
```
LISTLICENSE
LISTLICENSE,"4X218TRTRPNUB","High Resolution",4
LISTLICENSE,"JKHHFNH3RPNUB","Wave Mode",6
LISTLICENSE,"WF3CJR6PRPNUB","Current Profiler",1
OK
$PNOR,LISTLICENSE*76
$PNOR,LISTLICENSE,KEY="4X218TRTRPNUB",DESC="High
Resolution",TYPE=4*73
$PNOR,LISTLICENSE,KEY="JKHHFNH3RPNUB",DESC="Wave Mode",TYPE=6*00
```

```
$PNOR,LISTLICENSE,KEY="WF3CJR6PRPNUB",DESC="Current
Profiler",TYPE=1*1C
$PNOR,OK*2B
```

5 GETALL

The GETALL command retrieves all relevant configuration information for the instrument. This information can either be displayed on the command line or saved to a data file. For Nortek post-processing software to read a valid .ad2cp or .aqd file it must contain both the Header and Data Record. The Header information can be obtained by using the command GETALL.

Argument	Description
FN	Write the output to this file

Example: The result from a GETALL command on a 2D Horizontal Profiler 400 kHz with a default Average plan. Calibration values are not modifiable.

Parameter	Example	Description
GETCLOCKSTR	TIME="2021-06-08 16:12:44"	Current time on the instrument real-time clock
ID	STR="2D 400 S4SH",SN=450000	Instrument name and serial number
GETHW	FW=9001,FPGA=353,DIGITAL="I-6",INTERFACE="J-1",ANALOG="A-1(Low-Power EDR-LIMP)",SENSOR="J-1",BOOT=25,FWMINOR=8	Firmware and electronics revision numbers
BOARDSENSGET	AV=23,NB=2,HF=400,TTR=1.0000,TTRB5=1.0000,TTRB5AUX=0.0000,AUXRS=0	Production information
GETPWR	PLAN=50.64,BURST=0.00,AVG=47.93,PLAN1=0.00,BURST1=0.00,AVG1=0.00,TOTAL=50.64	Power consumption in mW for current plan
GETMEM	PLAN=0.003,BURST=0.000,AVG=0.003,PLAN1=0.000,BURST1=0.000,AVG1=0.000,TOTAL=0.003	Memory consumption in MB for current plan
GETPRECISION	AVGHORZ=-9.99,BURSTHORZ=-9.99,BEAM5=-9.99,AVGBEAM=0.53,BURSTBEAM=-9.99	Horizontal and along-beam precision values
GETPLAN	MIAVG=600,AVG=1,DIAVG=0,VD=0,MV=10,SA=35.0,BURST=0,MIBURST=3600,DIBURST=0,SV=0.00,FN="Data.ad2cp",SO=0,FREQ=400,NSTT=0	Current PLAN settings
GETAVG	NC=39,CS=2.50,BD=1.00,CY="XYZ",PL=0.0,AI=60,VP=0.000,VR=2.50,DF=7,NPING=60,NB=2,CH=0,MUX=0,BW="NARROW",ALTI=0,BT=0,ICE=0,ALTISTART=0.50,ALTIEND=100.00,RAWALTI=0	Current AVG settings
GETXFAVG	ROWS=2,COLS=2,M11=0.5518,M12=0.5518,M21=1.1831,M22=-1.1831	Transformation matrix values
GETUSER	POFF=9.50,DECL=0.00,HX=0,HY=0,HZ=0	Pressure and compass offset values

GETINST	BR=115200,RS=422,LED="ON24H",ORIENT="AUTOZUPDOWN",CMTOUT=300,DMTOUT=60,CFMTOUT=60	Instrument settings
GETCOMPASSCAL	DX=0,DY=0,DZ=0,M11=32767,M12=0,M13=0,M21=0,M22=32767,M23=0,M31=0,M32=0,M33=32767	Current compass calibration values
RECSTAT	SS=512,CS=32768,FC=15250161664,TC=15518924800,VS=15518924800,FF=4095,TF=4095	Recorder status
BEAMCFGLIST	BEAM=1,THETA=25.00,PHI=-90.00,FREQ=400,BW=6,BRD=1,HWBEAM=1,ZNOM=9.00,DIA=164.0	Instrument beam configuration
	BEAM=2,THETA=25.00,PHI=90.00,FREQ=400,BW=6,BRD=1,HWBEAM=2,ZNOM=9.00,DIA=164.0	
BEAMIMPLIST	BEAM=1,P0=1.00000e+02,P1=0.00000e+00,P2=0.00000e+00,P3=0.00000e+00,P4=0.00000e+00,T1=0.00000e+00	Instrument beam configuration
	BEAM=2,P0=1.00000e+02,P1=0.00000e+00,P2=0.00000e+00,P3=0.00000e+00,P4=0.00000e+00,T1=0.00000e+00	
LISTLICENSE	KEY="B4BHK6TF6BJ",DESC="16GB Recorder",TYPE=11	Installed instrument licenses
	KEY="7F9YW46Y6BJ",DESC="Averaging Mode",TYPE=1	
CALCOMPGET	DX=0,DY=0,DZ=0,M11=32767,M12=0,M13=0,M21=0,M22=32767,M23=0,M31=0,M32=0,M33=32767	Compass calibration values
CALTEMPGET	SC=0.99903	Temperature calibration value
CALTILTGET	PO=0.00,RO=0.00,MAGG=1,HO=0.00	Tilt sensor calibration values
CALACCLGET	AX=1.000000E+00,B0X=0.000000E+00,B1X=4.477940E-05,B2X=-1.101457E-06,B3X=3.221334E-08,A1X=5.413303E-05,A2X=8.092983E-07,A3X=-2.234600E-08	Accelerometer calibration values
	AY=1.000000E+00,B0Y=0.000000E+00,B1Y=-4.488621E-05,B2Y=-1.359965E-06,B3Y=2.713223E-08,A1Y=4.603504E-05,A2Y=5.183597E-07,A3Y=-1.793741E-08	
	AZ=1.000000E+00,B0Z=0.000000E+00,B1Z=7.714431E-05,B2Z=-1.434962E-06,B3Z=3.781547E-09,A1Z=-2.196427E-05,A2Z=1.452303E-06,A3Z=-9.148971E-09	
CALPRESSGET	MT=1,RREF=4.5275990000e+02,RB0=0.0000000000e+00,RB1=0.0000000000e+00,RB2=0.0000000000e+00,RB3=0.0000000000e+00,T0=-	Pressure sensor calibration values

	1.6127200000e+04,T1=1.4193100000e+01,T2=-4.1883218750e-03,T3=4.1506809380e-07,ID="I007408"	
CALPRESSCOEFFGET	A0=4.0661392500e+01,A1=-3.7689287500e-02,A2=1.1086410000e-05,A3=-1.0929538750e-09,B0=-2.0082089060e-01,B1=3.5114962500e-04,B2=-8.8997081250e-08,B3=7.0770606250e-12	
	C0=1.1984730000e-04,C1=-1.0565940000e-07,C2=3.1952409380e-11,C3=-3.2971271880e-15,D0=-4.4563965630e-06,D1=3.8111300000e-09,D2=-1.1022880000e-12,D3=1.0729378750e-16	
CALROTACCLGET	M11=1.00000,M12=0.00000,M13=0.00000,M21=0.00000,M22=1.00000,M23=0.00000,M31=0.00000,M32=0.00000,M33=1.00000	Accelerometer rotation matrix values
CALECHOGET	CHA0=0.00,CHB0=-15.31,CHC0=0.00	

6 Binary data formats

This chapter describes the binary data formats used by the Generation 2 2D Horizontal Profiler, including both the 1 MHz and the 400 kHz instrument. The two different instruments generate different file types (.ad2cp for the 400 kHz and .aqd for the 1 MHz), but they both follow the same structure. The binary data is stored in two files on the recorder:

[filename]_avgd.aqd (for 1 MHz) and **[filename]_avgd.ad2cp** (for 400 kHz) - contains the averaged current data. The data is averaged over the configured average interval and saved as Data Format 7. Note that internal quality control has been applied to this data. For more information about the data format and the internal quality control, please refer to the Quality Control chapter.

[filename].aqd (for 1 MHz) and **[filename].ad2cp** (for 400 kHz) – contains raw burst data saved as Data Format 3. These files are usually empty, since burst applications are typically not enabled.

The data formats are described in the following chapters. Note that the binary data formats all use a common header that specifies how the rest of the data block should be interpreted, containing information about data type, size etc. The header is documented separately as _HeaderData. A data block is the data from and including one header to the next, and can for example contain data from one velocity profile. We use the leading underscore in the chapter title to emphasize that this is not a complete data set, but is a part used by two or more other data formats. Note that for Data Format 3 we also have two other common data fields shared by several data formats (see illustration in the table below).

Binary data is always output as Little Endian.

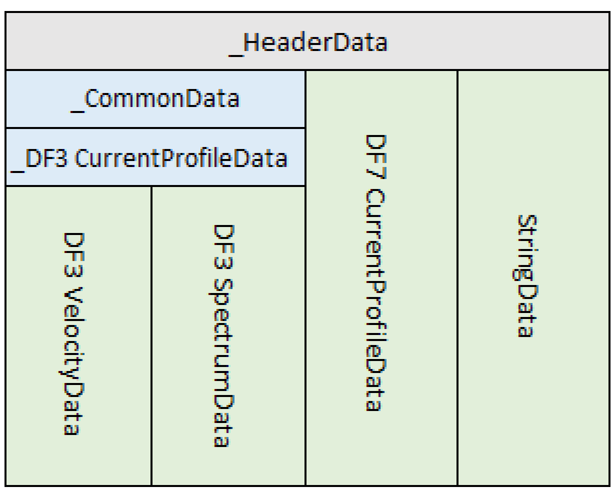

Figure: Showing how common data fields (gray for header and blue for other common's) relate to the sensor specific data fields (green). Note that we use a leading underscore (_) to emphasize that this is not a sensor data format but is common and used by two or more data formats.

About the tables

The tables have the columns **Field**, **Position/Size** and **Description**. While Field and Description provide the name and additional details about a specific field, **Position** and **Size** are described in more detail below:

Position indicates the location of a field within the data block. This refers to the byte position, counting from zero at the first byte. Note that there are two separate counts for the header and the data following the header, so the count starts from zero again after the header. For instance, the Data Series ID is at position 2 in the header of the data block.

Some positions are not fixed but depend on the fields in front of them. In these cases, Offset of Data (position 2 of the data following the header, see DF7 CurrentProfileData) can be used to determine the position of the following fields. When this is relevant, the position in the table is not shown as a number but as a variable name, such as OFFSET. Variable descriptions are listed below the tables where they are used.

Size refers to the data type of the field. For example, the Data Series ID in the header is an unsigned integer of 8 bits (uint8). Note that not all fields have a specific data type; some are objects requiring a certain number of bits. For instance, the status bit mask occupies 32 bits. For fields where the size is given in bits, a more detailed description of the field and each of the bits is listed separately below the table where they are used.

6.1 _HeaderData

The header definition for binary data formats. Note that the header may be verified without reading the rest of the data block since it has its own checksum.

Field	Position Size	Description
Sync byte	0 uint8	Always 0xA5.
Header size	1 uint8	Number of bytes in the headers. Normally it is 10 bytes, but in a few cases it may be 12 bytes to hold data size of 32 bytes.
Data series id	2 uint8	Defines the type of the following Data Record. 0x15 - Burst data as DF3. 0x16 - Average data as DF3. 0x17 - Bottom Track Data Record. 0x18 - Interleaved Burst Data Record (beam 5). 0x1E - Altimeter Record. 0x1F - Avg Altimeter Raw Record. 0x1A - Burst Altimeter Raw Record. 0x1B - DVL Bottom Track Record. 0x1C - Echo Sounder Record. 0x23 - Echo Sounder Raw Record.

		0x24 - Echo Sounder Raw Tx Record. 0x26 - Average data as DF7. 0x30 - Processed Wave Data Record. 0x1D - DVL Water Track Record. 0xC8 - Vector 2 data as DF8. 0xA0 - String Data Record, eg. GPS NMEA data, comment from the FWRITE command. 0x20 - Spectrum data as DF3.
Family id	3 uint8	Defines the Instrument Family. 0x10 is the Signature Family. 0x16 is the DVL family. 0x30 is the Aquadopp Generation 2 family. 0x40 is the Awac Generation 2 family.
Data size	4 unit16/uint32	Number of bytes in the following Data Record. If header size is 10, the data size is represented with a uint16. For large datasets, header may have 12 bytes giving room for a uint32 to represent data size.
Data checksum	6/8 uint16	Checksum of the following Data Record.
Header checksum	8/10 uint16	Checksum of all fields of the Header except the Header Checksum itself.

6.2 _CommonData

Used By: _DF3 CurrentProfileData

Common data definitions for parsing Nortek data formats.

Field	Position Size	Description
Version	0 uint8	Version number of the Data Record Definition.
Offset of data	1 uint8	Number of bytes from start of the record to start of the actual data. Unit: [# bytes]
Serial number	4 uint32	Instrument serial number from factory.
Year	8 uint8	Number of years since 1900.
Month	9 uint8	Month number counting from 0 which is January.
Day	10 uint8	Day of the month
Hour	11	24 hour of the day

	uint8	
Minutes	12 uint8	Minutes.
Seconds	13 uint8	Seconds.
Hundred micro seconds	14 uint16	Hundred micro seconds from last whole second. Unit: [100 µs]
Speed of sound	16 uint16	Speed of sound used by the instrument. Raw data given as 0.1 m/s Unit: [m/s]
Temperature	18 int16	Reading from the temperature sensor. Raw data given as 0.01 °C Unit: [°C]
Pressure	20 uint32	Raw data given as 0.001 dBar Unit: [dBar]
Heading	24 uint16	Raw data given as 0.01 degrees Unit: [deg]
Pitch	26 int16	Raw data given as 0.01 degrees Unit: [deg]
Roll	28 int16	Raw data given as 0.01 degrees Unit: [deg]
Cell size	32 uint16	Size of each cell (resolution) on the beam. Raw data given as mm Unit: [m]
Nominal correlation	36 uint8	The nominal correlation for the configured combination of cell size and velocity range Unit: [%]
Battery voltage	38 uint16	Raw value given in 0.1 Volt Unit: [V]
Magnetometer.X	40 int16	X axis flux raw value in last measurement interval
Magnetometer.Y	42 int16	Y axis flux raw value in last measurement interval
Magnetometer.Z	44 int16	Z axis flux raw value in last measurement interval
Accelerometer.X	46 int16	Raw accelerometer X axis value in last measurement interval Raw value divided by 16384 will give vector [x,y,z] of length 1 Note: The unit of the instrument is gravity [g]. Conversion of Accelerometer unit less raw

		measurements to m/s^2: divide measurement by 16384, then multiply by calibrated gravity in Oslo, 9.819 m/s^2.
Accelerometer.Y	48 int16	Raw Y axis value in last measurement interval Raw value divided by 16384 will give vector [x,y,z] of length 1
Accelerometer.Z	50 int16	Raw Z axis value in last measurement interval. Raw value divided by 16384 will give vector [x,y,z] of length 1
Data set description	54/56 uint16	Data set description. 0-3 Physical beam used for 1st data set. 4-7 Physical beam used for 2nd data set. 8-11 Physical beam used for 3th data set. 12-16 Physical beam used for 4th data set.
Transmitted energy	56/58 uint16	Transmitted energy.
Velocity scaling	58/60 int8	Velocity scaling used to scale velocity data.
Power level	59/61 int8	Configured power level Unit: [dB]
Magnetometer temperature	60/62 int16	Magnetometer temperature reading. Uncalibrated Raw data in 1/1000 °C Unit: [°C]
Real time clock temperature	62/64 int16	Real Time Clock temperature reading Unit: [°C]
Error status	64/66 2 * 8 bits	Error bit mask Object reference given in table below
Ensemble counter	72/74 uint32	Counts the number of ensembles in both averaged and burst data

Position and size variables:

Name	Description
54/56	The status field is at 54 or 56 depending on wether the ambiguity velocity (in DF3 CurrentProfileData) is 16bit or 32bit long.
56/58	The status field is at 56 or 58 depending on wether the ambiguity velocity (in DF3 CurrentProfileData) is 16bit or 32bit long.
58/60	The status field is at 56 or 58 depending on wether the ambiguity velocity (in DF3 CurrentProfileData) is 16bit or 32bit long.
59/61	The status field is at 59 or 61 depending on wether the ambiguity velocity (in DF3 CurrentProfileData) is 16bit or 32bit long.

60/62	The status field is at 60 or 62 depending on wether the ambiguity velocity (in DF3 CurrentProfileData) is 16bit or 32bit long.
62/64	The status field is at 62 or 64 depending on wether the ambiguity velocity (in DF3 CurrentProfileData) is 16bit or 32bit long.
64/66	The status field is at 64 or 66 depending on wether the ambiguity velocity (in DF3 CurrentProfileData) is 16bit or 32bit long.
72/74	The status field is at 72 or 74 depending on wether the ambiguity velocity (in DF3 CurrentProfileData) is 16bit or 32bit long.

Object reference: Error status

Error bit mask

Field	Position Size	Description
Data retrieval FIFO error	0 bit	Data retrieval FIFO error
Data retrieval overflow	1 bit	Data retrieval overflow
Data retrieval underrun	2 bit	Data retrieval Underrun
Data retrieval samples missing	3 bit	Data retrieval samples missing
Measurement not finished	4 bit	The Measurement and data storage/transmit didn't finish before next measurement started.
Sensor read failure	5 bit	Sensor read failure
Tag error beam 1 (In-phase)	8 bit	Tag error beam 1 (In-phase)
Tag error beam 1 (Quadrature-phase)	9 bit	Tag error beam 1 (Quadrature-phase)
Tag error beam 2 (In-phase)	10 bit	Tag error beam 2 (In-phase)
Tag error beam 2 (Quadrature-phase)	11 bit	Tag error beam 2 (Quadrature-phase)
Tag error beam 3 (In-phase)	12 bit	Tag error beam 3 (In-phase)

Tag error beam 3 (Quadrature-phase)	13 bit	Tag error beam 3 (Quadrature-phase)
Tag error beam 4 (In-Phase)	14 bit	Tag error beam 4 (In-phase)
Tag error beam 4 (Quadrature-phase)	15 bit	Tag Error Beam 4 (Quadrature-phase)

6.3 _DF3 CurrentProfileData

Extends: _CommonData
Used By: DF3 VelocityData, DF3 SpectrumData

Common data definitions for Nortek data format 3 (DF3).

Field	Position Size	Description
Configuration bit mask	2 2 * 8 bits	Record Configuration Bit Mask Object reference given in table below
Blanking	34 uint16	Distance from instrument to first data point on the beam. Raw data given as cm or mm depending on status.blankingDistanceScalingInCm Unit: [m]
Temperature PressureSensor	37 uint8	Temperature of pressure sensor: T=(Val/5)-4.0 Raw value given as 0.2 °C Unit: [°C]
Ambiguity Velocity	52 uint16	Ambiguity velocity, corrected for sound velocity, scaled according to Velocity scaling. 10^(Velocity scaling) m/s Unit: [m/s]
Extended status	66 2 * 8 bits	Extended status bit mask Object reference given in table below
Status	68 4 * 8 bits	Status bit mask. Note that bits 0, 2, 3, 4 are unused. Object reference given in table below

Object reference: Configuration bit mask

Record Configuration Bit Mask

Field	Position Size	Description

Has pressure sensor	0 bit	Pressure sensor value valid
Has temperature sensor	1 bit	Temperature sensor value valid
Has compass sensor	2 bit	Compass sensor value valid
Has tilt sensor	3 bit	Tilt sensor value valid
Has external sensor	4 bit	External sensor value valid
Has velocity data	5 bit	Velocity data included
Has amplitude data	6 bit	Amplitude data included
Has correlation data	7 bit	Correlation data included
Has altimeter data	8 bit	Altimeter data included
Has altimeter raw data	9 bit	Altimeter raw data included
Has AST data	10 bit	AST data included
Has echosounder data	11 bit	Echosounder data included
Has AHRS data	12 bit	AHRS data included
Has percentage good data	13 bit	Percentage data included
Has standard deviation data	14 bit	Standard deviation data included
Has spectrum data	15 bit	Amplitude spectrum data included.

Object reference: Extended status

Extended status bit mask

Field	Position Size	Description

Processor idles < 3%	0 bit	Indicates that the processor Idles less than 3 percent
Processor idles < 6%	1 bit	Indicates that the processor idles less than 6 percent
Processor idles < 12%	2 bit	Indicates that the processor idles less than 12 percent
External sound velocity probe	3 bit	Sound velocity probe velocity received
External heading, pitch, roll, and position	4 bit	External heading, pitch, roll and position received from NMEA NTKNAV
External heading	5 bit	External heading received from NMEA input
External pitch and roll	6 bit	External pitch and roll received from NMEA input
Ethernet powered system	7 bit	Indicates if the system is powered via Ethernet
File system flush	13 bit	File system flush in progress
Internal processing	14 bit	Internal processing in progress (e.g. wave processing)
Extended status should be interpreted	15 bit	If this bit is set the rest of the word/ extended status should be interpreted

Object reference: Status

Status bit mask. Note that bits 0, 2, 3, 4 are unused.

Field	**Position Size**	**Description**
Blanking distance scaling in cm	1 bit	Bit 1: Scaling of blanking distance 0: mm scaling 1: given in cm
Echosounder frequency bin	5-9 5 bit	Used only on the Signature100 which supports up to 5 packages pr frequency index. Processing is then done on the different frequency bands.
Boost running	10 bit	Boost running
Telemetry data	11 bit	Telemetry data

Echosounder index	12-15 4 bits	Echosounder frequency index. Valid numbers are 0, 1 and 2 (or 0000, 0001 and 0010) referring to frequencies 1, 2 or 3 as used in SET-/GETECHO.
Active configuration	16 bit	Bit 16: Active configuration 0: Settings for PLAN,BURST,AVG 1: Settings for PLAN1,BURST1,AVG1
Previous measurement skipped due to low voltage	17 bit	Bit 17: Last measurement low voltage skip 0: normal operation 1: last measurement skipped due to low input voltage
Previous wakeup state	18-21 4 bits	0 - Bad power 1 - Power on 2 - Break 3 - RTC 4 - Watchdog 5 - Low voltage 6 - Filesystem error
Auto orientation	22-24 3 bits	0: "Fixed" Fixed orientation 1: "Auto" Auto Up Down 3: "AHRS3D" AHRS3D
Orientation	25-27 3 bits	0: "XUP" Instrument x-axis defined up, heading reference axis is Z positive 1: "XDOWN" Instrument x-axis defined down, heading reference axis is Z positive 2: "YUP" Instrument y-axis defined up, heading reference axis is Z positive 3: "YDOWN" Instrument y-axis defined down, heading reference axis is Z positive 4: "ZUP" Instrument z-axis defined up, heading reference axis is X positive 5: "ZDOWN" Instrument z-axis defined down, heading reference axis is X positive 7: "AHRS" AHRS reports orientation any way it points. Example: Z down -> Roll = 180 deg.
Wake up state	28-31 4 bits	0 - Bad power 1 - Power on 2 - Break 3 - RTC 4 - Watchdog 5 - Low voltage 6 - Filesystem error

6.4 DF7 CurrentProfileData

ID: 0x26

Data definitions for parsing Nortek data format DF7.

Field	Position Size	Description
Version	0 uint8	Revision number.
Instrument type	1 uint8	Type of instrument. 0x30 - Gen. 2 Aquadopp and 1 MHz 2D Horizontal Profiler 0x40 - Gen. 2 AWAC and 400 kHz 2D Horizontal Profiler
Offset of data	2 uint16	Number of bytes from start of record to start of non-common data fields. Unit: [# bytes]
Configuration of instrument	4 4 * 8 bits	Configuration bit mask. Object reference given in table below
Types of data included	8 4 * 8 bits	Types of data included. Object reference given in table below
Valid data	12 8 * 8 bits	Bit is true if the chosen data is valid Object reference given in table below
Serial number	20 uint32	Instrument serial number from factory.
Year	24 uint8	Number of years since 1900.
Month	25 uint8	Month number counting from 0 which is January.
Day	26 uint8	Day of the month
Hour	27 uint8	24 hour of the day
Minutes	28 uint8	Minutes.
Seconds	29 uint8	Seconds.
Hundred micro seconds	30 uint16	Hundred micro seconds from last whole second. Unit: [100 µs]
Speed of sound	32 float	Configured or measured sound velocity. Unit: [m/s]

Temperature	36 float	Reading from the temperature sensor. Unit: [°C]
Pressure	40 float	Pressure measurement. Unit: [dBar]
Absolute pressure	44 float	Absolute pressure measurement.. Unit: [dBar]
Heading	48 float	Heading measurement Unit: [deg]
Pitch	52 float	Pitch measurement Unit: [deg]
Roll	56 float	Roll measurement Unit: [deg]
Standard deviation data.Pressure	60 float	Standard deviation on pressure data Raw data in 0.01 dBar Unit: [dBar]
Standard deviation data.Heading	64 float	Standard deviation on heading data Raw data in 0.01 degrees Unit: [deg]
Standard deviation data.Pitch	68 float	Standard deviation on pitch data Raw data in 0.01 degrees Unit: [deg]
Standard deviation data.Roll	72 float	Standard deviation on roll data Raw data in 0.01 degrees Unit: [deg]
Beams, coordinates and cells.Number of beams	76 uint8	Number of beams used.
Beams, coordinates and cells.Coordinate system	77 uint8	Coordinate system: b00 = ENU b01 = XYZ b10 = BEAM b11 = not used
Beams, coordinates and cells.Number of cells	78 uint16	The number of cells
Cell size	80 float	The size of each cell
Blanking	84 float	Blanking distance Unit: [m]

Battery voltage	88 float	Battery voltage
Pressure sensor temperature	92 float	Temperature measured by the pressure sensor Unit: [°C]
Magnetometer temperature	96 float	Temperature measured by the magnetometer Unit: [°C]
Real time clock temperature	100 float	Temperature measured by the real time clock Unit: [°C]
Magnetometer.X	104 float	raw X axis value in last measurement interval
Magnetometer.Y	108 float	raw Y axis value in last measurement interval
Magnetometer.Z	112 float	raw Z axis value in last measurement interval
Accelerometer.X	116 float	raw X axis value in last measurement interval
Accelerometer.Y	120 float	raw Y axis value in last measurement interval
Accelerometer.Z	124 float	raw Z axis value in last measurement interval
Ambiguity velocity	128 float	Ambiguity velocity Unit: [m/s]
Velocity scaling	132 float	Scale velocity data to m/s
Power level	136 float	Power level Unit: [dB]
Error status	140 4 * 8 bits	Error bit mask. Object reference given in table below
Status	144 4 * 8 bits	Status bit mask. Object reference given in table below
Average depth speed	148 float	Average depth speed Unit: [m/s]
Average depth direction	152 float	Average depth direction Unit: [deg]
Number average pings	156 uint16	The number of pings averaged over
Nominal correlation	158 uint16	Nominal correlation
Velocity data	OFFSET	This field exists if the Velocity data included bit of the Config byte is set.

	int16 * VEL_NB * VEL_NC	Data is scaled with Velocity Scaling float value in position 132. Unit: [m/s]
Amplitude data	AMP_POS uint8 * AMP_NB * AMP_NC	This field exists if the amplitude data included bit of the Config byte is set 0.5 dB/count Unit: [dB]
Correlation data	CORR_POS uint8 * CORR_NB * CORR_NC	This field exists if the Correlation data included bit of the Config byte is set [0 – 100 %] Unit: [%]
Percentage good data	PGD_START uint8 * PGD_LEN	Percent Good Estimate per cell These fields exist if the Percentage Good data included Unit: [%]
Quality data	QTY_START uint16 * QTY_LEN	Quality mask for each cell. If value is 0, all controls are passed. Otherwise, the masks show why the data is flagged. For more information see the Data Quality Control chapter. Bit 0 - Not used Bit 1 - Fish Bit 2 - Correlation Bit 3 - Pressure Bit 4 - Not used Bit 5 - Sidelobe Bit 6 - Percentage good Bit 7 - Not used Bit 8 - Bin mapping Bit 9 - Tilt
CTD data.Conductivity	CTD_START float	Conductivity measured by CTD sensor
CTD data.Temperature	CTD_START + 4 float	Temperature measured by CTD sensor
CTD data.Pressure	CTD_START + 8 float	Pressure measured by CTD sensor
STM data.Scattering	STM_START float	Scattering measured by STM sensor
STM data.HighRange	STM_START + 4 float	High range measured by STM sensor
STM data.StdDevScattering	STM_START + 8 float	Standard deviation of scattering measured by STM sensor

STM data.StdDevHighRange	STM_START + 12 float	Standard deviation of high range measured by STM sensor

Position and size variables:

Name	Description
VEL_NB	Primary dimension of velocity data is number of beams. Length 0 if correlation data in configuration bit map is false.
VEL_NC	Second dimension of velocity data is number of cells pr beam. Length 0 if correlation data in configuration bit map is false.
OFFSET	Offset of data given at position 1 in this dataset. Number of bytes from start of record to start of data.
AMP_NB	Primary dimension of amplitude data is number of beams. Length 0 if correlation data in configuration bit map is false.
AMP_NC	Second dimension of amplitude data is number of cells pr beam. Length 0 if correlation data in configuration bit map is false.
AMP_POS	Correlation data starts after the amplitude data.
CORR_NB	Primary dimension of correlation data is number of beams. Length 0 if correlation data in configuration bit map is false.
CORR_NC	Second dimension of correlation data is number of cells pr beam. Length 0 if correlation data in configuration bit map is false.
CORR_POS	Correlation data starts after the amplitude data.
PGD_LEN	The length of percent good data is the number of cells in the data sett. If the percent good data in the config bitmap is false, length is 0.
PGD_START	Percent good data follows the AHRS data.
QTY_LEN	The length of Quality data is the number of cells in the dataset. If the Quality in the config bitmap is false, length is 0.
QTY_START	Quality data.
CTD_LEN	CTD length.
CTD_START	CTD data.
STM_LEN	STM length.
STM_START	STM data.

Object reference: Configuration of instrument

Configuration bit mask.

Field	Position Size	Description

Pressure sensor	0 bit	Has pressure sensor
Temperature sensor	1 bit	Has temperature sensor
Compass	2 bit	Has compass
Tilt sensor	3 bit	Has tilt sensor
Recharge battery	4 bit	Need to recharge battery
Number of slanted beams	8-11 4 bits	Number of slanted beams
Total number of beams	12-15 4 bits	Total number of beams

Object reference: Types of data included

Types of data included.

Field	**Position Size**	**Description**
Velocity included	0 bit	Velocity data is included
Amplitude included	1 bit	Amplitude data is included
Correlation included	2 bit	Correlation data is included
Percent good data included	3 bit	Percent good data included
Quality data included	4 bit	Quality data included
Altimeter data included	5 bit	Altimeter data included
Recharge battery included	6 bit	Information if the battery needs recharging is included
CTD included	7 bit	Information if CTD data is included
STM included	8 bit	Information if STM data is included

Object reference: Valid data

Bit is true if the chosen data is valid

Field	Position Size	Description
Serial number	0 bit	Serial number is valid
Year, month, day, hour, second	1 bit	Year, month, day, hour, second is valid
Microseconds valid	2 bit	Microseconds data is valid
Speed of sound	3 bit	Speed of sound is valid
Water temperature is valid	4 bit	Water temperature is valid
Pressure	5 bit	Pressure data included
Absolute pressure	6 bit	Absolute pressure data included
Heading	7 bit	Heading is valid
Pitch	8 bit	Pitch is valid
Roll	9 bit	Roll is valid
Standard deviation pressure	10 bit	Standard deviation pressure is valid
Standard deviation heading	11 bit	Standard deviation heading is valid
Standard deviation pitch	12 bit	Standard deviation pitch is valid
Standard deviation roll	13 bit	Standard deviation roll is valid
Number of beams	14 bit	Number of beams is valid
Coordinate system	15 bit	Coordinate system is valid
Number of cells	16 bit	Number of cells is valid

Cell size	17 bit	Cell size is valid
Blanking	18 bit	Blanking is valid
Battery voltage	19 bit	Battery voltage is valid
Temperature in pressure sensor	20 bit	Temperature in pressure sensor data is valid
Temperature in magnetometer is valid	21 bit	Pressure data included
Temperature real time clock	22 bit	Temperature in real time clock is valid
Magnetometer raw	23 bit	Magnetometer raw data is valid
Accelerometer raw data	24 bit	Accelerometer raw data is valid
Ambiguity velocity	25 bit	Ambiguity velocity is valid
Velocity scale	26 bit	Velocity scale is valid
Power level	27 bit	Power level is valid
Error	28 bit	Error is valid
Status	29 bit	Status is valid
Depth average speed	30 bit	Depth average speed is valid
Depth average direction	31 bit	Depth average direction is valid
Number of averaged pings	32 bit	Number of averaged pings data is valid
Nominal correlation	33 bit	Nominal correlation is valid
CTDtemperature	34 bit	CTDtemperature is valid
CTDpressure	35 bit	CTDpressure is valid
CTDconductivity	36	CTDconductivity is valid

	bit	
STMvalues	37 bit	STMvalues is valid

Object reference: Error status

Error bit mask.

Field	Position Size	Description
Data retrieval FIFO error	0 bit	Data retrieval FIFO error
Data retrieval overflow	1 bit	Data retrieval overflow
Data retrieval underrun	2 bit	Data retrieval Underrun
Data retrieval samples missing	3 bit	Data retrieval samples missing
Measurement not finished	4 bit	Measurement not finished The Measurement and data storage/transmit didn’t finish before next measurement started.
Sensor read failure	5 bit	Sensor read failure
Tag error beam 1 (In-phase)	8 bit	Tag error beam 1 (In-phase)
Tag error beam 1 (Quadrature-phase)	9 bit	Tag error beam 1 (Quadrature-phase)
Tag error beam 2 (In-phase)	10 bit	Tag error beam 2 (In-phase)
Tag error beam 2 (Quadrature-phase)	11 bit	Tag error beam 2 (Quadrature-phase)
Tag error beam 3 (In-phase)	12 bit	Tag error beam 3 (In-phase)
Tag error beam 3 (Quadrature-phase)	13 bit	Tag error beam 3 (Quadrature-phase)
Tag error beam 4 (In-Phase)	14 bit	Tag error beam 4 (In-phase)

Tag error beam 4 (Quadrature-phase)	15 bit	Tag Error Beam 4 (Quadrature-phase)

Object reference: Status

Status bit mask.

Field	Position Size	Description
Is average data	0 bit	True if data is average
Active configuration	1 bit	True if active configuration
Last measured low voltage skip	2 bit	Last measured low voltage skip
In air	3 bit	True if instrument is in air
Previous wakeup state	16-19 4 bits	0 - Bad power 1 - Power on 2 - Break 3 - RTC 4 - Watchdog 5 - Low voltage 6 - Filesystem error
Auto orientation	20-22 3 bits	Bit 20-22: autoOrientation 0: "Fixed" Fixed orientation 1: "Auto" Auto Up Down 2: "Auto3D" 3: "AHRS3D" AHRS3D
Orientation	24-27 4 bits	Bit 24-27: Orientation 4: "UP" 5: "DOWN" 7: "AHRS"
Wake up state	28-31 4 bits	0 - Bad power 1 - Power on 2 - Break 3 - RTC 4 - Watchdog 5 - Low voltage 6 - Filesystem error

6.5 DF3 VelocityData

Extends: _DF3 CurrentProfileData
ID: 0x15, 0x16, 0x18, 0x1e, 0x1a, 0x1f

Data definitions for parsing Nortek velocity data format 3.

Field	Position Size	Description
Beams, coordinates and cells	30 2 bytes	Number of beams, coordinate system and number of cells. Object reference given in table below
STM data.Scattering	OFFSET float	Scattering measured by STM sensor
STM data.HighRange	OFFSET + 4 float	High range measured by STM sensor
STM data.reserved	OFFSET + 8 float * 4	4 reserved floats
Velocity data	VEL_START int16 * VEL_NB * VEL_NC	This field exists if the Velocity data included bit of the Config byte is set. 10^(Velocity Scaling) Unit: [m/s]
Amplitude data	AMP_POS uint8 * AMP_NB * AMP_NC	This field exists if the amplitude data included bit of the Config byte is set 0.5 dB/count Unit: [dB]
Correlation data	CORR_POS uint8 * CORR_NB * CORR_NC	This field exists if the Correlation data included bit of the Config byte is set [0 – 100 %] Unit: [%]
Altimeter data.Altimeter distance	ALTI_POS float	Distance to surface from Leading Edge algorithm Unit: [m]
Altimeter data.Altimeter quality	ALTI_POS + 4 uint16	Result of LE algorithm. When quality is deemed too low according to instrument specific limits both the distance and quality are set to 0.
Altimeter data.Altimeter status	ALTI_POS + 6 2 * 8 bits	Altimeter status bit mask
AST data.AST distance	AST_POS float	Distance to surface from Max Peak/AST algorithm Unit: [m]
AST data.AST quality	AST_POS + 4 uint16	Amplitude at which surface is detected with the Max Peak/AST algorithm.

		Raw data in steps of 0.01 dB, i.e. quality of 8000 = 80 dB Unit: [dB]
AST data.AST offset	AST_POS + 6 int16	Offset in step of measurement to velocity measurement Raw data given in 100 µs Unit: [s]
AST data.AST pressure	AST_POS + 8 float	Pressure value measured during the AST/altimeter ping Unit: [dbar]
Altimeter raw data.Num RawSamples	ALTIRAW_START + 8 uint32	Altimeter Raw Data – Number of Samples
Altimeter raw data.Samples distance	ALTIRAW_START + 12 uint16	Distance between samples Raw data given in 0.1mm Unit: [m]
Altimeter raw data.Data samples	ALTIRAW_START + 14 int16 * NRS	Altimeter Raw Data – Samples Raw data given as 16 bits Signed fract
AHRS data.Rotation matrix	AHRS_START float * 3 * 3	AHRS Rotation Matrix [3x3]
AHRS data.Quaternion W	AHRS_START + 36 float	W quaternion
AHRS data.Quaternion X	AHRS_START + 40 float	X quaternion
AHRS data.Quaternion Y	AHRS_START + 44 float	Y quaternion
AHRS data.Quaternion Z	AHRS_START + 48 float	Z quaternion
AHRS data.Gyro X	AHRS_START + 52 float	Gyro in X direction in degrees pr second Unit: [dps]
AHRS data.Gyro Y	AHRS_START + 56 float	Gyro in Y direction in degrees pr second Unit: [dps]
AHRS data.Gyro Z	AHRS_START + 60 float	Gyro in Z direction in degrees pr second Unit: [dps]
Percentage good data	PGD_START uint8 * PGD_LEN	Percent Good Estimate per cell This field exists if the Percentage Good data is included. For the Signature instruments, this will only be relevant for the _avgd.ad2cp file. Unit: [%]
Standard deviation data.Pitch	SD_START int16	Standard deviation on pitch data Raw data in 0.01 degrees

		Unit: [deg]
Standard deviation data.Roll	SD_START + 2 int16	Standard deviation on roll data Raw data in 0.01 degrees Unit: [deg]
Standard deviation data.Heading	SD_START + 4 int16	Standard deviation on heading data Raw data in 0.01 degrees Unit: [deg]
Standard deviation data.Pressure	SD_START + 6 int16	Standard deviation on pressure data Raw data in 0.001 Bar Unit: [bar]

Position and size variables:

Name	Description
STM_LEN	STM length.
OFFSET	Offset of data given at position 1 in this dataset. Number of bytes from start of record to start of data.
VEL_NB	Primary dimension of velocity data is number of beams. Length 0 if correlation data in configuration bit map is false.
VEL_NC	Second dimension of velocity data is number of cells pr beam. Length 0 if correlation data in configuration bit map is false.
VEL_START	Starting byte position of velocity data, which comes after the STM data section.
AMP_NB	Primary dimension of amplitude data is number of beams. Length 0 if correlation data in configuration bit map is false.
AMP_NC	Second dimension of amplitude data is number of cells pr beam. Length 0 if correlation data in configuration bit map is false.
AMP_POS	Correlation data starts after the amplitude data.
CORR_NB	Primary dimension of correlation data is number of beams. Length 0 if correlation data in configuration bit map is false.
CORR_NC	Second dimension of correlation data is number of cells pr beam. Length 0 if correlation data in configuration bit map is false.
CORR_POS	Correlation data starts after the amplitude data.
ALTI_POS	Altimeter data starts after the correlation data.
AST_POS	AST data starts after the altimeter data.
NRS	Number of raw samples given as first element of this object.
ALTIRAW_START	Altimeter raw data starts after the AST data.
AHRS_START	AHRS data follows the altimeter raw data.
PGD_START	Percent good data follows the AHRS data.

SD_START	The standard deviation data follows percent good data.

Object reference: Beams, coordinates and cells

Number of beams, coordinate system and number of cells.

Field	Position Size	Description
Number of beams	15-12 2 bits	Number of Beams (NB) Active beams represented as a 4 char string of 1s and 0s.
Coordinate system	11-10 2 bits	Coordinate system 00 = ENU 01 = XYZ 10 = BEAM 11 = not used
Number of cells	9-0 10 bits	Number of Cells (NC).

6.6 DF3 SpectrumData

Extends: _DF3 CurrentProfileData
ID: 0x20

Data definitions for parsing DF3 amplitude spectrum data.

Field	Position Size	Description
Beams and bins	30 2 bytes	Number of bins in the frequency spectrum. Object reference given in table below
Spectrum data.Start frequency	OFFSET float	Start frequency value Unit: [Hz]
Spectrum data.Step frequency	OFFSET + 4 float	Step frequency value Unit: [Hz]
Spectrum data.Frequency data	OFFSET + 64 int16 * BEAMS * BINS	Frequency spectrum amplitude data. There is room for 16 floating points for a spectrum header before the frequency data. Unit: [dB]

Position and size variables:

Name	Description

BEAMS	Matrix first dimension is number of beams. Eg: [[f_start, .., f_{start+step*(bins-1)}]_{beam1} [f_start, .., f_{start+step*(bins-1)}]_{beam2} .. [f_start, .., f_{start+step*(bins-1)}]_{beams}]
BINS	Per beam, frequencies are given as an array of length as number of bins. First element is the start frequency and frequencies increment by step frequency per element of the array. Eg: [[f_start, f_{start+step}, f_{start+step*2}, .., f_{start+step*(bins-1)}] _{beam1}, .., ..]
16+BEAMSxBINSx2	If configuration.hasSpectrumData is false, spectrum data is length 0. RAW: !this.configuration.hasSpectrumData ? 0 : this.beamsAndBins.numberOfBeams*this.beamsAndBins.numberOfBins*2 + 16*4
OFFSET	Number of bytes from start of record to start of data.

Object reference: Beams and bins

Number of bins in the frequency spectrum.

Field	Position Size	Description
Number of beams	15-13 3 bits	Number of active beams.
Number of bins	12-0 13 bits	Number of bins.

6.7 StringData

ID: 0xa0

String Data Record, eg. GPS NMEA data, comment from the FWRITE command.

Field	Position Size	Description
String	0 Size of record bytes	String data record.

7 Data Quality Control

When collecting average current data with your 2D Horizontal Profiler, the averaged data for each cell and each configured average interval is saved to the **[filename]_avgd.ad2cp** file for the 400 kHz version and to the **[filename]_avgd.aqd** file for the 1 MHz version. The data format used for the average current data is data format 7 (DF7, see DF7 CurrentProfileData). Single ping data is not saved. To avoid bias from poor-quality pings, the instrument applies an internal quality control before averaging.

The internal quality control process for the DF7 average current data can be outlined as follows:

1. **Correlation filtering:** All pings with correlation lower than 50 % are discarded. For more information about the correlation parameter, please refer to Correlation - theoretical background.
2. **Bin mapping:** the data is bin mapped on a ping-by-ping basis. For more information about bin mapping, please refer to the article Tilt
3. **Averaging:** the bin mapped pings that remains after the correlation filtering are averaged together within one average interval and one cell.
4. **Quality Code:** each averaged cell is given a specific Quality Code. The Quality Code flags potential data issues but does not discard or modify the data. Each bit indicates a specific quality check, as listed below.

Percent good

In the DF7 average current data there is also a Percentage Good value which indicates the percentage of valid pings retained in each cell during an average interval after the internal quality control. If the Percentage Good value is 0 %, no valid pings contributed to the average, and the resulting velocity is set to 0. The removal of pings with a correlation lower than 50% is irreversible as this is done prior to the averaging to ensure good data quality.

Example
You have configured your instrument to do 10 pings within each average interval. One ping in the first cell is discarded because its correlation is below 50 %. The resulting Percent Good value will be 90 % for this cell.

Quality Code

The Quality Code is a 16-bit status code. Each bit in the Quality Code indicates a specific quality check, as listed below:

Bit 0 (0000 0000 0001) - Not used

Bit 1 (0000 0000 0010) - Fish filter
Mask based on spike detection in each beam when compared to the mean values of the other beams.

Bit 2 (0000 0000 0100) - Correlation
Pings with correlation less than 50 % are masked.

Bit 3 (0000 0000 1000) - Pressure
Uses the pressure readings to detect if the collected data is out of water.

Bit 4 (0000 0001 0000) - Not used

Bit 5 (0000 0010 0000) - Sidelobe
Masks cells contaminated by surface sidelobes by comparing the minimum absolute pressure during the average interval to a conservative cutoff value of 1050 hPa for the corresponding surface pressure. For more information about sidelobes, refer to the following article: Sidelobe interference

Bit 6 (0000 0100 0000) - Percent good
Masks the cell if the Percent Good value is below 50 %. The Percent Good is calculated by dividing the number of pings that have passed the correlation test by the total number of pings within the cell and average interval.

Bit 7 (0000 1000 0000) - Not used

Bit 8 (0001 0000 0000) - Bin map
Used when no cells could be mapped to this vertical position, for example due to a tilted instrument.

Bit 9 (0010 0000 0000) - Tilt
Masks entire profile where either the average pitch or roll exceeds 40° through the average interval or the pitch or roll standard deviation exceeds 5°.

If the Quality Code is 0, all controls are passed. Otherwise, the code itself indicates why the data is flagged. If a data point is flagged by several control parameters, the code will be a combination of the respective bits. Note that these checks do not discard or alter the data in any way, only flag it. The exception is the Bin map flag which indicates there was not sufficient data coverage to map data to a specific cell position due to a tilted instrument. This will leave the velocity cell empty.

Example
One cell has one or several pings discarded due to low correlation and is also detected to be within the sidelobe interference area.
The resulting Quality Code is **36** which equals **0000 0010 0100** in binary. Here we can see that bit 2 and bit 5 are used.

During post-processing, you can either discard data flagged by one or more quality parameters, or retain it and perform manual quality control.

7.1 Telemetry Quality Control

When your 2D Horizontal Profiler is configured to output telemetry average current data, correlation filtering is applied before the data is averaged and output. All pings below the correlation threshold are discarded. By default, the lower correlation threshold is set to 50 %, but this can be adjusted using the correlation threshold argument. The threshold can be changed either directly in the Nortek Deployment software user interface or by using the **CORRTH** argument with the **SETTMAVG** command.

If more than 50 % of the pings within one cell and average interval are discarded due to correlation below the threshold, the entire cell will be flagged as invalid in the telemetry output. Flagged data will be represented by extreme placeholder values in the telemetry output:

- Velocity: -32.767 m/s
- Speed: 46.34 m/s
- Direction: 225°

8 Telemetry Data Formats

This section describes the Telemetry Data formats. For more information about the telemetry functionality in general, refer to the Telemetry chapter. Note that all averaged telemetry current data undergo an internal quality control as described in the Telemetry Quality Control chapter.

The checksum calculation is part of the NMEA standard. It is the representation of two hexadecimal characters of an XOR if all characters in the sentence between – but not including – the $ and the * character.

8.1 Average Telemetry Data Formats

The telemetry of the average current data is controlled by the **SET/GETTMAVG** command. The DF parameter of this command sets the data format, this can also be set in the deployment software.

Data format (DF)	Description
3	Binary format as described in the data format chapter.
7	Binary format as described in the data format chapter. Same as saved to the recorder.
100	Same NMEA format as previous generations of Aquadopp/AWAC (NMEA sentences: PNORI, PNORS and PNORC).
101	NMEA format 1 (without tags). (NMEA sentences: PNORI1, PNORS1 and PNORC1).
102	NMEA format 2 (with tags). (NMEA sentences: PNORI2, PNORS2 and PNORC2).
103	NMEA format 3 (with tags). (NMEA sentences: PNORH3, PNORS3 and PNORC3).
104	NMEA format 4 (without tags). (NMEA sentences: PNORH4, PNORS4 and PNORC4).
150	RDI Workhorse PD0 data format. Refer to RDI for documentation.
503	Only for the 400 kHz instrument. Legacy binary data format, same as for the previous generation of AWACs. Described in the Integrators Guide - Classic
504	Only for the 400 kHz instrument. Legacy ASCII data format, same as for the previous generation of AWACs. Described in the Integrators Guide - Classic
600	Only for the 1 MHz instrument. Legacy binary data format, same as for the previous generation of Aquadopp Profilers. Described in the Integrators Guide - Classic
603	Only for the 1 MHz instrument. Legacy ASCII data format, same as for the previous generation of Aquadopp Profilers. Described in the Integrators Guide - Classic

8.1.1 NMEA Format (DF=100)

The time stamp for DF100 for the 2D Horizontal Profilers will be set to the start of the average interval to be backwards compatible with the NMEA data format used by the previous generation.

Information (configuration): $PNORI

Column	Description	Data format	Example
0	Identifier	"$PNORI"	
1	Instrument type	N 0=Aquadopp 2=Aquadopp Profiler 4=Signature	2
2	Head ID	String	Aquadopp Profiler 2 MHz S2SP123456
3	Number of beams	N	3
4	Number of cells	N	20
5	Blanking (m)	dd.dd	0.20
6	Cell size (m)	dd.dd	1.00
7	Coordinate system	N 0=ENU 1=XYZ 2=BEAM	0
8	Checksum	*hh	2E

Example (DF=100): `$PNORI,2,Aquadopp Profiler 2 MHz S2SP123456,3,20,0.20,1.00,0*2E`

Sensor data: $PNORS

Column	Description	Data format	Example
0	Identifier	"$PNORS"	
1	Date	MMDDYY	102115
2	Time	HHMMSS	090715
3	Error Code (hex)	hh	00000000
4	Status Code (hex)	hh	2A480000
5	Battery Voltage	dd.d	14.4

6	Sound Speed	dddd.d	1523.0
7	Heading	ddd.d	275.9
8	Pitch (deg)	dd.d	15.7
9	Roll (deg)	dd.d	-2.3
10	Pressure (dBar)	ddd.ddd	0.000
11	Temperature (deg C)	dd.dd	22.45
12	Analog input #1	n/a	0
13	Analog input #2	n/a	0
14	Checksum	*hh	1C

Example (DF=100):

```
$PNORS,102115,090715,00000000,2A480000,14.4,1523.0,275.9,15.7,2.3,0.000,22.45,0,0*1C
```

Current velocity data: $PNORC

Column	Description	Data format	Example
0	Identifier	"$PNORC"	
1	Date	MMDDYY	102115
2	Time	HHMMSS	090715
3	Cell number	N	4
4	Velocity 1 (m/s) (Beam1/X/East)	dd.dd	0.56
5	Velocity 2 (m/s) (Beam2/Y/North)	dd.dd	-0.80
6	Velocity 3 (m/s) (Beam3/Z1/Up1)	dd.dd	-1.99
7	Velocity 4 (m/s) (Beam4/Z2/Up2) - not relevant for three beam systems, will be empty	dd.dd	
8	Speed (m/s)	dd.dd	0.98
9	Direction (deg)	ddd.d	305.2
10	Amplitude unit	C=Counts	C
11	Amplitude (Beam 1)	N	80
12	Amplitude (Beam 2)	N	88
13	Amplitude (Beam 3)	N	67

14	Amplitude (Beam 4) - not relevant for three beam systems, will be empty	N	
15	Correlation (%) (Beam1)	N	13
16	Correlation (%) (Beam2)	N	17
17	Correlation (%) (Beam3)	N	10
18	Correlation (%) (Beam4) - not relevant for three beam systems, will be empty	N	
19	Checksum	*hh	22

Example (DF=100): $PNORC,102115,090715,4,0.56,-0.80,-1.99,,0.98,305.2,C,80,88,67,,13,17,10,*22

8.1.2 NMEA Format 1 and 2 (DF=101/102)

The time stamp for DF101 and DF102 for the 2D Horizontal Profilers will be set to the time of the middle ping in the average interval.

Information Data:
Identifier:
PNORI1 for DF = 101 (without tags)
PNORI2 for DF = 102 (with tags)

Column	Description	Unit	TAG	Data format	Example
1	Instrument type		IT	N 0=Aquadopp 2=Aquadopp Profiler 4=Signature	IT=2
2	Head ID		SN	N	SN=123456
3	Number of Beams		NB	N	NB=3
4	Number of Cells		NC	N	NC=30
5	Blanking Distance	[m]	BD	dd.dd	BD=1.00
6	Cell Size	[m]	CS	dd.dd	CS=5.00
7	Coordinate System (ENU,BEAM,XYZ)		CY	N	CY=BEAM

Example (DF=101):
$PNORI1,2,123456,3,30,1.00,5.00,BEAM*5B
Example (DF=102):
$PNORI2,IT=2,SN=123456,NB=3,NC=30,BD=1.00,CS=5.00,CY=BEAM*68

Sensors Data:

Identifier:
PNORS1 for DF = 101 (without tags)
PNORS2 for DF = 102 (with tags)

Column	Description	Unit	TAG	Data format	Example
1	Date		DATE	MMDDYY	DATE=083013
2	Time		TIME	HHMMSS	TIME=132455
3	Error Code		EC	N	EC=0
4	Status Code		SC	hhhhhhhh	SC=34000034
5	Battery Voltage	[V]	BV	dd.d	BV=22.9
6	Sound Speed	[m/s]	SS	dddd.d	SS=1500.0
7	Heading Std.Dev.	[deg]	HSD	dd.dd	HSD=0.02
8	Heading	[deg]	H	ddd.d	H=123.4
9	Pitch	[deg]	PI	dd.d	PI=45.6
10	Pitch Std.Dev	[deg]	PISD	dd.dd	PISD=0.02
11	Roll	[deg]	R	dd.d	R=23.4
12	Roll Std.Dev.	[deg]	RSD	dd.dd	RSD=0.02
13	Pressure	[dBar]	P	ddd.ddd	P=123.456
14	Pressure StdDev	[dBar]	PSD	dd.dd	PSD=0.02
15	Temperature	[deg C]	T	dd.dd	T=24.56

Example (DF=101):
$PNORS1,083013,132455,0,34000034,22.9,1500.0,0.02,123.4,45.6,0.02,R=23.4,0.02,123.456,0.02,24.56*39
Example (DF=102):
$PNORS2,DATE=083013,TIME=132455,EC=0,SC=34000034,BV=22.9,SS=1500.0,HSD=0.02,H=123.4,PI=45.6,PISD=0.02,R=23.4,RSD=0.02,P=123.456,PSD=0.02,T=24.56*3F

Averaged Data:
Identifier:
PNORC1 for DF = 101 (without tags)
PNORC2 for DF = 102 (with tags)

The averaged data is repeated for each measurement cell.

Column	Description	Unit	TAG	Data format	Example
1	Date		DATE	MMDDYY	DATE=083013
2	Time		TIME	HHMMSS	TIME=132455
3	Cell Number		CN	N	CN=3
4	Cell Position	[m]	CP	dd.d	CP=11.0
5	Velocity East - only included if CY=ENU	[m/s]	VE	dd.ddd	VE=0.332
6	Velocity North - only included if CY=ENU	[m/s]	VN	dd.ddd	VN=0.332
7	Velocity Up - only included if CY=ENU	[m/s]	VU	dd.ddd	VU=0.332
9	Velocity X - only included if CY = XYZ	[m/s]	VX	dd.ddd	VX=0.332
10	Velocity Y - only included if CY = XYZ	[m/s]	VY	dd.ddd	VY=0.332
11	Velocity Z - only included if CY = XYZ	[m/s]	VZ	dd.ddd	VZ=0.332
13	Velocity Beam 1 - only included if CY=BEAM	[m/s]	V1	dd.ddd	V1=0.332
14	Velocity Beam 2 - only included if CY=BEAM	[m/s]	V2	dd.ddd	V2=0.332
15	Velocity Beam 3 - only included if CY=BEAM	[m/s]	V3	dd.ddd	V3=-0.332
17	Amplitude Beam 1	[dB]	A1	ddd.d	A1=78.9
18	Amplitude Beam 2	[dB]	A2	ddd.d	A2=78.9
19	Amplitude Beam 3	[dB]	A3	ddd.d	A3=78.9
21	Correlation Beam 1	[%]	C1	N	C1=78
22	Correlation Beam 2	[%]	C2	N	C2=78
23	Correlation Beam 3	[%]	C3	N	C3=78

Example (DF=101, CY=ENU):
```
$PNORC1,083013,132455,3,11.0,0.332,0.332,0.332,78.9,78.9,78.9,78,78,78*46
```
Example (DF=102, CY=ENU):
```
$PNORC2,DATE=083013,TIME=132455,CN=3,CP=11.0,VE=0.332,VN=0.332,VU=0.332,A1=
78.9,A2=78.9,A3=78.9,C1=78,C2=78,C3=78*6D
```

8.1.3 NMEA Format 3 and 4 (DF=103/104)

The time stamp for DF103 and DF104 for the 2D Horizontal Profilers will be set to the time of the middle ping in the average interval.

Header Data:

Identifier:
PNORH3 for DF = 103 (with tags)
PNORH4 for DF = 104 (without tags)

Column	Description	Unit	TAG	Data format	Example
1	Date		DATE	YYMMDD	DATE=141112
2	Time		TIME	HHMMSS	TIME=081946
3	Error Code		EC	N	EC=0
4	Status Code		SC	hhhhhhhh	SC=2A4C0000

Example (DF=103):
$PNORH3,DATE=141112,TIME=081946,EC=0,SC=2A4C0000*5F
Example (DF=104):
$PNORH4,141112,083149,0,2A4C0000*4A68

Sensors Data:
Identifier:
PNORS3 for DF = 103 (with tags)
PNORS4 for DF = 104 (without tags)

Column	Description	Unit	TAG	Data format	Example
1	Battery	[V]	BV	dd.d	BV=22.9
2	Sound Speed	[m/s]	SS	dddd.d	SS=1500.0
3	Heading	[deg]	H	ddd.d	H=123.4
4	Pitch	[deg]	PI	dd.d	PI=45.6
5	Roll	[deg]	R	dd.d	R=23.4
6	Pressure	[dBar]	P	ddd.ddd	P=123.456
7	Temperature	[deg C]	T	dd.dd	T=24.56

Example (DF=103):
$PNORS3,BV=22.9,SS=1546.1,H=151.1,PI=-12.0,R=-5.2,P=705.669,T=24.96*7A
Example (DF=104):
$PNORS4,22.9,1546.1,151.2,-11.9,-5.3,705.658,24.95*5A

Averaged Data:
Identifier:
PNORC3 for DF = 103 (with tags)
PNORC4 for DF = 104 (without tags)

The averaged data is repeated for each measurement cell.

Column	Description	Unit	TAG	Data format	Example
1	Cell position	[m]	CP	dd.d	CP=2.5
2	Speed	[m/s]	SP	dd.ddd	SP=0.751
3	Direction	[deg]	DIR	ddd.d	DIR=170.1
4	Averaged Correlation		AC	N	AC=5
5	Averaged Amplitude		AA	N	AA=28

Example (DF=103):
`$PNORC3,CP=4.5,SP=3.519,DIR=110.9,AC=6,AA=28*3B`
Example (DF=104):
`$PNORC4,27.5,1.815,322.6,4,28*70`