Sparton Inertial Systems

AHRS-M1/M2 Software Interface Control Document

Applicable to Sparton AHRS-M1 and AHRS-M2

REVISION HISTORY:

DATE	DESCRIPTION OF CHANGE	INITIALS	REV
05/25/2015	Original	RSW	A
07/13/2016	Updated Protocol Triggers section and added User Boot Program section	MJB	B
1/15/2018	Added AHRS-8 compatibility changes	MJB	C
6/7/2018	Merged with M1 document. Noted when M1 gained features	JTE	D
2/15/2023	Added sections for WMM and Mount Orientation, added XML description (Appendix A), added some troubleshooting (Appendix B). Spellcheck	MJB	E

Table of Contents

1 Introduction

The AHRS-M1_M2 software interface control document (ICD) is intended to be a reference manual. It is recommended the user start with the Sparton AHRS-M2 Quick Start Guide that gives a brief overview of the different protocol options and refers to sections in this document for more detail.

This User Manual describes how to communicate with, control, and obtain sensor status and configuration data from the Sparton family of inertial data products. This document is intended for system engineers, systems integrators or software developers who need to know how to communicate with, utilize, and obtain measurements from a Sparton inertial product.

This document applies to the AHRS-M1/M2, the device will be referred to as Inertial Systems and all references shall apply to all devices except where noted.

1.1 Scope

The AHRS-M2 ICD covers the following topics:

- Low level communications interfaces at the software level
- Packet level communications protocol
- Applications layer functions and the relevance to the function of the inertial system
- Examples of commands for user purposes

This document is a software ICD not a hardware or mechanical data sheet.

1.2 References

The ICD is an umbrella document in that some of the interface descriptions for the inertial systems are described in external documentation. This document describes the relationship of these documents. The additional documentation needed to complete the entire interface description for the inertial system family is listed below.

- *AHRS-M2 Quick Start Guide*
- *NorthTek™ System Programming Manual*
- *Remote Function Select (RFS) Protocol Suite, Interface Design Description*

1.3 Document Conventions

Descriptive text in this document appears in this font (Calibri 11).

Data sent transmitted to the inertial system appears in ***this font (Courier 11, Bold Italic).***

Data sent from the inertial system appears in **this font (Courier 11, Bold, grey background).**

Information that must be supplied in this spot is enclosed in **<>** pairs. The **<contents>** brace pair should be replaced with the user supplied data or will be filled in by the inertial system according to the description contained in the **<>**'s.

- A carriage return character is shown as **<CR>** or **<cr>.**
- A line feed character is shown as **<LF>** or **<lf>.**

- Control sequences are indicated as **<CTRL-S>.**

Nonprintable characters are spelled out in **<>** pairs, for example **<ESCAPE>** is equal to the escape character (0x1b). Additional text will be included when nonprintable characters might not be clear to the reader.

1.4 Applicability

The AHRS-M1/M2 Software ICD applies to the Sparton AHRS-M1 and AHRS-M2 Inertial Systems.

2 Overview

This document is intended to be a reference document. Please read through the Overview section and the AHRS-M2 Quick Start Guide and follow the references in the areas of interest. The AHRS-M2 Quick Start Guide gives step-by-step examples and points to sections in this document for more information.

The ICD details the available communication protocols as well as use of the inertial system from the programmer's point of view. Also included is a description of a unique capability where the user can program custom output from the inertial system using NorthTek™ scripts.

The AHRS-M2 provides high speed inertial data and attitude information in a small, lightweight package utilizing accelerometers, gyroscopes, magnetometers, and a temperature sensor.

2.1 Protocol Introduction

The interface to the inertial system is a serial communication link (User Port). The link is an asynchronous character-oriented interface operating in a bidirectional mode. The serial link transmits and receives asynchronous characters using nominal "UART" type framing with 8 data bits, no parity and a single stop bit, with a default baud rate of 115200. The baud rate may be changed by software command. No error detection is used with the individual characters.

The serial link is full duplex; however, in practice the information flow is largely half-duplex in nature. Initially, the client and the server operate in a poll-response fashion. For example, the device that desires to use the inertial system would be the master (client) and the inertial system would be the slave (server). The host device (master) sends a set of commands that define a set of desired data at a desired rate which the inertial device will then output periodically.

The inertial system provides the following protocols on the User Port:

- NorthTek™ -- A free form command line interpreter suitable for human operator usage or custom output requirements.
- NMEA --- A printable ASCII protocol that conforms to the general message formatting used by the National Marine Equipment Manufacturers (NMEA) association guidelines.
- Bit Stream ASCII – user customizable text based streaming data sets with optional checksum or CRC.

- Bit Stream Binary – user customizable binary streaming data sets with optional checksum or CRC.
- RFS -- A binary protocol called Remote Function Select (RFS). This protocol provides a robust, error checked method of communicating with the inertial system.

The RFS and NorthTek protocols provide a degree of backwards software compatibility with the Sparton DC-4E, GEDC-6E and AHRS-8 products.

The RFS protocol provides 1) Robust error checking via a 16 bit CRC as part of each packet, 2) User definable messages such that the user can generally put all required data into a single message and 3) A self-descriptive command structure such that new inertial system features are automatically accessible to hosts. 4) The binary representation is efficient for larger data sets

The NorthTek™ interface is an advanced user interface that allows the user to enter and perform small command scripts within the inertial system in addition to a standard set of interface commands.

2.2 In-Field Calibration

Sparton's AHRS-M2 utilizes innovative magnetic mapping and advanced adaptive algorithms to maintain overall system accuracy in real-time. The device offers two calibration modes: Precision Manual Calibration, and the new Autonomous or Auto-Calibration mode. In-field calibration can only take out magnetic disturbances that remain fixed relative to the inertial system while in motion.

2.2.1 AdaptCal

Autonomous calibration frees the user from the periodic calibrations necessary to keep a system at optimal performance over extended periods of time. During normal movement of the product, the autonomous calibration algorithm will automatically select points to ensure spherical diversity and start converging to a solution when an adequate number have been collected. The solution is further refined in real-time when new measurements are received without any intervention from the user.

```
// Prevent multiple loads of this script from overflowing the NorthTek program stack.
forget adaptCal
: adaptCal
  // User command to clear the point cal results
  clearPointCal 1 set drop
  300 delay    // milliseconds

  // Initialize the mag and accel calibration point buffer to all zeros
  initCalPointBuffer 1 set drop
  300 delay

  // begin the auto calibration mode
  autoFieldCalActive 1 set drop

  ." Calibration starting" cr
  350 delay

  ." Press any key to terminate" cr
```

```
  // This loop till continue to print the: (timing is print every ~0.5 seconds)
  //   1) number of points in the buffer used for calibration
  //   2) The calibration score in possible degrees of heading error (magFieldCalErr)
  //   3) The actual yaw error estimate which includes the calibration score
  begin
   ?key 0=
 while
   magBufferActiveIndex di.
   magFieldCalErr di.
   yawErrEst di.
   490 delay
 repeat
 key drop

  ." Calibration done!" cr

  // Turn off the auto calibration
  //   This is not necessary unless you specifically would like to turn off the auto calibration.
  //   The adaptcal is automatically being used once at least 4 points are collected.
  //   In this way, the adaptcal and manual point cal differ
  //   The adaptCcal needs to be switched off for the calibration to be used.
 autoFieldCalActive 0 set drop
 350 delay

  // Print out the calibration score one last time.
 magFieldCalErr di.
; //  End adaptCal
```

2.2.2 Manual Point Calibration

Precision manual calibration offers the user the ability to calibrate the system once and retain that calibration for future use. The maximum number of measurements has been increased from 12 to 32 allowing for more detailed calibrations in applications having a highly distorted magnetic environment. Users can take measurements in any orientation as long as the points are spatially diverse and not limited to one direction. For a good calibration, the device must be able to "see" the Earth's magnetic field from different orientations in order to know what the magnetic distortions look like in 3-dimensions. The precision calibration runs the same adaptive algorithm as the autonomous calibration to solve for magnetometer errors. The user can also select what action the algorithm takes once convergence to a solution has been achieved. The manual calibration can either turn itself off after convergence, stay in the manual calibration mode waiting for additional measurements while continuously converging, or transition to the autonomous calibration.

```
forget clearCals

// User command to clear the point cal results
// User command to default to the point cal
// Initialize the mag and accel calibration point buffer to all zeros

: clearCals
   clearPointCal 1 set drop
  350 delay

   initCalPointBuffer 1 set drop
  350 delay
;
```

```
// This loop till continue to print the: (timing is print every ~0.5 seconds)
//   1) number of points in the buffer used for calibration
//   2) The calibration score in possible degrees of heading error (magFieldCalErr)
//   3) The actual yaw error estimate which includes the calibration score

: convergenceLoop
  begin
    ?key 0=
  while
    ." t," cputime &di @ . ." ,pts," magBufferActiveIndex &di @ .
    ." ,err," magFieldCalErr &di @ f. ." ,YEE," yawErrEst &di @ f. cr
    490 delay
  repeat
  key drop
;

: cal3D
  clearCals

  ." Calibration starting" cr

  pointFieldCalActive 1 set drop
  350 delay

  ." Press any key to take next point, ESC to finish" cr

// Wait until key is pressed and either take a manual point or stop point capture
//  ESC key ends point capture
// Any other key captures a point and prints out the number of points captured
  begin
    key 27 = 0=
  while
    manualCalibrationPoint 1 set drop
    1000 delay
    magBufferActiveIndex di.
  repeat

  ." Starting error settling" cr
  ." Press any key to terminate" cr

  convergenceLoop

  ." Calibration done!" cr

// Turn off manualcalibration and print out calibration score.
// Depending on how the   variable is set, the calibration process ends or
//    1) transitions to auto calibration mode
//    2) or shuts off completely
  pointFieldCalActive 0 set drop
  350 delay // give compass time to process
  magFieldCalErr di.
;
```

2.3 Data Organization

The inertial systems incorporate an internal database. The database contains primitive elements. The primitive elements provide the storage for information that is accessed externally. The elements can be settings, commands or values. Regardless of the type of the element, the method of access to the value of the element is similar across all protocols.

Basic Data Types

The internal database elements are one of the following types:

- Int32 – A 32-bit signed integer.
- Float32 – An IEEE format 32-bit floating-point value.
- String – A counted string. Byte 0 is the length of the string that follows ***including*** the terminating null.
- Ordinal -- An 32 bit value that represents a selection from a discrete set of 0 based, named values (0..N).
- ArrayInt32 – An array of Int32 values
- ArrayFloat32 – An array of Float32 values.
- BitField – RFS only. A combined variable that includes multiple other data types in the same object.

The variable definitions for the inertial system are available as an XML file. This XML file describes each variable type, the dimensions of array variables, minimum and maximum values, and default values. Contact the manufacturer for availability.

The inertial system variables are listed in the table below. The variable names are the same names that would be entered in the commands. This is a shorthand table to remind the user of the variable names. Detailed descriptions of each variable are found in Appendix 0, Variable XML Descriptions. In general, all variables are available via any of the protocols except for the RFS composite variables.

2.3.1 Variable Summary Table

See Appendix A "Variable XML Descriptions" for the details on the variables in this table. Notes are given for variables that were introduced after the initial release. If there is not a notation listed and the variable is indicated to be in a product, then the variable existed in the product from the initial release.

Name	Description	Type
absoluteYawErr	Absolute yaw error	Float32
accelCalPoints	Array with up to calNumPoints of accelerometer field cal captured points	ArrayFloat32
accelCalTemperatureRaw	Temperature at which the Rate Table temperature comps are computed for each axis (256*C)	ArrayInt32
accelClearStatistics	Command to clear the accelerometer statistics (1=ON, 0=OFF)	Ordinal
accelCollectingStatistics	Collect data for accelerometer statistics (1=ON, 0=OFF)	Ordinal
accelDecimated	Averaged raw accel based on accelGyroDecimator	ArrayInt32
accelEst	Estimated accelerometer unit vector	ArrayFloat32
accelEstimatedNSamples	Estimated number of samples to reach accelTargetVariance	ArrayInt32
accelEstVariance	Estimated variance of the accel values	Float32
accelFCM	Accelerometer sensor factory calibration matrix	MatrixFloat32

accelGyroDecimator	raw accel and gyro is averaged every accelGyroDecimator number of samples	Int32
accelGyroDecimatorEndTime	Time the M0 sensor decimator task finished with the data	Int32
accelGyroDecimatorStartTime	Time the M0 sensor decimator task received the data	Int32
accelGyroM4Timestamp	Time the M4 received the accel and gyro raw values	Int32
accelGyroSMPLRT_DIV	Controls the sample rate divisor for the accel/gyro sensor	Int32
accelGyroTemperature	accel gyro sensor temperature - Celsius times 256	Int32
accelGyroTimestamp	Time of accel/gyro sensor data acquisition at decimation time	Int32
accelMean	Average raw accel since accelCollectingStatistics and/or accelClearStatistics	ArrayInt32
accelNSamples	Number of accel samples collected for statistical computations	Int32
accelp	Processed Accelerometer X,Y,Z array (milli-g)	ArrayFloat32
accelpClearStatistics	Command to clear the accelp statistics (1=ON, 0=OFF)	Ordinal
accelpCollectingStatistics	Collect data for accelp statistics (1=ON, 0=OFF)	Ordinal
accelpEstimatedNSamples	Estimated number of samples to reach accelpTargetVariance	ArrayInt32
accelpMean	Average accelp since accelpCollectingStatistics and/or accelpClearStatistics	ArrayFloat32
accelpNSamples	Number of accelp samples collected for statistical computations	Int32
accelpScaled	User scaled Accel X,Y,Z array	ArrayFloat32
accelpScaleFactor	Multiplied by accelp to get accelpScaled (change units)	Float32
accelpStatisticsInterval	Number of accelp samples between statistical result computations	Int32
accelpTargetVariance	Desired variance for the accelps used in computing accelpEstimatedNSamples	Float32
accelpTemperatureMean	Temperature average while the accelp statistics were being collected	Int32
accelpVarianceMean	Average variance of accelp data since accelpCollectingStatistics and/or accelpClearStatistics	ArrayFloat32
accelr	Raw Accelerometer X,Y,Z array	ArrayInt32
accelrange	Accelerometer scaling: 0=4g,1=8g,2=2g,3=16g	Ordinal
accelrangeFactor	Used to adjust the raw accel data according to the value of accelrange that is being used	Float32
accelRawTC	User copy of raw accel at factory calibration base temperature	ArrayFloat32
accelRawTCAdjustment	Temperature compensation vector computed using the accelTCM to be added to the raw accel vector	ArrayFloat32
accelSelfTestControlBits	Accel self-test control, bit 0 = Z, bit 1 = Y, bit 2 = X	Int32

accelStatisticsInterval	Number of accel samples between statistical result computations	Int32
accelTargetVariance	Desired variance for the accels used in computing accelEstimatedNSamples	Float32
accelTCM	accel sensor temperature compensation matrix	MatrixFloat32
accelTemperatureMean	Temperature average while the accel statistics were being collected	Int32
accelVariance	Initial value for the variance of the accel values	Float32
accelVarianceMean	Average variance of accel data since accelCollectingStatistics and/or accelClearStatistics	ArrayFloat32
accelXFactorySelfTest	Accel X self-test value generated by the sensor's factory	Int32
accelXSelfTest	Raw Accelerometer X self-test value	Int32
accelYFactorySelfTest	Accel Y self-test value generated by the sensor's factory	Int32
accelYSelfTest	Raw Accelerometer Y self-test value	Int32
accelZFactorySelfTest	Accel Z self-test value generated by the sensor's factory	Int32
accelZSelfTest	Raw Accelerometer Z self-test value	Int32
adaptNavActive	Users use this to turn on/off AdaptNav computations (normally on)	Ordinal
adaptNavIIICalCovEnd	Time the adaptNavIII calConvergence task ends (milliseconds)	Int32
adaptNavIIICalCovStart	Time the adaptNavIII calConvergence task starts (milliseconds)	Int32
adaptNavIIIDeltaTime	Time to run the AdaptNavIII algorithm	Int32
adaptNavIIIEndTime	Time the adaptNavIII task ends (milliseconds)	Int32
adaptNavIIIStartTime	Time the adaptNavIII task starts (milliseconds)	Int32
aEqAveraged	Used for monitoring error convergence	Float32
aEstimate	Gyro advanced accelerometer unit vector (unused)	ArrayFloat32
AHRS8_Compatible	Makes the AHRS-M2 largely compatible with AHRS-8 interface	Ordinal
alt	Altitude	Float32
angularMagP2PErr	Magnetic angle peak to peak error in field cal points	Float32
angularP2PErr	Angular peak to peak error in field cal points	Float32
autoFieldCalActive	User can set this to activate auto cal mode	Ordinal
baud	User Port Baud rate	Ordinal
boresightMatrix	Converts this unit’s orientation to host devices orientation	MatrixFloat32
boresightMatrixX	Boresight orientation for the X component	ArrayFloat32
boresightMatrixY	Boresight orientation for the Y component	ArrayFloat32
boresightMatrixZ	Boresight orientation for the Z component	ArrayFloat32
bornOn	Born on date	char
bufferFlushActive	Users use this to turn on/off clearing of field cal points	Ordinal
calBufferSwitch	Allows user selection of accel type to buffer into accelCalPoints for field cal	Ordinal

calCommand	Specific State Within 3D Calibration Mode	Ordinal
calCovCnt	Number of iterations of the calibration convergence algo	Int32
calDeqDelta	Field calibration derivative delta (unused)	Float32
calDeqDeltaCAC	Field calibration derivative delta	Float32
calDeqDeltaCAC2	Field calibration derivative delta for off axis terms (unused)	Float32
calmode	Calibration Mode	Ordinal
calNumPoints	Number of points collected for 3D/2D Calibration Mode	Int32
calPointBufferAccel	Accelerometer vectors captured with the mag cal points	MatrixFloat32
calPointBufferMag	Magnetic vectors captured as cal points	MatrixFloat32
calPointBufferOccupied	Indicates whether the corresponding point in the calPointBuffer is occupied	ArrayInt32
calPointDistribution	Factor indicating quality of in-field calibration points	Float32
calPointDistSigma	Calibration point distribution indicator factor	Float32
calPointsNumToAvg	Number of samples to be averaged for a 3D/2D Calibration Mode capture point	Int32
calStatus	Provide in-field calibration status	Ordinal
centripetalCompensation	Enable Centripetal acceleration correction of acceleration	Ordinal
centripetalCorrection	Enable Centripetal acceleration correction of acceleration	Ordinal
chan0Command	NorthTek or NMEA command string invoked by the channel0 trigger	char
chan0DisableBit	Simple function to disable a single bit in the chan0Enables variable	Int32
chan0EnableBit	Simple function to enable a single bit in the chan0Enables variable	Int32
chan0Enables	Bit mask the enables vids for output via channel 0	ArrayInt32
chan0Format	Channel 0's format for automatic output of variables	Ordinal
chan0Postamble	Postfix for channel 0's BitstreamAscii string	char
chan0Preamble	Prefix for channel 0's BitstreamAscii string	char
chan0Trigger	Event that triggers automatic preprogrammed output via Channel 0	Ordinal
chan0TriggerDivisor	Defines the number of channel 0 triggers per channel 0 output	Int32
chan1Command	NorthTek or NMEA command string invoked by the channel0 trigger	char
chan1DisableBit	Simple function to disable a single bit in the chan1Enables variable	Int32

chan1EnableBit	Simple function to enable a single bit in the chan1Enables variable	Int32
chan1Enables	Bit mask the enables vids for output via channel 1	ArrayInt32
chan1Format	Channel 1's format for automatic output of variables	Ordinal
chan1Postamble	Postfix for channel 1's BitstreamAscii string	char
chan1Preamble	Prefix for channel 1's BitstreamAscii string	char
chan1Trigger	Event that triggers automatic preprogrammed output via Channel 1	Ordinal
chan1TriggerDivisor	Defines the number of channel 1 triggers per channel 1 output	Int32
chan2Command	NorthTek or NMEA command string invoked by the channel0 trigger	char
chan2DisableBit	Simple function to disable a single bit in the chan2Enables variable	Int32
chan2EnableBit	Simple function to enable a single bit in the chan2Enables variable	Int32
chan2Enables	Bit mask the enables vids for output via channel 2	ArrayInt32
chan2Format	Channel 2's format for automatic output of variables	Ordinal
chan2Postamble	Postfix for channel 2's BitstreamAscii string	char
chan2Preamble	Prefix for channel 2's BitstreamAscii string	char
chan2Trigger	Event that triggers automatic preprogrammed output via Channel 2	Ordinal
chan2TriggerDivisor	Defines the number of channel 2 triggers per channel 2 output	Int32
chanTimerX100us	Defines the timer trigger event period which is this number times 100 microseconds	Int32
clearEEPROMLog	Clears the eeLog data including the startUpCount	Ordinal
clearFieldCal	Command variable used to override any AdaptCal field cal data with the most recent manual point cal data. This variable clears when the action has completed.	Ordinal
clearPointCal	User command to clear the point cal results	Ordinal
comp1Rate	composite1 report period (milliseconds)	Int32
comp2Rate	composite2 report period (milliseconds)	Int32
comp3Rate	composite3 report period (milliseconds)	Int32
compassCalcRate	Rate at which the compass computations run, with range 90-110 Hz	Int32
compassCollectingStatistics	Collect data for pitch/roll/yaw statistics (1=ON, 0=OFF)	Ordinal
compassDataReady	Event that indicates that the computed compass data is ready	Ordinal
compassOnForCalConvergence	Turns on compass computations for calibration convergence, 0=Off, 1=On	Ordinal
compDipAngle	complement of the Dip Angle (90-dipAngle, radians)	Float32

composite1	RFS only - User modifiable composite object with Roll,Pitch,Yaw,Lat,Lon,Alt,Temp	BitFieldElementType
composite1Dimension	Number of elements in array to follow	Byte
composite2	RFS only - User modifiable composite object with Roll,Pitch,Yaw,Lat,Lon,Alt,Temp	BitFieldElementType
composite2Dimension	Number of elements in array to follow	Byte
composite3	RFS only - User modifiable composite object with Roll,Pitch,Yaw,Lat,Lon,Alt,Temp	BitFieldElementType
composite3Dimension	Number of elements in array to follow	Byte
cp1	AccelEst crossed with MagEst unit vector	ArrayFloat32
cp2	cp1 crossed with AccelEst unit vector	ArrayFloat32
cputime	Processor time from startup (milliseconds)	Int32
currentTime	Processor time of the request for this variable	Int32
databaseReady	Indicator for the Cortex-M4 process that the database is ready (0x0F0FA5A5)	Int32
day	Estimated date expressed to the tenth of a year	Float32
debugMask	Turns on/off (bitwise) debug print statements controlled by *_mask commands	Int32
deltaDBias	Derivative approximation used in bias computation	Float32
deltaDCAC	Derivative approximation used in Cross Axis Correction computation	Float32
eeLogDatabaseWrite	Info indicating number of database writes to user EEPROM area	ArrayInt32
eeLogRestoreFactoryAuto	Failure info for an automatic restoration due to failure of both user DB buffersr	ArrayInt32
eeLogRestoreFactoryUser	Info indicating a user command of restorefactorycal	ArrayInt32
eeLogUserDB1Fail	Failure info for the database EEPROM buffer 1	ArrayInt32
eeLogUserDB2Fail	Failure info for the database EEPROM buffer 2	ArrayInt32
eepromWriteInProgress	1 if an EEPROM write is in progress, 0 otherwise	Int32
estimatedPk2PkYawError	estimate the Peak-to-Peak yaw error based on the infield calibration results (degrees)	Float32
estimatedRMSYawError	Estimated Root Mean Square yaw error	Float32
f_accelCalTemperatureRaw	Temperature at which the Rate Table temperature comps are computed for each axis (256*C)	ArrayInt32
f_accelFCM	Accelerometer sensor factory calibration matrix	MatrixFloat32
f_accelrange	Accelerometer scaling: 0=4g,1=8g,2=2g,3=16g	Ordinal
f_accelRawTC	User copy of raw accel at factory calibration base temperature	ArrayFloat32
f_accelTCM	accel sensor temperature compensation matrix	MatrixFloat32
f_calPointsNumToAvg	Number of samples to be averaged for a 3D/2D Calibration Mode capture point	Int32

f_gyroCalTemperatureRaw	Temperature at which the Rate Table temperature comps are computed for each axis (256*C)	ArrayInt32
f_gyroFCM	Gyro sensor factory calibration matrix	MatrixFloat32
f_gyroRawTC	User copy of raw gyro at factory calibration base temperature	ArrayFloat32
f_gyroTCM	gyro sensor temperature compensation matrix	MatrixFloat32
f_km0	Algorithm constant - magnetic gain	Float32
f_magCalTemperatureRaw	Temperature at which the Dycome temperature comps are computed for each mag X,Y,Z (256*C)	ArrayInt32
f_magFCM	Mag sensor factory calibration matrix	MatrixFloat32
f_magRawTC	User copy of raw mag at factory calibration base temperature	ArrayFloat32
f_magTCM	User copy of mag sensor temperature compensation matrix	MatrixFloat32
gyroBias	Real-time bias correction applied to gyrop to get gyropCorrected	ArrayFloat32
gyroBiasEstVariance	Estimated variance of the gyro bias	Float32
gyroBiasEstVarianceInit	Initial value of the estimated variance of the gyro bias	Float32
gyroBiasStability	Gyro Bias Stability in rads/sec	Float32
gyroBiasThermalStability	Gyro Bias Thermal Stability in rad/sec/DegC	Float32
gyroCalTemperatureRaw	Temperature at which the Rate Table temperature comps are computed for each axis (256*C)	ArrayInt32
gyroClearStatistics	Command to clear the gyro statistics (1=ON, 0=OFF)	Ordinal
gyroCollectingStatistics	Collect data for gyro statistics (1=ON, 0=OFF)	Ordinal
gyroConditioned	Factory calibrated gyro X,Y,Z array	ArrayFloat32
gyroCrossAxisCorrectionX	Calibration cross axis correction for the gyro X value (unused)	ArrayFloat32
gyroCrossAxisCorrectionY	Calibration cross axis correction for the gyro Y value (unused)	ArrayFloat32
gyroCrossAxisCorrectionZ	Calibration cross axis correction for the gyro Z value (unused)	ArrayFloat32
gyroDecimated	Averaged raw gyror based on accelGyroDecimator	ArrayInt32
gyroEstimatedNSamples	Estimated number of samples to reach gyroTargetVariance	ArrayInt32
gyroFCM	Gyro sensor factory calibration matrix	MatrixFloat32
gyroFieldCalX	In field calibration offset and scaler for the gyro X value (unused)	ArrayFloat32
gyroFieldCalY	In field calibration offset and scaler for the gyro Y value (unused)	ArrayFloat32
gyroFieldCalZ	In field calibration offset and scaler for the gyro Z value (unused)	ArrayFloat32
gyroMean	Average raw gyro since gyroCollectingStatistics and/or gyroClearStatistics	ArrayInt32

gyroNSamples	Number of gyro samples collected for statistical computations	Int32
gyrop	Processed gyro X,Y,Z array (radians per second), unless AHRS8_Compatible is 1, then radians per sample	ArrayFloat32
gyropBiasCorrected	gyrop with gyroBias correction	ArrayFloat32
gyropClearStatistics	Command to clear the gyrop statistics (1=ON, 0=OFF)	Ordinal
gyropCollectingStatistics	Collect data for gyrop statistics (1=ON, 0=OFF)	Ordinal
gyropCorrected	gyropBiasCorrected after gyroThreshold condition	ArrayFloat32
gyropEstimatedNSamples	Estimated number of samples to reach gyropTargetVariance	ArrayInt32
gyropMean	Average gyrop since gyropCollectingStatistics and/or gyropClearStatistics	ArrayFloat32
gyropNSamples	Number of gyrop samples collected for statistical computations	Int32
gyropScaled	User scaled Gyro X,Y,Z array	ArrayFloat32
gyropScaleFactor	Multiplied by gyrop to get gyropScaled (change units)	Float32
gyropStatisticsInterval	Number of gyrop samples between statistical result computations	Int32
gyropTargetVariance	Desired variance for the gyrops used in computing gyropEstimatedNSamples	Float32
gyropTemperatureMean	Temperature average while the gyrop statistics were being collected	Int32
gyropTimeStamp	Gyro time stamp taken after the gyrop data was computed	Int32
gyropVarianceMean	Average variance of gyrop data since gyropCollectingStatistics and/or gyropClearStatistics	Float32
gyror	Raw gyro X,Y,Z array	ArrayInt32
gyroRawTC	User copy of raw gyro at factory calibration base temperature	ArrayFloat32
gyroRawTCAdjustment	Temperature compensation vector computed using the gyroTCM to be added to the raw gyro vector	ArrayFloat32
gyroSampleRate	Sample rate of gyro data as observed during calibration	Float32
gyroSigma	Gyro stability factor	Float32
gyroStatisticsInterval	Number of gyro samples between statistical result computations	Int32
gyroTargetVariance	Desired variance for the gyros used in computing gyroEstimatedNSamples	Float32
gyroTCM	gyro sensor temperature compensation matrix	MatrixFloat32
gyroTemperatureMean	Temperature average while the gyro statistics were being collected	Int32
gyroThreshold	gyropCorrected is zeroed under this threshold	Float32
gyroTimeStamp	Time stamp taken at the arrival of the last gyro data item being averaged	Int32

gyroVariance	Initial value for the variance of the gyro values	Float32
gyroVarianceMean	Average variance of gyro data since gyroCollectingStatistics and/or gyroClearStatistics	ArrayFloat32
infieldCalMu	Infield cal magBias configuration parameter	Float32
infieldCalMu2	Infield cal magCrossAxisCorrection configuration parameter	Float32
infieldCalMu3	In-field calibration factor controlling the off-axis terms	Float32
infieldCalMu3_2D	In-field calibration factor controlling the off-axis terms for 2D (unused)	Float32
initCalPointBuffer	User command to initialize the in-field cal point buffer	Ordinal
initCalPointBufferRequest	Clears the field cal point buffers	Ordinal
InvokeGyroOffsetCal	Command the gyro offset calibration to start (unused)	Ordinal
InvokeTare	Command to update the boresight matrix based on current orientation	Ordinal
Ka	Algorithm accel factor (unused)	Float32
kaccelpAveraged		Float32
kdeviation	Algorithm constant - gain for single-pole IIR filter	Float32
kgyroOffsetCalFactor	Gyro offset field cal factor (unused)	Float32
km0	Algorithm constant - magnetic gain	Float32
kmag	Factor for estimated mag	Float32
Kw	Algorithm gyro factor (unused)	Float32
lat	Latitude	Float32
lonG	Longitude	Float32
lowpower	Low Power Mode	Ordinal
magAiding	User can clear this to ignore the mag data during AdaptNav computations	Ordinal
magAverageWindow	Number of samples which will be averaged during point selection	Int32
magBias	In-field calibration offset for the mags	ArrayFloat32
magBridgeBias	For Mode 2 only, adjustment to sensor raw data	ArrayInt32
magBufferActive	Provides the index into the calPointBuffer	ArrayInt32
magBufferActiveIndex	Current index into cal point buffer	Int32
magBW	Bandwidth (measurement time): 0=7.92 mS, 1=4.08 mS	Ordinal
magCalExpDecay	Current field cal decay factor	Float32
magCalExpDecayRate	Rate of field cal decay	Float32
magCalPoints	Array with up to calNumPoints of mag field cal captured points	ArrayFloat32
magCalTemperatureRaw	Temperature at which the Dycome temperature comps are computed for each mag X,Y,Z (256*C)	ArrayInt32
magClearStatistics	Command to clear the raw magnetometer statistics (1=ON, 0=OFF)	Ordinal
magCollectingStatistics	Collect data for magnetometer statistics (1=ON, 0=OFF)	Ordinal

magConditioned	Factory calibrated magnetometer X,Y,Z array	ArrayFloat32
magConversionDelayuS	The delay between the start mag conversion and the data read (uSecs)	Int32
magCrossAxisCorrection	Corrects the mag for mounting imperfections and in-field calibration	MatrixFloat32
magCrossAxisCorrectionX	Calibration cross axis correction for the magnetometer X value	ArrayFloat32
magCrossAxisCorrectionY	Calibration cross axis correction for the magnetometer Y value	ArrayFloat32
magCrossAxisCorrectionZ	Calibration cross axis correction for the magnetometer Z value	ArrayFloat32
magDecimated	Averaged magr based on magDecimator	ArrayInt32
magDecimator	magr is averaged every magDecimator number of raw samples	Int32
magErr	Magnetic error term resulting from calibration calcs	Float32
magErr2	Magnetic heading error estimate	Float32
magEst	Estimated magnetometer unit vector	ArrayFloat32
magEstimatedNSamples	Estimated number of samples to reach magTargetVariance	ArrayInt32
magEstVariance	Estimated variance of the mag values	Float32
magFCM	Mag sensor factory calibration matrix	MatrixFloat32
magFieldCalErr	Estimated Field Cal Error in Deg due to variability in buffered magnetics	Float32
magFieldCalTot	Average of magnetic field calibration point vector magnitudes	Float32
magFieldCalX	In field calibration offset and scalers for the magnetometer X value	ArrayFloat32
magFieldCalXYZ	Realtime mag offsets computed during field cal convergence phase (unused)	ArrayFloat32
magFieldCalY	In field calibration offset and scalers for the magnetometer Y value	ArrayFloat32
magFieldCalZ	In field calibration offset and scalers for the magnetometer Z value	ArrayFloat32
magHorzFieldCalTot	Average of the horizontal component of the magnetic field calibration points	Float32
magInstantaneousVariance		Float32
magLinearized	Calibrated mag before field cal and cross-axis correction (unused)	ArrayFloat32
magMean	Average raw mag since magCollectingStatistics and/or magClearStatistics	ArrayInt32
magMode	0=no auto setReset, 1= set or reset, 2= alternating set reset with bridge bias	Ordinal

magNSamples	Number of mag samples collected for statistical computations	Int32
magp	Processed Magnetometer X,Y,Z array (milli-Gauss)	ArrayFloat32
magP2PErr	Magnetic peak to peak error in field cal points	Float32
magpClearStatistics	Command to clear the magp statistics (1=ON, 0=OFF)	Ordinal
magpCollectingStatistics	Collect data for magp statistics (1=ON, 0=OFF)	Ordinal
magpCorrected	Processed Magnetometer X,Y,Z array	ArrayFloat32
magpEstimatedNSamples	Estimated number of samples to reach magpTargetVariance	ArrayInt32
magpMean	Average magp since magpCollectingStatistics and/or magpClearStatistics	ArrayFloat32
magpNSamples	Number of magp samples collected for statistical computations	Int32
magPointQF	Quality factor for the captured magnetic calibration points	Float32
magpScaled	User scaled Magnetometer X,Y,Z array	ArrayFloat32
magpScaleFactor	Multiplied by magp to get magpScaled (change units)	Float32
magpStatisticsInterval	Number of magp samples between statistical result computations	Int32
magpTargetVariance	Desired variance for the magps used in computing magpEstimatedNSamples	Float32
magpTemperatureMean	Temperature average while the magp statistics were being collected	Int32
magpVarianceMean	Average variance of magp data since magpCollectingStatistics and/or magpClearStatistics	ArrayFloat32
magr	Raw Magnetometer X,Y,Z,Total array	ArrayInt32
magRawTC	User copy of raw mag at factory calibration base temperature	ArrayFloat32
magRawTCAdjustment	Temperature compensation vector computed using the magTCM to be added to the raw mag vector	ArrayFloat32
magRealTimeVariance		Float32
magResetToSetMs	Controls the time from a Reset to a Set in milliseconds(Mode 2)	Int32
magSetResetDelayuS	The delay between a set or reset and a refill cap (uSecs)	Int32
magSetResetModulus	Controls the Set or Reset rate relative to number of samples (Modes 1)	Int32
magSetResetSetting	For Mode 1 only, selects whether the Set or Reset is used	Ordinal
magSetToResetMs	Controls the time from a Set to a Reset in milliseconds(Mode 2)	Int32
magStatisticsInterval	Number of mag samples between statistical result computations	Int32
magTargetVariance	Desired variance for the mags used in computing magEstimatedNSamples	Float32

magTCM	User copy of mag sensor temperature compensation matrix	MatrixFloat32
magTemperatureMean	Temperature average while the raw mag statistics were being collected	Int32
magTimestamp	Time of mag sensor data acquisition	Int32
magvar	Magnetic Variation - difference between true and magnetic	Float32
magVariance	Initial value for the variance of the mag values	Float32
magVarianceMean	Average variance of mag data since magCollectingStatistics and/or magClearStatistics	ArrayFloat32
manualCalibrationPoint	User command to capture a cal point	Ordinal
maxRFSRevision	Puts an upper limit on the RFS revision	Int32
mEstimate	Gyro advanced magnetometer unit vector (unused)	ArrayFloat32
minCalPoints	Number of points where the field calibration starts	Int32
name	Device Name	char
nmeaecho	Echoes NMEA characters	Ordinal
nmeaignorechecksum	Ignore received NMEA checksum	Ordinal
orientation	Physical orientation of compass relative to the host. This variable changes the boresightMatrix variable. Set this variable to 0 when changing the boresightMatrix by any other method to avoid interference between the settings.	Ordinal
pitch	Pitch angle in degrees, positive up	Float32
pitchClearStatistics	Command to clear the pitch statistics (1=ON, 0=OFF)	Ordinal
pitchEstimatedNSamples	Estimated number of samples to reach pitchTargetVariance	Int32
pitchMean	Average pitch since compassCollectingStatistics and/or compassClearStatistics	Float32
pitchNSamples	Number of pitch samples collected for statistical computations	Int32
pitchScaled	pitch * pitchScaleFactor	Float32
pitchScaleFactor	Multiplied by pitch to get pitchScaled (change units)	Float32
pitchStatisticsInterval	Number of pitch samples between statistical result computations	Int32
pitchTargetVariance	Desired variance for pitch used in computing pitchEstimatedNSamples	Float32
pitchTemperatureMean	Temperature average while the pitch statistics were being collected	Int32
pitchVarianceMean	Average variance of pitch data since compassCollectingStatistics and/or pitchClearStatistics	Float32
pMatrix	Platform rotation matrix (orientation of unit)	MatrixFloat32
pMatrix0	Platform Matrix Row 0	ArrayFloat32
pMatrix1	Platform Matrix Row 1	ArrayFloat32

pMatrix2	Platform Matrix Row 2	ArrayFloat32
pointCalTermination	Commands the cal mode after a manual point calibration (0=off, 1=point, 2=auto)	Ordinal
pointFieldCalActive	This is automatically set when a command is issued to collect a manual cal point	Ordinal
pointMagBias	Backup mag bias for point calibration	ArrayFloat32
pointMagCrossAxisCorrection	Backup of magCrossAxisCorrection for point calibration	MatrixFloat32
position	(RFS only) User modifiable composite object with Roll,Pitch,Yaw,Lat,Lon,Alt,Temp	BitFieldElementType
positionDimension	Number of elements in array to follow	Byte
positionrate	RFS “position” variable report period (milliseconds)	Int32
positionRateSwitch	Allows user selection of the trigger for the RFS Bitfield position variable	Ordinal
possibleMagPointQF	Possible quality factor if the current magnetic point is added to the captured calibration points	Float32
powerMode	Controls the power mode settings	Ordinal
printFieldCalIndexes	Streams the infield cal indexes	Ordinal
printmask	Items that get printed	Int32
printmodulus	How many counts of the print trigger cause the print	Int32
printtrigger	Event that causes sampled data to print	Int32
protectedAreaWriteEnable	Enables a write to the factory settings on the next attempt	Ordinal
quaternion	alternate attitude representation, components are W, X, Y, Z	ArrayFloat32
quiet	Debug Suppress	Ordinal
relCalBackupTime	Relative time since last backup of calibration parameters	Float32
resetGyroBias	Set this to 1 to set gyroBias to zero	Ordinal
resetMagEstVariance	Set this to 1 to restore magEstVariance back to its default value	Ordinal
restartCompassCalcs	Command to start compass calcs fresh (forget previous expected position)	Ordinal
restorefactorycal	Command to restore factory cal to user values	Ordinal
restoreFieldCal	Set Magnetic field calibration back to factory default	Ordinal
revertFieldCal	Command variable used to override any AdaptCal field cal data with the most recent manual point cal data. Note that Version 3.3.2 has a bug where this variable does not automatically clear, so use clearFieldCal instead or the user should set revertFieldCal back to 0.	Ordinal
REVISION	Subversion revision number of this software	char
RFSVersion	Highest RFS version that this software supports	Int32
roll	Roll angle in degrees, positive rightside down	Float32
rollClearStatistics	Command to clear the roll statistics (1=ON, 0=OFF)	Ordinal

rollEstimatedNSamples	Estimated number of samples to reach rollTargetVariance	Int32
rollMean	Average roll since compassCollectingStatistics and/or compassClearStatistics	Float32
rollNSamples	Number of roll samples collected for statistical computations	Int32
rollScaled	roll * rollScaleFactor	Float32
rollScaleFactor	Multiplied by roll to get rollScaled (change units)	Float32
rollStatisticsInterval	Number of roll samples between statistical result computations	Int32
rollTargetVariance	Desired variance for roll used in computing rollEstimatedNSamples	Float32
rollTemperatureMean	Temperature average while the roll statistics were being collected	Int32
rollVarianceMean	Average variance of roll data since compassCollectingStatistics and/or rollClearStatistics	Float32
roomTempOffset	Value of the raw temperature sensor at 25 degC	Int32
saveAll	Force database to save all values, even if default	Ordinal
sdotpmode	Mode for s.p output (unused)	Ordinal
senPwrCntrlToggleTime	On start-up, toggle the sensors' power for this many milliseconds	Int32
sensorConditionedDataReady	Event that indicates that the conditioned data is ready	Ordinal
sensorDecimatedDataReady	Event that triggers the sensor data conditioning	Ordinal
sensorsRunning	Controls whether the sensors are enabled or not, for reset or so cpu clock can be changed	Ordinal
serialnumber	Serial number for this navigation unit	char
set_wmm	Command to update the World Magnetic Model data, uses lat,long,alt,date	Ordinal
sphericalCalCoverage	Reports spherical coverage of buffered magnetic measurements as a percent	Float32
sphericalPoints	Idealized points used in adaptcal	MatrixFloat32
sphericalPointSeparationLimit	Used in selecting adaptcal points	Float32
startUpCount	Number of power up and reset start ups	Int32
temperature	Temperature - Celcius	Float32
timeStamp	Time stamp taking at beginning of compass calculations	Int32
usertofactorycal	Command to put user cal in factory cal	Ordinal
VERSION	Subversion tagged ID for this software	char
VERSION_M4	Subversion tagged ID for the M4 micro software	char
vidCount	Variable Count	Int32
WMMepoch	Base year for the World Magnetic Model coefficients being used	Float32
wmmGD	World Magnetic Model geodetic magnetic field X, Y, Z at Lat/Long	ArrayFloat32

wmmGDMagnitude	Magnitude of the World Magnetic Model computed Earth Field	Float32
wmmHasRun	World Magnetic Model has been run (wmmGD is valid)	Ordinal
yaw	Magnetic heading in degrees, positive East	Float32
yawClearStatistics	Command to clear the yaw statistics (1=ON, 0=OFF)	Ordinal
yawErrEst	Estimated yaw error	Float32
yawEstimatedNSamples	Estimated number of samples to reach yawTargetVariance	Int32
yawMean	Average yaw since compassCollectingStatistics and/or compassClearStatistics	Float32
yawNSamples	Number of yaw samples collected for statistical computations	Int32
yawScaled	yaw * yawScaleFactor	Float32
yawScaleFactor	Multiplied by yaw to get yawScaled (change units)	Float32
yawStatisticsInterval	Number of yaw samples between statistical result computations	Int32
yawt	True magnetic heading in degrees, positive East	Float32
yawTargetVariance	Desired variance for yaw used in computing yawEstimatedNSamples	Float32
yawTemperatureMean	Temperature average while the yaw statistics were being collected	Int32
yawtScaled	yawt * yawtScaleFactor	Float32
yawtScaleFactor	Multiplied by yawt to get yawtScaled (change units)	Float32
yawVarianceMean	Average variance of yaw data since compassCollectingStatistics and/or yawClearStatistics	Float32

2.4 World Magnetic Model for True Heading

The inertial systems contain a World Magnetic Model (geodetic coordinate system referenced to the WGS 84 ellipsoid) which is needed to determine the magnetic variation (difference between true heading and magnetic heading). This model requires the user to input a physical location and time (lat, lonG, alt and day[1]) and cause the compass to compute the magnetic variation (aka declination).

The world magnetic model is valid for 5 years at which point it must be updated. *NOTE: TO RETAIN MAGNETIC VARIATION ACCURACY, THE MAGNETIC MODEL MUST BE UPDATED EVERY FIVE YEARS ON THE NOAA SCHEDULE (see http://www.ngdc.noaa.gov/geomag/WMM/DoDWMM.shtml). THIS ONLY AFFECTS THE CALCULATION OF TRUE HEADING (yawt) AND DOES NOT AFFECT MAGNETIC HEADING ACCURACY (yaw).*

Users who have a product lifetime expectation that crosses a 5 year boundary (2020, 2025, …), and that need true heading output should plan to be able to update the WMM data. Hardware designers should

[1] These are NorthTek variable names, hence the odd capitalization.

refer to the reference design to ensure that the proper software interface is available to the compass for performing this update.

2.4.1 Invoking the WMM

This model is invoked by the user (using the NMEA $PSPA,AUTOVAR or by setting set_wmm using NorthTek or RFS) and should be rerun with updated position and date data whenever they change enough to affect the resulting magnetic variation (magvar). NOTE: Resetting the magnetic parameters to the factory default does not affect the magnetic variation information. To clear out the magnetic variation, manually set magvar to zero.

Here is a NorthTek example script:

```
lat  f29.13  set drop  // latitude - deg
lonG f-81.33 set drop  // longitude - deg
alt  f20.0   set drop  // altitude - meters
day  f2011.4 set drop  // date - year and 1/tenth year
set_wmm 1 set drop     // run the WMM
100 delay              // give it time to run
wmmHasRun di.    // magvar and wmmGD are set by WMM
magvar di.       // delta between magnetic North and true North
wmmGD di.        // earth magnetic vector at the given location
yawt di.         // true north
yaw di.          // magnetic north
```

2.4.2 WMM Model Data Update Procedure

The WMM model data is stored in flash and can be updated using NorthTek commands. The following NorthTek words (commands) provide the RAM addresses of specific WMM data items which can be used to load the WMM model data. They are:

- &numcoeff
- &gnm
- &hnm
- &dhnm
- &epoch
- &radius
- &major
- &minor

The NorthTek word “wmmcrc!” can then be used to compute the CRC for the WMM data and the NorthTek word “wmm!” should be used to store the WMM data from RAM to flash. The NorthTek word “wmm.” will display the data with elements of the arrays gnm, hnm, dgnm and dhnm displayed across the unlabeled lines (in the order listed).

Sparton will provide a script to load the WMM using the above NorthTek commands.

2.5 Mount Orientation Accommodation

The inertial system's orientation output (pitch/roll/yaw, pMatrix and quaternion) can be adjusted to be that of the host application by changing the "boresightMatrix". There are a number of ways this can be accomplished. When using any option other than using the "orientation" variable, make sure that orientation is set to 0 to avoid conflict. The following is a list summarizing the options:

- Set the orientation variable to one of the predefined 90-degree orientations. See the Sparton Navigation Sensor Product Guide available in the web site downloads for physical orientation diagrams. The options for the orientation variable are:
  - 0 = horizontal
  - 1 = vertical
  - 2 = left edge
  - 3 = right edge
  - 4 = inverted
- Orient the host device towards magnetic North in a level condition and set the InvokeTare command to 1 using NorthTek, NMEA $PSRFS or RFS.
- Use the mechanical design of the host device to determine the boresight matrix and set the boresightMatrix directly using NorthTek, NMEA $PSRFS or RFS.
- Run a NorthTek tare script (see Application Note AN 1003 and a sample script is in the downloads section on the website).

2.6 DC-4E, GEDC-6E and AHRS-8 Compatibility

Starting in version 3.0.0, the AHRS-M1/M2 is largely compatible with the DC-4E, GEDC-6E and AHRS-8 as far as customers are concerned. For short-hand, we call this AHRS-8 compatibility.

Many of the AHRS-8 variables and commands were added with a focus on adding those that are most commonly used by customers. This includes the Print Trigger and all of the *.p streaming (such as "1 accelp.p") and *_mask streaming (such as "accelp_mask d.on"). The cal3D (field calibration) NorthTek script for the AHRS-8 will work on AHRS-M1/M2. The AHRS-8 NMEA commands used for field calibration are also supported. See the table at the end of this document for a list of the new AHRS-8 compatible variables.

2.6.1 AHRS8_Compatible Flag

Since the AHRS-8 compatibility requirement was added after the AHRS-M2 was already in use, some variables and commands with the same name already had different formats, so changing either one would affect current users.

To deal with these differences, a new variable named **AHRS8_Compatible** was created to select one form or the other for these cases. Most of the AHRS-8 features that were ported to the AHRS-M1/M2 are independent of this flag. The cases where the flag takes affect are detailed in the following table.

The **AHRS8_Compatible** flag defaults to 0, is user settable, and maintains its setting across power cycles

Command or Variable Name	**AHRS8_Compatible = 1**	**AHRS8_Compatible = 0**
magp_mask output format	magp,time,magpX,magpY,magpZ	MP:time,magrX,magrY,magrZ,magpX,magpY, magpZ,yaw
compass_mask output format	2 decimal places are printed	3 decimal places are printed
gyrop	rads/sample	rads/sec
magp	magp is corrected by the field calibration (same as magpCorrected)	not corrected by field calibration (use magpCorrected as a field calibration corrected value)
centripetalCompensation verses centripetalCorrection	Use centripetalCorrection	Use centripetalCompensation

.

2.6.2 Start-Up Output by Default

In order to be AHRS-8 compatible, as of 3.0.0, the AHRS-M2 does not output any bytes upon start-up unless programmed to by a User Boot Progam. The AHRS-M1, however, still outputs one “OK”.

2.6.3 Streaming Commands - *.p

Unlike the AHRS-M1 and AHRS-M2 prior to Version 3.0.0, the AHRS-M1/M2 fully supports the commands listed in the section named Streaming Sensor Data Operations in the NorthTek Programming Manual. These are commands such as accel.p, accelp.p, compass.p, etc. For example, to set the output rate to be every 50 ms (default is 10 ms) and to turn on streaming of the compass data, command the following:

```
5 compass.mod  // output every 5*10ms = 50 ms
1 compass.p    // turn on streaming
…
0 compass.p  // turn off streaming
```

2.6.4 Streaming Commands - *_mask

See the section titled “Mask Commands for Streaming Output - *_mask” for the AHRS-8 compatible _mask commands that were added in version 3.0.0.

2.6.5 AHRS-8 Database Variable Compatibility

All AHRS-8 database variables that were considered to be of user interest (excludes factory calibration variables and many configuration variables) were added in version 3.0.0.

3 Protocols

All the serial port protocols co-exist on the same serial port at all times. The user may use whichever protocol is most desirable.

The factory default baud rate is 115200 baud but may be changed to any of the allowable baud rates. Note that the same baud rate applies to all protocols simultaneously. Once changed, the inertial system will maintain the new baud rate setting through a power cycle.

To change the baud rate to 9600 for example using NorthTek™, enter:

```
baud 4 set drop<cr>
```

The "drop" removes the result status of the "set" command off of the NorthTek™ stack. So, every "set" should be followed by a "drop" to prevent stack overflow.

3.1 NorthTek™

NorthTek™ is not a protocol per se. NorthTek is a programming language, a command interpreter, and an execution environment. NorthTek is described in detail in the NorthTek Programming Manual. NorthTek™ provides the user with a command line interface allowing direct interaction with the inertial system. NorthTek™ also allows the user to load custom programs into the inertial system that will execute a custom user application. The user may create programs that cause some of the standard protocol outputs but at user defined points or may create custom output depending on the specific need. The user specific algorithms may be used to filter the output data, create specific output formats and control the reporting rate.

Because NorthTek™ is an environment unto itself; an entire manual is dedicated to the NorthTek System as listed in the references. The user should refer to that manual for detailed descriptions of the commands being used in this manual.

Note that NorthTek™ script lines should be limited to 80 characters to prevent an internal buffer overflow. Carriage returns and line feeds are ignored as white space and so can be used to limit line lengths.

NorthTek™ provides a command line access to the internal database variables. NorthTek also provides raw and processed sensor data streamed at the acquisition rate. The sections that follow illustrate some of the NorthTek™ functionality that can be used on the inertial system. The reader should refer to the *NorthTek™ Programming Manual* for detailed descriptions on syntax and semantics of the commands used in the examples that follow.

3.1.1 Stack Based Language

NorthTek is a stack-based language where any number entered is pushed on the operand stack. Any operators operate on the items on the operand stack and results are pushed onto the stack. The stack can be viewed using the ".s" command.

Here is a simple example:
// place '2' on the stack and display

```
2OK
.s
Stack (top down)
[@Depth 0]    2                          (0x00000002)
OK
// '.' prints the integer value on the top of the stack
. 2OK
.s
Stack (top down)
OK
// add 2 + 3
2 3 + OK
.s
Stack (top down)
[@Depth 0]    5                          (0x00000005)
OK
// '.' prints the integer value on the top of the stack
. 5OK

// add 2.0 + 3.0 in floating point with f+
f2.0 f3.0OK
.s
Stack (top down)
[@Depth 0]    1077936128                 (0x40400000)
[@Depth 1]    1073741824                 (0x40000000)
OK
f+OK
.s
Stack (top down)
[@Depth 0]    1084227584                 (0x40A00000)
OK
// 'f.' prints the floating-point value on the top of the stack
f.5.000000OK
.s
Stack (top down)
OK
```

3.1.2 Displaying Database Elements

The internal database is available to the NorthTek™ command line. See Appendix A for a list of the database elements. All database elements may be printed to the output using the NorthTek™ command "di.". This command expects a reference to a data element on the stack, and then prints out the current value in a human readable form. The syntax is the name of the data element (case sensitive) followed by a space followed by "di." (also case sensitive). Some specific values are typically more interesting than others to the user.

Note that help is available for the database variables. The NorthTekSystem (NTS) outputs a longer, more informative string for the variable when

help <variable>

is entered.

Example:

```
help gyrop
Processed gyro X,Y,Z array
OK
```

Some of the common values are:

```
gyrop di.
gyrop=
00-- -5.210986e-03
01-- -7.198157e-04
02-- -2.699813e-02

OK

magp di.
magp =
00-- -295.907318
01-- -306.281860
02—445.833679

OK
```

This example also shows that multiple commands can be entered on a single input line (up to 80 characters).

```
magp di. gyrop di.
magp =
00-- -295.907318
01-- -306.281860
02— 445.833679
```

```
gyrop=
00-- -5.210986e-03
01-- -7.198157e-04
02-- -2.699813e-02

OK
```

3.1.3 Outputting values in User Defined Format

The user can control the output formatting if desired. The example below shows how to first print the default formatted output for a variable, and then print the output in a fixed format (equivalent to %8.3f in "C"). At any point, a ".s" command can be used to see what is on the stack

```
accelp di.
accelp =
00-- 140.380493
01-- 41.137589
02-- 972.043396

7 2 accelp di@  // read the address of accelp X value
8 +             // pointer to the Z value (add 8 bytes)
                // read and print the Z in %7.2f format
@ ff. 969.97OK



8       // total characters in output field
3       // digits to right of decimal point
gyrop   // load pointer to database object
di@     // load the address of the object's value
@       // read the value at that address
// print that value in the %8.3f C format, justs prints X
ff. 0.006OK
```

3.1.4 Reset

The NorthTek™ "reset" command will invoke a reset of the processor. Allow a few seconds for the processor to come back up. All variables will be reset to their default value or read from non-volatile memory if the variable is marked "permanent" and has been changed from the default. Any NorthTek™ script which has been loaded (except for a user boot program) will be lost. Note that if an EEPROM update is in progress, then the reset will be delayed until the update completes. Also consider that the power up sequence takes time to reload configuration and calibration data from EEPROM, so the reset command has a delay that is dependent on the amount of data in EEPROM.

3.1.5 Canceling NorthTek Programs

A <CTRL-C> will cancel a NorthTek™ program.

3.1.6 Inhibit NorthTek™ Echoed Characters and "OK"/"Huh" Responses

NorthTek™ responses (echoed characters, OK, Huh) can be inhibited on the user port. The requested data will still be returned. Use "0 echo!" to turn off responses (they are on by default). Use "1 echo!" to turn them back on.

3.1.7 Display Ordinal Options

For ordinal type variables, you can display their option choices using the command "di.choice". For each option, its text name is displayed plus the numeric value that must be used to select that option is in brackets.

<variableName> di.choice

Example:

baud di.choice

baud di.choice
Alt[0]
1200[1]
2400[2]
4800[3]
9600[4]
19200[5]
38400[6]
57600[7]
115200[8]
OK

3.1.8 Setting Database Elements – Integers and Ordinals

Integers and Ordinals can be set using the following syntax. Note that ordinals are set with an integer but display with the ordinal name. The "drop" removes the status result that is left on the stack after the "set" command. If the "drop" command is not used, then the stack may overflow.

<variableName> <integerValue> set drop

Example:

clearFieldCal 1 set drop

clearFieldCal di.
// note that the firmware clears this particular variable when the action is complete
`clearFieldCal = ClearDone`

OK

Setting Database Elements – Floats

Floating point values can be set using the following syntax:

<variableName> F<floatValue> set drop

Example:

gyropScaleFactor F1000.0 set drop // the F is not case sensitive

gyropScaleFactor di.

gyropScaleFactor = 1000.000000

OK

3.1.9 Setting Database Elements – Arrays

Arrays can be set using the following syntax:

```
<variableName>  array[ <startIndex> <endIndex> <sequence of numbers> ]array set drop
```

Example:

// clear the entire array

chan0Enables array[0 15 0 0 0 0 0 0 0 0 0 0 0 0 0 0]array set drop

// set elements 2, 3 and 4 in the array

chan0Enables array[2 4 1111 2222 3333]array set drop

chan0Enables di.

chan0Enables =
00-- 0
01-- 0
02-- 1111
03-- 2222
04-- 3333
05-- 0
06-- 0
07-- 0
08-- 0
09-- 0
10-- 0
11-- 0
12-- 0
13-- 0
14-- 0
15-- 0

3.1.10 Setting Database Elements – Matrices

Matrices can be set using the following syntax:

<variableName> m[<startRowIndex> <startColIndex> <endRowIndex> <endColIndex> <sequence of numbers>]m set drop

In the example below, <boresightMatrix> is a 3x3 matrix array. So, the user values will be set in the array elements in the following order :
[0,0][0,1][0,2][1,0][1,1][1,2] [2,0][2,1][2,2]

Example:

boresightMatrix
m[decimal
0 0 2 2
f1.000000e+00 f0.000000e+00 f0.000000e+00
f0.000000e+00 f1.000000e+00 f0.000000e+00
f0.000000e+00 f0.000000e+00 f1.000000e+00
*]m set drop***OK**

boresightMatrix di.

```
boresightMatrix =
 1.000000e+00 0.000000e+00 0.000000e+00
 0.000000e+00 1.000000e+00 0.000000e+00
 0.000000e+00 0.000000e+00 1.000000e+00

OK
```

3.1.11 Mask Commands for Streaming Output - *_mask

There are a few commands that will enable the user to get quick and easy output data from the compass. We call these the mask commands.

- compass_mask - Data is time (ms), pitch (deg), roll (deg), yaw (deg) – update rate can be modified using compass.mod, post 3.0.0 – if the AHRS8_Compatible variable is 1 then 2 decimal places are printed and 3 decimal places are printed if AHRS8_Compatible is 0.
- magp_mask - Data is time (ms), magp X, Y, Z (mGauss), post 3.0.0 – if AHRS8_Compatible is 1, then the format is “magp,time,magpX,magpY,magpZ" and if AHRS8_Compatible is 0, then the format is "MP:time,magrX,magrY,magrZ,magpX,magpY,magpZ,yaw".
- accelGyrop_mask - Data is time (ms), accelp X, Y, Z (mg), gyrop X, Y, Z (rads/sec), temperature (degC)

- accel_mask (3.0.0 and later) – raw accel data – update rate can be modified using accel.mod
- accel.mod (3.0.0 and later) – modifies the output rate of accel_mask and accel.p where the user commands “N accel.mod” (N an integer > 0). Subsequently, the data is output every N samples of accel data.
- accelp_mask (3.0.0 and later) - Data is time (ms), accelp X, Y, Z (mg)
- gyro mask (3.0.0 and later) – raw gyro data – update rate can be modified using gyro.mod
- gyrop_mask (3.0.0 and later) – Data is time (ms), gyrop X, Y, Z
- mag_mask (3.0.0 and later) – raw mag data
- magp_mask (3.0.0 and later) – Data is time (ms), magp X, Y, Z (mG)
- quaternion_mask (3.0.0 and later) – w, x, y, z

To turn on mask data, command "xxx_mask d.on". In all above cases, use "d.off" instead of "d.on" to turn off the streaming data. Also, ctrl-s will suspend the output and ctrl-q will re-enable it.

Example:

```
25 compass.mod  // set to output 25*10ms = 250 ms (4 Hz)
compass_mask d.on  // turn on streaming
…
compass_mask .doff  // turn off streaming
```

3.1.12 Accessing an RFS VID Using NorthTek – dvid@

An RFS VID (Variable ID) can be accessed using the NorthTek™ command, dvid@. The syntax is “<variableName> dvid@ .” (note there is a space and dot at the end to display the result).

3.1.13 Variable Search Using grep

A string may be used to search for variable names containing that string using the NorthTek command “grep”. This search is case sensitive.

```
grep <searchString>
```

3.1.14 User Boot Program

The AHRS-M2 start-up configuration and output can be programmed by the user in the inertial system’s EEPROM. The AHRS-M1 lacks this capability. Upon start-up, the inertial system checks an EEPROM sector located at 0x10000 for a NorthTek™ script and will execute it if finds one. See Application Note AN1002 for a detailed example.

To load a script, enter:

```
0x10000 userOpen       // Opens the EEPROM section at 0x10000 for writing
: put start: userWrite ;  // Defines a “put” function to write a subsequent  string to EEPROM
```

Now take the NorthTek™ script that is to be run on start-up and insert the string "put " at the beginning of each line. Each "put" line will load the script line into EEPROM. **Make sure that the script works** before putting it into EEPROM or the inertial system could be rendered unusable! If the script fails on start-up, try <ctrl-c> to stop it. After entering the modified script, enter:

userClose // completes the EEPROM "file"

To try it out, cycle power or enter:

reset

To remove the script from EEPROM, enter:

0x10000 eesectorerase

Here is an example that will print the version number upon start-up:

```
0x10000 userOpen
: put start: userWrite ;
put VERSION di.
userClose
```

3.1.15 Custom Combination Print Streaming

The customizable print streaming commands allows selection of the data to be streamed and provides control of when the data is streamed. This capability is available in units DC-4E 4.2.1, GEDC-6E 4.0.2, AHRS-8 2.2.1 or later.

For this capability, there are three configuration variables: printmodulus, printtrigger and printmask plus the following set of constants:

magr_trigger
magp_trigger
accelr_trigger
accelp_trigger
gyror_trigger
gyrop_trigger
yaw_trigger
yawt_trigger
quat_trigger
temp_trigger
pitch_trigger
roll_trigger

time_trigger

Each of the above trigger definitions is a single bit. If a bit is set in the printmask, then when a print trigger event occurs, the data that is associated with that bit will print in the output. Note that, although the triggers occur in interrupt handlers or high priority tasks, the actual printing occurs in a low priority task. This may cause a small delay in the print from the point of the trigger but is necessary to avoid interfering with the data acquisition and computations.

The printmodulus indicates how many trigger events must take place before the printing occurs.

For example, suppose we want to print the raw and processed magnetometer data and the temperature and we want to print on every 100th processed magnetometer calculation. To accomplish this, we would set the print mask using this sequence of NorthTek:

```
>>> printmask
>>> magr_trigger magp_trigger or temp_trigger or set drop
```

The “drop” is needed because the “set” operator puts a status result on the top of the stack. If the “drop” is not used, then eventually, the stack may overflow.

To cause it to print on every 100th trigger event:

```
>>> printmodulus 100 set drop
```

The trigger condition is set

```
>>> printtrigger magp_trigger set drop
```

Note that this kind of printing is on by default, but can be disabled with printer_mask d.off or suspended with CTRL-S, or by setting the printtrigger to 0

```
>>> printtrigger 0 set drop
```

Note that we can cause the print to be output with the word "p.”. With the “p.” command, we can write our own trigger condition in NorthTek and then evoke this printout.

Given the above configuration, the output is:

```
P:,659539,mr,5890,-712,7323,mp,327.566681,94.818001,385.820099,T,33.06
P:,660481,mr,5968,-695,7342,mp,314.355835,93.428726,386.015045,T,33.06
P:,661423,mr,5850,-661,7316,mp,330.460510,93.706581,385.495178,T,33.06
P:,662365,mr,5995,-742,7337,mp,314.104187,93.914970,386.534943,T,33.00
```

```
P:,663306,mr,5965,-699,7328,mp,324.169617,93.150871,385.625153,T,33.00
P:,664248,mr,5902,-694,7337,mp,320.206360,92.803551,386.210022,T,33.00
P:,665190,mr,5900,-677,7326,mp,320.080536,91.622673,385.495178,T,33.00
```

The interpretation of this output is P:,TIME,mr(indicates mag raw),x,y,z,mp(indicates mag processed),x,y,z,T(temp),value.

The printmodulus for accelr and gyror is applied to the RAW sample rates, i.e., 400 Hz on gyro/accel, NOT on the divided down values.

This is what we get when we set printmask to all of the trigger constants:

```
P:,878979,mr,6174,-751,7403,mp,337.317566,96.762985,390.499084,ar,22,38,2052,ap,-
6.347656,-22.460938,1011.718750,gr,-8,-4,26,gp,0.000010,0.000039,-
0.000102,y,342.53,yt,342.53,q,0.988351,-0.009816,0.004613,-0.151807,T,32.19
P:,879920,mr,5985,-763,7416,mp,327.126343,95.165321,389.914185,ar,21,34,2049,ap,-
6.591797,-20.874023,1010.375977,gr,-3,-16,9,gp,-0.000003,0.000004,-
0.000050,y,342.53,yt,342.53,q,0.988343,-0.009981,0.004853,-0.151841,T,32.19
P:,880862,mr,6045,-735,7397,mp,334.172119,97.874405,391.148926,ar,21,35,2046,ap,-
6.835938,-21.118164,1009.033203,gr,6,-25,6,gp,-0.000035,-0.000023,-
0.000042,y,342.54,yt,342.54,q,0.988360,-0.010000,0.004931,-0.151726,T,32.19
```

The time_trigger is triggered by the 1msec clock tick, so a time-based output can be created if necessary.

Note that if too much data is specified to print for the baud rate, the inertial system will drop whole lines of output. The algorithm will not let another trigger occur until the previous output has completed. Dropping whole lines avoids the problem of erroneous data due to dropped digits.

Here is a sample script which is similar to gyrop.p but with temperature added:

```
printmodulus 4 set drop  // every 4th raw gyro
printmask          // variable that gets set in the next line
gyror_trigger gyrop_trigger or temp_trigger or set drop
printtrigger printmask set  drop
```

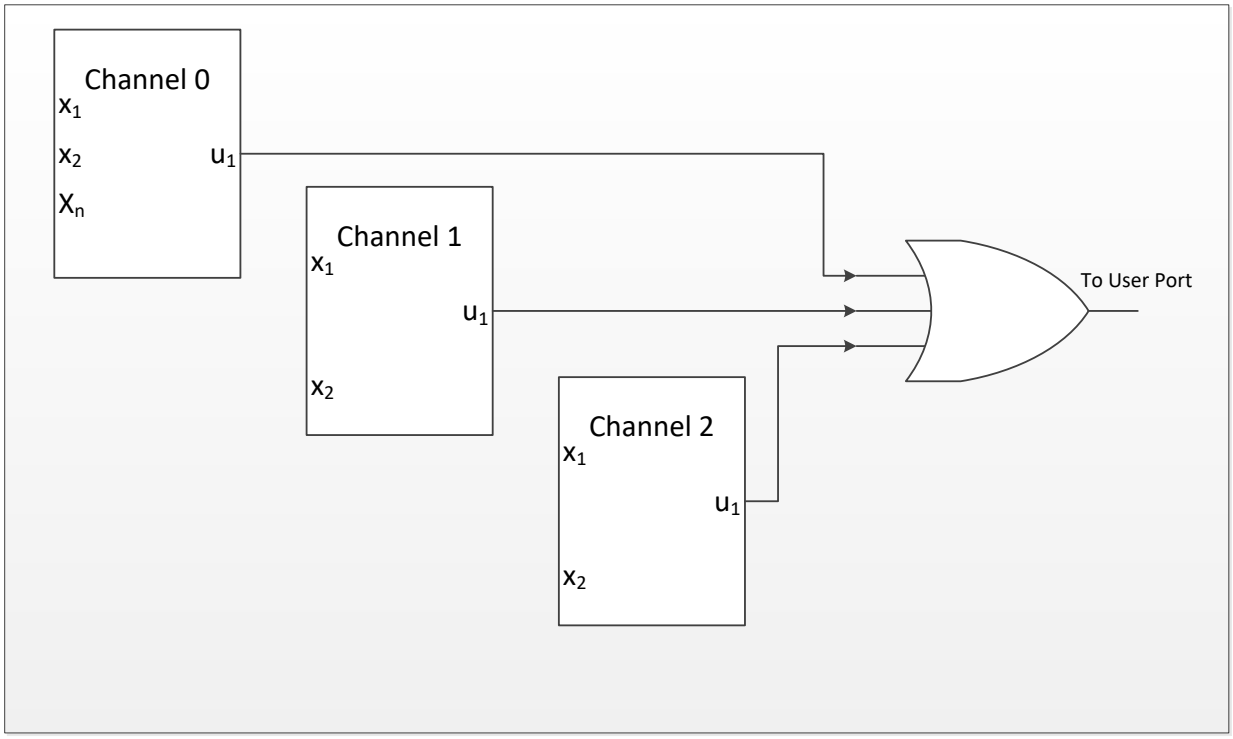
3.2 Protocol Triggers

Protocol Triggers are designed to give the user multiple, independent, output data messages at a repeating rate. The user may also obtain occasional data via query using any of the other protocols. These triggers provide automatically repeating outputs that can be synced with the internal computations thus providing the freshest possible data. The triggers may be used to periodically invoke

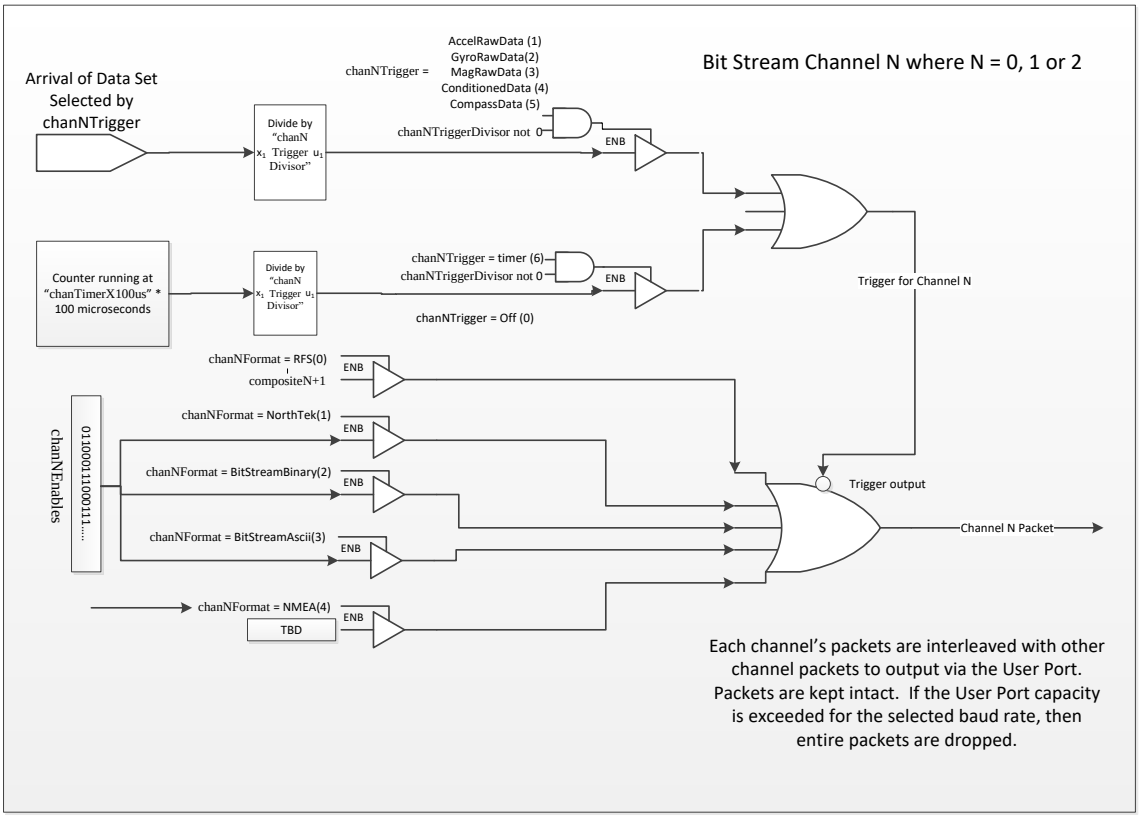
a NorthTek script, NMEA command, RFS composite message or a BitStream ACSII message or a BitStream Binary message.
See also the section named Custom Combination Print Streaming for the AHRS-8 compatible printtrigger method.

3.2.1 Triggered Serial Output Channels

There are three independent "streams" of user output as shown in the figure below via virtual channels. Zero, one, two or all three channels may be enabled. When triggered, packets of each channel will stream out out the user (serial) port according to the format (protocol) selected. Various formats may be multiplexed such as the SAPP framed binary protocols (RFS and BitStream Binary), text delimited protocols (BitStream ASCII and NMEA), and the free form protocol NorthTek.



All three channels use a separate trigger condition, a separate format and a separate user configured output data set. The figure below illustrates a channel. The user has great flexibility in configuring the trigger that causes a packet to be output and configuring the format and content of the packets.



°

3.2.1.1 Triggers

The output data can be synced with data availability with the use of a trigger. The trigger options are listed below. A trigger can be selected by setting the variable chanNTrigger (N=0..2) to a value from 0 to 6. The rate that the trigger occurs varies depending on its source (see list below). The output rate can be reduced by chanNTriggerDivisor (N=0..2). The chanNTrigger variable changes are saved in non-volatile memory.

For example, if chan0Trigger = 4 (ConditionedData), and the chan0TriggerDivisor = 10, then every 10th time the conditioned data is ready (100/10 = 10 Hz), then the selected data will be output via the user port at 10 Hz. Note that this data is not averaged so 9 sets of data will be dropped for every one that is output.

- Off[0] – for unused channels
- AccelRawData[1] -- (factory use) Triggers when accelerometer raw data is ready
- GyroRawData[2] -- (factory use) Triggers when gyro raw data is ready

- MagRawData[3] -- (factory use) Triggers when magnetometer raw data is ready
- ConditionedData[4] -- Triggers when the conditioned data is ready (magp, accelp, gyrop)
- CompassData[5] -- Triggers at the when the pitch, roll, yaw, quaternion, pMatrix data is ready
- Timer[6] – Triggers off a timer configured by the user using chanTimerX100us.

If you specify an amount of data that is more than the serial port can transmit at the chosen baud rate, the effective rate will be serial port limited and not be the actual rate you set. An overload of data will cause packets to be dropped to ensure the integrity of the packets that do get sent.

3.2.1.1.1 Accel Raw Data Trigger (1)

This trigger signals the computation of decimated raw accelerometer data. The actual updates of the data in this set varies depending on the data's individual update rate. The retrieval rate of the data set is defined by the variable named accelGyroDecimator which is defaulted to 1 (100 Hz).

3.2.1.1.2 Gyro Raw Data Trigger (2)

This trigger signals the computation of decimated raw gyrometer data. The actual updates of the data in this set varies depending on the data's individual update rate. The retrieval rate of the data set is defined by the variable named accelGyroDecimator which is defaulted to 1 (100 Hz).

3.2.1.1.3 Mag Raw Data Trigger (3)

This trigger signals computation of decimated raw magnetometer data. The actual updates of the data in this set varies depending on the data's individual update rate. The retrieval rate of the data set is defined by the variable named magDecimator which is defaulted to 2 (100 Hz).

3.2.1.1.4 Conditioned Data Trigger (4)

The Conditioned trigger is based on a signal generated immediately after the 100 Hz accel and gyro data is processed into engineering unit values. This data includes only accelp and gyrop but any other data can be streamed out the channel at the same time and they will have the most recently updated values.

3.2.1.1.5 Timer Trigger (6)

The Timer trigger is based on a free running timer. This timer is user configurable, using the variable chanTimerX100us, to tick at chanTimerX100us multiples of 100 microseconds. For example, setting chanTimerX100us to 50 yields a 5-millisecond tick rate (50*100 microseconds = 5,000 microseconds).

This timer is shared between the three channels. A channel's trigger can be customized further by further dividing the timer ticks by chanNTriggerDivisor. Thus, the timer tick should be set to a common denominator for the channels that use a timer trigger. For example, if the timer tick is set for 10,000 microseconds, and the user desires a 1Hz trigger for Channel 0, and a 20 Hz trigger for Channel 2, then the user should set chan0TriggerDivisor to 100 and chan2TriggerDivisor to 5.

3.2.1.2 Formatted (Protocol) Output

Once a trigger occurs, the selected packet formats will be output via a channel to the serial User Port. The output produced is a function of the format for that channel and is controlled by setting the variable chanNFormat (N=0, 1 or 2) to the number representing the desired format. The choices are 0(RFS), 1(NorthTek), 2(BitStreamBinary), 3(BitStreamAscii) and 4(NMEA). The format choices are saved in eeprom.

3.2.1.2.1 Triggered RFS (0)

The RFS composite messages can be synced according to when the data is freshest. The composite variables are the user customizable RFS bit field variables named composite1, composite2 and composite3.

Each channel will only stream one particular RFS composite variable. For example, if chan0Format is set to 0 (RFS), then the RFS bit field variable named composite1 will stream out channel 0. Similarly, the composite2 variable will stream from channel 1 and the composite3 variable will stream from channel 2.

The format for the composite variables is specified in the RFSProtocolSuite user manual and will be carried within a SAPP packet (defined in the same manual). See the RFS manual for details.

Here is an example to send the composite2 variable for every other compass calculation (50 Hz):

```
chan1Format 0 set drop   // send RFS composite2
chan1Trigger 5 set drop  // trigger on new compass data
chan1TriggerDivisor 2 set drop  // 100 Hz/2 => 50 Hz
```

To stop:

```
chan1TriggerDivisor 0 set drop
```

3.2.1.2.2 Triggered NorthTek (1)

The user can set up a trigger to periodically invoke a NorthTek user defined function given in a previously loaded script. The NorthTek script can be one that is programmed in EEPROM (see the User Boot Program section) or one that is sent to the inertial sensor after start-up. A NorthTek script function can be hooked up to a Protocol Trigger channel by setting the corresponding database string variable autoNorthTekN where N is 0, 1 or 2. The period is defined by the trigger for the Nth channel allowing the user to process freshly computed data. For example (note space after first quote):

```
chan1Format 1 set drop   // select NorthTek auto update
chan1Command " scriptname" set drop
```

The following is an example script that will print pitch and roll 4 times per second:

```
forget stack
variable stack          // set up for printf
: startArgs sp@ stack ! ; // set up start for printf
: endArgs stack @ ;      // set up end for printf

: printdata  // define function to print time,pitch,roll,yaw
startArgs "PRY,%d,%f,%f,%f"  cputime di@ pitch di@ roll di@ yaw di@ endArgs 80 printf
cr
;

chan1Command " printdata" set drop
chan1Format 1 set drop  // set to the NorthTek output format
chan1Trigger 5 set drop // trigger on new compass data
chan1TriggerDivisor 25 set drop  // 100 Hz/25 => 4 Hz
```

Example output:

```
PRY,66494,3.831054,33.307381,203.576431
PRY,66744,3.822923,33.314877,203.579666
PRY,66994,3.826396,33.311905,203.586319
PRY,67244,3.835386,33.281403,203.563629
PRY,67494,3.837020,33.309330,203.543472
PRY,67744,3.835978,33.309742,203.563599
…
```

```
// To stop the stream (must include carriage return and line feed after drop)
    chan1TriggerDivisor 0 set drop
```

3.2.1.2.3 Triggered Bit Stream Binary (2) and Bit Stream Ascii (3)

Both BitStreamBinary and BitStreamAscii's output contents are selected by the "chanNEnables" variables (chan0Enables, chan1Enables and chan2Enables). Each of these variables is a 512-bit array (16 32-bit integers), where each bit represents the VID (variable ID) number in the database. Each bit set selects a variable that will be packed in the output.

For Bit Stream Binary, it is up to the user to both select the desired bits in the chanNEnables and be aware of the size of each object being packed into the packet. Ordinals consume 1 byte, Int32's and Float32's take 4, arrays take multiples of 4, etc.

While the chanNEnables arrays can be set directly by computing the bit offset into the array, for convenience, there are commands to do that offset computation automatically. The commands are chanNEnableBit and chanNDisableBit (N=0, 1, 2). Simply set the command to the desired VID number. These command variables will not change when set, instead they cause the corresponding chanNEnables array to change. Note that the NorthTek "dvid@" command will return the VID for a given variable name.

A usage example is:

```
chan0EnableBit accelp dvid@ set drop
chan0EnableBit temperature dvid@ set drop
```

The ordering of the resulting output data is determined by the data's VID number. The output order can be determined by asking the device to provide the BitStream configuration. To examine the expected format, use the "chanNEnable." (N=0,1,2) command. Three axis vectored variables such as accelp, gyrop, etc. are output in the order of X, Y and then Z.

For example, to view the channel 0 configuration given the above enableBit example (including the terminating '.' which is a NorthTek print indicator):

```
chan0Enable.
```

Response:

```
<item name="accelp">
<vid>26</vid>
<offset>0</offset>
<type>Float32[3]</type>
</item>
<item name="temperature">
<vid>78</vid>
<offset>12</offset>
<type>Float32</type>
</item>
```

Note that the output is in XML format and describes the variable, VID number, byte offset from the beginning of the data in the packet, and the size of the object.

The variables chan0EnableBit, chan0DisableBit, chan1EnableBit, chan1DisableBit, chan2EnableBit and chan2DisableBit will enable or disable a single bit (specified by vid number) in the respective enables array. Note that all bits in an enable can be easily cleared with the following sequence, for example:
chan0Enables array[0 15 0 0 0 0 0 0 0 0 0 0 0 0 0 0]array set drop

The line above which clears an enables variable is recommended at the beginning of each NorthTek script using an enables variable.

3.2.1.2.3.1 Bit Stream Binary (2)

For BitStreamBinary, the output values are packed binary and are sorted in order of VID number. Floats are represented by the IEEE-754 32-bit format. It is up to the user to be aware of the size of each object being packed into the packet. Ordinals consume 1 byte, Int32's and Float32's take 4, arrays take multiples of 4, etc. The ordering of the data can be determined as described above using the "chanNEnable." command. The data is sent in a SAPP packet using a protocol identifier of 9 (see Appendix E Standard Asynchronous Packet Protocol in the RFS Protocol Suite document). The first byte of the SAPP packet payload is the channel number.

```
chan2Enables array[ 0 15 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ]array set drop
chan2Format 2 set drop  // Bit Stream Binary
chan2EnableBit accelrange dvid@ set drop
chan2EnableBit cputime dvid@ set drop
chan2EnableBit accelp dvid@ set drop
chan2Trigger 5 set drop  // decimated data
chan2TriggerDivisor 200 set drop // 100Hz/200=>0.5Hz

// Received the following hex bytes (ignore spaces between them)
01 10 95 69 02 3A AD 88 C2 5F C7 08 44 74 5D 4B 44 00 5F 9D 0E 00 C1 91
03

// here is the message parsed out
01      SAPP SOH Start
10 95  SAPP ByteCount, DLE translates to 0x15 = 21 bytes in SAPP payload (between ByteCount and ETX)
69      SAPP Error/Proto, Error=(don't ignore errors, no Ack, no Nak), Proto=9=Bit Stream Binary
02              Channel 2
3A AD 88 C2   Float32 accelp X
5F C7 08 44     Float32 accelp Y
74 5D 4B 44     Float32 accelp Z
00              Ordinal accelrange
5F 9D 0E 00    Int32   cputime
C1 91 SAPP CRC
03     SAPP ETX End
```

To view data order

chan2Enable.

Results:

```
<item name="accelp">
<vid>31</vid>
<offset>0</offset>
<type>Float32[3]</type>
</item>
<item name="accelrange">
<vid>66</vid>
<offset>12</offset>
<type>Byte</type>
</item>
<item name="cputime">
<vid>78</vid>
<offset>13</offset>
<type>Int32</type>
</item>
```

3.2.1.2.3.2 Bit Stream Ascii (3)

For BitStreamAscii, the values are sent in ASCII format and are also sorted in order of VID number.

```
// To get an Ascii display, select the BitStreamAscii
chan0Format 3 set drop

// Select the data
chan0EnableBit accelp dvid@ set drop
chan0EnableBit temperature dvid@ set drop

// Select the trigger and update rate
chan0Trigger 5 set drop // trigger on new decimated data computations
chan0TriggerDivisor 100 set drop  // only print 4 times a second (400 Hz/100)
```

Response example:

```
$CUS0,14.608162,-7.686007,1018.513184,30.625000,*
$CUS0,13.824555,-9.223790,1021.316406,30.625000,*
```

Note: "$CUS0" is the default preamble. The "*" is the default postamble. See next section for more info.

To stop the stream (must include CR LF after drop)

```
chan0TriggerDivisor 0 set drop
```

3.2.1.2.3.2.1 Bit Stream Ascii Prefix and Postfix

The BitStreamAscii format prepends a custom prefix and appends a custom postfix to each line of data. The above example shows the default prefix of “$CUS0” and the default postfix of “*”. These can be customized by setting the string variable chanNPreamble and chanNPostamble. These each have an 80 character limit.

Here is an example to change them for channel 0 (there must be a space after the first quote for any strings):

```
chan0Preamble “ Accels:” set drop
chan0Postamble “ ;” set drop
```

Response:

```
Accels:16.522532,-10.340560,1026.988770,30.937500,;
Accels:18.240021,-14.073236,1024.551147,30.875000,;
```

The BitStreamAscii format can be customized to output a CRC or checksum. See Appendix C for source code containing the CRC and Checksum used.

To add a CRC, include the character '\0x01' in the chanNPreamble which indicates the start of the CRC. To mark the end of the CRC'd bytes, include the character '\x02' in the chanNPostamble.

To select a checksum, include the character '\0x03' in the chanNPostamble and to select a CRC, include the character '\0x04' in the chanNPostamble. The CRC or checksum will appear in the place of the '\0x01'. For example:

// Show the use of embedded format characters to put CRC/checksums in user output

This one does checksum:

```
chan0Preamble "$\x01CUS0,"  set drop   // specify where to start the checksum
chan0Postamble "\x02*\x03" set drop // specify where to end checksum and where to put it
chan0Format 3 set drop  // Bit Stream Ascii
chan0Trigger 4 set drop  // Conditioned Data Trigger
```

This one does CRC:

```
chan1Preamble "$\x01CUS1,"  set drop   // specify where to start the CRC
chan1Postamble "\x02*\x04" set drop // specify where to end CRC and where to put it
chan1Format 3 set drop // Bit Stream ASCII
chan1Trigger 4 set drop // Conditioned Data Trigger
```

Full Bit Stream ASCII example:

```
// Clear any previous data selections for channels 0 and 1
chan0Enables array[ 0 15 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ]array set drop
chan1Enables array[ 0 15 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ]array set drop

// Select the data for channel 0
chan0EnableBit accelp dvid@ set drop
chan0EnableBit cputime dvid@ set drop
// Select the data for channel 1
chan1EnableBit gyrop dvid@ set drop
chan1EnableBit cputime dvid@ set drop

// Slow down the output to 2 Hz (2000 Hz/1000 = 2 Hz)
chan0TriggerDivisor 50 set drop  // 100Hz/50 => 2Hz
chan1TriggerDivisor 25 set drop // 100Hz/25=>4Hz
```

Example output from above script:

```
$CUS0,2.612818e-01,-1.617098e-02,10.077161,3177458,*63
$CUS1,1.715285e-02,2.853286e-02,-4.727963e-02,3177478,*C4B4
$CUS0,2.249641e-01,2.542049e-03,10.094501,3177958,*49
$CUS1,2.079850e-02,-5.304283e-03,-3.368925e-02,3177978,*F0FF
…
```

To stop the output:

```
chan0TriggerDivisor 0 set drop
chan1TriggerDivisor 0 set drop
```

3.2.1.2.4 Triggered NMEA (4)

The Bit Stream NMEA format allows the user to set up a trigger to periodically invoke an inertial sensor implemented NMEA command. A NMEA command can be hooked up to a Trigger channel by setting the corresponding database string variable chanNCommand where N is 0, 1 or 2. The period is defined by the trigger for the Nth channel allowing the user to process freshly computed data.

For example (note space after first quote):

```
chan2Trigger 4 set drop      // trigger when conditioned data ready
chan2TriggerDivisor 100 set drop  // output at 4100 Hz/100  = 1 Hz
chan2Format 4 set drop   // select NMEA protocol
chan2Command " $PSPA,A\n" set drop
```

To stop the stream (must include CR LF after drop)

```
chan2TriggerDivisor 0 set drop
```

3.3 RFS

The RFS protocol suite is described in a separate document titled Remote Function Select (RFS) Protocol Suite.

RFS assigns a unique Variable Identifier (VID) for each database element (see Appendix A for the list of available database elements). Other than the reserved VIDs[2] the user application should not depend on a particular numeric for a particular VID for the database elements from one release of the inertial system software to another. The string names will remain the same from release to release and can be used to resolve the same variable to a VID number.

A VID can be obtained for a variable name using the NorthTek™ command, dvid@ or the RFS command "Vid?". The NorthTek syntax is "<variableName> dvid@ ." (note there is a space and a dot at the end to display the result).

An alternative is to collect all VIDs upon start-up of the inertial system. To do this, the user application should use VID 0 to obtain the number of elements, and then obtain the names of all of the database elements by looping through all of the VIDS and using the "Get_Next" commands. The user application can then obtain the current VID for the desired named element listed in Appendix A. From then on, with that VID, the user application may use the "Set" command or the "Get_Value" command and the inertial system will respond with "Value_Is".

[2] VID 0 is always the count of the number of VIDs in the device. VID 1 is always the name of the device. Vid 2 is always the major version number. Vid 3 is always the minor version number.

The RFS protocol is layered on top of a packet transfer mechanism called Standard Asynchronous Packet Protocol (SAPP). SAPP is also described in the Remote Function Select (RFS) Protocol Suite's Appendix E. Please refer to the referenced document for more information.

3.3.1 RFS Get/Get_Response Example

Below is an inertial system specific example for which the user will need to have the RFS Protocol Suite document open for reference. In this example, the user application is obtaining the serial number of the inertial system unit. All figure numbers in parenthesis below refer to those found in the *RFS Protocol Suite document*.

The examples below were created using the SPARTACUS GUI application which was set to capture the SAPP Debug Output for the RFS command Get serialnumber (VID 5). SPARTACUS can be used to create more examples of the RFS/SAPP protocol. On the SPARTACUS Admin page, pull down Spartacus Settings->Debugging->Display SAPP Layer. Then select a VID and click the "Get" button. The bytes that are sent and received for the "Get" command are displayed in the Debug Output window. The blue buttons on the upper left side generate RFS commands for the selected VID.

This describes the steps taken in the user's RFS command example:

- Create a command to get information about the variable with Variable ID (VID) = 9:
    - RFS Command: revision, payload size = 1, get command = 1, message sequence = 0xab, VID = 5
    - This command translates to the following RFS payload (Figure 4-15 RFS Outer Layer Protocol Diagram):
        - 00 00 00 00 01 01 ab 05
- Add the SAPP layer for framing and error protection (Figures E.4-36 – Packet Frame and E.4-37 – Packet Body). Note that "10 81" is substituted for "01" in 2 places to prevent "01" from being confused with the start character (SOH) (reference E.3.4 Packet Frame)
    - 01 0B 00 00 00 00 00 10 81 10 81 ab 05 c8 d7 03

This is a detailed explanation of the bytes that are sent in the user's RFS command:

```
01          SOH - SAPP packet frame
0B          SAPP packet body size
00          Error Options/Protocol
            SAPP Payload - Start of RFS packet
00             RFS Revision 0
0000001081     RFS Payload size - 1081 = DLE substitution for 01
1081           DLE substitution for 01 - Get Command
ab             sequence number
05             VID for serial number
c8d7        2 CRC bytes for SAPP packet body
03          ETX for SAPP packet frame
```

Once the above command is received, the inertial system creates the response message Payload:

RFS Response: RFS Rev = 0, payload size=23(0x17), **getResponse** command = 00, seq=16(0x0F), Vid = 5
(see Figure 4-16 Data Description Object) Scalar 0x1X,
(see Figure 4-17 Scalar Description Field) Type 2=String, Field Size = 20 (0x14),
Name="serialnumber", String Length Max = 20 (0x14),
(see Figure 4-8 String Data Format) String Length=4, String Value="B13"
00 00 00 00 17 00 ab 05 10 02 14 0d 73 65 72 69 61 6c 6e 75 6d 62 65 72 00 14 04 42 31 33 00

The inertial system then adds the SAPP layer for framing and error detection resulting in the following output (RFS Protocol Suite Figures 4-36 – Packet Frame and 4-37 – Packet Body):

01 22 60 00 00 00 00 17 00 ab 05 10 90 02 14 0d 73 65 72 69 61 6c 6e 75 6d 62 65 72 00 14 04 42 31 33 00 8b a5 03

This is a detailed explanation of the bytes that are sent in the inertial system's RFS response:

```
01                              SOH - SAPP packet frame
22                              SAPP packet body size
60                              Error Options/Protocol
                                SAPP Payload - Start of RFS packet
00                                 RFS Revision 0
00000017                           RFS Payload size
00                                 GetResponse (followed by Data Description plus Value)
ab                                 sequence number of associated get request
05                                 VID for serial number
                                Data Description Object
1090                               DLE substitution for 10hex - Scalar (Data Desc Object)
                                Scalar Description Field
02                                 Type 2: String
14                                 Description Field size 14hex = 20 decimal
0D                                 Text field length 0Dhex = 13 decimal
73657269616C6E756D65720O           Name (Text Field): "serialnumber" incl/nul char
14                                 String Length Max 14hex = 20 decimal
                                String Data Format
04                                 # of bytes in the string
42313300                            String Chars "B13" plus nul
                                End of RFS packet (SAPP payload)
8ba5                            2 CRC bytes for SAPP packet body
03                              ETX for SAPP packet frame
```

3.3.2 RFS BitFields

BitFields are user programmable as to content allowing the user to configure a single message containing multiple useful values into a single report (similar to packed structures). BitFields minimize communications required to obtain the necessary data for the application. Full details of the BitFields and their usage can be found in the *RFS Protocol Suite* document.

A BitField is a variable type that contains meta-data instead of data. A BitField, for example, may contain a reference to an Int32 variable, an Ordinal Variable and a Float32 Variable. When setting or getting the variable, the application program would be able set or read all three values in the same message. The BitField variable contains the information to describe what three other variables would be set or read in

a message to/from the BitField variable. The BitField thus contains not the actual data to be transmitted, but information describing what information will be transmitted. RFS messages may be used to manipulate the BitField meta-data or operate on the actual data referred to by the BitField. Some RFS messages are modal in that they automatically determine if the intended message is for the meta-data or the data itself. Full details are described in the *RFS Protocol Suite* document. BitFields are only used with the RFS protocol. (RFS protocol technical support is available from the manufacturer).

3.3.3 RFS BitField Example

The Bit Field variables named composite1 (or "position" in some cases), composite2, and composite3 are defined by the user using the RFS Construct command. Each of these composite variables has a corresponding variable used to specify the repetition rate for the composite variable. For example, the composite2 variable's repetition rate is specified using comp2Rate.

Note that the SPARTACUS GUI will customize composite1 for its own use. This setting is maintained through power cycles.

Below is an example that uses the Construct command (

Table 1) to define the value object reported by the composite2 variable. The change to composite2 is confirmed using the Show command (Table 2) with the corresponding Format message response (Table 3). Then the data is obtained using the Get_Value command (Table 4) and the corresponding Value_Is response (Table 5).

NOTE: The message bytes for RFS are in hexadecimal form. These bytes are expected to be sent by the host application in binary form. Spaces are placed between hex values for human readability purposes only and should NOT be included in the actual packet.

Table 1: (TX) RFS Bitfield Construct Packet Example

Frame Value (Hex)	Protocol Layer	Description
01	SAPP	SOH (SAPP Header)
4e	SAPP	Body Size
00	SAPP	Error Handling Options/Protocol
04	RFS	Revision
00 00 00 00	RFS	Packet Payload Size (Only in RFS Revisions < 5)
0c	RFS	Command: Construct
a3	RFS	Packet Sequence
20	RFS	VID (composite2 in this example)
80	RFS	SAP Type/Access (Bitfield/Temporary ReadWrite)
41	RFS	Field Size (Bytes)
10 90	RFS	Field Count (DLE substitution for 0x10)
00 02 00 08	RFS	Cross Reference 1: 000 = Start Bit; 20 = Size; 008 = VID (pitch in this example)
02 02 00 09	RFS	Cross Reference 2: 020 = Start Bit; 20 = Size; 009 = VID (roll in this example)
04 02 00 0a	RFS	Cross Reference 3: 040 = Start Bit; 20 = Size; 00a = VID (yaw in this example)
10 86 02 00 26	RFS	Cross Reference 4: 060 = Start Bite (10 86 DLE = 0x60); 20 = Size; 026 = VID (magErr in this example)
00 00 00 00	RFS	Cross Reference 5: Blank
00 00 00 00	RFS	Cross Reference 6: Blank
00 00 00 00	RFS	Cross Reference 7: Blank
00 00 00 00	RFS	Cross Reference 8: Blank
00 00 00 00	RFS	Cross Reference 9: Blank
00 00 00 00	RFS	Cross Reference 10: Blank
00 00 00 00	RFS	Cross Reference 11: Blank
00 00 00 00	RFS	Cross Reference 12: Blank
00 00 00 00	RFS	Cross Reference 13: Blank
00 00 00 00	RFS	Cross Reference 14: Blank
00 00 00 00	RFS	Cross Reference 15: Blank
00 00 00 00	RFS	Cross Reference 16: Blank
6e f2	SAPP	CRC
03	SAPP	ETX

Table 2: (TX) RFS Bitfield Show Packet Example

Frame Value (Hex)	Protocol Layer	Description
01	SAPP	SOH (SAPP Header)
0b	SAPP	Body Size
00	SAPP	Error Handling Options/Protocol
04	RFS	Revision
00 00 00 10 81	RFS	Payload Size (10 81 DLE = 0x01)
05	RFS	Command: Show
a4	RFS	Packet Sequence
20	RFS	VID (composite2 in this example)
7f e3	SAPP	CRC
03	SAPP	ETX

Table 3: (RX) RFS Bitfield Format Packet Example

Frame Value (Hex)	Protocol Layer	Description
01	SAPP	SOH (SAPP Header)
5a	SAPP	Body Size
60	SAPP	Error Handling Options/Protocl
10 81	RFS	Revision (10 81 DLE = 0x01)
00 00 00 4f	RFS	Packet Payload Size
10 86	RFS	Command: Format (10 86 DLE = 0x06)
a4	RFS	Packet Sequence
20	RFS	VID (composite2 in this example)
80	RFS	SAP Type/Access (Bitfield/Temporary ReadWrite)
4d	RFS	Field Size
0b 63 6f 6d 70 6f 73 69 74 65 32 00	┐ RFS ┘	Name: composite2 (in this example) (Number of Characters + “composite2” + NULL)
10 90	RFS	Field Count (10 90 DLE = 0x10)
00 02 00 08	RFS	Cross Reference 1: 000 = Start Bit; 20 = Size; 008 = VID (pitch in this example)
02 02 00 09	RFS	Cross Reference 2: 020 = Start Bit; 20 = Size; 009 = VID (roll in this example)
04 02 00 0a	RFS	Cross Reference 3: 040 = Start Bit; 20 = Size; 00a = VID (yaw in this example)
10 86 02 00 26	RFS	Cross Reference 4: 060 = Start Bite (10 86 DLE = 0x60); 20 = Size; 026 = VID (magErr in this example)
00 00 00 00	RFS	Cross Reference 5: Blank
00 00 00 00	RFS	Cross Reference 6: Blank
00 00 00 00	RFS	Cross Reference 7: Blank
00 00 00 00	RFS	Cross Reference 8: Blank
00 00 00 00	RFS	Cross Reference 9: Blank
00 00 00 00	RFS	Cross Reference 10: Blank
00 00 00 00	RFS	Cross Reference 11: Blank
00 00 00 00	RFS	Cross Reference 12: Blank
00 00 00 00	RFS	Cross Reference 13: Blank
00 00 00 00	RFS	Cross Reference 14: Blank
00 00 00 00	RFS	Cross Reference 15: Blank
00 00 00 00	RFS	Cross Reference 16: Blank
74 b6	SAPP	CRC
03	SAPP	ETX

Table 4: (TX) RFS Bitfield Get_Value Packet Example

Frame Value (Hex)	Protocol Layer	Description
01	SAPP	SOH (SAPP Header)
0b	SAPP	Body Size
00	SAPP	Error Handling Options/Protocol
04	RFS	Revision
00 00 00 10 81	RFS	Payload Size (10 81 DLE = 0x01)
08	RFS	Command: Get_Value
a5	RFS	Packet Sequence
20	RFS	VID (composite2 in this example)
c1 d8	SAPP	CRC
03	SAPP	ETX

Table 5: (RX) RFS Bitfield Value_Is Packet Example

Frame Value (Hex)	Protocol Layer	Frame Description
01	SAPP	SOH (SAPP Header)
1e	SAPP	Body Size
60	SAPP	Error Handling Options/Protocol
10 81	RFS	Revision (10 81 DLE = 0x01)
00 00 00 13	RFS	Payload Size
09	RFS	Command: Value_Is
a5	RFS	Packet Sequence
20	RFS	VID (composite2 in this example)
80	RFS	SAP Type/Access (Bitfield/Temporary ReadWrite)
11	RFS	Field Size
04	RFS	Word Count
bf ea b9 27	RFS	Word 1: Value for pitch (bf ea b9 27 = -1.833775)
3f 96 5b 54	RFS	Word 2: Value for roll (3f 96 5b 54 = 1.174662)
43 3d 4f a2	RFS	Word 3: Value for yaw (43 3d 4f a2 = 189.311066)
3f 7b 4d 1e	RFS	Word 4: Value for magErr (3f 7b 4d 1e = 0.981645)
68 96	SAPP	CRC
03	SASPP	ETX

3.4 Legacy Binary

Legacy Binary was originally intended to be only used by users of the older Sparton compasses such as the SP3002, SP3003, and SP3004. As a result, it does not have access to any of the newer data such as the gyro data and so is not recommended for new designs.

For the AHRS-M2, it has been added to be compatibile with the DC-4E/GEDC-6E/AHRS-8 products.

All Legacy Binary commands sent to the compass begin with a header byte (0xA4) and end with a termination byte (0xA0). Since the I-Field calibration routines have been modified and improved, the In-Field Calibration commands of Legacy Binary are no longer supported.

If the compass does not recognize a sequence of bytes as being a legacy command, no response is produced. If the compass recognizes the command but, for some reason, cannot execute it, it will respond with an error code. All error codes begin with a header byte (0xAE) and end with a termination byte (0xA0). Commands which commence and do not conclude within a timely manner will be rejected. All multi-byte values are sent most significant byte (MSByte) first.

Error Code Format: 3 Bytes (***0xAE,<8-bit error code>,0xA0***)

8-bit Error Codes:

0xFF = Improper command termination (i.e., no 0xA0 found)
0xFE = Receive buffer overflow
0xFD = Invalid parameter associated with given command

Raw Magnetics

Description: Reads current magnetics directly from magnetometers (Mx, My, and Mz). These are raw sensor readings and do not yet have any calibration parameters applied.

Send: 3 Byte (***0xA4,0x01,0xA0***)
Response: 9 Bytes (**0xA4,0x01,<Mx>,<My>,<Mz as 16-bit integers MS byte first>,0xA0**)

Heading - TRUE

Description: Reads the current true heading. The heading is compensated for platform tilt. True heading is the magnetic heading corrected for magnetic variation.

Send: 3 Byte (***0xA4,0x02,0xA0***)
Response: 5 Bytes (**0xA4,0x02,<Heading as a 16-bit signed integer>,0xA0**)

Heading (degrees) = (16-bit Heading value)*360/4096
Heading Range = 0.0 to +359.9

Heading - Magnetic

Description: Reads the current magnetic heading. The heading is compensated for platform tilt.

Send: 3 Bytes (***0xA4,0x09,0xA0***)

Response: 5 Bytes
(**0xA4,0x09,<Heading as a 16-bit signed integer>,0xA0**)

Heading (degrees) = (16-bit Heading value)*360/4096
Heading Range = 0.0 to +359.9

Magnetic variation

Description: Set the magnetic variation angle. The heading will be adjusted to indicate true north instead of magnetic north. Magnetic variation angles >+180 and <-180 will be limited to +180 and –180 respectively.

Send: 5 Bytes (***0xA4,0x83,<16-bit signed integer value MSB first>,0xA0***)
Response: 5 Bytes
(**0xA4,0x83,<16-bit signed Variation>,0xA0**)

16-bit signed integer = (Magnetic Variation)*10.0

Automatic Magnetic Variation

Description: Latitude, Longitude, Altitude, and Day should be programmed separately using their respective commands before issuing this command. Automatic variation will compute the local magnetic variation based on the device's current geographical location (geodetic coordinate system referenced to the WGS 84 ellipsoid). Once the computation is complete, the magnetic variation will be updated in the compass. *NOTE: TO RETAIN MAGNETIC VARIATION ACCURACY, THE MAGNETIC MODEL MUST BE UPDATED EVERY FIVE YEARS. A SEPARATE* PROGRAM *IS AVAILABLE ON THE SUPPLIED CD WHICH WILL ASSIST IN DOWNLOADING NEW COEFFICIENTS INTO THE DIGITAL COMPASS. THIS ONLY AFFECTS THE CALCULATION OF TRUE HEADING AND DOES NOT AFFECT MAGNETIC HEADING ACCURACY.* NOTE: Resetting the magnetic parameters to the factory default does not affect the magnetic variation information. To clear out the magnetic variation, manually set the magnetic variation to zero
Send: 3 Byte (***0xA4,0x0F,0xA0***)
Response: 5 Bytes
(**0xA4,0x0F,<Variation as a 16-bit signed integer>,0xA0**)

16-bit signed integer = (Magnetic Variation)*10.0

Latitude

Description: Set the geodetic latitude angle in degrees (geodetic coordinate system referenced to the WGS 84 ellipsoid). The magnetic variation will not change until latitude, longitude, altitude, and day have been programmed and the Automatic Variation command is issued. Latitude >+90 or <-90 will be limited to +90 and –90 respectively.

Send: 5 Bytes
(***0xA4,0x8B,<16-bit signed integer value MSB first>,0xA0***)

Response: 5 Bytes
(**0xA4,0x8B,<16-bit Latitude>,0xA0**)

16-bit signed integer = (North(+) or South(-) Latitude in degrees)*100.0

Longitude

Description: Set the geodetic longitude angle in degrees (geodetic coordinate system referenced to the WGS 84 ellipsoid). The magnetic variation will not change until latitude, longitude, altitude, and day have been programmed and the Automatic Variation command is issued. . Latitude >+180 or <-180 will be limited to +180 and –180 respectively.

Send: 5 Bytes
(***0xA4,0x8C, 16-bit signed integer value MSB first,0xA0***)

Response: 5 Bytes
(**0xA4,0x8C, 16-bit signed Longitude,0xA0**)

16-bit signed integer = (East(+) or West(-) Longitude in degrees)*100.0

Altitude

Description: Set the geodetic altitude in meters above sea level (geodetic coordinate system referenced to the WGS 84 ellipsoid). The magnetic variation will not change until latitude, longitude, altitude, and day have been programmed and the Automatic Variation command is issued. Altitude >+32767 or <-32767 will be limited to +32767 and –32767 respectively.

Send: 5 Bytes
(***0xA4,0x8D, 16-bit signed integer value MSB first,0xA0***)

Response: 5 Bytes
(**0xA4,0x8D, 16-bit signed Altitude,0xA0**)

16-bit signed integer = +/- Altitude in meters

Day

Description: The day is entered as a fractional year based on the current day of the year (i.e., February 15 is the 46th day of the 2008. In fractional terms, this would be 46/365 = 0.126. The Fractional Day value for February 15, 2008 would then be 2008.1 (resolution beyond a tenth causes negligible change in variation). The magnetic variation will not change until latitude, longitude, altitude, and day have been programmed and the Automatic Variation command is issued. Day < 2005 will be limited to 2005.

Send: 5 Bytes
(***0xA4,0x8E, 16-bit unsigned integer value MSB first,0xA0***)
Response: 5 Bytes
(**0xA4,0x8E, 16-bit unsigned Day,0xA0**)

16-bit unsigned integer = (Fractional Day)*10.0

Magnetic Vector

Description: Measures the magnetic field strength along each axis (X, Y, and Z) and total absolute field strength (MAtotal) in milligauss.

Send: 3 Bytes
(***0xA4,0x04,0xA0)***
Response: 11 Bytes
(**0xA4,0x04, MAx, MAy, MAz, MAtotal as 16-bit integers,0xA0**)

Raw Acceleration

Description: Reads current acceleration directly from accelerometers (AccelX, AccelY, AccelZ). These are raw sensor readings and do not yet have any calibration parameters applied.

Send: 3 Bytes
(***0xA4,0x05,0xA0***)

Response: 9 Bytes
(**0xA4,0x05, AccelX, AccelY, AccelZ as 16-bit integers,0xA0**)

Pitch and Roll Output

Description: Reads the current platform orientation (Pitch and Roll).

Send: 3 Byte
(***0xA4,0x06,0xA0***)
Response: 7 Bytes
(**0xA4,0x06, Pitch, Roll as 16-bit signed integers,0xA0**)

Pitch (in degrees) = (Response Value)*90/4096
Pitch Range = -90 to +90

Roll (in degrees) = (Response Value)*180/4096
Acceleration Vector Roll Range = -180 to +180

Acceleration Vector

Description: Measures the acceleration along each axis (X, Y, and Z) and total absolute strength (Atotal) in milli-g.

Send: 3 Byte
(***0xA4,0x07,0xA0***)
Response: 11 Bytes
(**0xA4,0x07, Ax, Ay, Az, Atotal as 16-bit integers,0xA0**)

Temperature

Description: Reads the internal temperature channel of the on-board microcontroller. This measurement is calibrated at the factory, though not required by the compass in determining an accurate heading.

Send: 3 Bytes
(***0xA4,0x11,0xA0***)
Response: 5 Bytes
(**0xA4,0x11, Temperature as 16-bit signed integer MSB first,0xA0**)

Temperature_C = (Temperature_MSB*256 + Temperature_LSB)/10.0

BAUD RATE

Description: The factory default BAUD setting is 115200 Baud (0x08). When the baud rate command is issued on the UART, the compass will echo back the command once at the current baud rate and then again at the new baud rate. The baud rate will be stored in EEPROM and will become the new operating communication rate for the UART. The baud rate will apply to Legacy, RFS, NorthTek and NMEA commands issued on the UART.

Send: 4 Bytes
(***0xA4,0x57, 8-bit BAUD value MSB first,0xA0***)

Response: 4 Bytes
(**0xA4,0x57, 8-bit BAUD value,0xA0**)

Acceptable Baud Rate Values:
0x00 = 115.2 Baud, unless **AHRS8_Compatible** is 1 (post-3.0.0) then 300 Baud
0x01 = 1200 Baud
0x02 = 2400 Baud
0x03 = 4800 Baud
0x04 = 9600 Baud
0x05 = 19.2 kBaud
0x06 = 38.4 kBaud
0x07 = 57.6 kBaud
0x08 = 115.2 kBaud

Mounting Configuration

Description: Sets the mounting orientation of the compass platform. The default orientation is horizontal. For vertical orientations consult the particular device's data sheet. To determine the orientation setting, read the acceleration vector. When in a static level condition, Az should be approximately +1000mg and Ax and Ay should be close to zero.

Send: 4 Bytes
(***0xA4,0x4A, 8-bit orientation {0x00=Horizontal,0x01=Vertical},0xA0***)

Response: 4 Bytes
(**0xA4,0x4A, 8-bit orientation,0xA0**)

Appendices

A Variable XML Descriptions

XML File for database data showing the defaults, range, type, ordinal options and descriptions. DATA_PERMANENT indicates that the value will be stored in non-volatile memory and so is maintained across power cycles and software resets. DATA_READONLY indicates that the user should not set the value. DATA_PROTECTED indicates that a password is required to change the value.

Below is the XML file for the AHRS-M1. The AHRS-M2 has all the same variables except for the magnetometer related variables which the user should not be using.

```
<?xml version="1.0"?>
<embeddedDB name="config">
	<item name="vidCount">
		<longname>Variable Count</longname>
		<type>Int32</type>
		<flags>DATA_READONLY</flags>
		<defaultValue>0</defaultValue>
		<minValue>0</minValue>
		<getfunction>N</getfunction>
		<setfunction>N</setfunction>
		<maxValue>1000</maxValue>
		<units>None</units>
	</item>
	<item name="name">
		<longname>Device Name</longname>
		<type>String</type>
		<flags>DATA_PERMANENT|DATA_PROTECTED</flags>
		<dimension>80</dimension>
		<defaultValue>AHRS-M1</defaultValue>
		<units>None</units>
```

```
        </item>
        <item name="quiet">
                <longname>Debug Suppress</longname>
                <type>Ordinal</type>
                <defaultValue>1</defaultValue>
                <maxValue>1</maxValue>
                <labels>
                        <label>Debug On</label>
                        <label>Debug Off</label>
                </labels>
    <units>None</units>
</item>
<item name="VERSION">
    <longname>Subversion tagged ID for this software</longname>
    <type>String</type>
    <flags>DATA_READONLY</flags>
    <dimension>80</dimension>
    <defaultValue>undefined</defaultValue>
    <units>None</units>
</item>
<item name="REVISION">
    <longname>Subversion revision number of this software</longname>
    <type>String</type>
    <flags>DATA_READONLY</flags>
    <dimension>80</dimension>
    <defaultValue>undefined</defaultValue>
    <units>None</units>
</item>
        <item name="serialnumber">
                <longname>Serial number for this navigation unit</longname>
```

```
        <type>String</type>
        <flags>DATA_PERMANENT|DATA_PROTECTED</flags>
        <dimension>20</dimension>
        <defaultValue>undefined</defaultValue>
        <units>None</units>
</item>
<item name="orientation">
        <longname>Physical orientation of compass</longname>
        <type>Ordinal</type>
        <flags>DATA_PERMANENT</flags>
        <defaultValue>0</defaultValue>
        <maxValue>4</maxValue>
        <labels>
                <label>Horizontal</label>
                <label>Vertical</label>
                <label>Left Edge</label>
                <label>Right Edge</label>
                <label>Inverted</label>
        </labels>
        <setfunction>Y</setfunction>
        <units>Furlongs</units>
</item>
<item name="baud">
        <longname>User Port Baud rate</longname>
        <type>Ordinal</type>
        <flags>DATA_PERMANENT</flags>
        <defaultValue>8</defaultValue>
        <maxValue>8</maxValue>
        <labels>
                <label>Alt</label>
```

```
                <label>1200</label>
                <label>2400</label>
                <label>4800</label>
                <label>9600</label>
                <label>19200</label>
                <label>38400</label>
                <label>57600</label>
                <label>115200</label>
        </labels>
        <units>Bits Per Second</units>
</item>
<item name="pitch">
        <longname>Pitch angle in degrees, positive up</longname>
        <type>Float32</type>
        <flags>DATA_READONLY</flags>
        <defaultValue>0.0</defaultValue>
        <minValue>-180.0</minValue>
        <maxValue>180.0</maxValue>
        <units>degrees</units>
</item>
<item name="roll">
        <longname>Roll angle in degrees, positive rightside down</longname>
        <type>Float32</type>
        <flags>DATA_READONLY</flags>
        <defaultValue>0.0</defaultValue>
        <minValue>-180.0</minValue>
        <maxValue>180.0</maxValue>
        <units>degrees</units>
</item>
<item name="yaw">
```

```
                <longname>Magnetic heading in degrees, positive East</longname>
                <type>Float32</type>
                <flags>DATA_READONLY</flags>
                <defaultValue>0.0</defaultValue>
                <minValue>0.0</minValue>
                <maxValue>360.0</maxValue>
                <units>degrees</units>
        </item>
        <item name="yawt">
                <longname>True magnetic heading in degrees, positive East</longname>
                <type>Float32</type>
                <flags>DATA_READONLY</flags>
                <defaultValue>0.0</defaultValue>
                <minValue>0.0</minValue>
                <maxValue>360.0</maxValue>
                <units>degrees</units>
        </item>
        <item name="quaternion">
                <longname>alternate attitude representation, components are W, X, Y,
Z</longname>
                <type>ArrayFloat32</type>
                <dimension>4</dimension>
                <flags>DATA_READONLY</flags>
                <defaultValue>0.0</defaultValue>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <units>None</units>
        </item>
        <item name="lat">
                <longname>Latitude</longname>
```

```
        <type>Float32</type>
        <flags>DATA_PERMANENT</flags>
        <defaultValue>0.0</defaultValue>
        <minValue>-90.0</minValue>
        <maxValue>90.0</maxValue>
        <units>degrees</units>
</item>
<item name="lonG">
        <longname>Longitude</longname>
        <type>Float32</type>
        <flags>DATA_PERMANENT</flags>
        <defaultValue>0.0</defaultValue>
        <minValue>-180.0</minValue>
        <maxValue>180.0</maxValue>
        <units>degrees</units>
</item>
<item name="alt">
        <longname>Altitude</longname>
        <type>Float32</type>
        <flags>DATA_PERMANENT</flags>
        <defaultValue>0.0</defaultValue>
        <minValue>-10000.0</minValue>
        <maxValue>1000000.0</maxValue>
        <units>meters</units>
</item>
<item name="day">
        <longname>Estimated date expressed to the tenth of a year</longname>
        <type>Float32</type>
        <flags>DATA_PERMANENT</flags>
        <defaultValue>2015.0</defaultValue>
```

```
                <minValue>2000.0</minValue>
                <maxValue>3000.0</maxValue>
                <units>Years</units>
        </item>
        <item name="restorefactorycal">
                <longname>Command restore factory cal to user values</longname>
                <type>Ordinal</type>
                <defaultValue>0</defaultValue>
                <maxValue>1</maxValue>
                <labels>
                        <label>OFF</label>
                        <label>ON</label>
                </labels>
                <setfunction>N</setfunction>
                <units>None</units>
        </item>
        <item name="set_wmm">
                <longname>Command to update the World Magnetic Model data, uses
lat,long,alt,date</longname>
                <type>Ordinal</type>
                <defaultValue>0</defaultValue>
                <maxValue>1</maxValue>
                <labels>
                        <label>OFF</label>
                        <label>ON</label>
                </labels>
                <setfunction>N</setfunction>
                <units>None</units>
        </item>
        <item name="magvar">
```

```
            <longname>Magnetic Variation - difference between true and
magnetic</longname>

            <flags>DATA_PERMANENT</flags>

            <type>Float32</type>

            <defaultValue>0.0</defaultValue>

            <minValue>-360.0</minValue>

            <maxValue>360.0</maxValue>

            <setfunction>N</setfunction>

            <units>degrees</units>

      </item>

      <item name="wmmGD">

            <longname>World Magnetic Model geodetic magnetic field X, Y, Z at
Lat/Long</longname>

            <type>ArrayFloat32</type>

            <flags>DATA_PERMANENT</flags>

            <dimension>4</dimension>

            <minValue>-3.0E38</minValue>

            <maxValue>3.0E38</maxValue>

            <values>

                  <value>0.521215</value>

                  <value>0.0</value>

                  <value>0.853426</value>

                  <value>1.0</value>

            </values>

            <units>NONE</units>

      </item>

      <item name="wmmHasRun">

            <longname>World Magnetic Model has been run (wmmGD is valid)</longname>

       <type>Ordinal</type>

       <flags>DATA_PERMANENT</flags>

       <defaultValue>0</defaultValue>
```

```
            <minValue>0</minValue>
            <maxValue>1</maxValue>
        <labels>
                <label>false</label>
                <label>true</label>
        </labels>
            <units>NONE</units>
    </item>
        <item name="magr">
            <longname>Raw Magnetometer X,Y,Z,Total array</longname>
            <flags>DATA_READONLY</flags>
            <type>ArrayInt32</type>
            <dimension>3</dimension>
            <minValue>0xFFFF8000</minValue>
            <maxValue>0x00007FFF</maxValue>
            <values>
                <value>0</value>
                <value>0</value>
                <value>0</value>
            </values>
        </item>
        <item name="magp">
            <longname>Processed Magnetometer X,Y,Z array</longname>
            <type>ArrayFloat32</type>
            <flags>DATA_READONLY</flags>
            <dimension>3</dimension>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <units>milli-gauss</units>
        </item>
```

```
<item name="accelr">
	<longname>Raw Accelerometer X,Y,Z array</longname>
	<flags>DATA_READONLY</flags>
	<type>ArrayInt32</type>
	<dimension>3</dimension>
	<minValue>0xFFFF8000</minValue>
	<maxValue>0x00007FFF</maxValue>
	<values>
		<value>0</value>
		<value>0</value>
		<value>0</value>
	</values>
	<units>counts</units>
</item>
<item name="accelp">
	<longname>Processed Accelerometer X,Y,Z array</longname>
	<type>ArrayFloat32</type>
	<flags>DATA_READONLY</flags>
	<dimension>3</dimension>
	<minValue>-3.0E38</minValue>
	<maxValue>3.0E38</maxValue>
	<units>milli-g</units>
</item>
<item name="gyror">
	<longname>Raw gyro X,Y,Z array</longname>
	<flags>DATA_READONLY</flags>
	<type>ArrayInt32</type>
	<dimension>3</dimension>
	<minValue>0xFFFF8000</minValue>
	<maxValue>0x00007FFF</maxValue>
```

```
                <values>
                        <value>0</value>
                        <value>0</value>
                        <value>0</value>
                </values>
                <units>NONE</units>
        </item>
        <item name="gyrop">
                <longname>Processed gyro X,Y,Z array</longname>
                <type>ArrayFloat32</type>
                <flags>DATA_READONLY</flags>
                <dimension>3</dimension>
                <minValue>-999999999.0</minValue>
                <maxValue>999999999.0</maxValue>
                <units>NONE</units>
        </item>
        <item name="gyroSampleRate">
                <longname>Sample rate of gyro data as observed during
calibration</longname>
                <flags>DATA_READONLY</flags>
                <type>Float32</type>
                <defaultValue>100.0</defaultValue>
                <minValue>1.0</minValue>
                <maxValue>1000.0</maxValue>
                <units>Hz</units>
        </item>
        <item name="positionrate">
                <longname>Composite 1 report rate</longname>
                <type>Int32</type>
                <flags>0</flags>
```

```
                <defaultValue>0</defaultValue>

                <minValue>0</minValue>

                <maxValue>180000</maxValue>

                <units>reports per second</units>

        </item>

        <!-- A Bit field definition -->

        <item name="position">

                <longname>User modifiable composite object with
Roll,Pitch,Yaw,Lat,Lon,Alt,Temp</longname>

                <type>BitField</type>

                <flags>DATA_PERMANENT</flags>

                <dimension>16</dimension>

                <!-- This is the maximum number of fields possible -->

                <fields>

                        <!-- This determines the default number of fields -->

                        <field>

                                <vidName>pitch</vidName>

                                <bits>32</bits>

                        </field>

                        <field>

                                <vidName>roll</vidName>

                                <bits>32</bits>

                        </field>

                        <field>

                                <vidName>yawt</vidName>

                                <bits>32</bits>

                        </field>

                        <field>

                                <vidName>magErr</vidName>

                                <bits>32</bits>
```

```
                </field>
                <field>
                        <vidName>temperature</vidName>
                        <bits>32</bits>
                </field>
                <field>
                        <vidName>magp</vidName>
                        <bits>96</bits>
                </field>
                <field>
                        <vidName>accelp</vidName>
                        <bits>96</bits>
                </field>
                <field>
                        <vidName>gyrop</vidName>
                        <bits>96</bits>
                </field>
                <field>
                        <vidName>currentTime</vidName>
                        <bits>32</bits>
                </field>
    </fields>
          <getfunction>N</getfunction>
          <setfunction>N</setfunction>
          <units>None</units>
</item>
<item name="comp2Rate">
          <longname>Composite 2 report rate</longname>
          <type>Int32</type>
          <defaultValue>0</defaultValue>
```

```
                <minValue>0</minValue>

                <maxValue>400</maxValue>

                <units>reports per second</units>

        </item>

        <item name="composite2">

                <longname>User modifiable composite object with
Roll,Pitch,Yaw,Lat,Lon,Alt,Temp</longname>

                <type>BitField</type>

                <flags>DATA_PERMANENT</flags>

                <dimension>16</dimension>

                <!-- This is the maximum number of fields possible -->

                <fields>

                        <!-- This determines the default number of fields -->

                        <field>

                                <vidName>pitch</vidName>

                                <bits>32</bits>

                        </field>

                        <field>

                                <vidName>roll</vidName>

                                <bits>32</bits>

                        </field>

                        <field>

                                <vidName>yaw</vidName>

                                <bits>32</bits>

                        </field>

                        <field>

                                <vidName>lat</vidName>

                                <bits>32</bits>

                        </field>

                        <field>
```

```
                        <vidName>lonG</vidName>
                        <bits>32</bits>
                    </field>
                    <field>
                        <vidName>alt</vidName>
                        <bits>32</bits>
                    </field>
                    <field>
                        <vidName>temperature</vidName>
                        <bits>32</bits>
                    </field>
                </fields>
                <getfunction>N</getfunction>
                <setfunction>N</setfunction>
                <units>None</units>
        </item>
        <item name="comp3Rate">
                <longname>Composite 3 report rate</longname>
                <type>Int32</type>
                <defaultValue>0</defaultValue>
                <minValue>0</minValue>
                <maxValue>400</maxValue>
                <units>reports per second</units>
        </item>
        <item name="composite3">
                <longname>User modifiable composite object with
Roll,Pitch,Yaw,Lat,Lon,Alt,Temp</longname>
                <type>BitField</type>
                <flags>DATA_PERMANENT</flags>
                <dimension>16</dimension>
```

```
<!-- This is the maximum number of fields possible -->
<fields>
	<!-- This determines the default number of fields -->
	<field>
		<vidName>pitch</vidName>
		<bits>32</bits>
	</field>
	<field>
		<vidName>roll</vidName>
		<bits>32</bits>
	</field>
	<field>
		<vidName>yaw</vidName>
		<bits>32</bits>
	</field>
	<field>
		<vidName>lat</vidName>
		<bits>32</bits>
	</field>
	<field>
		<vidName>lonG</vidName>
		<bits>32</bits>
	</field>
	<field>
		<vidName>alt</vidName>
		<bits>32</bits>
	</field>
	<field>
		<vidName>temperature</vidName>
		<bits>32</bits>
```

```
                </field>
        </fields>
        <getfunction>N</getfunction>
        <setfunction>N</setfunction>
        <units>None</units>
</item>
<item name="calmode">
        <longname>Calibration Mode</longname>
        <type>Ordinal</type>
        <defaultValue>0</defaultValue>
        <maxValue>3</maxValue>
        <labels>
                <label>CalModeOff</label>
                <label>CalMode3D</label>
                <label>CalMode2D</label>
                <label>User</label>
        </labels>
        <units>NONE</units>
</item>
<item name="calCommand">
        <longname>Specific State Within 3D Calibration Mode</longname>
        <type>Ordinal</type>
        <defaultValue>0</defaultValue>
        <maxValue>4</maxValue>
        <labels>
                <label>CALIBRATION_COMMAND_START</label>
                <label>CALIBRATION_COMMAND_CAPTURE_POINT</label>
                <label>CALIBRATION_COMMAND_END_CAPTURE</label>
                <label>CALIBRATION_COMMAND_END_CALIBRATION</label>
                <label>CALIBRATION_RESET</label>
```

```
                </labels>
                <setfunction>N</setfunction>
        </item>
        <item name="calNumPoints">
                <longname>Number of points collected for 3D/2D Calibration
Mode</longname>
                <type>Int32</type>
                <flags>DATA_READONLY</flags>
          <defaultValue>0</defaultValue>
          <minValue>0</minValue>
                <maxValue>12</maxValue>
                <units>point count</units>
    </item>
        <item name="magErr">
                <longname>Magnetic error term resulting from calibration calcs</longname>
        <type>Float32</type>
                <flags>DATA_READONLY</flags>
                <defaultValue>0.0</defaultValue>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <units>none</units>
    </item>
        <item name="yawErrEst">
                <longname>Estimated yaw error</longname>
                <type>Float32</type>
                <defaultValue>180.0f</defaultValue>
                <minValue>-9999.0</minValue>
                <maxValue>9999.0</maxValue>
                <setfunction>N</setfunction>
                <units>degrees</units>
```

```
</item>
<item name="absoluteYawErr">
	<longname>Absolute yaw error</longname>
	<type>Float32</type>
	<defaultValue>0.0</defaultValue>
	<minValue>-9999.0</minValue>
	<maxValue>9999.0</maxValue>
	<setfunction>N</setfunction>
	<units>degrees</units>
</item>
<item name="usertofactorycal">
	<longname>Command to put user cal in factory cal</longname>
	<type>Ordinal</type>
	<defaultValue>0</defaultValue>
	<maxValue>1</maxValue>
	<labels>
		<label>OFF</label>
		<label>ON</label>
	</labels>
	<setfunction>N</setfunction>
	<units>None</units>
</item>
<item name="km0">
	<longname>Algorithm constant - magnetic gain</longname>
	<type>Float32</type>
	<defaultValue>0.01</defaultValue>
	<minValue>-3.0E38</minValue>
	<maxValue>3.0E38</maxValue>
	<units>None</units>
</item>
```

```
        <item name="kaccelpAveraged">
                <longname></longname>
                <type>Float32</type>
                <defaultValue>0.01f</defaultValue>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <units>None</units>
        </item>
    <item name="magTimestamp">
         <longname>Time of mag sensor data acquisition</longname>
         <type>Int32</type>
         <defaultValue>0</defaultValue>
         <minValue>0</minValue>
         <maxValue>0x7FFFFFFF</maxValue>
         <units>ms</units>
     </item>
     <item name="accelGyroDecimator">
         <longname>raw accel and gyro is averaged every accelGyroDecimator number of
samples</longname>
         <type>Int32</type>
         <flags>DATA_PERMANENT</flags>
         <defaultValue>1</defaultValue>
         <minValue>0</minValue>
         <maxValue>0x7FFFFFFF</maxValue>
         <units>sample count</units>
     </item>
     <item name="accelDecimated">
                <longname>Averaged raw accel based on accelGyroDecimator</longname>
                <type>ArrayInt32</type>
                <dimension>3</dimension>
```

```
                <minValue>0xFFFF8000</minValue>
                <maxValue>0x00007FFF</maxValue>
                <values>
                        <value>0</value>
                        <value>0</value>
                        <value>0</value>
                </values>
                <units>counts</units>
        </item>
    <item name="magpTemperatureMean">
                <longname>Temperature average while the magp statistics were being
collected</longname>
        <type>Int32</type>
         <defaultValue>0</defaultValue>
         <minValue>0</minValue>
         <maxValue>0x7FFFFFFF</maxValue>
         <units>Celsius*256</units>
        </item>
    <item name="gyroDecimated">
                <longname>Averaged raw gyror based on accelGyroDecimator</longname>
                <type>ArrayInt32</type>
                <dimension>3</dimension>
                <minValue>0xFFFF8000</minValue>
                <maxValue>0x00007FFF</maxValue>
                <values>
                        <value>0</value>
                        <value>0</value>
                        <value>0</value>
                </values>
                <units>counts</units>
```

```
        </item>
    <item name="accelGyroTemperature">
            <longname>accel gyro sensor temperature - Celsius times 256</longname>
       <type>Int32</type>
        <defaultValue>1</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>Celsius*256</units>
       </item>
   <item name="accelGyroTimestamp">
        <longname>Time of accel/gyro sensor data acquisition at decimation
time</longname>
        <type>Int32</type>
        <defaultValue>0</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>ms</units>
    </item>
    <item name="magMean">
            <longname>Average raw mag since magCollectingStatistics and/or
magClearStatistics </longname>
            <type>ArrayInt32</type>
            <dimension>3</dimension>
            <minValue>0xFFFF8000</minValue>
            <maxValue>0x00007FFF</maxValue>
            <values>
                  <value>0</value>
                  <value>0</value>
                  <value>0</value>
            </values>
            <units>counts</units>
```

```
        </item>
    <item name="magVarianceMean">
            <longname>Average variance of mag data since magCollectingStatistics
and/or magClearStatistics </longname>
            <type>ArrayFloat32</type>
            <dimension>3</dimension>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <values>
                  <value>0.0</value>
                  <value>0.0</value>
                  <value>0.0</value>
            </values>
            <units>magrCounts</units>
       </item>
    <item name="magTemperatureMean">
            <longname>Temperature average while the raw mag statistics were being
collected</longname>
       <type>Int32</type>
        <defaultValue>0</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>Celsius*256</units>
       </item>
    <item name="magCollectingStatistics">
        <longname>Collect data for magnetometer statistics (1=ON, 0=OFF)</longname>
          <type>Ordinal</type>
          <defaultValue>0</defaultValue>
          <maxValue>1</maxValue>
          <labels>
                <label>OFF</label>
```

```
                    <label>ON</label>

                </labels>

            <units>None</units>

        </item>

        <item name="magClearStatistics">

            <longname>Command to clear the raw magnetometer statistics (1=ON,
0=OFF)</longname>

                <type>Ordinal</type>

                <defaultValue>0</defaultValue>

                <maxValue>1</maxValue>

                <labels>

                        <label>OFF</label>

                        <label>ON</label>

                </labels>

            <units>None</units>

        </item>

        <item name="magStatisticsInterval">

                <longname>Number of mag samples between statistical result
computations</longname>

            <type>Int32</type>

             <defaultValue>5</defaultValue>

             <minValue>0</minValue>

             <maxValue>0x7FFFFFFF</maxValue>

             <units>magrSamples</units>

            </item>

        <item name="magNSamples">

                <longname>Number of mag samples collected for statistical
computations</longname>

            <type>Int32</type>

             <defaultValue>0</defaultValue>

             <minValue>0</minValue>
```

```
            <maxValue>0x7FFFFFFF</maxValue>
            <units>magrSamples</units>
          </item>
      <item name="magTargetVariance">
                <longname>Desired variance for the mags used in computing
magEstimatedNSamples</longname>
            <flags>DATA_PERMANENT</flags>
            <type>Float32</type>
            <defaultValue>0.023f</defaultValue>
            <minValue>0.0</minValue>
            <maxValue>1000.0</maxValue>
            <units>None</units>
          </item>
      <item name="magEstimatedNSamples">
                <longname>Estimated number of samples to reach
magTargetVariance</longname>
                <type>ArrayInt32</type>
                <dimension>3</dimension>
                <minValue>0</minValue>
                <maxValue>0x7FFFFFFF</maxValue>
                <values>
                      <value>0</value>
                      <value>0</value>
                      <value>0</value>
                </values>
                <units>counts</units>
          </item>
      <item name="compassCollectingStatistics">
            <longname>Collect data for pitch/roll/yaw statistics (1=ON, 0=OFF)</longname>
              <type>Ordinal</type>
              <defaultValue>0</defaultValue>
```

```
            <maxValue>1</maxValue>
            <labels>
                  <label>OFF</label>
                  <label>ON</label>
            </labels>
        <units>None</units>
    </item>
   <item name="pitchMean">
            <longname>Average pitch since compassCollectingStatistics and/or
compassClearStatistics </longname>
            <type>Float32</type>
            <flags>DATA_READONLY</flags>
            <defaultValue>0.0</defaultValue>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <units>degrees</units>
       </item>
    <item name="pitchVarianceMean">
            <longname>Average variance of pitch data since
compassCollectingStatistics and/or pitchClearStatistics </longname>
            <type>Float32</type>
            <flags>DATA_READONLY</flags>
            <defaultValue>0.0</defaultValue>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <units>degrees</units>
       </item>
    <item name="pitchTemperatureMean">
            <longname>Temperature average while the pitch statistics were being
collected</longname>
       <type>Int32</type>
```

```
            <defaultValue>0</defaultValue>
            <minValue>0</minValue>
            <maxValue>0x7FFFFFFF</maxValue>
            <units>Celsius*256</units>
        </item>
        <item name="magCrossAxisCorrectionX">
                <longname>Calibration cross axis correction for the magnetometer X
value</longname>
                <type>ArrayFloat32</type>
                <flags>DATA_PERMANENT</flags>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>1.0</value>
                        <value>0.0</value>
                        <value>0.0</value>
                </values>
                <units>NONE</units>
        </item>
        <item name="magCrossAxisCorrectionY">
                <longname>Calibration cross axis correction for the magnetometer Y
value</longname>
                <type>ArrayFloat32</type>
                <flags>DATA_PERMANENT</flags>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0</value>
                        <value>1.0</value>
```

```
                        <value>0.0</value>
                </values>
                <units>NONE</units>
        </item>
        <item name="magCrossAxisCorrectionZ">
                <longname>Calibration cross axis correction for the magnetometer Z
value</longname>
                <type>ArrayFloat32</type>
                <flags>DATA_PERMANENT</flags>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0</value>
                        <value>0.0</value>
                        <value>1.0</value>
                </values>
                <units>NONE</units>
        </item>
        <!-- END of user versions of factory calibration -->
        <item name="magFieldCalX">
                <longname>In field calibration offset and scalers for the magnetometer X
value</longname>
                <type>ArrayFloat32</type>
                <flags>DATA_PERMANENT</flags>
                <dimension>2</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0f</value>
                        <value>1.0f</value>
```

```
                </values>
                <units>NONE</units>
        </item>
        <item name="magFieldCalY">
                <longname>In field calibration offset and scalers for the magnetometer Y
value</longname>
                <type>ArrayFloat32</type>
                <flags>DATA_PERMANENT</flags>
                <dimension>2</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0f</value>
                        <value>1.0f</value>
                </values>
                <units>NONE</units>
        </item>
        <item name="magFieldCalZ">
                <longname>In field calibration offset and scalers for the magnetometer Z
value</longname>
                <type>ArrayFloat32</type>
                <flags>DATA_PERMANENT</flags>
                <dimension>2</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0f</value>
                        <value>1.0f</value>
                </values>
                <units>NONE</units>
        </item>
```

```
    <item name="accelrange">
        <longname>Accelerometer scaling: 0=4g,1=8g,2=2g,3=16g</longname>
        <type>Ordinal</type>
        <flags>DATA_PERMANENT</flags>
        <defaultValue>0</defaultValue>
        <maxValue>3</maxValue>
        <labels>
            <label>4g</label>
            <label>8g</label>
            <label>2g</label>
            <label>16g</label>
        </labels>
        <units>None</units>
    </item>
    <item name="pitchClearStatistics">
        <longname>Command to clear the pitch statistics (1=ON, 0=OFF)</longname>
          <type>Ordinal</type>
          <defaultValue>0</defaultValue>
          <maxValue>1</maxValue>
          <labels>
                <label>OFF</label>
                <label>ON</label>
          </labels>
      <units>None</units>
    </item>
    <item name="pitchStatisticsInterval">
            <longname>Number of pitch samples between statistical result
computations</longname>
       <type>Int32</type>
        <defaultValue>5</defaultValue>
```

```
          <minValue>0</minValue>

          <maxValue>0x7FFFFFFF</maxValue>

          <units>pitchSamples</units>

        </item>

    <item name="pitchNSamples">

              <longname>Number of pitch samples collected for statistical
computations</longname>

        <type>Int32</type>

          <defaultValue>0</defaultValue>

          <minValue>0</minValue>

          <maxValue>0x7FFFFFFF</maxValue>

          <units>pitchSamples</units>

        </item>

    <item name="pitchTargetVariance">

              <longname>Desired variance for pitch used in computing
pitchEstimatedNSamples</longname>

          <flags>DATA_PERMANENT</flags>

          <type>Float32</type>

          <defaultValue>0.023f</defaultValue>

          <minValue>0.0</minValue>

          <maxValue>1000.0</maxValue>

          <units>None</units>

        </item>

    <item name="pitchEstimatedNSamples">

              <longname>Estimated number of samples to reach
pitchTargetVariance</longname>

        <type>Int32</type>

          <defaultValue>0</defaultValue>

          <minValue>0</minValue>

          <maxValue>0x7FFFFFFF</maxValue>

               <units>counts</units>
```

```
        </item>
    <item name="rollMean">
                <longname>Average roll since compassCollectingStatistics and/or
compassClearStatistics </longname>
                <type>Float32</type>
                <flags>DATA_READONLY</flags>
                <defaultValue>0.0</defaultValue>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <units>degrees</units>
        </item>
    <item name="rollVarianceMean">
                <longname>Average variance of roll data since compassCollectingStatistics
and/or rollClearStatistics </longname>
                <type>Float32</type>
                <flags>DATA_READONLY</flags>
                <defaultValue>0.0</defaultValue>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <units>degrees</units>
        </item>
    <item name="rollTemperatureMean">
                <longname>Temperature average while the roll statistics were being
collected</longname>
        <type>Int32</type>
         <defaultValue>0</defaultValue>
         <minValue>0</minValue>
         <maxValue>0x7FFFFFFF</maxValue>
         <units>Celsius*256</units>
        </item>
    <item name="rollClearStatistics">
```

```
        <longname>Command to clear the roll statistics (1=ON, 0=OFF)</longname>
           <type>Ordinal</type>
           <defaultValue>0</defaultValue>
           <maxValue>1</maxValue>
           <labels>
                  <label>OFF</label>
                  <label>ON</label>
           </labels>
       <units>None</units>
    </item>
    <item name="rollStatisticsInterval">
            <longname>Number of roll samples between statistical result
computations</longname>
       <type>Int32</type>
        <defaultValue>5</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>rollSamples</units>
       </item>
    <item name="rollNSamples">
            <longname>Number of roll samples collected for statistical
computations</longname>
       <type>Int32</type>
        <defaultValue>0</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>rollSamples</units>
       </item>
    <item name="rollTargetVariance">
            <longname>Desired variance for roll used in computing
rollEstimatedNSamples</longname>
```

```
            <flags>DATA_PERMANENT</flags>

            <type>Float32</type>

            <defaultValue>0.023f</defaultValue>

            <minValue>0.0</minValue>

            <maxValue>1000.0</maxValue>

            <units>None</units>

          </item>

    <item name="rollEstimatedNSamples">

                <longname>Estimated number of samples to reach
rollTargetVariance</longname>

          <type>Int32</type>

            <defaultValue>0</defaultValue>

            <minValue>0</minValue>

            <maxValue>0x7FFFFFFF</maxValue>

                <units>counts</units>

          </item>

    <item name="yawMean">

                <longname>Average yaw since compassCollectingStatistics and/or
compassClearStatistics </longname>

                <type>Float32</type>

                <flags>DATA_READONLY</flags>

                <defaultValue>0.0</defaultValue>

                <minValue>-3.0E38</minValue>

                <maxValue>3.0E38</maxValue>

                <units>degrees</units>

          </item>

    <item name="magpMean">

                <longname>Average magp since magpCollectingStatistics and/or
magpClearStatistics </longname>

                <type>ArrayFloat32</type>

                <dimension>3</dimension>
```

```
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0</value>
                        <value>0.0</value>
                        <value>0.0</value>
                </values>
                <units>milliGauss</units>
        </item>
    <item name="magpVarianceMean">
                <longname>Average variance of magp data since magpCollectingStatistics
and/or magpClearStatistics </longname>
                <type>ArrayFloat32</type>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0</value>
                        <value>0.0</value>
                        <value>0.0</value>
                </values>
                <units>g</units>
        </item>
   <item name="gyroEstimatedNSamples">
      <longname>Estimated number of samples to reach gyroTargetVariance</longname>
      <type>ArrayInt32</type>
      <dimension>3</dimension>
      <minValue>0</minValue>
      <maxValue>0x7FFFFFFF</maxValue>
      <values>
```

```
            <value>0</value>

            <value>0</value>

            <value>0</value>

        </values>

        <units>counts</units>

    </item>

      <item name="yawStatisticsInterval">

                <longname>Number of yaw samples between statistical result
computations</longname>

          <type>Int32</type>

           <defaultValue>5</defaultValue>

           <minValue>0</minValue>

           <maxValue>0x7FFFFFFF</maxValue>

           <units>yawSamples</units>

          </item>

      <item name="yawNSamples">

                <longname>Number of yaw samples collected for statistical
computations</longname>

          <type>Int32</type>

           <defaultValue>0</defaultValue>

           <minValue>0</minValue>

           <maxValue>0x7FFFFFFF</maxValue>

           <units>yawSamples</units>

          </item>

      <item name="yawTargetVariance">

                <longname>Desired variance for yaw used in computing
yawEstimatedNSamples</longname>

           <flags>DATA_PERMANENT</flags>

           <type>Float32</type>

           <defaultValue>0.023f</defaultValue>

           <minValue>0.0</minValue>
```

```
            <maxValue>1000.0</maxValue>

            <units>None</units>

        </item>

    <item name="yawEstimatedNSamples">

            <longname>Estimated number of samples to reach
yawTargetVariance</longname>

            <type>Int32</type>

            <defaultValue>0</defaultValue>

            <minValue>0</minValue>

            <maxValue>0x7FFFFFFF</maxValue>

            <units>counts</units>

        </item>

    <item name="accelMean">

            <longname>Average raw accel since accelCollectingStatistics and/or
accelClearStatistics </longname>

            <type>ArrayInt32</type>

            <dimension>3</dimension>

            <minValue>0xFFFF8000</minValue>

            <maxValue>0x00007FFF</maxValue>

            <values>

                <value>0</value>

                <value>0</value>

                <value>0</value>

            </values>

            <units>counts</units>

        </item>

    <item name="accelVarianceMean">

            <longname>Average variance of accel data since accelCollectingStatistics
and/or accelClearStatistics </longname>

            <type>ArrayFloat32</type>

            <dimension>3</dimension>
```

```
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0</value>
                        <value>0.0</value>
                        <value>0.0</value>
                </values>
                <units>accelrCounts</units>
        </item>
    <item name="accelTemperatureMean">
                <longname>Temperature average while the accel statistics were being
collected</longname>
        <type>Int32</type>
         <defaultValue>0</defaultValue>
         <minValue>0</minValue>
         <maxValue>0x7FFFFFFF</maxValue>
         <units>Celsius*256</units>
        </item>
    <item name="accelCollectingStatistics">
         <longname>Collect data for accelerometer statistics (1=ON, 0=OFF)</longname>
            <type>Ordinal</type>
            <defaultValue>0</defaultValue>
            <maxValue>1</maxValue>
            <labels>
                   <label>OFF</label>
                   <label>ON</label>
            </labels>
        <units>None</units>
    </item>
    <item name="accelClearStatistics">
```

```
        <longname>Command to clear the accelerometer statistics (1=ON,
0=OFF)</longname>

           <type>Ordinal</type>

           <defaultValue>0</defaultValue>

           <maxValue>1</maxValue>

           <labels>

                  <label>OFF</label>

                  <label>ON</label>

           </labels>

       <units>None</units>

    </item>

    <item name="accelStatisticsInterval">

             <longname>Number of accel samples between statistical result
computations</longname>

       <type>Int32</type>

        <defaultValue>5</defaultValue>

        <minValue>0</minValue>

        <maxValue>0x7FFFFFFF</maxValue>

        <units>accelrSamples</units>

       </item>

    <item name="accelTargetVariance">

             <longname>Desired variance for the accels used in computing
accelEstimatedNSamples</longname>

        <flags>DATA_PERMANENT</flags>

        <type>Float32</type>

        <defaultValue>0.023f</defaultValue>

        <minValue>0.0</minValue>

        <maxValue>1000.0</maxValue>

        <units>None</units>

       </item>

    <item name="accelNSamples">
```

```
            <longname>Number of accel samples collected for statistical
computations</longname>

       <type>Int32</type>

        <defaultValue>0</defaultValue>

        <minValue>0</minValue>

        <maxValue>0x7FFFFFFF</maxValue>

        <units>accelrSamples</units>

      </item>

   <item name="accelEstimatedNSamples">

            <longname>Estimated number of samples to reach
accelTargetVariance</longname>

            <type>ArrayInt32</type>

            <dimension>3</dimension>

            <minValue>0</minValue>

            <maxValue>0x7FFFFFFF</maxValue>

            <values>

                  <value>0</value>

                  <value>0</value>

                  <value>0</value>

            </values>

            <units>counts</units>

      </item>

   <item name="gyroMean">

     <longname>Average raw gyro since gyroCollectingStatistics and/or
gyroClearStatistics </longname>

     <type>ArrayInt32</type>

     <dimension>3</dimension>

     <minValue>0xFFFF8000</minValue>

     <maxValue>0x00007FFF</maxValue>

     <values>

        <value>0</value>
```

```
            <value>0</value>
            <value>0</value>
        </values>
        <units>counts</units>
    </item>
     <item name="gyroVarianceMean">
        <longname>Average variance of gyro data since gyroCollectingStatistics and/or
gyroClearStatistics </longname>
        <type>ArrayFloat32</type>
        <dimension>3</dimension>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <values>
            <value>0.0</value>
            <value>0.0</value>
            <value>0.0</value>
        </values>
        <units>gyroCounts</units>
    </item>
     <item name="gyroTemperatureMean">
        <longname>Temperature average while the gyro statistics were being
collected</longname>
         <type>Int32</type>
          <defaultValue>0</defaultValue>
          <minValue>0</minValue>
          <maxValue>0x7FFFFFFF</maxValue>
          <units>Celsius*256</units>
    </item>
     <item name="gyroCollectingStatistics">
          <longname>Collect data for gyro statistics (1=ON, 0=OFF)</longname>
         <type>Ordinal</type>
```

```
        <defaultValue>0</defaultValue>
        <maxValue>1</maxValue>
        <labels>
           <label>OFF</label>
           <label>ON</label>
        </labels>
  <units>None</units>
   </item>
   <item name="gyroClearStatistics">
        <longname>Command to clear the gyro statistics (1=ON, 0=OFF)</longname>
        <type>Ordinal</type>
        <defaultValue>0</defaultValue>
        <maxValue>1</maxValue>
        <labels>
           <label>OFF</label>
           <label>ON</label>
        </labels>
  <units>None</units>
   </item>
   <item name="gyroStatisticsInterval">
     <longname>Number of gyro samples between statistical result
computations</longname>
        <type>Int32</type>
        <defaultValue>5</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>gyroSamples</units>
  </item>
   <item name="gyroNSamples">
     <longname>Number of gyro samples collected for statistical
computations</longname>
```

```
        <type>Int32</type>
         <defaultValue>0</defaultValue>
         <minValue>0</minValue>
         <maxValue>0x7FFFFFFF</maxValue>
         <units>gyroSamples</units>
    </item>
    <item name="gyroTargetVariance">
       <longname>Desired variance for the gyros used in computing
gyroEstimatedNSamples</longname>
         <flags>DATA_PERMANENT</flags>
         <type>Float32</type>
         <defaultValue>0.023f</defaultValue>
         <minValue>0.0</minValue>
         <maxValue>1000.0</maxValue>
         <units>None</units>
    </item>
        <item name="gyroFieldCalX">
               <longname>In field calibration offset and scaler for the gyro X value
(unused)</longname>
               <type>ArrayFloat32</type>
               <flags>DATA_PERMANENT</flags>
               <dimension>2</dimension>
               <minValue>-3.0E38</minValue>
               <maxValue>3.0E38</maxValue>
               <values>
                      <value>0.0</value>
                      <value>1.0</value>
               </values>
               <units>NONE</units>
        </item>
        <item name="gyroFieldCalY">
```

```
            <longname>In field calibration offset and scaler for the gyro Y value
(unused)</longname>

            <type>ArrayFloat32</type>

            <flags>DATA_PERMANENT</flags>

            <dimension>2</dimension>

            <minValue>-3.0E38</minValue>

            <maxValue>3.0E38</maxValue>

            <values>

                  <value>0.0</value>

                  <value>1.0</value>

            </values>

            <units>NONE</units>

      </item>

      <item name="gyroFieldCalZ">

            <longname>In field calibration offset and scaler for the gyro Z value
(unused)</longname>

            <type>ArrayFloat32</type>

            <flags>DATA_PERMANENT</flags>

            <dimension>2</dimension>

            <minValue>-3.0E38</minValue>

            <maxValue>3.0E38</maxValue>

            <values>

                  <value>0.0</value>

                  <value>1.0</value>

            </values>

            <units>NONE</units>

      </item>

      <item name="gyroCrossAxisCorrectionX">

            <longname>Calibration cross axis correction for the gyro X value
(unused)</longname>

            <type>ArrayFloat32</type>
```

```
                <flags>DATA_PERMANENT</flags>

                <dimension>3</dimension>

                <minValue>-3.0E38</minValue>

                <maxValue>3.0E38</maxValue>

                <values>

                        <value>1.0</value>

                        <value>0.0</value>

                        <value>0.0</value>

                </values>

                <units>NONE</units>

        </item>

        <item name="gyroCrossAxisCorrectionY">

                <longname>Calibration cross axis correction for the gyro Y value
(unused)</longname>

                <type>ArrayFloat32</type>

                <flags>DATA_PERMANENT</flags>

                <dimension>3</dimension>

                <minValue>-3.0E38</minValue>

                <maxValue>3.0E38</maxValue>

                <values>

                        <value>0.0</value>

                        <value>1.0</value>

                        <value>0.0</value>

                </values>

                <units>NONE</units>

        </item>

        <item name="gyroCrossAxisCorrectionZ">

                <longname>Calibration cross axis correction for the gyro Z value
(unused)</longname>

                <type>ArrayFloat32</type>

                <flags>DATA_PERMANENT</flags>
```

```
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0</value>
                        <value>0.0</value>
                        <value>1.0</value>
                </values>
                <units>NONE</units>
        </item>
        <item name="InvokeGyroOffsetCal">
                <longname>Command the gyro offset calibration to start
(unused)</longname>
                <type>Ordinal</type>
                <defaultValue>0</defaultValue>
                <maxValue>1</maxValue>
                <labels>
                        <label>OFF</label>
                        <label>ON</label>
                </labels>
                <units>None</units>
        </item>
        <item name="boresightMatrixX">
                <longname>Boresight orientation for the X component</longname>
                <type>ArrayFloat32</type>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>1.0</value>
```

```
                <value>0.0</value>
                <value>0.0</value>
        </values>
        <setfunction>Y</setfunction>
        <units>NONE</units>
</item>
<item name="boresightMatrixY">
        <longname>Boresight orientation for the Y component</longname>
        <type>ArrayFloat32</type>
        <dimension>3</dimension>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <values>
                <value>0.0</value>
                <value>1.0</value>
                <value>0.0</value>
        </values>
        <setfunction>Y</setfunction>
        <units>NONE</units>
</item>
<item name="boresightMatrixZ">
        <longname>Boresight orientation for the Z component</longname>
        <type>ArrayFloat32</type>
        <dimension>3</dimension>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <values>
                <value>0.0</value>
                <value>0.0</value>
                <value>1.0</value>
```

```
            </values>
            <setfunction>Y</setfunction>
            <units>NONE</units>
      </item>
      <item name="lowpower">
            <longname>Low Power Mode</longname>
       <type>Ordinal</type>
       <defaultValue>0</defaultValue>
            <maxValue>1</maxValue>
       <labels>
                  <label>off</label>
                  <label>on</label>
       </labels>
   <units>None</units>
   </item>
    <item name="temperature">
            <longname>Temperature - Celcius</longname>
            <type>Float32</type>
        <flags>DATA_READONLY</flags>
       <defaultValue>0</defaultValue>
            <minValue>-1000</minValue>
            <maxValue>1000</maxValue>
   <units>None</units>
   </item>
      <item name="kgyroOffsetCalFactor">
            <longname>Gyro offset field cal factor (unused)</longname>
            <type>Float32</type>
            <flags>DATA_PERMANENT</flags>
            <defaultValue>0.996</defaultValue>
            <minValue>-3.0E38</minValue>
```

```
        <maxValue>3.0E38</maxValue>
        <units>None</units>
    </item>
    <item name="Kw">
        <longname>Algorithm gyro factor (unused)</longname>
        <type>Float32</type>
        <flags>DATA_READONLY</flags>
        <defaultValue>0.0</defaultValue>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>None</units>
    </item>
    <item name="Ka">
        <longname>Algorithm accel factor (unused)</longname>
        <type>Float32</type>
        <flags>DATA_READONLY</flags>
        <defaultValue>0.0</defaultValue>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>None</units>
    </item>
    <item name="cp1">
        <longname>AccelEst crossed with MagEst unit vector</longname>
        <type>ArrayFloat32</type>
        <flags>DATA_READONLY</flags>
        <dimension>3</dimension>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>NONE</units>
    </item>
```

```
<item name="cp2">
        <longname>cp1 crossed with AccelEst unit vector</longname>
        <type>ArrayFloat32</type>
        <flags>DATA_READONLY</flags>
        <dimension>3</dimension>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>NONE</units>
</item>
<item name="accelEst">
        <longname>Estimated accelerometer unit vector</longname>
        <type>ArrayFloat32</type>
        <flags>DATA_READONLY</flags>
        <dimension>3</dimension>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>NONE</units>
</item>
<item name="magEst">
        <longname>Estimated magnetometer unit vector</longname>
        <type>ArrayFloat32</type>
        <flags>DATA_READONLY</flags>
        <dimension>3</dimension>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>NONE</units>
</item>
<item name="mEstimate">
        <longname>Gyro advanced magnetometer unit vector (unused)</longname>
        <type>ArrayFloat32</type>
```

```
        <flags>DATA_READONLY</flags>
        <dimension>3</dimension>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>NONE</units>
</item>
<item name="aEstimate">
        <longname>Gyro advanced accelerometer unit vector (unused)</longname>
        <type>ArrayFloat32</type>
        <flags>DATA_READONLY</flags>
        <dimension>3</dimension>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>NONE</units>
</item>
<item name="nmeaecho">
        <longname>Echoes NMEA characters</longname>
        <type>Ordinal</type>
        <defaultValue>0</defaultValue>
        <maxValue>1</maxValue>
        <labels>
                <label>off</label>
                <label>on</label>
        </labels>
        <units>None</units>
</item>
<item name="nmeaignorechecksum">
        <longname>Ignore received NMEA checksum</longname>
        <type>Ordinal</type>
        <defaultValue>1</defaultValue>
```

```
                <maxValue>1</maxValue>
                <labels>
                        <label>off</label>
                        <label>on</label>
                </labels>
                <units>None</units>
        </item>
        <item name="restoreFieldCal">
                <longname>Set Magnetic field calibration back to factory
default</longname>
                <type>Ordinal</type>
                <defaultValue>0</defaultValue>
                <maxValue>1</maxValue>
                <labels>
                        <label>off</label>
                        <label>on</label>
                </labels>
                <units>None</units>
                <setfunction>N</setfunction>
        </item>
    <item name="pMatrix0">
         <longname>Platform Matrix Row 0</longname>
                <type>ArrayFloat32</type>
         <flags>DATA_READONLY</flags>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
               <value>1.0f</value>
                        <value>0.0f</value>
```

```
                    <value>0.0f</value>
                </values>
            <units>NONE</units>
        </item>
    <item name="pMatrix1">
            <longname>Platform Matrix Row 1</longname>
                <type>ArrayFloat32</type>
            <flags>DATA_READONLY</flags>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0f</value>
                    <value>1.0f</value>
                        <value>0.0f</value>
                </values>
            <units>NONE</units>
        </item>
    <item name="pMatrix2">
            <longname>Platform Matrix Row 2</longname>
            <type>ArrayFloat32</type>
            <flags>DATA_READONLY</flags>
                <dimension>3</dimension>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
                <values>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>1.0f</value>
                </values>
```

```
            <units>NONE</units>
          </item>
        <item name="printmask">
            <longname>Items that get printed </longname>
            <type>Int32</type>
            <defaultValue>0</defaultValue>
            <minValue>0</minValue>
            <maxValue>0x7fffffff</maxValue>
            <units>Bits</units>
        </item>
        <item name="printtrigger">
            <longname>Event that causes sampled data to print </longname>
            <type>Int32</type>
            <defaultValue>0</defaultValue>
            <minValue>0</minValue>
            <maxValue>0x7fffffff</maxValue>
            <units>Bits</units>
        </item>
        <item name="printmodulus">
            <longname>How many counts of the print trigger cause the print</longname>
            <type>Int32</type>
            <defaultValue>4</defaultValue>
            <minValue>0</minValue>
            <maxValue>10000</maxValue>
            <units>Counts</units>
        </item>


    <item name="centripetalCorrection">
                <longname>Enable Centripetal acceleration correction of acceleration
    </longname>
```

```
            <type>Ordinal</type>
            <flags>DATA_PERMANENT</flags>
            <defaultValue>1</defaultValue>
            <maxValue>1</maxValue>
            <labels>
                    <label>Off</label>
                    <label>On</label>
            </labels>
            <units>None</units>
    </item>
    <item name="calPointsNumToAvg">
       <longname>Number of samples to be averaged for a 3D/2D Calibration Mode capture point</longname>
       <type>Int32</type>
       <defaultValue>10</defaultValue>
       <minValue>1</minValue>
       <maxValue>6000</maxValue>
        <setfunction>Y</setfunction>
       <units>sample count</units>
    </item>
    <item name="InvokeTare">
       <longname>Command to update the boresight matrix based on current orientation</longname>
       <type>Ordinal</type>
       <defaultValue>0</defaultValue>
       <maxValue>1</maxValue>
       <labels>
              <label>OFF</label>
              <label>ON</label>
       </labels>
       <units>None</units>
```

```
    </item>
    <item name="restartCompassCalcs">
      <longname>Command to start compass calcs fresh (forget previous expected
position)</longname>
      <type>Ordinal</type>
      <defaultValue>0</defaultValue>
      <maxValue>1</maxValue>
      <labels>
            <label>OFF</label>
            <label>ON</label>
      </labels>
      <units>None</units>
      </item>
    <item name="gyroSigma">
            <longname>Gyro stability factor</longname>
            <type>Float32</type>
           <flags>DATA_PERMANENT</flags>
            <defaultValue>8.9e-5</defaultValue>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <units>N/A</units>
      </item>
    <item name="magErr2">
      <longname>Magnetic heading error estimate</longname>
      <type>Float32</type>
      <defaultValue>999.0</defaultValue>
      <minValue>0.0</minValue>
      <maxValue>360.0</maxValue>
      <units>degrees</units>
    </item>
```

```
    <item name="magFieldCalErr">
       <longname>Estimated Field Cal Error in Deg due to variability in buffered
magnetics</longname>
       <type>Float32</type>
          <flags>DATA_PERMANENT</flags>
       <defaultValue>180.0f</defaultValue>
       <minValue>0.0</minValue>
       <maxValue>360.0</maxValue>
       <units>degrees</units>
    </item>
    <item name="magFieldCalTot">
       <longname>Average of magnetic field calibration point vector
magnitudes</longname>
       <type>Float32</type>
       <flags>DATA_PERMANENT</flags>
       <defaultValue>99999.0</defaultValue>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
       <units>N/A</units>
    </item>
    <item name="magHorzFieldCalTot">
       <longname>Average of the horizontal component of the magnetic field calibration
points</longname>
       <type>Float32</type>
       <flags>DATA_PERMANENT</flags>
       <defaultValue>99999.0</defaultValue>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
       <units>N/A</units>
    </item>
    <item name="calPointDistribution">
```

```
            <longname>Factor indicating quality of in-field calibration points</longname>
            <flags>DATA_PERMANENT</flags>
            <type>Float32</type>
            <defaultValue>0.0</defaultValue>
            <minValue>0.0</minValue>
            <maxValue>1000.0</maxValue>
                  <units>None</units>
         </item>
      <item name="calPointDistSigma">
                  <longname>Calibration point distribution indicator factor</longname>
                  <type>Float32</type>
                 <flags>DATA_PERMANENT</flags>
                  <defaultValue>30.0</defaultValue>
                  <minValue>-3.0E38</minValue>
                  <maxValue>3.0E38</maxValue>
                  <units>N/A</units>
         </item>
      <item name="magCalPoints">
         <longname>Array with up to calNumPoints of mag field cal captured
points</longname>
         <type>ArrayFloat32</type>
         <flags>DATA_READONLY</flags>
         <dimension>36</dimension>
                  <minValue>-3.0E38</minValue>
                  <maxValue>3.0E38</maxValue>
                              <value>0.0f</value>
         <units>NONE</units>
         </item>
      <item name="accelCalPoints">
         <longname>Array with up to calNumPoints of accelerometer field cal captured
points</longname>
```

```
        <type>ArrayFloat32</type>
        <flags>DATA_READONLY</flags>
        <dimension>36</dimension>
               <minValue>-3.0E38</minValue>
               <maxValue>3.0E38</maxValue>
                  <value>0.0f</value>
               <units>NONE</units>
        </item>
        <item name="infieldCalMu">
               <longname>Infield cal magBias configuration parameter</longname>
               <type>Float32</type>
                  <flags>DATA_PERMANENT</flags>
               <defaultValue>6.0e-2f</defaultValue>
               <minValue>-3.0E38</minValue>
               <maxValue>3.0E38</maxValue>
               <units>None</units>
        </item>
        <item name="infieldCalMu2">
               <longname>Infield cal magCrossAxisCorrection configuration
parameter</longname>
               <type>Float32</type>
                  <flags>DATA_PERMANENT</flags>
               <defaultValue>6.0e-7f</defaultValue>
               <minValue>-3.0E38</minValue>
               <maxValue>3.0E38</maxValue>
               <units>None</units>
        </item>
    <item name="infieldCalMu3">
         <longname>In-field calibration factor controlling the off-axis
terms</longname>
         <type>Float32</type>
```

```
            <flags>DATA_PERMANENT</flags>

            <defaultValue>6.0e-7</defaultValue>

            <minValue>0.0</minValue>

            <maxValue>100.0</maxValue>

            <units>none</units>

        </item>

           <item name="magLinearized">

                  <longname>Calibrated mag before field cal and cross-axis correction
(unused)</longname>

                  <type>ArrayFloat32</type>

                  <dimension>3</dimension>

                  <minValue>-3.0E38</minValue>

                  <maxValue>3.0E38</maxValue>

                  <values>

                         <value>0.0</value>

                         <value>0.0</value>

                         <value>0.0</value>

                  </values>

                  <units>NONE</units>

           </item>

        <item name="magPointQF">

            <longname>Quality factor for the captured magnetic calibration
points</longname>

            <flags>DATA_PERMANENT</flags>

            <type>Float32</type>

            <defaultValue>0.0</defaultValue>

            <minValue>0.0</minValue>

            <maxValue>50000.0</maxValue>

            <units>None</units>

        </item>

        <item name="possibleMagPointQF">
```

```
        <longname>Possible quality factor if the current magnetic point is added to
the captured calibration points</longname>

        <type>Float32</type>

        <defaultValue>0.0</defaultValue>

        <minValue>0.0</minValue>

        <maxValue>50000.0</maxValue>

        <units>None</units>

    </item>

    <item name="wmmGDMagnitude">

        <longname>Magnitude of the World Magnetic Model computed Earth
Field</longname>

        <type>Float32</type>

        <flags>DATA_PERMANENT</flags>

        <defaultValue>450.0</defaultValue>

        <minValue>0.1</minValue>

        <maxValue>50000.0</maxValue>

        <units>milliGauss</units>

    </item>


    <item name="timeStamp">

        <longname>Time stamp taking at beginning of compass calcuations</longname>

        <type>Int32</type>

        <defaultValue>0</defaultValue>

        <minValue>0</minValue>

        <maxValue>0x7FFFFFFF</maxValue>

        <units>milliseconds</units>

    </item>

    <item name="currentTime">

        <longname>Processor time of the request for this variable</longname>

        <type>Int32</type>

        <defaultValue>0</defaultValue>
```

```
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <getfunction>Y</getfunction>
        <units>milliseconds</units>
    </item>
    <item name="calDeqDelta">
        <longname>Field calibration derivative delta (unused)</longname>
              <type>Float32</type>
        <flags>DATA_PERMANENT</flags>
        <defaultValue>1.0e-2</defaultValue>
              <minValue>-3.0E38</minValue>
              <maxValue>3.0E38</maxValue>
        <units>N/A</units>
    </item>
    <item name="calStatus">
       <longname>Provide in-field calibration status</longname>
       <type>Ordinal</type>
         <flags>DATA_PERMANENT</flags>
       <defaultValue>0</defaultValue>
       <maxValue>3</maxValue>
       <labels>
              <label>DEFAULT</label>
              <label>INPROGRESS</label>
              <label>COMPLETE</label>
              <label>ABORTED</label>
       </labels>
       <units>None</units>
    </item>
       <item name="kdeviation">
              <longname>Algorithm constant - gain for single-pole IIR filter</longname>
```

```
            <type>Float32</type>
            <flags>DATA_PERMANENT</flags>
            <defaultValue>0.002</defaultValue>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <units>None</units>
   </item>



<item name="magpCollectingStatistics">
    <longname>Collect data for magp statistics (1=ON, 0=OFF)</longname>
       <type>Ordinal</type>
       <defaultValue>0</defaultValue>
       <maxValue>1</maxValue>
       <labels>
              <label>OFF</label>
              <label>ON</label>
       </labels>
   <units>None</units>
</item>
<item name="magpClearStatistics">
    <longname>Command to clear the magp statistics (1=ON, 0=OFF)</longname>
       <type>Ordinal</type>
       <defaultValue>0</defaultValue>
       <maxValue>1</maxValue>
       <labels>
              <label>OFF</label>
              <label>ON</label>
       </labels>
```

```
        <units>None</units>
    </item>
    <item name="magpStatisticsInterval">
            <longname>Number of magp samples between statistical result
computations</longname>
        <type>Int32</type>
         <defaultValue>5</defaultValue>
         <minValue>0</minValue>
         <maxValue>0x7FFFFFFF</maxValue>
         <units>magpSamples</units>
        </item>
    <item name="magpNSamples">
            <longname>Number of magp samples collected for statistical
computations</longname>
        <type>Int32</type>
         <defaultValue>0</defaultValue>
         <minValue>0</minValue>
         <maxValue>0x7FFFFFFF</maxValue>
         <units>magpSamples</units>
        </item>
    <item name="magpTargetVariance">
            <longname>Desired variance for the magps used in computing
magpEstimatedNSamples</longname>
         <flags>DATA_PERMANENT</flags>
         <type>Float32</type>
         <defaultValue>0.023f</defaultValue>
         <minValue>0.0</minValue>
         <maxValue>1000.0</maxValue>
         <units>None</units>
        </item>
    <item name="magpEstimatedNSamples">
```

```
            <longname>Estimated number of samples to reach
magpTargetVariance</longname>

            <type>ArrayInt32</type>

            <dimension>3</dimension>

            <minValue>0</minValue>

            <maxValue>0x7FFFFFFF</maxValue>

            <values>

                  <value>0</value>

                  <value>0</value>

                  <value>0</value>

            </values>

            <units>counts</units>

      </item>


   <item name="accelpMean">

            <longname>Average accelp since accelpCollectingStatistics and/or
accelpClearStatistics </longname>

            <type>ArrayFloat32</type>

            <dimension>3</dimension>

            <minValue>-3.0E38</minValue>

            <maxValue>3.0E38</maxValue>

            <values>

                  <value>0.0</value>

                  <value>0.0</value>

                  <value>0.0</value>

            </values>

            <units>counts</units>

      </item>

   <item name="accelpVarianceMean">

            <longname>Average variance of accelp data since
accelpCollectingStatistics and/or accelpClearStatistics </longname>
```

```
                <type>ArrayFloat32</type>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0</value>
                        <value>0.0</value>
                        <value>0.0</value>
                </values>
                <units>g</units>
        </item>
    <item name="accelpTemperatureMean">
                <longname>Temperature average while the accelp statistics were being
collected</longname>
        <type>Int32</type>
         <defaultValue>0</defaultValue>
         <minValue>0</minValue>
         <maxValue>0x7FFFFFFF</maxValue>
         <units>Celsius*256</units>
        </item>
    <item name="accelpCollectingStatistics">
         <longname>Collect data for accelp statistics (1=ON, 0=OFF)</longname>
            <type>Ordinal</type>
            <defaultValue>0</defaultValue>
            <maxValue>1</maxValue>
            <labels>
                   <label>OFF</label>
                   <label>ON</label>
            </labels>
        <units>None</units>
```

```
    </item>
    <item name="accelpClearStatistics">
        <longname>Command to clear the accelp statistics (1=ON, 0=OFF)</longname>
           <type>Ordinal</type>
           <defaultValue>0</defaultValue>
           <maxValue>1</maxValue>
           <labels>
                  <label>OFF</label>
                  <label>ON</label>
           </labels>
       <units>None</units>
    </item>
    <item name="accelpStatisticsInterval">
              <longname>Number of accelp samples between statistical result
computations</longname>
       <type>Int32</type>
        <defaultValue>5</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>accelpSamples</units>
       </item>
    <item name="accelpNSamples">
              <longname>Number of accelp samples collected for statistical
computations</longname>
       <type>Int32</type>
        <defaultValue>0</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>accelpSamples</units>
       </item>
    <item name="accelpTargetVariance">
```

```
            <longname>Desired variance for the accelps used in computing
accelpEstimatedNSamples</longname>

        <flags>DATA_PERMANENT</flags>

        <type>Float32</type>

        <defaultValue>0.023f</defaultValue>

        <minValue>0.0</minValue>

        <maxValue>1000.0</maxValue>

        <units>None</units>

      </item>

    <item name="accelpEstimatedNSamples">

            <longname>Estimated number of samples to reach
accelpTargetVariance</longname>

            <type>ArrayInt32</type>

            <dimension>3</dimension>

            <minValue>0</minValue>

            <maxValue>0x7FFFFFFF</maxValue>

            <values>

                  <value>0</value>

                  <value>0</value>

                  <value>0</value>

            </values>

            <units>counts</units>

      </item>

    <item name="gyropMean">

            <longname>Average gyrop since gyropCollectingStatistics and/or
gyropClearStatistics </longname>

            <type>ArrayFloat32</type>

            <dimension>3</dimension>

            <minValue>-3.0E38</minValue>

            <maxValue>3.0E38</maxValue>

            <values>
```

```
                        <value>0.0</value>
                        <value>0.0</value>
                        <value>0.0</value>
                </values>
                <units>counts</units>
        </item>
    <item name="gyropVarianceMean">
                <longname>Average variance of gyrop data since gyropCollectingStatistics
and/or gyropClearStatistics </longname>
                <type>ArrayFloat32</type>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0</value>
                        <value>0.0</value>
                        <value>0.0</value>
                </values>
                <units>g</units>
        </item>
    <item name="gyropTemperatureMean">
                <longname>Temperature average while the gyrop statistics were being
collected</longname>
        <type>Int32</type>
         <defaultValue>0</defaultValue>
         <minValue>0</minValue>
         <maxValue>0x7FFFFFFF</maxValue>
         <units>Celsius*256</units>
        </item>
        <!-- user calibration gyroscope values -->
    <item name="gyropCollectingStatistics">
```

```
        <longname>Collect data for gyrop statistics (1=ON, 0=OFF)</longname>
           <type>Ordinal</type>
           <defaultValue>0</defaultValue>
           <maxValue>1</maxValue>
           <labels>
                  <label>OFF</label>
                  <label>ON</label>
           </labels>
      <units>None</units>
    </item>
    <item name="gyropClearStatistics">
        <longname>Command to clear the gyrop statistics (1=ON, 0=OFF)</longname>
           <type>Ordinal</type>
           <defaultValue>0</defaultValue>
           <maxValue>1</maxValue>
           <labels>
                  <label>OFF</label>
                  <label>ON</label>
           </labels>
      <units>None</units>
    </item>
    <item name="gyropStatisticsInterval">
             <longname>Number of gyrop samples between statistical result
computations</longname>
       <type>Int32</type>
        <defaultValue>5</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>gyropSamples</units>
       </item>
```

```
    <item name="gyropNSamples">
            <longname>Number of gyrop samples collected for statistical
computations</longname>
       <type>Int32</type>
        <defaultValue>0</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>gyropSamples</units>
      </item>
    <item name="gyropTargetVariance">
            <longname>Desired variance for the gyrops used in computing
gyropEstimatedNSamples</longname>
        <flags>DATA_PERMANENT</flags>
        <type>Float32</type>
        <defaultValue>0.023f</defaultValue>
        <minValue>0.0</minValue>
        <maxValue>1000.0</maxValue>
        <units>None</units>
      </item>
    <item name="gyropEstimatedNSamples">
            <longname>Estimated number of samples to reach
gyropTargetVariance</longname>
            <type>ArrayInt32</type>
            <dimension>3</dimension>
            <minValue>0</minValue>
            <maxValue>0x7FFFFFFF</maxValue>
            <values>
                  <value>0</value>
                  <value>0</value>
                  <value>0</value>
            </values>
```

```
                <units>counts</units>
        </item>
              <item name="powerMode">
          <longname>Controls the power mode settings</longname>
      <type>Ordinal</type>
      <defaultValue>0</defaultValue>
          <maxValue>2</maxValue>
        <labels>
               <label>NormalPower</label>
               <label>LowPower</label>
               <label>StandyPower</label>
        </labels>
      <units>None</units>
         </item>

        <item name="magRawTC">
               <longname>User copy of raw mag at factory calibration base temperature
</longname>
               <type>ArrayFloat32</type>
               <flags>DATA_PERMANENT|DATA_SUBSYSTEM</flags>
               <dimension>3</dimension>
               <minValue>-3.0E38</minValue>
               <maxValue>3.0E38</maxValue>
               <values>
                      <value>0.0f</value>
                      <value>0.0f</value>
                      <value>0.0f</value>
               </values>
               <units>None</units>
        </item>
```

```
        <item name="accelRawTC">
                <longname>User copy of raw accel at factory calibration base temperature
</longname>
                <type>ArrayFloat32</type>
                <flags>DATA_PERMANENT|DATA_SUBSYSTEM</flags>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                </values>
                <units>None</units>
        </item>
        <item name="gyroRawTC">
                <longname>User copy of raw gyro at factory calibration base temperature
</longname>
                <type>ArrayFloat32</type>
                <flags>DATA_PERMANENT|DATA_SUBSYSTEM</flags>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                </values>
                <units>None</units>
        </item>
```

```
        <!-- user magnetic offset temperature correction values -->
        <item name="magCalTemperatureRaw">
                <longname>Temperature at which the Dycome temperature comps are computed
for each mag X,Y,Z (256*C)</longname>
                <type>ArrayInt32</type>
                <dimension>3</dimension>
       <minValue>-32768</minValue>
       <maxValue>32767</maxValue>
                <flags>DATA_PERMANENT|DATA_SUBSYSTEM</flags>
                <values>
              <value>7450</value>
              <value>7450</value>
              <value>7450</value>
                </values>
                <units>256timesC</units>
        </item>
        <item name="magTCM">
                <longname>User copy of mag sensor temperature compensation
matrix</longname>
                <type>MatrixFloat32</type>
```

```
<flags>DATA_PERMANENT|DATA_SUBSYSTEM</flags>
<rows>3</rows>
<columns>8</columns>
<minValue>-3.0E38</minValue>
<maxValue>3.0E38</maxValue>
<rowData>
        <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
            </columnData>
        <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
            </columnData>
        <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
```

```
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                    </columnData>
            </rowData>
        <units>none</units>
</item>
<item name="accelTCM">
        <longname>accel sensor temperature compensation matrix</longname>
        <type>MatrixFloat32</type>
        <flags>DATA_PERMANENT|DATA_SUBSYSTEM</flags>
        <rows>3</rows>
        <columns>8</columns>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <rowData>
                <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
    </columnData>
                <columnData>
                        <value>0.0f</value>
```

```
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
    </columnData>
                <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
    </columnData>
</rowData>
        <units>none</units>
 </item>
 <item name="gyroTCM">
        <longname>gyro sensor temperature compensation matrix</longname>
        <type>MatrixFloat32</type>
        <flags>DATA_PERMANENT|DATA_SUBSYSTEM</flags>
        <rows>3</rows>
        <columns>8</columns>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <rowData>
```

```
        <columnData>
                <value>0.0f</value>
<value>0.0f</value>
<value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
        </columnData>
        <columnData>
<value>0.0f</value>
                <value>0.0f</value>
<value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
        </columnData>
        <columnData>
<value>0.0f</value>
<value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
        </columnData>
```

```
            </rowData>
            <units>NONE</units>
        </item>
    <item name="magFCM">
            <longname>Mag sensor factory calibration matrix</longname>
            <type>MatrixFloat32</type>
        <flags>DATA_PERMANENT|DATA_SUBSYSTEM</flags>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <rows>3</rows>
            <columns>10</columns>
            <rowData>
                    <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0488f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                    </columnData>
                    <columnData>
                <value>0.0f</value>
                <value>-0.0488f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
```

```
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                    </columnData>
                    <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0488f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                    </columnData>
             </rowData>
            <values/>
            <units>NONE</units>
   </item>
<item name="accelFCM">
   <longname>Accelerometer sensor factory calibration matrix</longname>
   <type>MatrixFloat32</type>
    <flags>DATA_PERMANENT|DATA_SUBSYSTEM</flags>
   <minValue>-3.0E38</minValue>
   <maxValue>3.0E38</maxValue>
   <rows>3</rows>
   <columns>10</columns>
```

```
<rowData>
<columnData>
          <value>0.0f</value>
          <value>0.12207f</value>
          <value>0.0f</value>
          <value>0.0f</value>
          <value>0.0f</value>
          <value>0.0f</value>
          <value>0.0f</value>
          <value>0.0f</value>
          <value>0.0f</value>
          <value>0.0f</value>
</columnData>
<columnData>
          <value>0.0f</value>
          <value>0.0f</value>
          <value>0.0f</value>
          <value>0.0f</value>
          <value>-0.12207f</value>
          <value>0.0f</value>
          <value>0.0f</value>
          <value>0.0f</value>
          <value>0.0f</value>
          <value>0.0f</value>
  </columnData>
 <columnData>
          <value>0.0f</value>
          <value>0.0f</value>
          <value>0.0f</value>
          <value>0.0f</value>
```

```
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.12207f</value>
                <value>0.0f</value>
                <value>0.0f</value>
          </columnData>
               </rowData>
               <values/>
               <units>NONE</units>
     </item>
  <item name="gyroFCM">
               <longname>Gyro sensor factory calibration matrix</longname>
               <type>MatrixFloat32</type>
       <flags>DATA_PERMANENT|DATA_SUBSYSTEM</flags>
               <minValue>-3.0E38</minValue>
               <maxValue>3.0E38</maxValue>
               <rows>3</rows>
               <columns>10</columns>
               <rowData>
     <columnData>
                <value>0.0f</value>
                <value>-2.66462e-4f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
```

```
            <value>0.0f</value>
        </columnData>
        <columnData>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>2.66462e-4f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
        </columnData>
        <columnData>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>-2.66462e-4f</value>
            <value>0.0f</value>
            <value>0.0f</value>
        </columnData>
    </rowData>
        <values/>
        <units>NONE</units>
</item>
```

```
        <item name="f_magRawTC">
                <longname>User copy of raw mag at factory calibration base temperature
</longname>
                <type>ArrayFloat32</type>
                <flags>DATA_PERMANENT|DATA_PROTECTED</flags>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                </values>
                <units>None</units>
        </item>

        <item name="f_accelRawTC">
                <longname>User copy of raw accel at factory calibration base temperature
</longname>
                <type>ArrayFloat32</type>
                <flags>DATA_PERMANENT|DATA_PROTECTED</flags>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                </values>
                <units>None</units>
        </item>
```

```
        <item name="f_gyroRawTC">
                <longname>User copy of raw gyro at factory calibration base temperature
</longname>
                <type>ArrayFloat32</type>
                <flags>DATA_PERMANENT|DATA_PROTECTED</flags>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                </values>
                <units>None</units>
        </item>
    <item name="f_accelCalTemperatureRaw">
        <longname>Temperature at which the Rate Table temperature comps are computed
for each axis (256*C)</longname>
        <type>ArrayInt32</type>
                <flags>DATA_PERMANENT|DATA_PROTECTED</flags>
        <dimension>3</dimension>
        <minValue>-32768</minValue>
        <maxValue>32767</maxValue>
        <values>
            <value>7450</value>
            <value>7450</value>
            <value>7450</value>
        </values>
        <setfunction>N</setfunction>
        <units>256timesC</units>
    </item>
```

```
        <!-- factory values for user magnetic offset temperature correction values -->
        <item name="f_magCalTemperatureRaw">
              <longname>Temperature at which the Dycome temperature comps are computed
for each mag X,Y,Z (256*C)</longname>
              <type>ArrayInt32</type>
              <dimension>3</dimension>
       <minValue>-32768</minValue>
       <maxValue>32767</maxValue>
              <flags>DATA_PERMANENT|DATA_PROTECTED</flags>
              <values>
             <value>7450</value>
             <value>7450</value>
             <value>7450</value>
              </values>
              <units>256timesC</units>
        </item>
    <item name="f_gyroCalTemperatureRaw">
        <longname>Temperature at which the Rate Table temperature comps are computed
for each axis (256*C)</longname>
        <type>ArrayInt32</type>
              <flags>DATA_PERMANENT|DATA_PROTECTED</flags>
        <dimension>3</dimension>
        <minValue>-32768</minValue>
        <maxValue>32767</maxValue>
        <values>
            <value>7450</value>
            <value>7450</value>
            <value>7450</value>
        </values>
        <setfunction>N</setfunction>
        <units>256timesC</units>
```

```
        </item>
        <item name="f_magTCM">
              <longname>User copy of mag sensor temperature compensation
matrix</longname>
           <type>MatrixFloat32</type>
              <flags>DATA_PERMANENT|DATA_PROTECTED</flags>
              <rows>3</rows>
              <columns>8</columns>
              <minValue>-3.0E38</minValue>
              <maxValue>3.0E38</maxValue>
              <rowData>
               <columnData>
                           <value>0.0f</value>
                           <value>0.0f</value>
                           <value>0.0f</value>
                           <value>0.0f</value>
                           <value>0.0f</value>
                           <value>0.0f</value>
                           <value>0.0f</value>
                           <value>0.0f</value>
               </columnData>
               <columnData>
                           <value>0.0f</value>
                           <value>0.0f</value>
                           <value>0.0f</value>
                           <value>0.0f</value>
                           <value>0.0f</value>
                           <value>0.0f</value>
                           <value>0.0f</value>
                           <value>0.0f</value>
```

```
            </columnData>
            <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
            </columnData>
              </rowData>
           <units>none</units>
    </item>
    <item name="f_accelTCM">
           <longname>accel sensor temperature compensation matrix</longname>
           <type>MatrixFloat32</type>
           <flags>DATA_PERMANENT|DATA_PROTECTED</flags>
           <rows>3</rows>
           <columns>8</columns>
           <minValue>-3.0E38</minValue>
           <maxValue>3.0E38</maxValue>
           <rowData>
            <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
```

```
                <value>0.0f</value>
                <value>0.0f</value>
            </columnData>
            <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
            </columnData>
            <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
            </columnData>
        </rowData>
        <units>none</units>
    </item>
    <item name="f_gyroTCM">
        <longname>gyro sensor temperature compensation matrix</longname>
        <type>MatrixFloat32</type>
        <flags>DATA_PERMANENT|DATA_PROTECTED</flags>
```

```
<rows>3</rows>
<columns>8</columns>
<minValue>-3.0E38</minValue>
<maxValue>3.0E38</maxValue>
<rowData>
 <columnData>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
 </columnData>
 <columnData>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
 </columnData>
 <columnData>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
```

```
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
            </columnData>
    </rowData>
            <units>none</units>
    </item>
<item name="f_magFCM">
            <longname>Mag sensor factory calibration matrix</longname>
            <type>MatrixFloat32</type>
            <flags>DATA_PERMANENT|DATA_PROTECTED</flags>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <rows>3</rows>
            <columns>10</columns>
            <rowData>
             <columnData>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0488f</value>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
             </columnData>
             <columnData>
```

```
            <value>0.0f</value>
            <value>-0.0488f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
          </columnData>
          <columnData>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0488f</value>
            <value>0.0f</value>
            <value>0.0f</value>
          </columnData>
          </rowData>
        <values/>
        <units>NONE</units>
    </item>
  <item name="f_accelFCM">
        <longname>Accelerometer sensor factory calibration matrix</longname>
        <type>MatrixFloat32</type>
```

```
            <flags>DATA_PERMANENT|DATA_PROTECTED</flags>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <rows>3</rows>
            <columns>10</columns>
        <rowData>
             <columnData>
               <value>0.0f</value>
               <value>0.12207f</value>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
             </columnData>
             <columnData>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>-0.12207f</value>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
             </columnData>
```

```
            <columnData>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.0f</value>
              <value>0.12207f</value>
              <value>0.0f</value>
              <value>0.0f</value>
            </columnData>
      </rowData>
            <values/>
            <units>NONE</units>
      </item>
   <item name="f_gyroFCM">
            <longname>Gyro sensor factory calibration matrix</longname>
            <type>MatrixFloat32</type>
            <flags>DATA_PERMANENT|DATA_PROTECTED</flags>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <rows>3</rows>
            <columns>10</columns>
      <rowData>
            <columnData>
              <value>0.0f</value>
              <value>-2.66462e-4f</value>
              <value>0.0f</value>
              <value>0.0f</value>
```

```
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
          </columnData>
          <columnData>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>2.66462e-4f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
        </columnData>
        <columnData>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>0.0f</value>
            <value>-2.66462e-4f</value>
            <value>0.0f</value>
            <value>0.0f</value>
```

```
        </columnData>
                </rowData>
               <values/>
               <units>NONE</units>
        </item>
        <item name="magRawTCAdjustment">
               <longname>Temperature compensation vector computed using the magTCM to be
added to the raw mag vector</longname>
               <type>ArrayFloat32</type>
               <flags>DATA_READONLY</flags>
               <dimension>3</dimension>
               <minValue>-3.0E38</minValue>
               <maxValue>3.0E38</maxValue>
               <values>
                      <value>0.0</value>
                      <value>0.0</value>
                      <value>0.0</value>
               </values>
               <units>None</units>
        </item>
        <item name="magRealTimeVariance">
               <longname></longname>
               <type>Float32</type>
               <minValue>-3.0E38</minValue>
               <maxValue>3.0E38</maxValue>
               <units>None</units>
        </item>
        <item name="magInstantaneousVariance">
               <longname></longname>
               <type>Float32</type>
```

```
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <units>None</units>
      </item>

      <item name="saveAll">
            <longname>Force database to save all values, even if default</longname>
            <type>Ordinal</type>
            <defaultValue>0</defaultValue>
            <maxValue>1</maxValue>
            <labels>
                  <label>off</label>
                  <label>on</label>
            </labels>
            <units>None</units>
            <setfunction>N</setfunction>
      </item>
   <item name="magBW">
       <longname>Bandwidth (measurement time): 0=7.92 mS, 1=4.08 mS</longname>
          <type>Ordinal</type>
       <flags>DATA_PERMANENT</flags>
          <defaultValue>1</defaultValue>
          <maxValue>1</maxValue>
          <labels>
                <label>7_92ms</label>
                <label>4_08ms</label>
          </labels>
      <units>None</units>
   </item>
   <item name="magMode">
```

```
        <longname>0=no auto setReset, 1= set or reset, 2= alternating set reset with
bridge bias</longname>
           <type>Ordinal</type>
        <flags>DATA_PERMANENT</flags>
           <defaultValue>2</defaultValue>
           <maxValue>2</maxValue>
           <labels>
                  <label>NO_AUTO_SET_RESET</label>
                  <label>SET_OR_RESET</label>
                  <label>ALTERNATE_SET_RESET</label>
           </labels>
      <units>None</units>
    </item>
    <item name="magSetResetModulus">
        <longname>Controls the Set or Reset rate relative to number of samples (Modes
1)</longname>
        <type>Int32</type>
        <flags>DATA_PERMANENT</flags>
        <defaultValue>1</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>sample count</units>
    </item>

      <item name="sdotpmode">
             <longname>Mode for s.p output (unused)</longname>
             <type>Ordinal</type>
             <defaultValue>0</defaultValue>
             <maxValue>4</maxValue>
             <labels>
                    <label>Mode 0</label>
```

```
                <label>Mode 1</label>
                <label>Mode 2</label>
                <label>Mode 3</label>
                <label>Mode 4</label>
            </labels>
            <units>None</units>
    </item>

<item name="f_accelrange">
    <longname>Accelerometer scaling: 0=4g,1=8g,2=2g,3=16g</longname>
    <type>Ordinal</type>
    <flags>DATA_PERMANENT|DATA_PROTECTED</flags>
    <defaultValue>0</defaultValue>
    <maxValue>3</maxValue>
    <labels>
        <label>4g</label>
        <label>8g</label>
        <label>2g</label>
        <label>16g</label>
    </labels>
    <units>None</units>
</item>
    <item name="kmag">
            <longname>Factor for estimated mag</longname>
            <type>Float32</type>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <units>None</units>
    </item>
    <item name="magBias">
```

```
                <longname>In-field calibration offset for the mags</longname>
                <type>ArrayFloat32</type>
        <flags>DATA_PERMANENT</flags>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0</value>
                        <value>0.0</value>
                        <value>0.0</value>
                </values>
                <units>milliGauss</units>
         </item>
         <item name="pointMagBias">
                <longname>Backup mag bias for point calibration</longname>
                <type>ArrayFloat32</type>
        <flags>DATA_PERMANENT</flags>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0</value>
                        <value>0.0</value>
                        <value>0.0</value>
                </values>
                <units>milliGauss</units>
         </item>
    <item name="pointMagCrossAxisCorrection">
                <longname>Backup of magCrossAxisCorrection for point
calibration</longname>
```

```
            <type>MatrixFloat32</type>
        <flags>DATA_PERMANENT</flags>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <rows>3</rows>
            <columns>3</columns>
            <rowData>
                  <columnData>
               <value>1.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
                  </columnData>
                  <columnData>
               <value>0.0f</value>
               <value>1.0f</value>
               <value>0.0f</value>
                  </columnData>
                  <columnData>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>1.0f</value>
                  </columnData>
             </rowData>
            <values/>
            <units>NONE</units>
       </item>


    <item name="f_calPointsNumToAvg">
      <longname>Number of samples to be averaged for a 3D/2D Calibration Mode capture
point</longname>
```

```
        <type>Int32</type>

        <flags>DATA_PERMANENT</flags>

        <defaultValue>10</defaultValue>

        <minValue>1</minValue>

        <maxValue>6000</maxValue>

        <units>sample count</units>

    </item>


    <item name="yawVarianceMean">

            <longname>Average variance of yaw data since compassCollectingStatistics
and/or yawClearStatistics </longname>

            <type>Float32</type>

            <flags>DATA_READONLY</flags>

            <defaultValue>0.0</defaultValue>

            <minValue>-3.0E38</minValue>

            <maxValue>3.0E38</maxValue>

            <units>degrees</units>

       </item>

    <item name="yawTemperatureMean">

            <longname>Temperature average while the yaw statistics were being
collected</longname>

       <type>Int32</type>

        <defaultValue>0</defaultValue>

        <minValue>0</minValue>

        <maxValue>0x7FFFFFFF</maxValue>

        <units>Celsius*256</units>

       </item>

    <item name="magCrossAxisCorrection">

            <longname>Corrects the mag for mounting imperfections and in-field
calibration</longname>

            <type>MatrixFloat32</type>
```

```
<flags>DATA_PERMANENT</flags>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <rows>3</rows>
        <columns>3</columns>
        <rowData>
               <columnData>
           <value>1.0f</value>
           <value>0.0f</value>
           <value>0.0f</value>
               </columnData>
               <columnData>
           <value>0.0f</value>
           <value>1.0f</value>
           <value>0.0f</value>
               </columnData>
               <columnData>
           <value>0.0f</value>
           <value>0.0f</value>
           <value>1.0f</value>
               </columnData>
         </rowData>
        <values/>
  <setfunction>Y</setfunction>
        <units>NONE</units>
 </item>
```

```
    <item name="WMMepoch">
        <longname>Base year for the World Magnetic Model coefficients being
used</longname>
        <type>Float32</type>
        <flags>DATA_PERMANENT</flags>
        <defaultValue>2015.0f</defaultValue>
        <minValue>2000.0f</minValue>
        <maxValue>3000.0f</maxValue>
        <units>year</units>
    </item>
    <item name="gyroTimeStamp">
        <longname>Time stamp taken at the arrival of the last gyro data item being
averaged</longname>
        <type>Int32</type>
        <defaultValue>0</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>milliseconds</units>
    </item>
    <item name="eepromWriteInProgress">
        <longname>1 if an EEPROM write is in progress, 0 otherwise</longname>
        <type>Int32</type>
        <defaultValue>0</defaultValue>
        <minValue>0</minValue>
        <maxValue>1</maxValue>
```

```
            <units>None</units>
        </item>
    <item name="calDeqDeltaCAC">
        <longname>Field calibration derivative delta</longname>
            <type>Float32</type>
            <flags>DATA_PERMANENT</flags>
        <defaultValue>1.0e-4</defaultValue>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
        <units>N/A</units>
        </item>
    <item name="compassCalcRate">
            <longname>Rate at which the compass computations run, with range 90-110
Hz</longname>
            <type>Int32</type>
            <flags>DATA_PERMANENT</flags>
            <defaultValue>100</defaultValue>
            <minValue>90</minValue>
            <maxValue>110</maxValue>
            <units>Hz</units>
        </item>
        <item name="compassOnForCalConvergence">
            <longname>Turns on compass computations for calibration convergence,
0=Off, 1=On</longname>
            <type>Ordinal</type>
            <flags>DATA_PERMANENT</flags>
            <defaultValue>1</defaultValue>
            <maxValue>1</maxValue>
            <labels>
                <label>Off</label>
                <label>On</label>
```

```
                </labels>
                <units>None</units>
        </item>
        <item name="clearEEPROMLog">
                <longname>Clears the eeLog data including the startUpCount</longname>
                <type>Ordinal</type>
                <defaultValue>0</defaultValue>
                <maxValue>1</maxValue>
                <labels>
                        <label>off</label>
                        <label>on</label>
                </labels>
                <units>None</units>
        </item>
        <item name="protectedAreaWriteEnable">
                <longname>Enables a write to the factory settings on the next
attempt</longname>
                <type>Ordinal</type>
                <defaultValue>0</defaultValue>
                <maxValue>1</maxValue>
                <labels>
                        <label>off</label>
                        <label>on</label>
                </labels>
                <units>None</units>
        </item>
          <item name="startUpCount">
          <longname>Number of power up and reset start ups</longname>
          <type>Int32</type>
          <defaultValue>0</defaultValue>
```

```
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>None</units>
    </item>
    <item name="eeLogUserDB1Fail">
            <longname>Failure info for the database EEPROM buffer 1</longname>
            <type>ArrayInt32</type>
            <flags>DATA_READONLY</flags>
            <dimension>3</dimension>
            <minValue>0</minValue>
            <maxValue>0x7FFFFFFF</maxValue>
            <values>
                  <value>0</value>
                  <value>0</value>
                  <value>0</value>
            </values>
            <units>NONE</units>
      </item>
      <item name="eeLogUserDB2Fail">
            <longname>Failure info for the database EEPROM buffer 2</longname>
            <type>ArrayInt32</type>
            <flags>DATA_READONLY</flags>
            <dimension>3</dimension>
            <minValue>0</minValue>
            <maxValue>0x7FFFFFFF</maxValue>
            <values>
                  <value>0</value>
                  <value>0</value>
                  <value>0</value>
            </values>
```

```
                <units>NONE</units>
        </item>
        <item name="eeLogRestoreFactoryAuto">
                <longname>Failure info for an automatic restoration due to failure of
both user DB buffersr</longname>
                <type>ArrayInt32</type>
                <flags>DATA_READONLY</flags>
                <dimension>3</dimension>
                <minValue>0</minValue>
                <maxValue>0x7FFFFFFF</maxValue>
                <values>
                        <value>0</value>
                        <value>0</value>
                        <value>0</value>
                </values>
                <units>NONE</units>
        </item>
        <item name="eeLogRestoreFactoryUser">
                <longname>Info indicating a user command of restorefactorycal</longname>
                <type>ArrayInt32</type>
                <flags>DATA_READONLY</flags>
                <dimension>3</dimension>
                <minValue>0</minValue>
                <maxValue>0x7FFFFFFF</maxValue>
                <values>
                        <value>0</value>
                        <value>0</value>
                        <value>0</value>
                </values>
                <units>NONE</units>
```

```
        </item>

        <item name="estimatedPk2PkYawError">

              <longname>estimate the Peak-to-Peak yaw error based on the infield
calibration results (degrees)</longname>

              <type>Float32</type>

              <flags>DATA_PERMANENT</flags>

              <defaultValue>180.0f</defaultValue>

              <minValue>-3.0E38</minValue>

              <maxValue>3.0E38</maxValue>

              <units>degrees</units>

    </item>

    <item name="calDeqDeltaCAC2">

        <longname>Field calibration derivative delta for off axis terms
(unused)</longname>

              <type>Float32</type>

        <flags>DATA_PERMANENT</flags>

        <defaultValue>1.0e-4</defaultValue>

              <minValue>-3.0E38</minValue>

              <maxValue>3.0E38</maxValue>

        <units>N/A</units>

       </item>

    <item name="infieldCalMu3_2D">

        <longname>In-field calibration factor controlling the off-axis terms for 2D
(unused)</longname>

        <type>Float32</type>

      <flags>DATA_PERMANENT</flags>

        <defaultValue>0</defaultValue>

        <minValue>0.0</minValue>

        <maxValue>100.0</maxValue>

        <units>none</units>

       </item>
```

```
        <item name="compDipAngle">
                <longname>complement of the Dip Angle (90-dipAngle, radians)</longname>
                <type>Float32</type>
                <flags>DATA_READONLY</flags>
                <defaultValue>180.0f</defaultValue>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <units>radians</units>
        </item>
        <item name="magFieldCalXYZ">
                <longname>Realtime mag offsets computed during field cal convergence
phase (unused)</longname>
                <type>ArrayFloat32</type>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0</value>
                        <value>0.0</value>
                        <value>0.0</value>
                </values>
                <units>NONE</units>
        </item>
        <item name="calBufferSwitch">
                <longname>Allows user selection of accel type to buffer into
accelCalPoints for field cal</longname>
                <type>Ordinal</type>
                   <flags>DATA_PERMANENT</flags>
                <defaultValue>1</defaultValue>
                <maxValue>1</maxValue>
                <labels>
```

```
                        <label>accelpSelection</label>
                        <label>accelEstSelection</label>
                </labels>
                <units>None</units>
        </item>
        <item name="positionRateSwitch">
                <longname>Allows user selection of the trigger for the RFS Bitfield
position variable</longname>
                <type>Ordinal</type>
                   <flags>DATA_PERMANENT</flags>
                <defaultValue>0</defaultValue>
                <maxValue>1</maxValue>
                <labels>
                        <label>timerInMilliseconds</label>
                        <label>compassDataTrigger</label>
                </labels>
                <units>None</units>
        </item>
    <item name="gyropTimeStamp">
        <longname>Gyro time stamp taken after the gyrop data was computed</longname>
        <type>Int32</type>
        <defaultValue>0</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>milliseconds</units>
    </item>



        <!-- Scout specific data -->
    <item name="cputime">
```

```
        <longname>Processor time from startup (milliseconds)</longname>
        <type>Int32</type>
        <flags>DATA_READONLY</flags>
        <defaultValue>0</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>milliseconds</units>
    </item>
    <item name="pMatrix">
            <longname>Platform rotation matrix (orientation of unit)</longname>
            <type>MatrixFloat32</type>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <rows>3</rows>
            <columns>3</columns>
            <rowData>
                  <columnData>
               <value>1.0f</value>
               <value>0.0f</value>
               <value>0.0f</value>
                  </columnData>
                  <columnData>
               <value>0.0f</value>
               <value>1.0f</value>
               <value>0.0f</value>
                  </columnData>
                  <columnData>
               <value>0.0f</value>
               <value>0.0f</value>
               <value>1.0f</value>
```

```
                    </columnData>
                 </rowData>
                <values/>
                <units>NONE</units>
        </item>
          <item name="boresightMatrix">
                <longname>Converts this units orientation to host devices
orientation</longname>
            <type>MatrixFloat32</type>
            <flags>DATA_PERMANENT</flags>
               <minValue>-3.0E38</minValue>
               <maxValue>3.0E38</maxValue>
               <rows>3</rows>
               <columns>3</columns>
               <rowData>
                     <columnData>
                        <value>1.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
                     </columnData>
                     <columnData>
                        <value>0.0f</value>
                        <value>1.0f</value>
                        <value>0.0f</value>
                      </columnData>
                      <columnData>
                   <value>0.0f</value>
                   <value>0.0f</value>
                   <value>1.0f</value>
                      </columnData>
```

```
                </rowData>
                <values/>
                <setfunction>Y</setfunction>
                <units>NONE</units>
        </item>
        <item name="aEqAveraged">
                <longname>Used for monitoring error convergence</longname>
                <type>Float32</type>
                <defaultValue>1000.0f</defaultValue>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <units>None</units>
        </item>
        <item name="relCalBackupTime">
                <longname>Relative time since last backup of calibration
parameters</longname>
                <type>Float32</type>
                <defaultValue>0.0</defaultValue>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <units>None</units>
        </item>
    <item name="magBufferActiveIndex">
        <longname>Current index into cal point buffer</longname>
        <type>Int32</type>
        <defaultValue>0</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7fffffff</maxValue>
        <units>None</units>
    </item>
```

```
        <item name="sphericalCalCoverage">
                <longname>Reports spherical coverage of buffered magnetic measurements as
a percent</longname>
                <type>Float32</type>
                <defaultValue>0.0</defaultValue>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <units>None</units>
        </item>
        <item name="sphericalPointSeparationLimit">
                <longname>Used in selecting adaptcal points</longname>
                <type>Float32</type>
                <defaultValue>0.947f</defaultValue>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <units>None</units>
        </item>
        <item name="sphericalPoints">
                <longname>Idealized points used in AdaptCal</longname>
            <type>MatrixFloat32</type>
            <flags>DATA_PERMANENT</flags>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <rows>32</rows>
                <columns>3</columns>
                <rowData>
                 <columnData>
                    <value>  7.623577990635768100e-01</value>
                    <value>  3.083251657260023800e-01</value>
                    <value> -5.689869755864119800e-01</value>
```

```
</columnData>
<columnData>
   <value>  8.775183418402466900e-01</value>
   <value> -2.953121248763028700e-01</value>
   <value> -3.778257649168289500e-01</value>
</columnData>
<columnData>
   <value>  1.422040069486279800e-01</value>
   <value>  3.655422995039560600e-01</value>
   <value> -9.198678425084305700e-01</value>
</columnData>
<columnData>
   <value> -7.935130481616248900e-03</value>
   <value>  9.999685163565099600e-01</value>
   <value> -3.685464294963838000e-16</value>
</columnData>
<columnData>
   <value>  2.771796259627252200e-01</value>
   <value> -7.416031098856175100e-01</value>
   <value> -6.108979312120353900e-01</value>
</columnData>
<columnData>
   <value>  4.712279125067833100e-01</value>
   <value> -1.833180383880957900e-01</value>
   <value> -8.627506889455609500e-01</value>
</columnData>
<columnData>
   <value>  1.079649811379038600e-01</value>
   <value>  3.031323098395067300e-01</value>
   <value>  9.468127405032409300e-01</value>
```

```
</columnData>
<columnData>
   <value> -5.137529514094382200e-01</value>
   <value>  1.829805860314377400e-01</value>
   <value>  8.381980732820156800e-01</value>
</columnData>
<columnData>
   <value>  3.992996357642775200e-01</value>
   <value> -7.406340411636113100e-01</value>
   <value>  5.403896908233663000e-01</value>
</columnData>
<columnData>
   <value> -7.211644647895408800e-01</value>
   <value> -3.701365709250235600e-01</value>
   <value>  5.855943421761179400e-01</value>
</columnData>
<columnData>
   <value> -6.587733062935752200e-01</value>
   <value> -7.490302523612281100e-01</value>
   <value>  7.050824038866983200e-02</value>
</columnData>
<columnData>
   <value>  2.406265181878727800e-01</value>
   <value>  7.955510642607794800e-01</value>
   <value>  5.560551977082154300e-01</value>
</columnData>
<columnData>
   <value>  6.113022021013234700e-01</value>
   <value> -7.887906858629050200e-01</value>
   <value> -6.417843564625667900e-02</value>
```

```
</columnData>
<columnData>
   <value>  8.720432131220212700e-01</value>
   <value> -3.574938620779444800e-01</value>
   <value>  3.342735003323127900e-01</value>
</columnData>
<columnData>
   <value>  9.805280292184503900e-01</value>
   <value>  1.948382739121349600e-01</value>
   <value>  2.455261566354529500e-02</value>
</columnData>
<columnData>
   <value> -3.517833297458684600e-01</value>
   <value> -7.964331358352396700e-01</value>
   <value> -4.918767620619583500e-01</value>
</columnData>
<columnData>
   <value> -7.623577990635767000e-01</value>
   <value> -3.083251657260029400e-01</value>
   <value> -5.689869755864119800e-01</value>
</columnData>
<columnData>
   <value> -8.775183418402466900e-01</value>
   <value>  2.953121248763026500e-01</value>
   <value> -3.778257649168292900e-01</value>
</columnData>
<columnData>
   <value> -1.422040069486278900e-01</value>
   <value> -3.655422995039566700e-01</value>
   <value> -9.198678425084303400e-01</value>
```

```
</columnData>
<columnData>
   <value>  7.935130481616248900e-03</value>
   <value> -9.999685163565099600e-01</value>
   <value>  2.648737386723612200e-16</value>
</columnData>
<columnData>
   <value> -2.771796259627251100e-01</value>
   <value>  7.416031098856170600e-01</value>
   <value> -6.108979312120359500e-01</value>
</columnData>
<columnData>
   <value> -4.712279125067833100e-01</value>
   <value>  1.833180383880952700e-01</value>
   <value> -8.627506889455611700e-01</value>
</columnData>
<columnData>
   <value> -1.079649811379037300e-01</value>
   <value> -3.031323098395062900e-01</value>
   <value>  9.468127405032411600e-01</value>
</columnData>
<columnData>
   <value>  5.137529514094384400e-01</value>
   <value> -1.829805860314371300e-01</value>
   <value>  8.381980732820155700e-01</value>
</columnData>
<columnData>
   <value> -3.992996357642776300e-01</value>
   <value>  7.406340411636116400e-01</value>
   <value>  5.403896908233659600e-01</value>
```

```
</columnData>
<columnData>
   <value>  7.21164464789540990e-01</value>
   <value>  3.70136570925023940e-01</value>
   <value>  5.85594342176117500e-01</value>
</columnData>
<columnData>
   <value>  6.58773306293575110e-01</value>
   <value>  7.49030252361228110e-01</value>
   <value>  7.05082403886693470e-02</value>
</columnData>
<columnData>
   <value> -2.40626518187872560e-01</value>
   <value> -7.95551064260779260e-01</value>
   <value>  5.56055197708215990e-01</value>
</columnData>
<columnData>
   <value> -6.11302202101323580e-01</value>
   <value>  7.88790685862905020e-01</value>
   <value> -6.41784356462571650e-02</value>
</columnData>
<columnData>
   <value> -8.72043213122021150e-01</value>
   <value>  3.57493862077944650e-01</value>
   <value>  3.34273500323127900e-01</value>
</columnData>
<columnData>
   <value> -9.80528029218450390e-01</value>
   <value> -1.94838273912135010e-01</value>
   <value>  2.45526156635454930e-02</value>
```

```
            </columnData>
            <columnData>
               <value>  3.517833297458685700e-01</value>
               <value>  7.964331358352392300e-01</value>
               <value> -4.918767620619589100e-01</value>
            </columnData>
            </rowData>
          <values/>
          <units>NONE</units>
 </item>
  <item name="magVariance">
          <longname>Initial value for the variance of the mag values</longname>
          <type>Float32</type>
<flags>DATA_PERMANENT</flags>
          <defaultValue>36.0f</defaultValue>
          <minValue>-3.0E38</minValue>
          <maxValue>3.0E38</maxValue>
          <units>None</units>
 </item>
 <item name="accelVariance">
          <longname>Initial value for the variance of the accel values</longname>
          <type>Float32</type>
<flags>DATA_PERMANENT</flags>
          <defaultValue>36.0f</defaultValue>
          <minValue>-3.0E38</minValue>
          <maxValue>3.0E38</maxValue>
          <units>None</units>
 </item>
 <item name="gyroVariance">
          <longname>Initial value for the variance of the gyro values</longname>
```

```
        <type>Float32</type>
<flags>DATA_PERMANENT</flags>
        <defaultValue>1.0966227e-4f</defaultValue>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>None</units>
 </item>
 <item name="gyroBiasStability">
        <longname>Gyro Bias Stability in rads/sec</longname>
        <type>Float32</type>
<flags>DATA_PERMANENT</flags>
        <defaultValue>1.939255e-5f</defaultValue>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>rads/sec</units>
 </item>
 <item name="gyroBiasThermalStability">
        <longname>Gyro Bias Thermal Stability in rad/sec/DegC</longname>
        <type>Float32</type>
<flags>DATA_PERMANENT</flags>
        <defaultValue>2.617994e-3f</defaultValue>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>rad/sec/DegC</units>
 </item>
 <item name="magEstVariance">
        <longname>Estimated variance of the mag values</longname>
        <type>Float32</type>
        <defaultValue>1000000.0f</defaultValue>
        <minValue>-3.0E38</minValue>
```

```
        <maxValue>3.0E38</maxValue>
        <units>None</units>
</item>
<item name="accelEstVariance">
        <longname>Estimated variance of the accel values</longname>
        <type>Float32</type>
        <defaultValue>1000000.0f</defaultValue>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>None</units>
</item>
<item name="gyroBiasEstVariance">
        <longname>Estimated variance of the gyro bias</longname>
        <type>Float32</type>
        <defaultValue>1.096621e-4f</defaultValue>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>None</units>
</item>
<item name="deltaDBias">
        <longname>Derivative approximation used in bias computation</longname>
        <type>Float32</type>
  <flags>DATA_PERMANENT</flags>
        <defaultValue>1.0E-2f</defaultValue>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>None</units>
</item>
<item name="deltaDCAC">
```

```
            <longname>Derivative approximation used in Cross Axis Correction
computation</longname>

        <flags>DATA_PERMANENT</flags>

            <type>Float32</type>

            <defaultValue>1.0E-4f</defaultValue>

            <minValue>-3.0E38</minValue>

            <maxValue>3.0E38</maxValue>

            <units>None</units>

      </item>

      <item name="magCalExpDecay">

            <longname>Current field cal decay factor</longname>

            <type>Float32</type>

            <defaultValue>0.5f</defaultValue>

            <minValue>-3.0E38</minValue>

            <maxValue>3.0E38</maxValue>

            <units>None</units>

      </item>

      <item name="magCalExpDecayRate">

            <longname>Rate of field cal decay</longname>

            <type>Float32</type>

            <defaultValue>1.004f</defaultValue>

            <minValue>-3.0E38</minValue>

            <maxValue>3.0E38</maxValue>

            <units>None</units>

      </item>

   <item name="pointCalTermination">

      <longname>Commands the cal mode after a manual point calibration (0=off,
1=point, 2=auto)</longname>

      <type>Ordinal</type>

      <defaultValue>1</defaultValue>

      <maxValue>2</maxValue>
```

```
        <labels>
           <label>TURN_OFF_CAL</label>
           <label>STAY_IN_POINT_CAL</label>
           <label>CHANGE_TO_AUTO_CAL</label>
        </labels>
        <setfunction>N</setfunction>
        <units>None</units>
    </item>
        <item name="accelRawTCAdjustment">
              <longname>Temperature compensation vector computed using the accelTCM to
be added to the raw accel vector</longname>
              <type>ArrayFloat32</type>
          <flags>DATA_READONLY</flags>
              <dimension>3</dimension>
              <minValue>-3.0E38</minValue>
              <maxValue>3.0E38</maxValue>
              <values>
                    <value>0.0</value>
                    <value>0.0</value>
                    <value>0.0</value>
              </values>
              <units>None</units>
        </item>
        <item name="gyroRawTCAdjustment">
              <longname>Temperature compensation vector computed using the gyroTCM to
be added to the raw gyro vector</longname>
              <type>ArrayFloat32</type>
          <flags>DATA_READONLY</flags>
              <dimension>3</dimension>
              <minValue>-3.0E38</minValue>
              <maxValue>3.0E38</maxValue>
```

```
        <values>
                <value>0.0</value>
                <value>0.0</value>
                <value>0.0</value>
        </values>
        <units>None</units>
</item>
<item name="magBufferActive">
        <longname>Provides the index into the calPointBuffer</longname>
        <type>ArrayInt32</type>
        <dimension>32</dimension>
        <minValue>0</minValue>
        <maxValue>500</maxValue>
        <values>
                <value>0</value>
                <value>0</value>
                <value>0</value>
                <value>0</value>
                <value>0</value>
                <value>0</value>
                <value>0</value>
                <value>0</value>
                <value>0</value>
                <value>0</value>
                <value>0</value>
                <value>0</value>
                <value>0</value>
                <value>0</value>
                <value>0</value>
                <value>0</value>
                <value>0</value>
                <value>0</value>
```

```
                    <value>0</value>
                    <value>0</value>
                    <value>0</value>
                    <value>0</value>
                    <value>0</value>
                    <value>0</value>
                    <value>0</value>
                    <value>0</value>
                    <value>0</value>
                    <value>0</value>
                    <value>0</value>
                    <value>0</value>
                    <value>0</value>
                    <value>0</value>
                    <value>0</value>
                    <value>0</value>
           </values>
                <units>none</units>
        </item>
        <item name="calPointBufferOccupied">
                <longname>Indicates whether the corresponding point in the calPointBuffer
is occupied</longname>
                <type>ArrayInt32</type>
                <dimension>32</dimension>
                <minValue>0</minValue>
                <maxValue>1</maxValue>
                <values>
                    <value>0</value>
                    <value>0</value>
                    <value>0</value>
```

```
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
            <value>0</value>
        </values>
```

```
        <units>none</units>
</item>
<item name="calPointBufferMag">
        <longname>Magnetic vectors captured as cal points</longname>
        <type>MatrixFloat32</type>
        <rows>32</rows>
        <columns>3</columns>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <rowData>
                <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
   </columnData>
                <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
   </columnData>
                <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
   </columnData>
                <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
   </columnData>
```

```
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
```

```
            <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
        </columnData>
            <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
        </columnData>
            <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
        </columnData>
            <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
        </columnData>
            <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
        </columnData>
            <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
        </columnData>
```

```
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
</columnData>
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
</columnData>
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
</columnData>
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
</columnData>
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
</columnData>
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
</columnData>
```

```
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
          <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
```

```
                        <columnData>
                                <value>0.0f</value>
                                <value>0.0f</value>
                                <value>0.0f</value>
            </columnData>
                        <columnData>
                                <value>0.0f</value>
                                <value>0.0f</value>
                                <value>0.0f</value>
            </columnData>
                        <columnData>
                                <value>0.0f</value>
                                <value>0.0f</value>
                                <value>0.0f</value>
            </columnData>
                        <columnData>
                                <value>0.0f</value>
                                <value>0.0f</value>
                                <value>0.0f</value>
            </columnData>
            </rowData>
                <units>none</units>
        </item>
        <item name="calPointBufferAccel">
                <longname>Accelerometer vectors captured with the mag cal
points</longname>
                <type>MatrixFloat32</type>
                <rows>32</rows>
                <columns>3</columns>
                <minValue>-3.0E38</minValue>
```

```
        <maxValue>3.0E38</maxValue>
        <rowData>
                <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
</columnData>
                <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
</columnData>
                <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
</columnData>
                <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
</columnData>
                <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
</columnData>
                <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
```

```
                <value>0.0f</value>
    </columnData>
            <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
            <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
            <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
            <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
            <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
                <value>0.0f</value>
    </columnData>
            <columnData>
                <value>0.0f</value>
                <value>0.0f</value>
```

```
                    <value>0.0f</value>
      </columnData>
            <columnData>
                    <value>0.0f</value>
                    <value>0.0f</value>
                    <value>0.0f</value>
      </columnData>
            <columnData>
                    <value>0.0f</value>
                    <value>0.0f</value>
                    <value>0.0f</value>
      </columnData>
            <columnData>
                    <value>0.0f</value>
                    <value>0.0f</value>
                    <value>0.0f</value>
      </columnData>
            <columnData>
                    <value>0.0f</value>
                    <value>0.0f</value>
                    <value>0.0f</value>
      </columnData>
            <columnData>
                    <value>0.0f</value>
                    <value>0.0f</value>
                    <value>0.0f</value>
      </columnData>
            <columnData>
                    <value>0.0f</value>
                    <value>0.0f</value>
```

```
                    <value>0.0f</value>
    </columnData>
              <columnData>
                    <value>0.0f</value>
                    <value>0.0f</value>
                    <value>0.0f</value>
    </columnData>
              <columnData>
                    <value>0.0f</value>
                    <value>0.0f</value>
                    <value>0.0f</value>
    </columnData>
              <columnData>
                    <value>0.0f</value>
                    <value>0.0f</value>
                    <value>0.0f</value>
    </columnData>
              <columnData>
                    <value>0.0f</value>
                    <value>0.0f</value>
                    <value>0.0f</value>
    </columnData>
              <columnData>
                    <value>0.0f</value>
                    <value>0.0f</value>
                    <value>0.0f</value>
    </columnData>
              <columnData>
                    <value>0.0f</value>
                    <value>0.0f</value>
```

```
                    <value>0.0f</value>
            </columnData>
                    <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
            </columnData>
                    <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
            </columnData>
                    <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
            </columnData>
                    <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
            </columnData>
                    <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
            </columnData>
                    <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
```

```
                        <value>0.0f</value>
            </columnData>
                    <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
            </columnData>
                    <columnData>
                        <value>0.0f</value>
                        <value>0.0f</value>
                        <value>0.0f</value>
            </columnData>
            </rowData>
        <units>none</units>
         </item>
   <item name="sensorsRunning">
        <longname>Controls whether the sensors are enabled or not, for reset or so cpu
clock can be changed</longname>
        <type>Ordinal</type>
        <defaultValue>1</defaultValue>
        <maxValue>1</maxValue>
        <labels>
           <label>No</label>
           <label>Yes</label>
        </labels>
        <setfunction>N</setfunction>
        <units>Furlongs</units>
   </item>
   <item name="minCalPoints">
        <longname>Number of points where the field calibration starts</longname>
```

```
        <type>Int32</type>
        <flags>DATA_PERMANENT</flags>
        <defaultValue>4</defaultValue>
        <maxValue>32</maxValue>
        <setfunction>N</setfunction>
        <units>None</units>
    </item>
    <item name="magAverageWindow">
        <longname>Number of samples which will be averaged during point
selection</longname>
        <type>Int32</type>
        <flags>DATA_PERMANENT</flags>
        <defaultValue>20</defaultValue>
        <minValue>1</minValue>
        <maxValue>1000</maxValue>
        <setfunction>Y</setfunction>
        <units>None</units>
    </item>
    <item name="debugMask">
        <longname>Turns on/off (bitwise) debug print statements controlled by *_mask
commands</longname>
        <type>Int32</type>
        <defaultValue>0</defaultValue>
        <minValue>0x80000000</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>furlong bits</units>
    </item>
        <item name="sensorDecimatedDataReady">
                <longname>Event that triggers the sensor data conditioning</longname>
                <type>Ordinal</type>
                <defaultValue>0</defaultValue>
```

```
        <maxValue>1</maxValue>
                <labels>
                        <label>NotReady</label>
                        <label>Ready</label>
                </labels>
                <units>None</units>
         </item>
         <item name="sensorConditionedDataReady">
                <longname>Event that indicates that the conditioned data is
ready</longname>
                <type>Ordinal</type>
                <defaultValue>0</defaultValue>
        <maxValue>1</maxValue>
                <labels>
                        <label>NotReady</label>
                        <label>Ready</label>
                </labels>
                <units>None</units>
         </item>
         <item name="autoFieldCalActive">
                <longname>User can set this to activate auto cal mode</longname>
                <type>Ordinal</type>
                <defaultValue>0</defaultValue>
        <maxValue>1</maxValue>
                <labels>
                        <label>False</label>
                        <label>True</label>
                </labels>
                <units>None</units>
         </item>
```

```
        <item name="pointFieldCalActive">
                <longname>This is automatically set when a command is issued to collect a
manual cal point</longname>
                <type>Ordinal</type>
                <defaultValue>0</defaultValue>
       <maxValue>1</maxValue>
                <labels>
                        <label>False</label>
                        <label>True</label>
                </labels>
                <units>None</units>
        </item>
        <item name="initCalPointBuffer">
                <longname>User command to initialize the in-field cal point
buffer</longname>
                <type>Ordinal</type>
                <defaultValue>0</defaultValue>
       <maxValue>1</maxValue>
                <labels>
                        <label>InitDone</label>
                        <label>InitCmd</label>
                </labels>
                <units>None</units>
        </item>
        <item name="clearPointCal">
                <longname>User command to clear the point cal results</longname>
                <type>Ordinal</type>
                <defaultValue>0</defaultValue>
       <maxValue>1</maxValue>
                <labels>
                        <label>ClearDone</label>
```

```
                <label>ClearCmd</label>
            </labels>
            <units>None</units>
        </item>
        <item name="clearFieldCal">
            <longname>Deprecated, use revertFieldCal instead, user command to default
to the point cal</longname>
            <type>Ordinal</type>
            <defaultValue>0</defaultValue>
               <maxValue>1</maxValue>
            <labels>
                <label>ClearDone</label>
                <label>ClearCmd</label>
            </labels>
            <units>None</units>
        </item>
        <item name="revertFieldCal">
            <longname>User command to default to the point cal</longname>
            <type>Ordinal</type>
            <defaultValue>0</defaultValue>
       <maxValue>1</maxValue>
            <labels>
                <label>ClearDone</label>
                <label>ClearCmd</label>
            </labels>
            <units>None</units>
        </item>
        <item name="manualCalibrationPoint">
            <longname>User command to capture a cal point</longname>
            <type>Ordinal</type>
```

```
                <defaultValue>0</defaultValue>
        <maxValue>1</maxValue>
                <labels>
                        <label>CaptureDone</label>
                        <label>CaptureCmd</label>
                </labels>
                <units>None</units>
         </item>
         <item name="magAiding">
                <longname>User can clear this to ignore the mag data during AdaptNav
computations</longname>
                <type>Ordinal</type>
                <defaultValue>1</defaultValue>
        <maxValue>1</maxValue>
                <labels>
                        <label>False</label>
                        <label>True</label>
                </labels>
                <units>None</units>
         </item>
         <item name="adaptNavActive">
                <longname>Users use this to turn on/off AdaptNav computations (normally
on)</longname>
                <type>Ordinal</type>
                <defaultValue>1</defaultValue>
        <maxValue>1</maxValue>
                <labels>
                        <label>False</label>
                        <label>True</label>
                </labels>
                <units>None</units>
```

```
        </item>
        <item name="bufferFlushActive">
              <longname>Users use this to turn on/off clearing of field cal
points</longname>
              <type>Ordinal</type>
              <defaultValue>1</defaultValue>
                 <maxValue>1</maxValue>
              <labels>
                    <label>False</label>
                    <label>True</label>
              </labels>
              <units>None</units>
        </item>
        <item name="printFieldCalIndexes">
              <longname>Streams the infield cal indexes</longname>
              <type>Ordinal</type>
              <defaultValue>0</defaultValue>
                 <maxValue>1</maxValue>
              <labels>
                    <label>False</label>
                    <label>True</label>
              </labels>
              <units>None</units>
        </item>
        <item name="compassDataReady">
              <longname>Event that indicates that the computed compass data is
ready</longname>
              <type>Ordinal</type>
              <defaultValue>0</defaultValue>
       <maxValue>1</maxValue>
              <labels>
```

```
                    <label>NotReady</label>
                    <label>Ready</label>
                </labels>
                <units>None</units>
            </item>
        <item name="maxRFSRevision">
            <longname>Puts an upper limit on the RFS revision</longname>
            <type>Int32</type>
            <defaultValue>3</defaultValue>
            <minValue>0</minValue>
            <maxValue>0x7FFFFFFF</maxValue>
            <units>None</units>
        </item>
         <item name="magResetToSetMs">
            <longname>Controls the time from a Reset to a Set in milliseconds(Mode
2)</longname>
            <type>Int32</type>
            <flags>DATA_PERMANENT</flags>
            <defaultValue>50</defaultValue>
            <minValue>0</minValue>
            <maxValue>0x7FFFFFFF</maxValue>
            <units>milliseconds</units>
         </item>
         <item name="roomTempOffset">
            <longname>Value of the raw temperature sensor at 25 degC</longname>
                <type>Int32</type>
                  <flags>DATA_PERMANENT</flags>
                <defaultValue>0</defaultValue>
                <minValue>-99999</minValue>
                <maxValue>99999</maxValue>
```

```
        <units>None</units>
    </item>
        <item name="databaseReady">
        <longname>Indicator for the Cortex-M4 process that the database is ready
(0x0F0FA5A5)</longname>
        <type>Int32</type>
        <defaultValue>0</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>None</units>
    </item>
    <item name="accelXFactorySelfTest">
                <longname>Accel X self-test value generated by the sensor's
factory</longname>
                <type>Int32</type>
                <defaultValue>0</defaultValue>
                <minValue>0</minValue>
                <maxValue>0xFF</maxValue>
                <units>NONE</units>
    </item>
    <item name="accelYFactorySelfTest">
                <longname>Accel Y self-test value generated by the sensor's
factory</longname>
                <type>Int32</type>
                <defaultValue>0</defaultValue>
                <minValue>0</minValue>
                <maxValue>0xFF</maxValue>
                <units>NONE</units>
    </item>
    <item name="accelZFactorySelfTest">
                <longname>Accel Z self-test value generated by the sensor's
factory</longname>
```

```
            <type>Int32</type>
            <defaultValue>0</defaultValue>
            <minValue>0</minValue>
            <maxValue>0xFF</maxValue>
            <units>NONE</units>
   </item>
   <item name="accelSelfTestControlBits">
            <longname>Accel self-test control, bit 0 = Z, bit 1 = Y, bit 2 = X
</longname>
            <type>Int32</type>
            <defaultValue>0</defaultValue>
            <minValue>0</minValue>
            <maxValue>0xFF</maxValue>
            <units>NONE</units>
   </item>
   <item name="accelXSelfTest">
            <longname>Raw Accelerometer X self-test value</longname>
            <flags>DATA_READONLY</flags>
            <type>Int32</type>
            <minValue>0xFFFF8000</minValue>
            <maxValue>0x00007FFF</maxValue>
            <units>counts</units>
   </item>
   <item name="accelYSelfTest">
            <longname>Raw Accelerometer Y self-test value</longname>
            <flags>DATA_READONLY</flags>
            <type>Int32</type>
            <minValue>0xFFFF8000</minValue>
            <maxValue>0x00007FFF</maxValue>
            <units>counts</units>
```

```
</item>
<item name="accelZSelfTest">
          <longname>Raw Accelerometer Z self-test value</longname>
          <flags>DATA_READONLY</flags>
          <type>Int32</type>
          <minValue>0xFFFF8000</minValue>
          <maxValue>0x00007FFF</maxValue>
          <units>counts</units>
</item>
<item name="RFSVersion">
   <longname>Highest RFS version that this software supports</longname>
   <type>Int32</type>
   <flags>DATA_READONLY</flags>
   <defaultValue>5</defaultValue>
   <minValue>0</minValue>
   <maxValue>0x7FFFFFFF</maxValue>
   <units>None</units>
</item>
<item name="VERSION_M4">
   <longname>Subversion tagged ID for the M4 micro software</longname>
   <type>String</type>
   <flags>DATA_READONLY</flags>
   <dimension>80</dimension>
   <defaultValue>undefined</defaultValue>
          <getfunction>Y</getfunction>
   <units>None</units>
</item>
<item name="bornOn">
   <longname>Born on date</longname>
   <type>String</type>
```

```
    <flags>DATA_READONLY|DATA_SUBSYSTEM</flags>
    <dimension>80</dimension>
    <defaultValue>1/1/1970</defaultValue>
    <units>None</units>
</item>
 <item name="yawClearStatistics">
     <longname>Command to clear the yaw statistics (1=ON, 0=OFF)</longname>
        <type>Ordinal</type>
        <defaultValue>0</defaultValue>
        <maxValue>1</maxValue>
        <labels>
               <label>OFF</label>
               <label>ON</label>
        </labels>
    <units>None</units>
 </item>
 <item name="magDecimator">
     <longname>magr is averaged every magDecimator number of raw samples</longname>
     <type>Int32</type>
     <flags>DATA_PERMANENT</flags>
     <defaultValue>2</defaultValue>
     <minValue>0</minValue>
     <maxValue>0x7FFFFFFF</maxValue>
     <units>sample count</units>
 </item>
    <item name="magpCorrected">
          <longname>Processed Magnetometer X,Y,Z array</longname>
          <type>ArrayFloat32</type>
          <flags>DATA_READONLY</flags>
          <dimension>3</dimension>
```

```
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <units>milli-gauss</units>
        </item>
        <item name="gyroBias">
                <longname>Real-time bias correction applied to gyrop to get
gyropCorrected</longname>
                <type>ArrayFloat32</type>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <values>
                        <value>0.0</value>
                        <value>0.0</value>
                        <value>0.0</value>
                </values>
                <units>NONE</units>
        </item>
        <item name="gyroThreshold">
                <longname>gyropCorrected is zeroed under this threshold</longname>
                <type>Float32</type>
       <flags>DATA_PERMANENT</flags>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
       <defaultValue>0.00349066f</defaultValue>
                <units>rads/sec</units>
        </item>
        <item name="gyropBiasCorrected">
                <longname>gyrop with gyroBias correction</longname>
                <type>ArrayFloat32</type>
```

```
        <dimension>3</dimension>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <values>
               <value>0.0</value>
               <value>0.0</value>
               <value>0.0</value>
        </values>
        <units>NONE</units>
</item>
<item name="gyropCorrected">
        <longname>gyropBiasCorrected after gyroThreshold condition</longname>
        <type>ArrayFloat32</type>
        <dimension>3</dimension>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <values>
               <value>0.0</value>
               <value>0.0</value>
               <value>0.0</value>
        </values>
        <units>NONE</units>
</item>
<item name="magpScaleFactor">
        <longname>Multiplied by magp to get magpScaled (change units)</longname>
        <type>Float32</type>
        <defaultValue>0.001</defaultValue>
           <flags>DATA_PERMANENT</flags>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
```

```
            <units>None</units>
        </item>
            <item name="magpScaled">
            <longname>User scaled Magnetometer X,Y,Z array</longname>
            <type>ArrayFloat32</type>
            <flags>DATA_READONLY</flags>
            <dimension>3</dimension>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <units>User configured</units>
        </item>
        <item name="accelpScaleFactor">
            <longname>Multiplied by accelp to get accelpScaled (change
units)</longname>
            <type>Float32</type>
            <defaultValue>0.001</defaultValue>
            <flags>DATA_PERMANENT</flags>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <units>None</units>
        </item>
            <item name="accelpScaled">
            <longname>User scaled Accel X,Y,Z array</longname>
            <type>ArrayFloat32</type>
            <flags>DATA_READONLY</flags>
            <dimension>3</dimension>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <units>User configured</units>
        </item>
```

```
        <item name="gyropScaleFactor">
                <longname>Multiplied by gyrop to get gyropScaled (change
units)</longname>
                <type>Float32</type>
                <defaultValue>57.2957795</defaultValue>
                <flags>DATA_PERMANENT</flags>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <units>None</units>
        </item>
                <item name="gyropScaled">
                <longname>User scaled Gyro X,Y,Z array</longname>
                <type>ArrayFloat32</type>
                <flags>DATA_READONLY</flags>
                <dimension>3</dimension>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <units>User configured</units>
        </item>
        <item name="pitchScaleFactor">
                <longname>Multiplied by pitch to get pitchScaled (change
units)</longname>
                <type>Float32</type>
                <defaultValue>0.017453292</defaultValue>
                <flags>DATA_PERMANENT</flags>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <units>None</units>
        </item>
                <item name="pitchScaled">
                <longname>pitch * pitchScaleFactor</longname>
```

```
        <type>Float32</type>
        <flags>DATA_READONLY</flags>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>User configured</units>
</item>
<item name="rollScaleFactor">
        <longname>Multiplied by roll to get rollScaled (change units)</longname>
        <type>Float32</type>
        <defaultValue>0.017453292</defaultValue>
        <flags>DATA_PERMANENT</flags>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>None</units>
</item>
        <item name="rollScaled">
        <longname>roll * rollScaleFactor</longname>
        <type>Float32</type>
        <flags>DATA_READONLY</flags>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
        <units>User configured</units>
</item>
<item name="yawScaleFactor">
        <longname>Multiplied by yaw to get yawScaled (change units)</longname>
        <type>Float32</type>
        <defaultValue>0.017453292</defaultValue>
        <flags>DATA_PERMANENT</flags>
        <minValue>-3.0E38</minValue>
        <maxValue>3.0E38</maxValue>
```

```
            <units>None</units>
        </item>
            <item name="yawScaled">
            <longname>yaw * yawScaleFactor</longname>
            <type>Float32</type>
            <flags>DATA_READONLY</flags>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <units>User configured</units>
        </item>
        <item name="yawtScaleFactor">
            <longname>Multiplied by yawt to get yawtScaled (change units)</longname>
            <type>Float32</type>
            <defaultValue>0.017453292</defaultValue>
            <flags>DATA_PERMANENT</flags>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <units>None</units>
        </item>
            <item name="yawtScaled">
            <longname>yawt * yawtScaleFactor</longname>
            <type>Float32</type>
            <flags>DATA_READONLY</flags>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <units>User configured</units>
        </item>
    <!-- A Bit field definition -->
    <item name="composite1">
```

```
            <longname>User modifiable composite object with
Roll,Pitch,Yaw,Lat,Lon,Alt,Temp</longname>

            <type>BitField</type>

            <flags>DATA_PERMANENT</flags>

            <dimension>16</dimension>

            <!-- This is the maximum number of fields possible -->

            <fields>

                  <!-- This determines the default number of fields -->

                  <field>

                        <vidName>pitch</vidName>

                        <bits>32</bits>

                  </field>

                  <field>

                        <vidName>roll</vidName>

                        <bits>32</bits>

                  </field>

                  <field>

                        <vidName>yawt</vidName>

                        <bits>32</bits>

                  </field>

                  <field>

                        <vidName>magEstVariance</vidName>

                        <bits>32</bits>

                  </field>

                  <field>

                        <vidName>temperature</vidName>

                        <bits>32</bits>

                  </field>

                  <field>

                        <vidName>magp</vidName>
```

```
                        <bits>96</bits>
                    </field>
                    <field>
                        <vidName>accelp</vidName>
                        <bits>96</bits>
                    </field>
                    <field>
                        <vidName>gyrop</vidName>
                        <bits>96</bits>
                    </field>
                    <field>
                        <vidName>currentTime</vidName>
                        <bits>32</bits>
                    </field>
            </fields>
                <getfunction>N</getfunction>
                <setfunction>N</setfunction>
                <units>None</units>
        </item>
    <item name="accelGyroSMPLRT_DIV">
        <longname>Controls the sample rate divisor for the accel/gyro
sensor</longname>
        <type>Int32</type>
        <flags>DATA_PERMANENT</flags>
        <defaultValue>9</defaultValue>
        <minValue>0</minValue>
        <maxValue>999</maxValue>
        <units>Fortnights</units>
    </item>
   <item name="comp1Rate">
```

```
            <longname>Composite 1 report period (milliseconds)</longname>

            <type>Int32</type>

            <flags>0</flags>

            <defaultValue>0</defaultValue>

            <minValue>0</minValue>

            <maxValue>180000</maxValue>

            <units>milliseconds per report</units>

        </item>

    <item name="accelrangeFactor">

        <longname>Used to adjust the raw accel data according to the value of
accelrange that is being used</longname>

        <type>Float32</type>

        <defaultValue>1.0</defaultValue>

        <minValue>0.5</minValue>

        <maxValue>4.0</maxValue>

        <units>None</units>

    </item>

    <item name="accelCalTemperatureRaw">

        <longname>Temperature at which the Rate Table temperature comps are computed
for each axis (256*C)</longname>

        <type>ArrayInt32</type>

             <flags>DATA_PERMANENT|DATA_SUBSYSTEM</flags>

        <dimension>3</dimension>

        <minValue>-32768</minValue>

        <maxValue>32767</maxValue>

        <values>

            <value>7450</value>

            <value>7450</value>

            <value>7450</value>

        </values>

        <setfunction>N</setfunction>
```

```
            <units>256timesC</units>
        </item>
        <item name="gyroCalTemperatureRaw">
            <longname>Temperature at which the Rate Table temperature comps are computed
for each axis (256*C)</longname>
            <type>ArrayInt32</type>
                 <flags>DATA_PERMANENT|DATA_SUBSYSTEM</flags>
            <dimension>3</dimension>
            <minValue>-32768</minValue>
            <maxValue>32767</maxValue>
            <values>
                <value>7450</value>
                <value>7450</value>
                <value>7450</value>
            </values>
            <setfunction>N</setfunction>
            <units>256timesC</units>
        </item>
          <item name="gyroConditioned">
                 <longname>Factory calibrated gyro X,Y,Z array</longname>
                 <type>ArrayFloat32</type>
                 <flags>DATA_READONLY</flags>
                 <dimension>3</dimension>
                 <minValue>-999999999.0</minValue>
                 <maxValue>999999999.0</maxValue>
                 <units>NONE</units>
          </item>
          <item name="magConditioned">
                 <longname>Factory calibrated magnetometer X,Y,Z array</longname>
                 <type>ArrayFloat32</type>
```

```
            <flags>DATA_READONLY</flags>
            <dimension>3</dimension>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <units>milli-gauss</units>
        </item>
    <item name="centripetalCompensation">
            <longname>Enable Centripetal acceleration correction of acceleration
</longname>
            <type>Ordinal</type>
            <flags>DATA_PERMANENT</flags>
            <defaultValue>0</defaultValue>
            <maxValue>1</maxValue>
            <labels>
                    <label>Off</label>
                    <label>On</label>
            </labels>
            <units>None</units>
    </item>
       <item name="f_km0">
            <longname>Algorithm constant - magnetic gain</longname>
            <type>Float32</type>
            <flags>DATA_PERMANENT</flags>
            <defaultValue>0.01</defaultValue>
            <minValue>-3.0E38</minValue>
            <maxValue>3.0E38</maxValue>
            <units>None</units>
       </item>
     <item name="eeLogDatabaseWrite">
            <longname>Info indicating number of database writes to user EEPROM
area</longname>
```

```
                <type>ArrayInt32</type>

                <flags>DATA_READONLY</flags>

                <dimension>3</dimension>

                <minValue>0</minValue>

                <maxValue>0x7FFFFFFF</maxValue>

                <values>

                        <value>0</value>

                        <value>0</value>

                        <value>0</value>

                </values>

                <units>NONE</units>

        </item>

        <!-- This is the individual bits that are enabled for a stream output -->

        <!-- A simplified bitfield setting. Needs to have one bit per variable
currently around 256 bits -->

        <!-- This takes 8 words, so double it to 16 words each bit is a variable so
this will allow for up to 512 -->

        <!-- Note that this function must have a set function to allow all 32 bits to
be set as the database treats as signed -->

        <item name="chan0TriggerDivisor">

                <longname>Defines the number of channel 0 triggers per channel 0 output
</longname>

                <type>Int32</type>

                <flags>0</flags>

                <defaultValue>0</defaultValue>

                <minValue>0</minValue>

                <maxValue>0x7fffffff</maxValue>

                <units>triggers per output</units>

        </item>

        <item name="chan1TriggerDivisor">

                <longname>Defines the number of channel 1 triggers per channel 1 output
</longname>
```

```
                <type>Int32</type>
                <flags>0</flags>
                <defaultValue>0</defaultValue>
                <minValue>0</minValue>
                <maxValue>0x7fffffff</maxValue>
                <units>triggers per output</units>
        </item>
        <item name="chan2TriggerDivisor">
                <longname>Defines the number of channel 2 triggers per channel 2 output
</longname>
                <type>Int32</type>
                <flags>0</flags>
                <defaultValue>0</defaultValue>
                <minValue>0</minValue>
                <maxValue>0x7fffffff</maxValue>
                <units>triggers per output</units>
        </item>
        <item name="chan0Enables">
                <longname>Bit mask the enables vids for output via channel 0</longname>
                <type>ArrayInt32</type>
                <dimension>16</dimension>
                <flags>DATA_PERMANENT</flags>
                <setfunction>Y</setfunction>
                <defaultValue>0</defaultValue>
                <minValue>0x80000000</minValue>
                <maxValue>0x7FFFFFFF</maxValue>
                <units>None</units>
        </item>
        <item name="chan1Enables">
                <longname>Bit mask the enables vids for output via channel 1</longname>
```

```
            <type>ArrayInt32</type>
            <dimension>16</dimension>
            <flags>DATA_PERMANENT</flags>
            <setfunction>Y</setfunction>
            <defaultValue>0</defaultValue>
            <minValue>0x80000000</minValue>
            <maxValue>0x7FFFFFFF</maxValue>
            <units>None</units>
      </item>
      <item name="chan2Enables">
            <longname>Bit mask the enables vids for output via channel 2</longname>
            <type>ArrayInt32</type>
            <dimension>16</dimension>
            <flags>DATA_PERMANENT</flags>
            <setfunction>Y</setfunction>
            <defaultValue>0</defaultValue>
            <minValue>0x80000000</minValue>
            <maxValue>0x7FFFFFFF</maxValue>
            <units>None</units>
      </item>
      <item name="chan0EnableBit">
            <longname>Simple function to enable a single bit in the chan0Enables
variable</longname>
            <type>Int32</type>
            <setfunction>Y</setfunction>
            <defaultValue>0</defaultValue>
            <minValue>0</minValue>
            <maxValue>512</maxValue>
            <units>None</units>
      </item>
```

```
        <item name="chan0DisableBit">

                <longname>Simple function to disable a single bit in the chan0Enables
variable</longname>

                <type>Int32</type>

                <setfunction>Y</setfunction>

                <defaultValue>0</defaultValue>

                <minValue>0</minValue>

                <maxValue>512</maxValue>

                <units>None</units>

        </item>

        <item name="chan1EnableBit">

                <longname>Simple function to enable a single bit in the chan1Enables
variable</longname>

                <type>Int32</type>

                <setfunction>Y</setfunction>

                <defaultValue>0</defaultValue>

                <minValue>0</minValue>

                <maxValue>512</maxValue>

                <units>None</units>

        </item>

        <item name="chan1DisableBit">

                <longname>Simple function to disable a single bit in the chan1Enables
variable</longname>

                <type>Int32</type>

                <setfunction>Y</setfunction>

                <defaultValue>0</defaultValue>

                <minValue>0</minValue>

                <maxValue>512</maxValue>

                <units>None</units>

        </item>

        <item name="chan2EnableBit">
```

```
            <longname>Simple function to enable a single bit in the chan2Enables
variable</longname>

            <type>Int32</type>

            <setfunction>Y</setfunction>

            <defaultValue>0</defaultValue>

            <minValue>0</minValue>

            <maxValue>512</maxValue>

            <units>None</units>

      </item>

      <item name="chan2DisableBit">

            <longname>Simple function to disable a single bit in the chan2Enables
variable</longname>

            <type>Int32</type>

            <setfunction>Y</setfunction>

            <defaultValue>0</defaultValue>

            <minValue>0</minValue>

            <maxValue>512</maxValue>

            <units>None</units>

      </item>

   <!-- Note that thes events have to match the triggers for the channels  and the
nav trigger since a common index is passed -->

   <!-- to one of the functions -->

   <item name="chan0Trigger">

      <longname>Event that triggers automatic preprogrammed output via Channel
0</longname>

      <type>Ordinal</type>

      <flags>DATA_PERMANENT</flags>

      <defaultValue>0</defaultValue>

      <maxValue>6</maxValue>

      <labels>

         <label>Off</label>

         <label>AccelRawData</label>
```

```
            <label>GyroRawData</label>
            <label>MagRawData</label>
            <label>ConditionedData</label>
            <label>CompassData</label>
            <label>Timer</label>
        </labels>
    <units>None</units>
    </item>
    <item name="chan1Trigger">
        <longname>Event that triggers automatic preprogrammed output via Channel
1</longname>
        <type>Ordinal</type>
        <flags>DATA_PERMANENT</flags>
        <defaultValue>0</defaultValue>
        <maxValue>6</maxValue>
        <labels>
            <label>Off</label>
            <label>AccelRawData</label>
            <label>GyroRawData</label>
            <label>MagRawData</label>
            <label>ConditionedData</label>
            <label>CompassData</label>
            <label>Timer</label>
        </labels>
    <units>None</units>
    </item>
    <item name="chan2Trigger">
        <longname>Event that triggers automatic preprogrammed output via Channel
2</longname>
        <type>Ordinal</type>
        <flags>DATA_PERMANENT</flags>
```

```
      <defaultValue>0</defaultValue>
      <maxValue>6</maxValue>
      <labels>
         <label>Off</label>
         <label>AccelRawData</label>
         <label>GyroRawData</label>
         <label>MagRawData</label>
         <label>ConditionedData</label>
         <label>CompassData</label>
         <label>Timer</label>
      </labels>
   <units>None</units>
   </item>
      <item name="chan0Format">
            <longname>Channel 0's format for automatic output of variables</longname>
            <type>Ordinal</type>
            <flags>DATA_PERMANENT</flags>
            <defaultValue>1</defaultValue>
   <maxValue>4</maxValue>
            <labels>
                  <label>RFS</label>
                  <label>NorthTek</label>
                  <label>BitstreamBinary</label>
                  <label>BitstreamAscii</label>
                  <label>NMEA</label>
            </labels>
            <units>None</units>
      </item>
      <item name="chan1Format">
            <longname>Channel 1's format for automatic output of variables</longname>
```

```
            <type>Ordinal</type>
            <flags>DATA_PERMANENT</flags>
            <defaultValue>1</defaultValue>
<maxValue>4</maxValue>
            <labels>
                  <label>RFS</label>
                  <label>NorthTek</label>
                  <label>BitstreamBinary</label>
                  <label>BitstreamAscii</label>
                  <label>NMEA</label>
            </labels>
            <units>None</units>
   </item>
   <item name="chan2Format">
            <longname>Channel 2's format for automatic output of variables</longname>
            <type>Ordinal</type>
            <flags>DATA_PERMANENT</flags>
            <defaultValue>1</defaultValue>
<maxValue>4</maxValue>
            <labels>
                  <label>RFS</label>
                  <label>NorthTek</label>
                  <label>BitstreamBinary</label>
                  <label>BitstreamAscii</label>
                  <label>NMEA</label>
            </labels>
            <units>None</units>
   </item>
   <item name="chan0Preamble">
            <longname>Prefix for channel 0's BitstreamAscii string</longname>
```

```
            <type>String</type>
            <flags>DATA_PERMANENT</flags>
            <dimension>80</dimension>
            <defaultValue>$CUS0,</defaultValue>
            <units>None</units>
      </item>
      <item name="chan0Postamble">
            <longname>Postfix for channel 0's BitstreamAscii string</longname>
            <type>String</type>
            <flags>DATA_PERMANENT</flags>
            <dimension>80</dimension>
            <defaultValue>*</defaultValue>
            <units>None</units>
      </item>
   <item name="chan1Preamble">
       <longname>Prefix for channel 1's BitstreamAscii string</longname>
            <type>String</type>
            <flags>DATA_PERMANENT</flags>
            <dimension>80</dimension>
            <defaultValue>$CUS1,</defaultValue>
            <units>None</units>
      </item>
   <item name="chan1Postamble">
       <longname>Postfix for channel 1's BitstreamAscii string</longname>
            <type>String</type>
            <flags>DATA_PERMANENT</flags>
            <dimension>80</dimension>
            <defaultValue>*</defaultValue>
            <units>None</units>
      </item>
```

```
    <item name="chan2Preamble">
        <longname>Prefix for channel 2's BitstreamAscii string</longname>
            <type>String</type>
            <flags>DATA_PERMANENT</flags>
            <dimension>80</dimension>
            <defaultValue>$CUS2,</defaultValue>
            <units>None</units>
      </item>
    <item name="chan2Postamble">
        <longname>Postfix for channel 2's BitstreamAscii string</longname>
            <type>String</type>
            <flags>DATA_PERMANENT</flags>
            <dimension>80</dimension>
            <defaultValue>*</defaultValue>
            <units>None</units>
      </item>
      <item name="chan0Command">
            <longname>NorthTek or NMEA command string invoked by the channel0
trigger</longname>
            <type>String</type>
            <flags>0</flags>
            <dimension>80</dimension>
            <defaultValue>.ok</defaultValue>
            <units>None</units>
      </item>
      <item name="chan1Command">
            <longname>NorthTek or NMEA command string invoked by the channel0
trigger</longname>
            <type>String</type>
            <flags>0</flags>
            <dimension>80</dimension>
```

```
                <defaultValue>.ok</defaultValue>

                <units>None</units>

        </item>

        <item name="chan2Command">

                <longname>NorthTek or NMEA command string invoked by the channel0
trigger</longname>

                <type>String</type>

                <flags>0</flags>

                <dimension>80</dimension>

                <defaultValue>.ok</defaultValue>

                <units>None</units>

        </item>

    <item name="chanTimerX100us">

     <longname>Defines the timer trigger event period which is this number times 100
microseconds</longname>

     <type>Int32</type>

     <flags>DATA_PERMANENT</flags>

     <defaultValue>10000</defaultValue>

     <minValue>5</minValue>

     <maxValue>600000</maxValue>

     <setfunction>Y</setfunction>

     <units>NONE</units>

   </item>

        <item name="initCalPointBufferRequest">

                <longname>Clears the field cal point buffers</longname>

                <type>Ordinal</type>

                <defaultValue>1</defaultValue>

                <maxValue>1</maxValue>

                <labels>

                        <label>OFF</label>

                        <label>ON</label>
```

```
            </labels>
            <setfunction>N</setfunction>
            <units>None</units>
     </item>
      <item name="AHRS8_Compatible">
    <longname>Makes the AHRS-M2 largely compatible with AHRS-8 interface</longname>
    <type>Ordinal</type>
    <flags>DATA_PERMANENT</flags>
    <defaultValue>0</defaultValue>
    <maxValue>1</maxValue>
    <labels>
       <label>No</label>
       <label>Yes</label>
    </labels>
    <setfunction>N</setfunction>
    <units>None</units>
</item>
     <item name="magDecimated">
            <longname>Averaged magr based on magDecimator</longname>
            <type>ArrayInt32</type>
            <dimension>3</dimension>
            <minValue>0xFFFF8000</minValue>
            <maxValue>0x00007FFF</maxValue>
            <values>
                  <value>0</value>
                  <value>0</value>
                  <value>0</value>
            </values>
            <units>counts</units>
     </item>
```

```
        <item name="magP2PErr">
                <longname>Magnetic peak to peak error in field cal points</longname>
                <type>Float32</type>
                <flags>DATA_READONLY</flags>
                <defaultValue>9999999.0</defaultValue>
                <minValue>0.0</minValue>
                <maxValue>9999999.0</maxValue>
                <units>mG</units>
        </item>
        <item name="angularP2PErr">
                <longname>Angular peak to peak error in field cal points</longname>
                <type>Float32</type>
                <flags>DATA_READONLY</flags>
                <defaultValue>9999999.0</defaultValue>
                <minValue>0.0</minValue>
                <maxValue>9999999.0</maxValue>
                <units>radians</units>
        </item>
        <item name="angularMagP2PErr">
                <longname>Magnetic angle peak to peak error in field cal
points</longname>
                <type>Float32</type>
                <flags>DATA_READONLY</flags>
                <defaultValue>9999999.0</defaultValue>
                <minValue>0.0</minValue>
                <maxValue>9999999.0</maxValue>
                <units>radians</units>
        </item>
        <item name="adaptNavIIIStartTime">
                <longname>Time the adaptNavIII task starts (milliseconds)</longname>
```

```
                <type>Int32</type>
                <flags>0</flags>
                <defaultValue>0</defaultValue>
                <minValue>0</minValue>
                <maxValue>0x7FFFFFFF</maxValue>
                <units>ms</units>
        </item>
        <item name="adaptNavIIIEndTime">
                <longname>Time the adaptNavIII task ends (milliseconds)</longname>
                <type>Int32</type>
                <flags>0</flags>
                <defaultValue>0</defaultValue>
                <minValue>0</minValue>
                <maxValue>0x7FFFFFFF</maxValue>
                <units>ms</units>
        </item>
        <item name="adaptNavIIICalCovStart">
                <longname>Time the adaptNavIII calConvergence task starts
(milliseconds)</longname>
                <type>Int32</type>
                <flags>0</flags>
                <defaultValue>0</defaultValue>
                <minValue>0</minValue>
                <maxValue>0x7FFFFFFF</maxValue>
                <units>reports per second</units>
        </item>
        <item name="adaptNavIIICalCovEnd">
                <longname>Time the adaptNavIII calConvergence task ends
(milliseconds)</longname>
                <type>Int32</type>
                <flags>0</flags>
```

```
                <defaultValue>0</defaultValue>

                <minValue>0</minValue>

                <maxValue>0x7FFFFFFF</maxValue>

                <units>reports per second</units>

        </item>

        <item name="calCovCnt">

                <longname>Number of iterations of the calibration convergence
algo</longname>

                <type>Int32</type>

                <flags>0</flags>

                <defaultValue>0</defaultValue>

                <minValue>0</minValue>

                <maxValue>0x7FFFFFFF</maxValue>

                <units>reports per second</units>

        </item>

        <item name="resetGyroBias">

                <longname>Set this to 1 to set gyroBias to zero</longname>

                <type>Ordinal</type>

                <defaultValue>0</defaultValue>

                <maxValue>1</maxValue>

                <labels>

                        <label>OFF</label>

                        <label>ON</label>

                </labels>

                <setfunction>N</setfunction>

                <units>None</units>

        </item>

        <item name="resetMagEstVariance">

                <longname>Set this to 1 to restore magEstVariance back to its default
value</longname>

                <type>Ordinal</type>
```

```
                <defaultValue>0</defaultValue>
                <maxValue>1</maxValue>
                <labels>
                        <label>OFF</label>
                        <label>ON</label>
                </labels>
                <setfunction>N</setfunction>
                <units>None</units>
        </item>
        <item name="accelGyroM4Timestamp">
                <longname>Time the M4 received the accel and gyro raw values</longname>
                <type>Int32</type>
                <defaultValue>0</defaultValue>
        <maxValue>0x7FFFFFFF</maxValue>
                <units>ms</units>
        </item>
        <item name="accelGyroDecimatorStartTime">
                <longname>Time the M0 sensor decimator task received the data</longname>
                <type>Int32</type>
                <defaultValue>0</defaultValue>
        <maxValue>0x7FFFFFFF</maxValue>
                <units>ms</units>
        </item>
        <item name="accelGyroDecimatorEndTime">
                <longname>Time the M0 sensor decimator task finished with the
data</longname>
                <type>Int32</type>
                <defaultValue>0</defaultValue>
        <maxValue>0x7FFFFFFF</maxValue>
                <units>ms</units>
```

```
        </item>
    <item name="magSetToResetMs">
        <longname>Controls the time from a Set to a Reset in milliseconds(Mode
2)</longname>
        <type>Int32</type>
        <flags>DATA_PERMANENT</flags>
        <defaultValue>50</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>milliseconds</units>
    </item>
    <item name="magBridgeBias">
            <longname>For Mode 2 only, adjustment to sensor raw data</longname>
            <type>ArrayInt32</type>
            <dimension>3</dimension>
            <minValue>0xFFFF8000</minValue>
            <maxValue>0x00007FFF</maxValue>
            <values>
                <value>32768</value>
                <value>32768</value>
                <value>32768</value>
            </values>
            <units>counts</units>
       </item>
    <item name="magSetResetSetting">
        <longname>For Mode 1 only, selects whether the Set or Reset is used</longname>
        <type>Ordinal</type>
        <flags>DATA_PERMANENT</flags>
       <defaultValue>0</defaultValue>
       <maxValue>1</maxValue>
```

```
        <labels>
           <label>SET</label>
           <label>RESET</label>
        </labels>
        <units>None</units>
    </item>
    <item name="magConversionDelayuS">
        <longname>The delay between the start mag conversion and the data read
(uSecs)</longname>
        <type>Int32</type>
        <defaultValue>4100</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>uSecs</units>
    </item>
    <item name="magSetResetDelayuS">
        <longname>The delay between a set or reset and a refill cap (uSecs)</longname>
        <type>Int32</type>
        <defaultValue>1000</defaultValue>
        <minValue>0</minValue>
        <maxValue>0x7FFFFFFF</maxValue>
        <units>uSecs</units>
    </item>
       <item name="senPwrCntrlToggleTime">
             <longname>On start-up, toggle the sensors' power for this many
milliseconds</longname>
             <type>Int32</type>
      <flags>DATA_PERMANENT</flags>
             <defaultValue>0</defaultValue>
      <maxValue>9999999</maxValue>
             <units>ms</units>
```

```
        </item>
        <item name="gyroBiasEstVarianceInit">
                <longname>Initial value of the estimated variance of the gyro
bias</longname>
                <type>Float32</type>
                   <flags>DATA_PERMANENT</flags>
                <defaultValue>0.00010966227f</defaultValue>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <units>None</units>
        </item>
        <item name="estimatedRMSYawError">
                <longname>Estimated Root Mean Square yaw error</longname>
                <type>Float32</type>
                <defaultValue>9999.0f</defaultValue>
                <minValue>-3.0E38</minValue>
                <maxValue>3.0E38</maxValue>
                <units>degrees</units>
        </item>

      <item name="adaptNavIIIDeltaTime">
                <longname>Time to run the AdaptNavIII algorithm</longname>
                <type>Int32</type>
                <defaultValue>0</defaultValue>
                   <maxValue>0x7FFFFFFF</maxValue>
                <units>ms</units>
      </item>
      <item name="reserved">
                <longname>Reserved object</longname>
                <type>Int32</type>
```

```
            <flags>DATA_READONLY</flags>
            <defaultValue>305463004</defaultValue>
            <minValue>0</minValue>
            <getfunction>N</getfunction>
            <setfunction>N</setfunction>
            <maxValue>0x7FFFFFFF</maxValue>
            <units>None</units>
        </item>
</embeddedDB>
```

B Troubleshooting

B.1 Repeating output is not appearing.

- type <ctrl-Q> since you may have previously typed a <ctrl-S>

B.2 Repeating output is making typing a new command difficult.

- enter <ctrl-S> to pause the repeat display and type in your new command, restart any repeat displays with <ctrl-Q>

B.3 The inertial system does not output any response.

The inertial system does not output anything by default. It only responds on command (such as an <Enter> key). If there is no response to a command, try the following:

- Ensure that your terminal emulator has not lost its connection to the com port.
- Check that there is a TTL logic level connection to inertial system user port pins, USER_RXD and USER_TXD
- Check that the baud rate matches the baud rate setting on the inertial system (initial factory setting is 115200)
- Check that the other serial interface properties are 8 bit, no parity, 1 stop bit, no flow control
- inertial system has power (4-10Volts DC) and ground

B.4 No NMEA response

- Send ‘nmeaecho 1 set drop’ to echo characters back to terminal
- Make sure the terminal is configured to transmit ‘LF+CR’ for each line sent

B.5 Compass heading drifts then settles on a heading

- Compass magnetometers need to be calibrated. Follow the instructions in the user guide to do a manual or adaptive calibration.
- Insure magAiding is set to 1.

- Consider external magnetic interference

B.6 Compass constantly spinning through 360 degrees

- May be caused by inaccurate gyro biases.
- Keep compass stationary and send 'gyroThreshold f1000.0 set drop' and wait a few seconds for the spinning to stop and the gyros to recalibrate.
- After a few seconds, send 'gyroThreshold >default' and then 'save' to save the default threshold value and new gyro biases to non-volatile memory
- Insure magAiding is set to 1.

C Utility Code – CRCs and Checksums

C.1 CRC

```
/* A little table to bit reverse bytes easily */
const Byte flip[] =
{
   0x00,0x80,0x40,0xc0,0x20,0xa0,0x60,0xe0,
   0x10,0x90,0x50,0xd0,0x30,0xb0,0x70,0xf0,
   0x08,0x88,0x48,0xc8,0x28,0xa8,0x68,0xe8,
   0x18,0x98,0x58,0xd8,0x38,0xb8,0x78,0xf8,
   0x04,0x84,0x44,0xc4,0x24,0xa4,0x64,0xe4,
   0x14,0x94,0x54,0xd4,0x34,0xb4,0x74,0xf4,
   0x0c,0x8c,0x4c,0xcc,0x2c,0xac,0x6c,0xec,
   0x1c,0x9c,0x5c,0xdc,0x3c,0xbc,0x7c,0xfc,
   0x02,0x82,0x42,0xc2,0x22,0xa2,0x62,0xe2,
   0x12,0x92,0x52,0xd2,0x32,0xb2,0x72,0xf2,
   0x0a,0x8a,0x4a,0xca,0x2a,0xaa,0x6a,0xea,
   0x1a,0x9a,0x5a,0xda,0x3a,0xba,0x7a,0xfa,
   0x06,0x86,0x46,0xc6,0x26,0xa6,0x66,0xe6,
   0x16,0x96,0x56,0xd6,0x36,0xb6,0x76,0xf6,
   0x0e,0x8e,0x4e,0xce,0x2e,0xae,0x6e,0xee,
   0x1e,0x9e,0x5e,0xde,0x3e,0xbe,0x7e,0xfe,
   0x01,0x81,0x41,0xc1,0x21,0xa1,0x61,0xe1,
   0x11,0x91,0x51,0xd1,0x31,0xb1,0x71,0xf1,
   0x09,0x89,0x49,0xc9,0x29,0xa9,0x69,0xe9,
   0x19,0x99,0x59,0xd9,0x39,0xb9,0x79,0xf9,
   0x05,0x85,0x45,0xc5,0x25,0xa5,0x65,0xe5,
   0x15,0x95,0x55,0xd5,0x35,0xb5,0x75,0xf5,
   0x0d,0x8d,0x4d,0xcd,0x2d,0xad,0x6d,0xed,
   0x1d,0x9d,0x5d,0xdd,0x3d,0xbd,0x7d,0xfd,
```

```
    0x03,0x83,0x43,0xc3,0x23,0xa3,0x63,0xe3,
    0x13,0x93,0x53,0xd3,0x33,0xb3,0x73,0xf3,
    0x0b,0x8b,0x4b,0xcb,0x2b,0xab,0x6b,0xeb,
    0x1b,0x9b,0x5b,0xdb,0x3b,0xbb,0x7b,0xfb,
    0x07,0x87,0x47,0xc7,0x27,0xa7,0x67,0xe7,
    0x17,0x97,0x57,0xd7,0x37,0xb7,0x77,0xf7,
    0x0f,0x8f,0x4f,0xcf,0x2f,0xaf,0x6f,0xef,
    0x1f,0x9f,0x5f,0xdf,0x3f,0xbf,0x7f,0xff
};

const Byte remainderHigh[256] =
 {
    0x00,0x10,0x20,0x30,0x40,0x50,0x60,0x70,
    0x81,0x91,0xa1,0xb1,0xc1,0xd1,0xe1,0xf1,
    0x12,0x02,0x32,0x22,0x52,0x42,0x72,0x62,
    0x93,0x83,0xb3,0xa3,0xd3,0xc3,0xf3,0xe3,
    0x24,0x34,0x04,0x14,0x64,0x74,0x44,0x54,
    0xa5,0xb5,0x85,0x95,0xe5,0xf5,0xc5,0xd5,
    0x36,0x26,0x16,0x06,0x76,0x66,0x56,0x46,
    0xb7,0xa7,0x97,0x87,0xf7,0xe7,0xd7,0xc7,
    0x48,0x58,0x68,0x78,0x08,0x18,0x28,0x38,
    0xc9,0xd9,0xe9,0xf9,0x89,0x99,0xa9,0xb9,
    0x5a,0x4a,0x7a,0x6a,0x1a,0x0a,0x3a,0x2a,
    0xdb,0xcb,0xfb,0xeb,0x9b,0x8b,0xbb,0xab,
    0x6c,0x7c,0x4c,0x5c,0x2c,0x3c,0x0c,0x1c,
    0xed,0xfd,0xcd,0xdd,0xad,0xbd,0x8d,0x9d,
    0x7e,0x6e,0x5e,0x4e,0x3e,0x2e,0x1e,0x0e,
    0xff,0xef,0xdf,0xcf,0xbf,0xaf,0x9f,0x8f,
    0x91,0x81,0xb1,0xa1,0xd1,0xc1,0xf1,0xe1,
    0x10,0x00,0x30,0x20,0x50,0x40,0x70,0x60,
    0x83,0x93,0xa3,0xb3,0xc3,0xd3,0xe3,0xf3,
    0x02,0x12,0x22,0x32,0x42,0x52,0x62,0x72,
    0xb5,0xa5,0x95,0x85,0xf5,0xe5,0xd5,0xc5,
    0x34,0x24,0x14,0x04,0x74,0x64,0x54,0x44,
    0xa7,0xb7,0x87,0x97,0xe7,0xf7,0xc7,0xd7,
    0x26,0x36,0x06,0x16,0x66,0x76,0x46,0x56,
    0xd9,0xc9,0xf9,0xe9,0x99,0x89,0xb9,0xa9,
    0x58,0x48,0x78,0x68,0x18,0x08,0x38,0x28,
    0xcb,0xdb,0xeb,0xfb,0x8b,0x9b,0xab,0xbb,
    0x4a,0x5a,0x6a,0x7a,0x0a,0x1a,0x2a,0x3a,
    0xfd,0xed,0xdd,0xcd,0xbd,0xad,0x9d,0x8d,
    0x7c,0x6c,0x5c,0x4c,0x3c,0x2c,0x1c,0x0c,
```

```
    0xef,0xff,0xcf,0xdf,0xaf,0xbf,0x8f,0x9f,
    0x6e,0x7e,0x4e,0x5e,0x2e,0x3e,0x0e,0x1e,
  };


const Byte remainderLow[256] =
  {
    0x00,0x21,0x42,0x63,0x84,0xa5,0xc6,0xe7,
    0x08,0x29,0x4a,0x6b,0x8c,0xad,0xce,0xef,
    0x31,0x10,0x73,0x52,0xb5,0x94,0xf7,0xd6,
    0x39,0x18,0x7b,0x5a,0xbd,0x9c,0xff,0xde,
    0x62,0x43,0x20,0x01,0xe6,0xc7,0xa4,0x85,
    0x6a,0x4b,0x28,0x09,0xee,0xcf,0xac,0x8d,
    0x53,0x72,0x11,0x30,0xd7,0xf6,0x95,0xb4,
    0x5b,0x7a,0x19,0x38,0xdf,0xfe,0x9d,0xbc,
    0xc4,0xe5,0x86,0xa7,0x40,0x61,0x02,0x23,
    0xcc,0xed,0x8e,0xaf,0x48,0x69,0x0a,0x2b,
    0xf5,0xd4,0xb7,0x96,0x71,0x50,0x33,0x12,
    0xfd,0xdc,0xbf,0x9e,0x79,0x58,0x3b,0x1a,
    0xa6,0x87,0xe4,0xc5,0x22,0x03,0x60,0x41,
    0xae,0x8f,0xec,0xcd,0x2a,0x0b,0x68,0x49,
    0x97,0xb6,0xd5,0xf4,0x13,0x32,0x51,0x70,
    0x9f,0xbe,0xdd,0xfc,0x1b,0x3a,0x59,0x78,
    0x88,0xa9,0xca,0xeb,0x0c,0x2d,0x4e,0x6f,
    0x80,0xa1,0xc2,0xe3,0x04,0x25,0x46,0x67,
    0xb9,0x98,0xfb,0xda,0x3d,0x1c,0x7f,0x5e,
    0xb1,0x90,0xf3,0xd2,0x35,0x14,0x77,0x56,
    0xea,0xcb,0xa8,0x89,0x6e,0x4f,0x2c,0x0d,
    0xe2,0xc3,0xa0,0x81,0x66,0x47,0x24,0x05,
    0xdb,0xfa,0x99,0xb8,0x5f,0x7e,0x1d,0x3c,
    0xd3,0xf2,0x91,0xb0,0x57,0x76,0x15,0x34,
    0x4c,0x6d,0x0e,0x2f,0xc8,0xe9,0x8a,0xab,
    0x44,0x65,0x06,0x27,0xc0,0xe1,0x82,0xa3,
    0x7d,0x5c,0x3f,0x1e,0xf9,0xd8,0xbb,0x9a,
    0x75,0x54,0x37,0x16,0xf1,0xd0,0xb3,0x92,
    0x2e,0x0f,0x6c,0x4d,0xaa,0x8b,0xe8,0xc9,
    0x26,0x07,0x64,0x45,0xa2,0x83,0xe0,0xc1,
    0x1f,0x3e,0x5d,0x7c,0x9b,0xba,0xd9,0xf8,
    0x17,0x36,0x55,0x74,0x93,0xb2,0xd1,0xf0
  };
```

```
Word16 computeCRC(Byte *buffer,Int32 length,Word16 startingValue,Byte doLSB)
{
   Byte   crcH,crcL;
   Byte index;
   crcH = startingValue >> 8;
   crcL = startingValue & 0xff;
   if(doLSB)
   {
      while(length--)
      {
         index = ((flip[*buffer++]) ^ (crcH));
         crcH = (crcL) ^ remainderHigh[index];
         crcL = remainderLow[index];
      }//while
   }
   else
   {
      while(length--)
      {
         index = (*buffer++) ^ crcH;
         // crc = (crc<<8) ^ mt[index];
         crcH = (crcL) ^ remainderHigh[index];
         crcL = remainderLow[index];
      }//while
   }
   return((((Word16)crcH)<<8) | (Word16)crcL);
}

// use the MSB version so we don't have to bit flip the CRC value
// when sending, the other end will use the same source code
// to check the crc, so it does not have to match CCITT standards exactly.

#define NOMINAL_CRC_SEED  0xFFFF
Word16 calcCRC(Byte *buffer, Int32 length)
{
  return(computeCRC(buffer,length,NOMINAL_CRC_SEED,0));
}
```

C.2 Checksum

```
Byte computeChecksum(Byte *ptr,Int32 length)
```

```
{
  int i;
  Byte checksum = 0;
  for(i = 0;i < length;i++)
  {
    checksum ^= *ptr;
    ptr++;
  }
  return(checksum);
}
```