# CH Robotics

CHR-6dm Attitude and Heading Reference System
Product datasheet - Rev. 1.0, Preliminary

Device Overview
The CHR-6dm AHRS is a cost-effective orientation sensor providing yaw, pitch, and roll angle outputs at up to 300 Hz. An Extended Kalman Filter (EKF) combines data from onboard accelerometers, rate gyros, and magnetic sensors to produce yaw, pitch, and roll angle estimates.
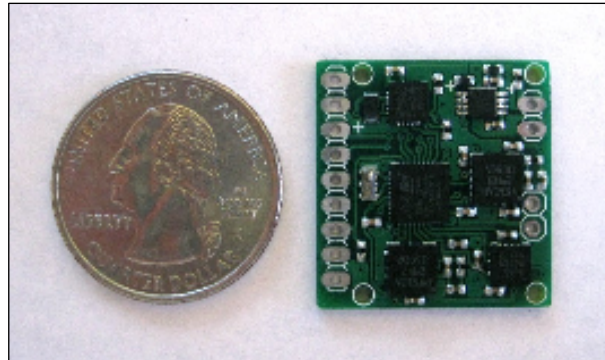
Communication with the CHR-6dm is performed over a TTL (3.3V) UART at 115200 Baud. The AHRS can be configured to transmit raw sensor in addition to angle estimates, and the transmission rate can be configured in 1 Hz increments from 20 Hz to 300 Hz. Alternatively, the CHR-6dm can operate in "silent mode," where data is transmitted only when specific requests are received over the UART. Regardless of the transmission mode and rate, internal angle estimates are maintained at over 500 Hz to ensure long-term accuracy.

The CHR-6dm simplifies integration by providing a number of automatic calibration routines, including rate gyro bias calibration, magnetometer hard and soft iron calibration, and accelerometer "zeroing" to compensate for AHRS-platform misalignment. All calibration routines are triggered by sending simple commands over the serial interface.

An onboard 3.3V regulator also simplifies integration. With possible input voltages ranging from +3.3V to +12V, and with 5V tolerant IO pins, the CHR-6dm is easily integrated with a wide range of systems. The CHR-6dm also provides +3.3V outputs that can supply up to 400mA of current for powering other peripheral devices.

For custom applications, the CHR-6dm makes 6 extra GPIO pins available, which can be configured as digital inputs and outputs and as a TTL UART and an SPI bus. The extra pins can be used for motor control, GPS integration, pressure sensor integration, and communication with other digital devices, among other things. Free development tools are available, and our firmware is available as a reference.

Summary of Features
- Onboard EKF produces yaw, pitch, and roll angle estimates
- Automatic gyro bias calibration
- Soft and hard iron calibration
- Cross-axis misalignment correction[1]
- Adjustable output rates (20 Hz - 300 Hz)
- Onboard 3.3V regulator simplifies integration
- +3.3V output sources up to 400mA for powering other peripherals (ie. GPS)
- Open-source firmware with free development tools
- Open-source PC software for data visualization and AHRS configuration
- Two UARTs and one SPI bus routed out for custom applications

---

[1] Soft and hard iron calibration matrices and cross-axis alignment matrices must be determined and set by the end user. The CHR-6dm configuration software includes routines for computing soft and hard iron calibration data.

CHR-6dm Attitude and Heading Reference System
Product datasheet - Rev. 1.0, Preliminary

Common Applications
- Robotics
- Platform Stabilization
- Motion Tracking
- Enhanced GPS Navigation
- General Motion Sensing
- Image Stabilization

CH Robotics

CHR-6dm Attitude and Heading Reference System
Product datasheet - Rev. 1.0, Preliminary

1. Revision History

Rev. 1.1 - Initial Release

2. Absolute Maximum Ratings

Table 1 - CHR-6dm Absolute Maximum Ratings

Symbol	Ratings	Maximum Value	Unit
Vdd	Supply voltage	-0.3 to +12V	V
Vin	Input voltage on any digital IO pin	-0.3 to 5.1 V	V
A	Acceleration	3000 g for .5 ms	
		10000 g for .1 ms	
Top	Operating temperature range	-40 to +85	˚C
Tstg	Storage temperature range	-40 to +125	˚C

3. Electrical Characteristics

Symbol	Parameter	Test condition	Min.	Typ.	Max.	Unit

Vdd	Supply voltage		3.3	3.3 - 9	12	V
Idd	Supply current		50	52	58	mA

Table 2 - Gyro Electrical Characteristics

Symbol	Parameter	Test Condition	Min.	Typ.	Max.	Unit
FSA	Measurement	4x OUT (amplified)		+/- 100		°/s
FS	Range	OUT (not amplified)		+/- 400		°/s
SoA	Sensitivity	4x OUT (amplified)		10		mV/°/s
So		OUT (not amplified)		2.5		mV/°/s
SoDr	Sensitivity change vs temperature	Delta from 25 °C		.03		%/°/s
Voff	Zero-rate level			1.23		V
Vref	Reference Voltage			1.23		V
OffDr	Zero-rate level change vs. temperature	Delta from 25 °C		.02		°/s/°C
NL	Non linearity	Best fit straight line		+/- 1		% FS
BW	Bandwidth			140		Hz
Rn	Rate noise density			0.017		°/s/rt Hz

Table 3 - Accelerometer Electrical Characteristics

Symbol	Parameter	Test Condition	Min.	Typ.	Max.	Unit
aFS	Measurement Range		+/- 3	+/- 3.6		g
SoXYZ	Sensitivity	Vdd = 3 V	270	300	330	mV/g
Stemp	Sensitivity change due to temperature	Vs = 3 V		0.01		%/°C
nXY	X,Y noise density			150		ug/rt Hz
nZ	Z noise density			300		ug/rt Hz
offXY	X,Y 0 g voltage	Vs = 3 V	1.35	1.5	1.65	V
offZ	Z 0 g voltage	Vs = 3 V	1.2	1.5	1.8	V
aNL	Non linearity			+- .03		% FS
aBW		With external filter		140		Hz

4. Pin Configuration and Functional Descriptions

Top View (note: drawing is not to scale)

CHR-6dm Attitude and Heading Reference System
Product datasheet - Rev. 1.0, Preliminary
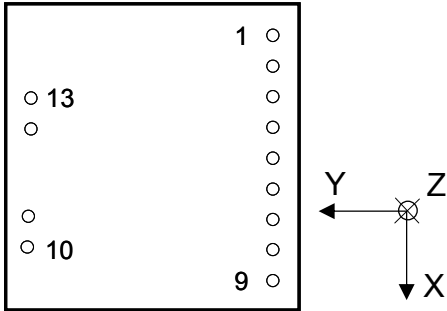
Table 4 - Pin Descriptions

Pin Description		
Pin #	Pin Name	Description
1	TX	TX (output)
2	RX	RX (input)
3	PGM	Bootloader activation pin. Pull high to activate bootloader - leave disconnected for normal operation.
4	D1	GPIO pin 1 (TX2)
5	D2	GPIO pin 2 (RX2)
6	D3	GPIO pin 3 (MOSI)
7	D4	GPIO pin 4 (MISO)
8	D5	GPIO pin 5 (SCK)
9	D6	GPIO pin 6 (SS)
10	Vdd	Input voltage (3.3 - 12 V)
11	GND	Ground
12	GND	Ground
13	+3.3V	+3.3V output.

5. Mechanical Drawing

Top View (Not to Scale)
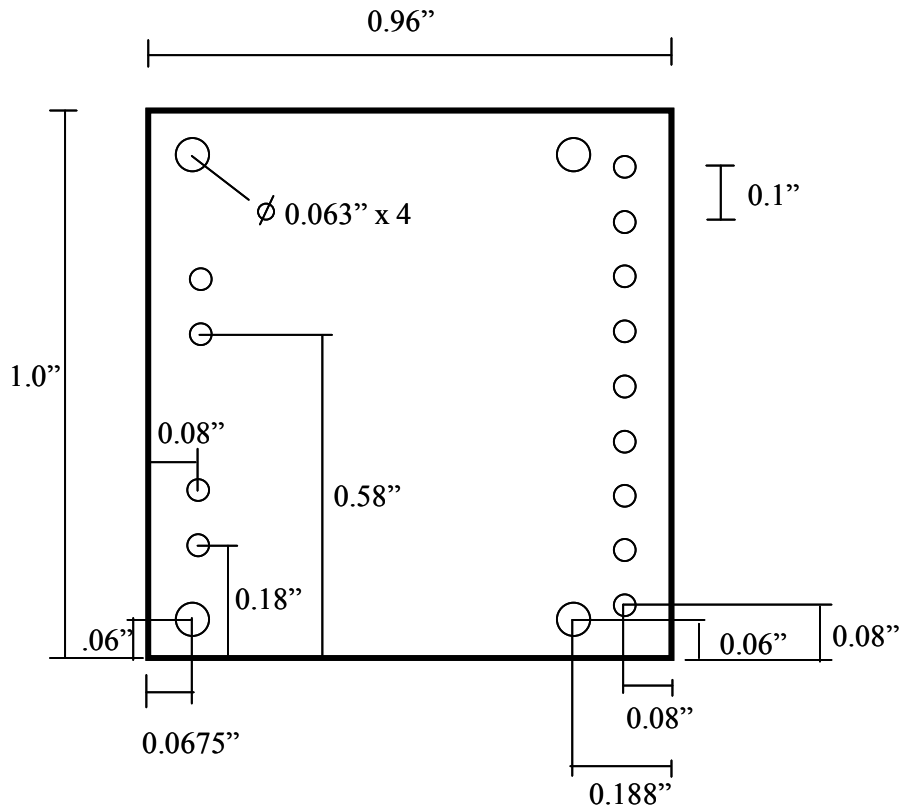
6. Basic Operation

The CHR-6dm is factory-configured to broadcast angles and angular rates at 200 Hz over a TTL UART at 115200 baud. The serial protocol uses 8 data bits, 1 stop bit, and no parity. The UART logic level is 3.3V, but the pins are 5V tolerant for simple integration with +5V systems.

Data from the AHRS is transmitted in a specific packet structure outlined in Section 7, "Communication with the CHR-6dm." On first power-up, the CHR-6dm will begin transmitting SENSOR_DATA packets at the aforementioned rate. See Section 7.2.9 for details about the SENSOR_DATA packet.

By default, the CHR-6dm only transmits yaw, pitch, and roll angles, and angular rates. However, the AHRS can also be configured to send sensor data from any set of sensors by sending a the sensor a SET_ACTIVE_CHANNELS packet. Only data from "active" channels will be transmitted by the AHRS.

Angle estimates and sensor measurements are returned as 16-bit 2's complement integers. To obtain actual angles and sensor measurements, the data returned by the AHRS should be multiplied by a scale factor. Scale factors for each channel are given below

Table 5 - Data output scale factors

Channel	Scale Factor	Units
Yaw	0.0109863	°/LSB
Pitch	0.0109863	°/LSB
Roll	0.0109863	°/LSB
Yaw rate	0.0137329	°/s/LSB
Pitch rate	0.0137329	°/s/LSB
Roll rate	0.0137329	°/s/LSB
Mag x	0.061035	mGauss/LSB
Mag y	0.061035	mGauss/LSB
Mag z	0.061035	mGauss/LSB
Gyro x	0.01812	°/s/LSB
Gyro y	0.01812	°/s/LSB
Gyro z	0.01812	°/s/LSB
Accel x	0.106812	mg/LSB
Accel y	0.106812	mg/LSB
Accel z	0.106812	mg/LSB

The CHR-6dm AHRS operates in one of two modes: in Broadcast Mode, the AHRS automatically transmits sensor data at a user-configurable frequency between 20 Hz and 300 Hz. In Silent Mode, the AHRS only transmits data when a GET_DATA packet is received over the UART. The CHR-6dm is set to Broadcast Mode by default. The default transmission frequency is 200 Hz.

Regardless of the broadcast frequency, the AHRS output registers are updated internally at a rate of 500 Hz. When the AHRS transmits sensor data and states, the most recent samples are sent. Sensor data is always sent in a SENSOR_DATA packet.

Broadcast Mode is enabled by sending a SET_BROADCAST_MODE packet to the AHRS. The SET_BROADCAST_MODE packet also configures the broadcast frequency. Silent Mode is enabled by sending a SET_SILENT_MODE packet to the AHRS.

All changes made to the AHRS configuration are stored in RAM. To make the configuration persistent, a WRITE_TO_FLASH packet should be sent to the AHRS.

6.1.   Sampling, Filtering, and Angle Estimation

The CHR-6dm oversamples and decimates the ADC data on all channels to reduce quantization noise and increase the effective ADC resolution to 16 bits. Decimated sensor data is then passed to the EKF for angle estimation.

6.1.1.   EKF Assumptions and Limitations

The CHR-6dm uses a continuous-discrete implementation of the EKF, where rate gyro measurements are used to drive the prediction step, and the accelerometer and magnetometer are used for drift correction in the update step. Two major assumptions are made:

1. On average, the accelerometers measure the gravity vector. The EKF "knows" which way is down by observing the direction of the gravity vector. There are a number of cases where this assumption breaks down, particularly in aircraft. When an aircraft makes a coordinated turn, the measured acceleration will always point in the direction of the bottom of the fuselage. Over a long turn, the AHRS pitch and roll angle estimates will therefore tend to zero. To correct this problem, additional sensors must be used for centripetal compensation. The CHR-6dm has no native support for centripetal compensation.

2. Distortion of the magnetic field measurement is predictable. Distortion from objects that rotate and translate with the AHRS can be corrected, but other distortions (from power lines, ferrous metal objects, etc.) will cause erroneous angle measurements.

As long as the two preceding assumptions are valid, the AHRS can provide better than .5 degree accuracy in pitch and roll, and 1 degree accuracy in yaw under static conditions.

All AHRS angle measurements are made with respect to a North-East-Down (NED) inertial frame. The inertial frame x-axis is aligned with magnetic north, the y-axis is aligned with magnetic east, and the z-axis points down toward the center of the Earth. The "pitch" angle represents positive rotation about the x-axis, "roll" represents positive rotation about the y-axis, and "yaw" represents positive rotation about the z-axis.

The sequence of rotations used to give the orientation of the AHRS is first yaw, then pitch, then roll.

Note that since Euler angles (yaw, pitch, roll) are used to represent the orientation of the AHRS, there is a singularity at pitch = +/-90 degrees. That is, when the pitch angle reaches +/-90 degrees, there is more than one way to represent the AHRS orientation, and the angle estimate may no longer be valid. **The CHR-6dm therefore cannot be used to measure orientation if it is expected to approach a pitch angle of +/-90 degrees.** To fix this problem, a quaternion representation must be used instead of Euler Angles. At this time, the CHR-6dm does not support quaternions.

In the event that the CHR-6dm encounters a singularity, the angle estimate and covariance can be reset by sending the AHRS an EKF_RESET packet.

In some cases, it may be necessary to ignore data received by the accelerometers or by the magnetometer. For example, if the AHRS is to be used in an environment where the magnetic field readings aren't reliable (ie. indoors or next to an RF transmitter), then magnetometer inputs to the EKF can be disabled. Accelerometer inputs can be similarly disabled, and the EKF can be turned off entirely if desired. The EKF is configured by sending a SET_EKF_CONFIG packet to the AHRS.

6.1.2.   EKF tuning

EKF performance can be tuned by adjusting the process noise covariance matrix and the measurement noise covariance matrices (there are two measurement noise matrices - one for the accels, and one for the magnetometer). Generally the measurement noise matrices should remain fixed at the factory default, and the process noise matrix should be adjusted. Measurement noise matrices can be adjusted by sending SET_MAG_MEASUREMENT_NOISE and SET_ACCEL_MEASUREMENT_NOISE packets. The process noise matrix can be adjusted by sending a SET_PROCESS_NOISE packet.

Loosely speaking, the process noise matrix is used to specify how much the EKF "trusts" data from the gyros with respect to data from the magnetic sensors and accelerometers. The lower the values along the diagonal of the matrix, the more the rate gyros are trusted. Conversely, if the diagonal elements are large, the gyros are trusted less and the accels and magnetometers are weighted more heavily.

Trusting gyros more heavily increases the effect of time-varying gyro output biases, essentially increasing DC errors in the angle estimates. Trusting accels and magnetic sensors more heavily increases AHRS sensitivity to vibrations and distorted magnetic field measurements.

Optimal selection of the process noise matrix depends on the specific environment in which the AHRS will be operating. If the AHRS will be subjected to large accelerations that interfere with gravity vector measurement, then the diagonal terms of the process noise matrix should be small. On a more stationary platform, on the other hand, the accels can be trusted more. Increase the diagonal terms of the process noise matrix to reduce bias errors.

6.2. Calibration

The CHR-6dm comes ready to report angles immediately, but in precision applications it is often beneficial to perform extra calibration to improve accuracy. There are a variety of calibration procedures that can be performed ranging from very simple to more complex. Possible calibration options are:

1. Rate gyro bias calibration
2. Rate gyro scale factor calibration
3. Accelerometer bias calibration
4. Magnetometer soft and hard iron calibration
5. Accelerometer and magnetometer reference vectors
6. Accelerometer and rate gyro cross-axis misalignment correction

Details about each calibration procedure are given below.

6.2.1. Gyro and Accelerometer Biases

The zero-rate output the rate gyros and the zero-acceleration output of the accelerometers is device-dependent, and the gyro biases vary with temperature. The 16-bit signed value returned by the AHRS for each axis is given by

data = ADC_data - bias

where *ADC_data* is the unsigned filtered ADC data, and *bias* is a configurable 16-bit signed value. By default, *bias* is set at 0 for both the rate gyros and the accelerometers. While the default biases may be sufficient for the accelerometers, the biases for the rate gyros should generally be reconfigured at run-time.

The bias for the x, y, and z gyro axes can be configured using the SET_ GYRO_BIAS packet. The AHRS also includes an auto-zero command that samples the rate gyro channels and automatically sets the bias to zero the output data. Automatic gyro calibration is triggered by sending a

ZERO_RATE_GYROS packet to the AHRS. During automatic calibration, which takes approximately three seconds, the AHRS should be stationary.

By default, the rate gyros are re-calibrated automatically on AHRS startup. Automatic startup calibration can be disabled and re-enabled by sending a SET_START_CAL packet to the AHRS.

The bias for the accelerometers is set using the SET_ ACCEL_BIAS packet.

6.2.2.	Magnetometer Hard and Soft Iron Calibration

Metal and magnetic objects near the AHRS distort magnetometer measurements, creating significant errors in estimated angles. Distortions from objects that are not in a fixed position relative to the AHRS cannot generally be corrected. On the other hand, distortions from objects that are in a fixed position with respect to the AHRS can be detected and corrected. For example, if the AHRS is mounted to a platform using steel screws, the magnetic field will be distorted; but, since the screws rotate and translate with the AHRS, the distortions can be corrected. In contrast, if a screwdriver were placed close to the AHRS, then the resulting distortions could not be corrected because the screwdriver does not move with the sensor.

For magnetometer soft and hard iron calibration the common procedure is to mount the AHRS in its final configuration and then perform calibration.

Magnetic field inputs to the EKF are first multiplied by a field correction matrix which can be set by sending a SET_MAG_CAL packet to the AHRS. By default, the correction matrix is the identity (ie. no calibration is performed). The CHR-6dm configuration utility can be used to determine the field correction matrix and write it to the AHRS.

6.2.3.	Gyro Scale Factor Calibration

For improved accuracy, the internal rate gyro scale factor can be changed by the user. The internal rate gyro scale factor is different from the scale factor described in Section 6. The scale factor described in Section 6 is used by the end user to convert the 16-bit signed integer output of the AHRS to an actual angular rate. The scale factor mentioned here is used internally to convert raw sensor data to a floating point angular rate used by the EKF. This rate is later converted to a 16-bit signed integer for transmission from the AHRS.

Internal gyro rate measurements are given by

$$rate\_output = scale\_factor*(ADC\_data - \text{bias})$$

in degrees per second. The CHR-6dm comes with a preset scale factor that is close to the true value. While the preset scale factor is accurate enough for most applications, better dynamic angle accuracy can be obtained by calibrating the device.

To calibrate the scale factor, a user would rotate the AHRS at a known rate (using a turn-table with an encoder or hall effect sensor for rate measurement) and measure the rate gyro outputs. The scale factor is given simply by

$$scale\_factor = actual\_rate \,/\, gyro\_output$$

where 'actual_rate' is given in degrees per second. The scale factor for each gyro axis can be set by sending a SET_X_GYRO_SCALE, SET_Y_GYRO_SCALE, or SET_Z_GYRO_SCALE packet. To revert to factory settings, a RESET_TO_FACTORY packet should be sent.

6.2.4. Gyro Cross-Axis Alignment Calibration

The CHR-6dm measures rotation rates about three axes. Under ideal circumstances, these three axes would be perfectly orthogonal, so that rotation about a single AHRS axis would only produce outputs on the rate gyro for that axis. Due to manufacturing tolerances, however, the axes are never perfectly aligned. While cross-axis misalignment is often small enough that it can be ignored, precision applications may require that misalignment be corrected.

Cross-axis misalignment correction can be understood in terms of a change of basis from a non-orthogonal basis formed by the axes of the rate gyros to an orthogonal basis formed by the body frame of the sensor (see Section 6.1 for details about coordinate frames used by the AHRS). We will say that the basis vectors for the body coordinate frame are given by the elementary basis vectors $\{e_1 \quad e_2 \quad e_3\}$. The basis vectors for the non-orthogonal rate gyro axes, expressed in terms of the elementary basis vectors, are given by $\{v_x \quad v_y \quad v_z\}$.

The rotation rates measured by the rate gyros are expressed in terms of the basis $\{v_x \quad v_y \quad v_z\}$. Let the vector **x** is be composed of the *x*, *y*, and *z* rotation rates measured by the rate gyros at any given time. Then the rotation rates expressed in the body frame are given by

$$\mathbf{x}_e = \begin{bmatrix} v_x & v_y & v_z \end{bmatrix} \mathbf{x} = T\mathbf{x}$$

This is simply a change of basis operation. By default, *T* is the identity matrix (ie. no cross-axis calibration is performed). If cross-axis calibration is to be performed, *T* must be determined experimentally. The matrix *T* can then be set be the user by sending a SET_GYRO_ALIGNMENT packet to the AHRS.

To determine *T* experimentally, the AHRS should be mounted on the center of a turn-table that can rotate the AHRS at a constant rate about any body frame axis. The exact rate is not important. It is important, however, that the AHRS is mounted so that when the turn-table is spinning, there is only rotation about one body frame axis at a time. The mechanical fixture used to mount the AHRS must therefore have tight tolerances.

By rotating the AHRS about one body frame axis and measuring the response of all the rate gyros, it is possible to determine the vectors needed to perform the change of basis. Let $\hat{v}_x$ be the measured response of the rate gyros to rotation about the body frame x-axis. Let $\hat{v}_y$ and $\hat{v}_z$ be similarly defined. Then the matrix *T* is given by

$$T = \begin{bmatrix} \hat{v}_x & \hat{v}_y & \hat{v}_z \end{bmatrix}^{-1}.$$

To prevent unwanted scaling, each column vector in T should be normalized.

6.2.5.	Accel Cross-Axis Alignment Calibration

The accelerometers on the CHR-6dm would ideally be perfectly aligned to the body frame of the device. Much like the onboard gyros, however, manufacturing tolerances cause small misalignments in the accelerometer axes with respect to the body frame. While cross-axis misalignment is often small enough that it can be ignored, precision applications may require that misalignment be corrected.

Cross-axis calibration of the accelerometers is performed in the same fashion as rate gyro cross-axis calibration, except that the AHRS should be aligned with the gravity vector for each axis instead of being rotated about the body frame axes.

The accelerometer cross-axis alignment matrix can be set by sending a SET_ACCEL_ALIGNMENT packet to the AHRS.

6.2.6.	Magnetometer and Accelerometer Reference Vectors

The AHRS uses two reference vectors to determine its orientation. The accelerometer reference vector corresponds to the measured gravity vector when pitch and roll angles are zero. The magnetometer reference vector represents the measured magnetic field of the Earth when yaw, pitch, and roll angles are zero. When the accelerometer and magnetometer reference vectors match the output of the sensors, the AHRS is oriented with zero yaw, pitch, and roll.

By default, the accelerometer reference vector corresponds to a zero reading on the X and Y accel axes, and the acceleration of gravity on the Z axis. Thus, if the sensor is perfectly "flat" with respect to the Earth, pitch and roll angles will remain at zero. In some cases, the AHRS will not be mounted perfectly level. The accelerometer reference vector can be reset so that the current orientation of the AHRS corresponds to zero pitch and roll. This is accomplished by sending the AHRS an AUTO_SET_ACCEL_REF packet. When the packet has been received, the AHRS will take the current accelerometer readings and use them as the reference vector. The accel reference vector can also be set manually by sending a SET_ACCEL_REF_VECTOR packet to the AHRS.

By default, the magnetometer reference vector is set as the magnetic field measurement when yaw, pitch, and roll angles are zero (ie. the x-axis of the sensor is aligned with magnetic north). Since the magnetic field of the Earth changes depending on location, the magnetometer reference vector should be reset to correspond to its location. This can be accomplished by orienting the AHRS so that it is pointed magnetic north (y-axis reading should be zero) and sending an AUTO_SET_MAG_REF packet to the unit. Alternatively, the reference vector can be set manually be sending a SET_MAG_REF_VECTOR packet to the AHRS.

6.3. Self-Test

The CHR-6dm includes a self-test feature that checks rate gyro and accelerometer channels for proper functionality. The self-test sequence is started by sending a SELF_TEST packet to the

AHRS. On completion, a STATUS_REPORT packet is returned to report whether each sensor channel passed the self-test.

7. Communication with the CHR-6dm

The CHR-6dm communicates over a logic-level UART at 115200 baud, 8 data bits, 1 stop bit, and no parity. While the logic level is 3.3 V, the IO pins are 5V tolerant for simple integration with 5V systems.

7.1.   CHR-6dm RX Packets

RX Packets are packets that can be received by the CHR-6dm. Each packet received by the AHRS must begin with the three byte sequence "snp" to signal the beginning of a new packet. The fourth byte is Packet Type indicator (PT), which identifies the packet being received. The fifth byte, N, is the number of bytes contained in the data section of the packet. The N-byte data section immediately follows data length byte, and the packet is finally ended with a two byte checksum, which contains the sum of all previous bytes in the packet.

If the Packet Type indicator is unrecognized, the AHRS will transmit an UNRECOGNIZED_COMMAND packet.

RX Packet Structure

Function	's'	'n'	'p'	PT	N	d_1	…	d_N	CHK	
Byte	1	2	3	4	5	6	…	N+5	N+6	N+7

RX Packet Description

1 - 3	Each received packet must begin with the three-byte (character) sequence "snp" to signal the beginning of a new packet.
4	**PT** specifies the packet type.
5	**N** specifies the number of data bytes to expect.
6 - (N+5)	d_1 through d_N contain the **N** data bytes in the packet.
(N+6) - (N+7)	**CHK** is a two-byte checksum.

The total size of a packet with **N** data bytes is (**N** + 7) bytes. After the CHR6-dm receives a full packet, it checks to ensure that the checksum given in the last two bytes matches the sum of all preceding bytes in the packet. If the checksum is invalid, a BAD_CHECKSUM packet is transmitted in response.

7.1.1.   CHR-6dm RX Packet Overview

PT	Packet Name	Description
0x80	SET_ACTIVE_CHANNELS	Specifies which channel data should be transmitted over the UART.
0x81	SET_SILENT_MODE	Enables "Silent Mode." In Silent Mode, the AHRS only reports data when a GET_DATA packet is

CHR-6dm Attitude and Heading Reference System
Product datasheet - Rev. 1.0, Preliminary

		received.
0x82	SET_BROADCAST_MODE	Enables "Broadcast Mode." In Broadcast Mode, the AHRS automatically transmits sensor data every **Ts** milliseconds, where **Ts** is encoded in the data section of the SET_BROADCAST_MODE packet.
0x83	SET_GYRO_BIAS	Manually sets the x-axis rate gyro bias term. The bias term can be automatically set for all gyro axes by sending a ZERO_RATE_GYROS packet.
0x84	SET_ACCEL_BIAS	Manually sets the x-axis accelerometer bias term.
0x85	SET_ACCEL_REF_VECTOR	Manually sets the accelerometer reference vector used by the EKF to determine "which way is down"
0x86	AUTO_SET_ACCEL_REF	Causes the AHRS to set the current accelerometer measurements as the reference vector (sets pitch and roll angles to zero for the given orientation).
0x87	ZERO_RATE_GYROS	Starts internal self-calibration of all three rate gyro axes. By default, rate gyros are zeroed on AHRS startup, but gyro startup calibration can be disabled (or re-enabled) by sending a SET_START_CAL packet.
0x88	SELF_TEST	Instructs the AHRS to perform a self-test of all sensor channels. A STATUS_REPORT packet is transmitted after the self-test is complete.
0x89	SET_START_CAL	Enables or disables automatic startup calibration of rate gyro biases.
0x8A	SET_PROCESS_COVARIANCE	Sets the 3x3 matrix representing the covariance of the process noise used in the prediction step of the EKF.
0X8B	SET_MAG_COVARIANCE	Sets the 3x3 matrix representing the covariance of the measurement noise used in the magnetometer update step of the EKF
0x8C	SET_ACCEL_COVARIANCE	Sets the 3x3 matrix representing the covariance of the measurement noise used in the accelerometer update step of the EKF
0x8D	SET_EKF_CONFIG	Sets the EKF_CONFIG register. Can be used to enable/disable the EKF, or to enable/disable accelerometer and magnetometer updates.
0x8E	SET_GYRO_ALIGNMENT	Sets the 3x3 matrix used to correct rate gyro cross-axis misalignment.
0x8F	SET_ACCEL_ALIGNMENT	Sets the 3x3 matrix used to correct accelerometer cross-axis misalignment
0x90	SET_MAG_REF_VECTOR	Sets the reference vector representing the expected output of the magnetometer when yaw, pitch, and roll angles are zero.
0x91	AUTO_SET_MAG_REF	Sets the current magnetometer output as the reference vector.
0x92	SET_MAG_CAL	Sets the 3x3 magnetic field distortion correction matrix to compensate for soft-iron distortions, axis misalignment, and sensor scale inconsistencies.
0x93	SET_MAG_BIAS	Sets the magnetic field measurement bias to

		compensate for hard iron distortions
0x94	SET_GYRO_SCALE	Sets the scale factors used to convert raw ADC data to rates for angle estimation on the EKF.
0x95	EKF_RESET	Sets all terms in the current state covariance matrix to zero and re-initializes angle estimates.
0x96	RESET_TO_FACTORY	Resets all AHRS setting to factory default values
0xA0	WRITE_TO_FLASH	Writes current AHRS configuration to on-board flash, so that the configuration persists when the power is cycled.
0x01	GET_DATA	In Listen Mode, causes the AHRS to transmit data from all active channels in a SENSOR_DATA packet.
0x02	GET_ACTIVE_CHANNELS	Reports which channels are "active" in an ACTIVE_CHANNELS_REPORT packet. Active channels are sensor channels that are measured and transmitted in response to a GET_DATA packet, or periodically in Broadcast Mode.
0x03	GET_BROADCAST_MODE	Returns the BROADCAST_MODE_REPORT packet, which specifies whether the AHRS is in Broadcast Mode or Silent Mode.
0x04	GET_ACCEL_BIAS	Return the bias values for all three accel axes in a ACCEL_BIAS_REPORT packet.
0x05	GET_ACCEL_REF_VECTOR	Returns the accelerometer reference vector in an ACCEL_REF_VECTOR_REPORT packet
0x06	GET_GYRO_BIAS	Returns the bias values for all three rate gyros in a GYRO_BIAS_REPORT packet.
0x07	GET_GYRO_SCALE	Returns the rate gyro scale factors used internally to convert raw rate gyro data to actual rates in a GYRO_SCALE_REPORT packet.
0x08	GET_START_CAL	Reports whether the AHRS is configured to calibrate rate gyro biases automatically on startup in a START_CAL_REPORT packet.
0x09	GET_EKF_CONFIG	Returns the one byte EKF configuration register in an EKF_CONFIG_REPORT packet.
0X0A	GET_ACCEL_COVARIANCE	Returns the variance of the accelerometer measurements used by the EKF update step
0x0B	GET_MAG_COVARIANCE	Returns the variance of the magnetometer measurements used by the EKF update step
0x0C	GET_PROCESS_COVARIANCE	Returns the variance of the EKF prediction step
0x0D	GET_STATE_COVARIANCE	Returns the 3x3 matrix representing the covariance of the current EKF state estimates in a STATE_COVARIANCE_REPORT packet.
0x0E	GET_GYRO_ALIGNMENT	Returns the 3x3 matrix used to correct rate gyro cross-axis misalignment
0x0F	GET_ACCEL_ALIGNMENT	Returns the 3x3 matrix used to correct accelerometer cross-axis misalignment
0x10	GET_MAG_REF_VECTOR	Returns the magnetometer reference vector in a MAG_REF_VECTOR_REPORT packet

| 0x11 | GET_MAG_CAL | Returns the 3x3 magnetometer correction matrix in an MAG_CAL_REPORT packet |
| 0x12 | GET_MAG_BIAS | Returns the magnetometer bias correction used by the EKF. The bias is reported in a MAG_BIAS_REPORT packet. |

7.1.2.   SET_ACTIVE_CHANNELS

Description

Specifies which channel data should be transmitted over the UART in response to a GET_DATA packet, or periodically in Broadcast Mode. Any combination of sensor channels can be set as active or inactive.

Packet Definition

PT = 0x80
N = 2

D1								D2							
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
yaw	pitch	roll	yaw rate	pitch rate	roll rate	mx	my	mz	gx	gy	gz	ax	ay	az	0

If the bit corresponding to a sensor channel is set (i.e. gz = 1), then the channel is active. If the bit is cleared (i.e. gz = 0), then the channel is inactive, and data from that channel will not be transmitted.

yaw = yaw angle estimate
pitch = pitch angle estimate
roll = roll angle estimate
yaw rate = yaw rate estimate
pitch rate = pitch rate estimate
roll rate = roll rate estimate
mx = magnetometer x-axis data
my = magnetometer y-axis data
mz = magnetometer z-axis data
gx = gyro x-axis data
gy = gyro y-axis data
gz = gyro z-axis data
ax = accel x-axis data
ay = accel y-axis data
az = accel z-axis data

7.1.3.   SET_SILENT_MODE

Description

Enables "Silent Mode." In Silent Mode, the AHRS only reports data when a GET_DATA packet is received.

Packet Definition

PT = 0x81
N = 0

The SET_SILENT_MODE packet includes no data bytes (the data length byte is followed immediately by the checksum).

7.1.4.   SET_BROADCAST_MODE

Description

Enables "Broadcast Mode." In Broadcast Mode, the AHRS automatically transmits sensor data at regular time intervals.

Packet Definition

PT = 0x82
N = 1

D1							
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
x							

The broadcast frequency is given by
$$f = ((280/255)*x + 20) \text{ Hz}$$
where f is the broadcast frequency, and x is the data byte passed in the SET_BROADCAST_MODE packet. The broadcast frequency thus ranges between 300 Hz ($x = 255$) and 20 Hz ($x = 0$).

7.1.5.   SET_GYRO_BIAS

Description

Manually sets the rate gyro bias on the X, Y, and Z rate gyros. The bias can be automatically set for all gyro axes by sending a ZERO_RATE_GYROS packet. The biases must be 16-bit 2's complement integers.

Packet Definition

PT = 0x83
N = 6

D1	D2	D3	D4	D5	D6

Z Gyro Bias	Y Gyro Bias	X Gyro Bias

7.1.6. SET_ACCEL_BIAS

Description

Manually sets the accelerometer biases on the X, Y, and Z accelerometers. The biases must be 16-bit 2's complement integers.

Packet Definition

PT = 0x84
N = 6

D1	D2	D3	D4	D5	D6
Z Accel Bias		Y Accel Bias		X Accel Bias	

7.1.7. SET_ACCEL_REF_VECTOR

Description

Manually sets the accelerometer reference vector. The data in the SET_ACCEL_REF_VECTOR packet corresponds to the raw accelerometer measurements expected when the pitch and roll angles are zero. The data for each axis should be a 16-bit 2's complement integer.

Packet Definition

PT = 0x85
N = 6

D1	D2	D3	D4	D5	D6
Z ref		Y ref		X ref	

7.1.8. AUTO_SET_ACCEL_REF

Description

Sets the accel reference vector to the most recent data acquired by the accelerometers. Returns the new reference vector in a ACCEL_REF_VECTOR_REPORT packet.

Packet Definition

PT = 0x86
N = 0

The AUTO_SET_ACCEL_REF packet includes no data bytes (the data length byte is followed immediately by the checksum).

7.1.9.   ZERO_RATE_GYROS

Description

Starts internal self-calibration of all three rate gyro axes. By default, rate gyros are zeroed on AHRS startup, but gyro startup calibration can be disabled (or re-enabled) by sending a SET_START_CAL packet

Packet Definition

PT = 0x87
N = 0

There are no data bytes associated with the ZERO_RATE_GYROS packet. The checksum immediately follows the data length byte.

7.1.10.   SELF_TEST

Description

Instructs the AHRS to perform a self-test of the accelerometer and gyro sensor channels. The self-test sequence takes approximately 570 milliseconds to complete. During this time, the AHRS should be kept stationary. A STATUS_REPORT packet is transmitted after the self-test is complete.

Packet Definition

PT = 0x88
N = 0

There are no data bytes associated with the SELF_TEST packet. The checksum immediately follows the data length byte.

7.1.11.   SET_START_CAL

Description

Enables or disables automatic startup calibration of rate gyros.

Packet Definition

PT = 0x89
N = 1

D1							
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
NA	NA	NA	NA	NA	NA	NA	en

If 'en' = 1, then startup calibration will be enabled.

7.1.12.   SET_PROCESS_COVARIANCE

Description

Sets the process covariance to be used in the prediction step of the EKF. The unit assumes that the process covariance will be a diagonal matrix with equivalent diagonal entries. The SET_PROCESS_COVARIANCE packet thus includes only one 32-bit floating point value for the covariance matrix.

Packet Definition

PT = 0x8A
N = 4

D1	D2	D3	D4
32-bit IEEE floating point value			

7.1.13.   SET_MAG_COVARIANCE

Description

Sets the covariance to be used in the magnetometer update step of the EKF. The unit assumes that the magnetometer covariance will be a diagonal matrix with equivalent diagonal entries. The SET_MAG_COVARIANCE packet thus includes only one 32-bit floating point value for the covariance matrix.

Packet Definition

PT = 0x8B
N = 4

D1	D2	D3	D4
32-bit IEEE floating point value			

7.1.14.   SET_ACCEL_COVARIANCE

Description

Sets the covariance to be used in the accelerometer update step of the EKF. The unit assumes that the accelerometer covariance will be a diagonal matrix with equivalent diagonal entries. The

SET_ACCEL_COVARIANCE packet thus includes only one 32-bit floating point value for the covariance matrix.

Packet Definition

PT = 0x8C
N = 4

D1	D2	D3	D4

32-bit IEEE floating point value

7.1.15.   SET_EKF_CONFIG

Description

Sets the EKF configuration register. This packet is used to enable/disable accelerometer and magnetometer updates to the angle estimates.

Packet Definition

PT = 0x8D
N = 1

D1							
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
NA	NA	NA	NA	NA	NA	Accel_EN	Mag_EN

If Mag_EN = 1, then magnetometer updates are used for yaw angle correction. If 0, then magnetometer correction is disabled.
If Accel_EN = 1, then accelerometer updates are used for pitch and roll angle correction. If 0, then accelerometer correction is disabled.

7.1.16.   SET_GYRO_ALIGNMENT

Description

Sets the 3x3 calibration matrix used to correct cross-axis misalignment of the rate gyro outputs. Each element in the matrix is a 32-bit IEEE floating point value.

Packet Definition

PT = 0x8E
N = 36

D1-D4	D5-D8	D9-D12	D13-D16	D17-D20	D21-D24	D25-D28	D29-D32	D33-D36
mat[0,0]	mat[0,1]	mat[0,2]	mat[1,0]	mat[1,1]	mat[1,2]	mat[2,0]	mat[2,1]	mat[2,2]

mat[i,j] corresponds to the i-th row and j-th column of the correction matrix.

7.1.17. SET_ACCEL_ALIGNMENT

Description

Sets the 3x3 calibration matrix used to correct cross-axis misalignment of the acclerometer outputs. Each element in the matrix is a 32-bit IEEE floating point value.

Packet Definition

PT = 0x8F
N = 36

D1-D4	D5-D8	D9-D12	D13-D16	D17-D20	D21-D24	D25-D28	D29-D32	D33-D36
mat[0,0]	mat[0,1]	mat[0,2]	mat[1,0]	mat[1,1]	mat[1,2]	mat[2,0]	mat[2,1]	mat[2,2]

mat[i,j] corresponds to the i-th row and j-th column of the correction matrix.

7.1.18. SET_MAG_REF_VECTOR

Description

Manually sets the magnetometer reference vector. The data in the SET_MAG_REF_VECTOR packet corresponds to the raw magnetometer measurements expected when the pitch, roll, and yaw angles are zero. The data for each axis should be a 16-bit 2's complement integer.

Packet Definition

PT = 0x90
N = 6

D1	D2	D3	D4	D5	D6
Z ref		Y ref		X ref	

7.1.19. AUTO_SET_MAG_REF

Description

Sets the magnetic field reference vector to the most recent magnetic sensor measurement. Returns a MAG_REF_VECTOR_REPORT PACKET with the new reference vector.

Packet Definition

PT = 0x91
N = 0

The AUTO_SET_MAG_REF packet includes no data bytes (the data length byte is followed immediately by the checksum).

7.1.20.   SET_MAG_CAL

Description

Sets the 3x3 calibration matrix used for soft and hard iron calibration, axis misalignment calibration, and scale calibration of the magnetometer.

Packet Definition

PT = 0x92
N = 36

D1-D4	D5-D8	D9-D12	D13-D16	D17-D20	D21-D24	D25-D28	D29-D32	D33-D36
mat[0,0]	mat[0,1]	mat[0,2]	mat[1,0]	mat[1,1]	mat[1,2]	mat[2,0]	mat[2,1]	mat[2,2]

mat[i,j] corresponds to the i-th row and j-th column of the correction matrix.

7.1.21.   SET_MAG_BIAS

Description

Sets the magnetic field bias term to compensate for hard-iron distortions of the magnetic field. Each bias term is a 16-bit 2's complement integer.

Packet Definition

PT = 0x93
N = 6

D1	D2	D3	D4	D5	D6
Z mag bias		Y mag bias		X mag bias	

7.1.22.   SET_GYRO_SCALE

Description

Sets the scale factors used to compute rates from raw gyro data on all axes. Scale factors in the SET_GYRO_SCALE packet must be 32-bit IEEE floating-point values

Packet Definition

PT = 0x94

N = 12

D1-D4	D5-D8	D9-D12
Z gyro scale	Y gyro scale	X gyro scale

7.1.23.   EKF_RESET

Description

Sets each term in the state covariance matrix to zero and re-initializes the EKF. This command can be used for recovery if the state is corrupted by passing too close to the singularity at pitch = 90 degrees.

Packet Definition

PT = 0x95
N = 0

There are no data bytes in the EKF_RESET packet. The checksum immediately follows the data length byte.

7.1.24.   RESET_TO_FACTORY

Description

Resets all AHRS configuration to the factory defaults. This includes calibration parameters, biases, communication settings, etc. The factory defaults are written to RAM. To make them persistent, a WRITE_TO_FLASH packet must be sent following the RESET_TO_FACTORY command.

Packet Definition

PT = 0x96
N = 0

There are no data bytes in the EKF_RESET packet. The checksum immediately follows the data length byte.

7.1.25.   WRITE_TO_FLASH

Description

Writes AHRS configuration to on-board flash so that the configuration persists when the power is cycled.

Packet Definition

PT = 0xA0
N = 0

There are no data bytes associated with the WRITE_TO_FLASH packet. The checksum immediately follows the data length byte.

7.1.26.　　　GET_DATA

Description

In Silent Mode, the AHRS waits to receive a GET_DATA packet before transmitting sensor data. The most recent data from all active sensor channels is transmitted in response to a GET_DATA packet. In Broadcast Mode, a GET_DATA packet is ignored.

Packet Definition

PT = 0x01
N = 0

There are no data bytes associated with the GET_DATA packet. The checksum immediately follows the data length byte.

7.1.27.　　　GET_ACTIVE_CHANNELS

Description

Reports which channels are "active." Active channels are sensor channels that are measured and transmitted in response to a GET_DATA packet, or periodically in Broadcast Mode.   Active channels are reported in an ACTIVE_CHANNEL_REPORT packet.

Packet Definition

PT = 0x02
N = 0

There are no data bytes associated with the GET_ACTIVE_CHANNELS packet. The checksum immediately follows the data length byte.

7.1.28.　　　GET_BROADCAST_MODE

Description

Causes the AHRS to send a BROADCAST_MODE_REPORT packet, which specifies whether the AHRS is in Broadcast Mode or Silent Mode.

Packet Definition

PT = 0x03
N = 0

There are no data bytes associated with the GET_BROADCAST_MODE packet. The checksum immediately follows the data length byte.

7.1.29.   GET_ACCEL_BIAS

Description

Return the bias values for all three accel axes in an ACCEL_BIAS_REPORT packet.

Packet Definition

PT = 0x04
N = 0

There are no data bytes associated with the GET_ACCEL_BIAS packet. The checksum immediately follows the data length byte.

7.1.30.   GET_ACCEL_REF_VECTOR

Description

Return the accelerometer reference vector in a ACCEL_REF_VECTOR_REPORT.

Packet Definition

PT = 0x05
N = 0

There are no data bytes associated with the GET_ACCEL_REF_VECTOR packet. The checksum immediately follows the data length byte.

7.1.31.   GET_GYRO_BIAS

Description

Returns the bias values for all three rate gyros in a GYRO_BIAS_REPORT packet.

Packet Definition

PT = 0x06
N = 0

There are no data bytes associated with the GET_GYRO_BIAS packet. The checksum immediately follows the data length byte.

7.1.32.   GET_GYRO_SCALE

Description

Returns the scale factors used to convert raw gyro measurements to angular rates. Data is returned in a GYRO_SCALE_REPORT packet.

Packet Definition

PT = 0x07
N = 0

There are no data bytes associated with the GET_GYRO_SCALE packet. The checksum immediately follows the data length byte.

7.1.33.   GET_START_CAL

Description

Reports whether gyro startup calibration is enabled by sending a START_CAL_REPORT packet.

Packet Definition

PT = 0x08
N = 0

There are no data bytes associated with the GET_START_CAL packet. The checksum immediately follows the data length byte.

7.1.34.   GET_EKF_CONFIG

Description

Returns the value of the EKF configuration register in a EKF_CONFIG_REPORT packet.

Packet Definition

PT = 0x09
N = 0

There are no data bytes associated with the GET_EKF_CONFIG packet. The checksum immediately follows the data length byte.

7.1.35.   GET_ACCEL_COVARIANCE

Description

Returns the covariance to be used in the accelerometer update step of the EKF. Covariance is transmitted in a ACCEL_COVARIANCE_REPORT packet.

Packet Definition

PT = 0x0A
N = 0

There are no data bytes associated with the GET_ACCEL_COVARIANCE packet. The checksum immediately follows the data length byte.

7.1.36.   GET_MAG_COVARIANCE

Description

Returns the covariance to be used in the magnetometer update step of the EKF. Covariance is transmitted in a MAG_COVARIANCE_REPORT packet.

Packet Definition

PT = 0x0B
N = 0

There are no data bytes associated with the GET_MAG_COVARIANCE packet. The checksum immediately follows the data length byte.

7.1.37.   GET_PROCESS_COVARIANCE

Description

Returns the covariance to be used in the prediction step of the EKF. Covariance is transmitted in a PROCESS_COVARIANCE_REPORT packet.

Packet Definition

PT = 0x0C
N = 0

There are no data bytes associated with the GET_PROCESS_COVARIANCE packet. The checksum immediately follows the data length byte.

7.1.38. GET_STATE_COVARIANCE

Description

Returns the 3x3 covariance matrix of the current state estimates in the EKF. The covariance matrix is sent in a STATE_COVARIANCE_REPORT packet.

Packet Definition

PT = 0x0D
N = 0

There are no data bytes associated with the GET_STATE_COVARIANCE packet. The checksum immediately follows the data length byte.

7.1.39. GET_GYRO_ALIGNMENT

Description

Returns the 3x3 matrix used to correct gyro cross-axis misalignment. The alignment matrix is returned in a GYRO_ALIGNMENT_REPORT packet.

Packet Definition

PT = 0x0E
N = 0

There are no data bytes associated with the GET_GYRO_ALIGNMENT packet. The checksum immediately follows the data length byte.

7.1.40. GET_ACCEL_ALIGNMENT

Description

Returns the 3x3 matrix used to correct accelerometer cross-axis misalignment. The alignment matrix is returned in an ACCEL_ALIGNMENT_REPORT packet.

Packet Definition

PT = 0x0F
N = 0

There are no data bytes associated with the GET_ACCEL_ALIGNMENT packet. The checksum immediately follows the data length byte.

7.1.41.   GET_MAG_REF_VECTOR

Description

Returns the magnetic field reference vector in a MAG_REF_VECTOR_REPORT packet.

Packet Definition

PT = 0x10
N = 0

There are no data bytes associated with the GET_MAG_REF_VECTOR packet. The checksum immediately follows the data length byte.

7.1.42.   GET_MAG_CAL

Description

Returns the 3x3 matrix used to correct magnetometer soft iron distortions, axis misalignment, and scale factors. The calibration matrix is returned in a MAG_CAL_REPORT packet.

Packet Definition

PT = 0x11
N = 0

There are no data bytes associated with the GET_MAG_CAL packet. The checksum immediately follows the data length byte.

7.1.43.   GET_MAG_BIAS

Description

Returns the magnetometer biases used to correct hard iron distortions. Biases are returned in a MAG_BIAS_REPORT packet.

Packet Definition

PT = 0x12
N = 0

There are no data bytes associated with the GET_MAG_BIAS packet. The checksum immediately follows the data length byte.

CHR-6dm Attitude and Heading Reference System
Product datasheet - Rev. 1.0, Preliminary

7.2. CHR-6dm TX Packets

TX Packets are packets that can be transmitted by the AHRS. The structure of transmitted packets is exactly the same as the structure of RX packets.

TX Packet Structure

Function	's'	'n'	'p'	PT	N	d$_1$	…	d$_N$	CHK	
Byte	1	2	3	4	5	6	…	N+5	N+6	N+7

TX Packet Description

1 - 3	Each received packet must begin with the three-byte (character) sequence "snp" to signal the beginning of a new packet.
4	**PT** specifies the packet type.
5	**N** specifies the number of data bytes to expect.
6 - (N+5)	**d$_1$** through **d$_N$** contain the **N** data bytes in the packet.
(N+6) - (N+7)	**CHK** is a two-byte checksum.

The total size of a packet with **N** data bytes is (**N** + 7) bytes.

7.2.1. CHR-6dm TX Packet Overview

PT	Packet Name	Description
0xB0	COMMAND_COMPLETE	Transmitted by the AHRS upon successful completion of a command that does not require data to be returned.
0xB1	COMMAND_FAILED	Transmitted by the AHRS when a command received over the UART could not be executed.
0xB2	BAD_CHECKSUM	Transmitted by the AHRS when a received packet checksum does not match the sum of the other bytes in the packet.
0xB3	BAD_DATA_LENGTH	Transmitted by the AHRS when a received packet contained more or less data than expected for a given packet type.
0xB4	UNRECOGNIZED_PACKET	Transmitted if the AHRS receives an unrecognized packet.
0xB5	BUFFER_OVERFLOW	Transmitted by the AHRS when the internal receive buffer overflows before a full packet is received.
0xB6	STATUS_REPORT	Transmitted at the end of a self-test procedure, triggered by a SELF_TEST command.
0xB7	SENSOR_DATA	Sent in response to a GET_DATA packet, or sent automatically in Broadcast Mode.
0xB8	GYRO_BIAS_REPORT	Sent in response to the GET_GYRO_BIAS command.
0xB9	GYRO_SCALE_REPORT	Sent in response to a GET_GYRO_SCALE command.

0xBA	START_CAL_REPORT	Sent in response to a GET_START_CAL command.
0xBB	ACCEL_BIAS_REPORT	Sent in response to the GET_ACCEL_BIAS command.
0xBC	ACCEL_REF_VECTOR_REPORT	Sent in response to a GET_ACCEL_REF_VECTOR command, or sent after an AUTO_SET_ACCEL_REF command.
0xBD	ACTIVE_CHANNEL_REPORT	Sent in response to a GET_ACTIVE_CHANNELS command.
0xBE	ACCEL_COVARIANCE_REPORT	Sent in response to a GET_ACCEL_COVARIANCE command
0xBF	MAG_COVARIANCE_REPORT	Sent in response to a GET_MAG_COVARIANCE command.
0xC0	PROCESS_COVARIANCE_REPORT	Sent in response to a GET_PROCESS_COVARIANCE command
0xC1	STATE_COVARIANCE_REPORT	Sent in response to a GET_STATE_COVARIANCE command
0xC2	EKF_CONFIG_REPORT	Sent in response to a GET_EKF_CONFIG command.
0xC3	GYRO_ALIGNMENT_REPORT	Sent in response to a GET_GYRO_ALIGNMENT command.
0xC4	ACCEL_ALIGNMENT_REPORT	Sent in response to a GET_ACCEL_ALIGNMENT command.
0xC5	MAG_REF_VECTOR_REPORT	Sent in response to a GET_MAG_REF_VECTOR command
0xC6	MAG_CAL_REPORT	Sent in response to a GET_MAG_CAL command
0xC7	MAG_BIAS_REPORT	Sent in response to a GET_MAG_BIAS command
0XC8	BROADCAST_MODE_REPORT	Sent in response to a GET_BROADCAST_MODE command.

7.2.2.   COMMAND_COMPLETE

Description
Transmitted by the AHRS upon successful completion of a command that does not require data to be returned.

Packet Definition
PT = 0xB0
N = 1

D1							
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Packet Type (TP) identifier for executed command							

7.2.3.   COMMAND_FAILED

Description
Transmitted by the AHRS when a command received over the UART could not be executed.

Packet Definition
PT = 0xB1
N = 1

D1							
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Packet Type (TP) identifier for command that could not be executed							

7.2.4. BAD_CHECKSUM

Description
Transmitted by the AHRS when a received packet checksum does not match the sum of the other bytes in the packet.

Packet Definition
PT = 0xB2
N = 0

7.2.5. BAD_DATA_LENGTH

Description
Transmitted by the AHRS when a received packet contained more or less data than expected for a given packet type.

Packet Definition
PT = 0xB3
N = 1

D1							
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Packet Type (TP) identifier of the received packet							

7.2.6. UNRECOGNIZED_PACKET

Description
Transmitted if the AHRS receives an unrecognized packet

Packet Definition
PT = 0xB4
N = 1

D1							
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Packet Type (TP) identifier of unrecognized command							

7.2.7.	BUFFER_OVERFLOW

Description
Transmitted by the AHRS when the internal receive buffer overflows before a full packet is received.

Packet Definition
PT = 0xB5
N = 0

7.2.8.	STATUS_REPORT

Description
Transmitted at the end of a self-test procedure, triggered by a SELF_TEST command.

Packet Definition
PT = 0xB6
N = 1

D1							
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
NA	NA	gyro_z error	gyro_y error	gyro_x error	accel_z error	accel_y error	accel_x error

If the self-test failed for any of the channels, then the corresponding bit will be set (i.e. if the z-axis gyro fails the self-test, then bit 5 will be set, etc.)

7.2.9.	SENSOR_DATA

Description
Sent in response to a GET_DATA packet, or sent automatically in Broadcast Mode. Only active channels are included in the packet.

Packet Definition
PT = 0xB7
N = 2 + 2*(# of active channels)

The first two bytes following the PT byte indicate which channels are active. The actual sensor data is contained in the remaining data bytes.

If all channels are active, then data is given in the following order: { yaw, pitch, roll, yaw_rate, pitch_rate, roll_rate, mag_z, mag_y, mag_x, gyro_z, gyro_y, gyro_x, accel_z, accel_y, accel_x }. Data bytes D3 and D4 correspond to the yaw angle, D5 and D6 to the pitch angle, etc. Data is returned as 16-bit two's-complement integers.

When one or more channel is inactive, then the data is returned in the same order, but skipping the inactive channels. For example, if all magnetic field and rate gyro channels are disabled, then the data is given in the following order: { yaw, pitch, roll, accel_z, accel_y, accel_x }

D1								D2								D3	D4	D5	D6	...
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	1st		2nd		
yaw	pitch	roll	yaw rate	pitch rate	roll rate	mx	my	mz	gx	gy	gz	ax	ay	az	0	active channel		active channel		...

yaw = 1 if yaw angle is active, 0 otherwise
pitch = 1 if pitch angle is active, 0 otherwise
roll = 1 if roll angle is active, 0 otherwise
yaw_rate = 1 if yaw rate is active, otherwise
pitch_rate = 1 if pitch rate is active, otherwise
roll_rate = 1 if roll rate is active, otherwise
mx = 1 if x-axis magnetometer is active, 0 otherwise
my = 1 if y-axis magnetometer is active, 0 otherwise
mz = 1 if z-axis magnetometer is active, 0 otherwise
gz = 1 if the z-axis gyro is active, 0 otherwise
gy = 1 if the y-axis gyro is active, 0 otherwise
gx = 1 if the x-axis gyro is active, 0 otherwise
az = 1 if the z-axis accelerometer is active, 0 otherwise
ay = 1 if the y-axis accelerometer is active, 0 otherwise
ax = 1 if the x-axis accelerometer is active, 0 otherwise

Note that yaw, pitch, and roll rates are not equivalent to gyro rate outputs - gyros measure rotation rates in the body frame, while yaw, pitch, and roll rates are specified in the inertial frame (see Section 6 for more details).

7.2.10.   GYRO_BIAS_REPORT

Description
Sent in response to the GET_GYRO_BIAS command. Returns the gyro biases for the X, Y, and Z gyro axes as 16-bit 2's complement signed integers.

Packet Definition
PT = 0xB8
N = 6

D1	D2	D3	D4	D5	D6
gyro_z bias		gyro_y bias		gyro_x bias	

7.2.11.   GYRO_SCALE_REPORT

Description
Sent in response to the GET_GYRO_SCALE command. Returns the scale factors used to convert the X, Y, and Z gyro raw data to angular rates. Scale factors are returns as 32-bit IEEE floating-point values.

Packet Definition
PT = 0xB9
N = 12

CH Robotics

CHR-6dm Digital AHRS datasheet

D1-D4	D5-D8	D9-D12
Z gyro scale	Y gyro scale	X gyro scale

7.2.12.   START_CAL_REPORT

Description
Sent in response to the GET_START_CAL command. Returns the gyro bias startup calibration status (enabled or disabled).

Packet Definition
PT = 0xBA
N = 1

D1							
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
NA	NA	NA	NA	NA	NA	NA	en

If 'en' = 1, then startup gyro bias calibration is enabled.

7.2.13.   ACCEL_BIAS_REPORT

Description
Sent in response to the GET_ACCEL_BIAS command. Returns the accelerometer biases for the X, Y, and Z accel axes as 16-bit 2's complement signed integers.

Packet Definition
PT = 0xBB
N = 6

D1	D2	D3	D4	D5	D6
accel_z bias		accel_y bias		accel_x bias	

7.2.14.   ACCEL_REF_VECTOR_REPORT

Description
Sent in response to the GET_ACCEL_REF_VECTOR command. Returns the expected accelerometer output when pitch and roll angles are zero.

Packet Definition
PT = 0xBC
N = 6

D1	D2	D3	D4	D5	D6
accel_z ref		accel_y ref		accel_x ref	

- 37 -

7.2.15.	ACTIVE_CHANNEL_REPORT

Description
Sent in response to a GET_ACTIVE_CHANNELS command

Packet Definition
PT = 0xBD
N = 2

D1								D2							
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
yaw	pitch	roll	yaw rate	pitch rate	roll rate	mx	my	mz	gx	gy	gz	ax	ay	az	0

If the bit corresponding to a sensor channel is set (i.e. gz = 1), then the channel is active. If the bit is cleared (i.e. gz = 0), then the channel is inactive, and data from that channel is not transmitted.

yaw = yaw angle estimate
pitch = pitch angle estimate
roll = roll angle estimate
yaw rate = yaw rate estimate
pitch rate = pitch rate estimate
roll rate = roll rate estimate
mx = magnetometer x-axis data
my = magnetometer y-axis data
mz = magnetometer z-axis data
gx = gyro x-axis data
gy = gyro y-axis data
gz = gyro z-axis data
ax = accel x-axis data
ay = accel y-axis data
az = accel z-axis data

7.2.16.	ACCEL_COVARIANCE_REPORT

Description

Returns the covariance used in the accelerometer update step of the EKF. The AHRS assumes that the accelerometer covariance is a diagonal matrix with equivalent diagonal entries. The ACCEL_COVARIANCE_REPORT packet thus includes only one 32-bit floating point value for the covariance matrix.

Packet Definition

PT = 0xBE
N = 4

D1	D2	D3	D4

D1	D2	D3	D4
32-bit IEEE floating point value			

7.2.17.　　MAG_COVARIANCE_REPORT

Description

Returns the covariance used in the magnetometer update step of the EKF. The AHRS assumes that the magnetometer covariance is a diagonal matrix with equivalent diagonal entries. The MAG_COVARIANCE_REPORT packet thus includes only one 32-bit floating point value for the covariance matrix.

Packet Definition

PT = 0xBF
N = 4

D1	D2	D3	D4
32-bit IEEE floating point value			

7.2.18.　　PROCESS_COVARIANCE_REPORT

Description

Returns the covariance used in the prediction step of the EKF. The AHRS assumes that the process covariance is a diagonal matrix with equivalent diagonal entries. The PROCESS_COVARIANCE_REPORT packet thus includes only one 32-bit floating point value for the covariance matrix.

Packet Definition

PT = 0xC0
N = 4

D1	D2	D3	D4
32-bit IEEE floating point value			

7.2.19.　　STATE_COVARIANCE_REPORT

Description

Returns the 3x3 covariance matrix of the current yaw, pitch, and roll state estimates from the EKF. Each matrix element is a 32-bit IEEE floating-point value.

Packet Definition

PT = 0xC1
N = 36

D1-D4	D5-D8	D9-D12	D13-D16	D17-D20	D21-D24	D25-D28	D29-D32	D33-D36
mat[0,0]	mat[0,1]	mat[0,2]	mat[1,0]	mat[1,1]	mat[1,2]	mat[2,0]	mat[2,1]	mat[2,2]

mat[i,j] corresponds to the i,jth element in the matrix.

7.2.20.　　　EKF_CONFIG_REPORT

Description
Sent in response to a GET_EKF_CONFIG report. Reports whether the accelerometer and magnetometer are used in the EKF update step.

Packet Definition

PT = 0xC2
N = 1

D1							
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
NA	NA	NA	NA	NA	NA	Accel_EN	Mag_EN

If Mag_EN = 1, then magnetometer updates are used for yaw angle correction. If 0, then magnetometer correction is disabled.
If Accel_EN = 1, then accelerometer updates are used for pitch and roll angle correction. If 0, then accelerometer correction is disabled.

7.2.21.　　　GYRO_ALIGNMENT_REPORT

Description

Returns the 3x3 matrix used to correct gyro cross-axis misalignment. Each value in the matrix is a 32-bit IEEE floating-point value.

Packet Definition

PT = 0xC3
N = 36

D1-D4	D5-D8	D9-D12	D13-D16	D17-D20	D21-D24	D25-D28	D29-D32	D33-D36
mat[0,0]	mat[0,1]	mat[0,2]	mat[1,0]	mat[1,1]	mat[1,2]	mat[2,0]	mat[2,1]	mat[2,2]

mat[i,j] corresponds to the i,jth element in the matrix.

7.2.22.　　　ACCEL_ALIGNMENT_REPORT

Description

Returns the 3x3 matrix used to correct accelerometer cross-axis misalignment. Each value in the matrix is a 32-bit IEEE floating-point value.

Packet Definition

PT = 0xC4
N = 36

D1-D4	D5-D8	D9-D12	D13-D16	D17-D20	D21-D24	D25-D28	D29-D32	D33-D36
mat[0,0]	mat[0,1]	mat[0,2]	mat[1,0]	mat[1,1]	mat[1,2]	mat[2,0]	mat[2,1]	mat[2,2]

mat[i,j] corresponds to the i,jth element in the matrix.

7.2.23.   MAG_REF_VECTOR_REPORT

Description
Sent in response to the GET_MAG_REF_VECTOR command. Returns the expected magnetometer output when yaw, pitch, and roll angles are zero. Each value returned is a 16-bit 2's complement signed integer representing the raw magnetometer output.

Packet Definition
PT = 0xC5
N = 6

D1	D2	D3	D4	D5	D6
mag_z ref		mag_y ref		mag_x ref	

7.2.24.   MAG_CAL_REPORT

Description

Returns the 3x3 matrix used to correct magnetometer axis misalignment and soft iron distortion effects. Each value in the matrix is stored as a 32-bit IEEE floating-point value.

Packet Definition

PT = 0xC6
N = 36

D1-D4	D5-D8	D9-D12	D13-D16	D17-D20	D21-D24	D25-D28	D29-D32	D33-D36
mat[0,0]	mat[0,1]	mat[0,2]	mat[1,0]	mat[1,1]	mat[1,2]	mat[2,0]	mat[2,1]	mat[2,2]

mat[i,j] corresponds to the i,jth element in the matrix.

7.2.25. MAG_BIAS_REPORT

Description

Sent in response to the GET_MAG_BIAS command. Returns the bias terms used to compensate for hard-iron magnetic field distortions. Each value returned is a 16-bit 2's complement signed integer representing the raw magnetometer output.

Packet Definition

PT = 0xC7

N = 6

D1	D2	D3	D4	D5	D6
mag_z bias		mag_y bias		mag_x bias	

7.2.26. BROADCAST_MODE_REPORT

Description

Sent in response to a GET_BROADCAST_MODE packet. Specifies whether the AHRS is in Broadcast Mode or Silent Mode. Also specifies the broadcast frequency if the AHRS is in Broadcast Mode.

Packet Definition

PT = 0xC8

N = 2

D1								D2							
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
x								NA	NA	NA	NA	NA	NA	NA	*mode*

The broadcast frequency is given by

$$f = ((280/255)*x + 20) \text{ Hz}$$

where f is the broadcast frequency, and x is the byte stored in D1. The broadcast frequency thus ranges between 300 Hz (when x = 255) and 20 Hz (when x = 0).

If *mode* = 1, then the AHRS is in Broadcast Mode.
If *mode* = 0, then the AHRS is in Silent Mode.

8. Disclaimer

This document is provided as reference only; "typical" device specifications must be evaluated by the end-user. CH Robotics reserves the right to modify this document and the products it describes without notice.

CH Robotics products are not intended for use in weapons systems, aircraft, life-saving or life-sustaining systems, automobiles, or any other application where failure could result in injury, death, property damage, or environmental damage.