CXM547 User's Manual

Micro Orientation Sensor
Rev A November, 2001
PN 6001-0020
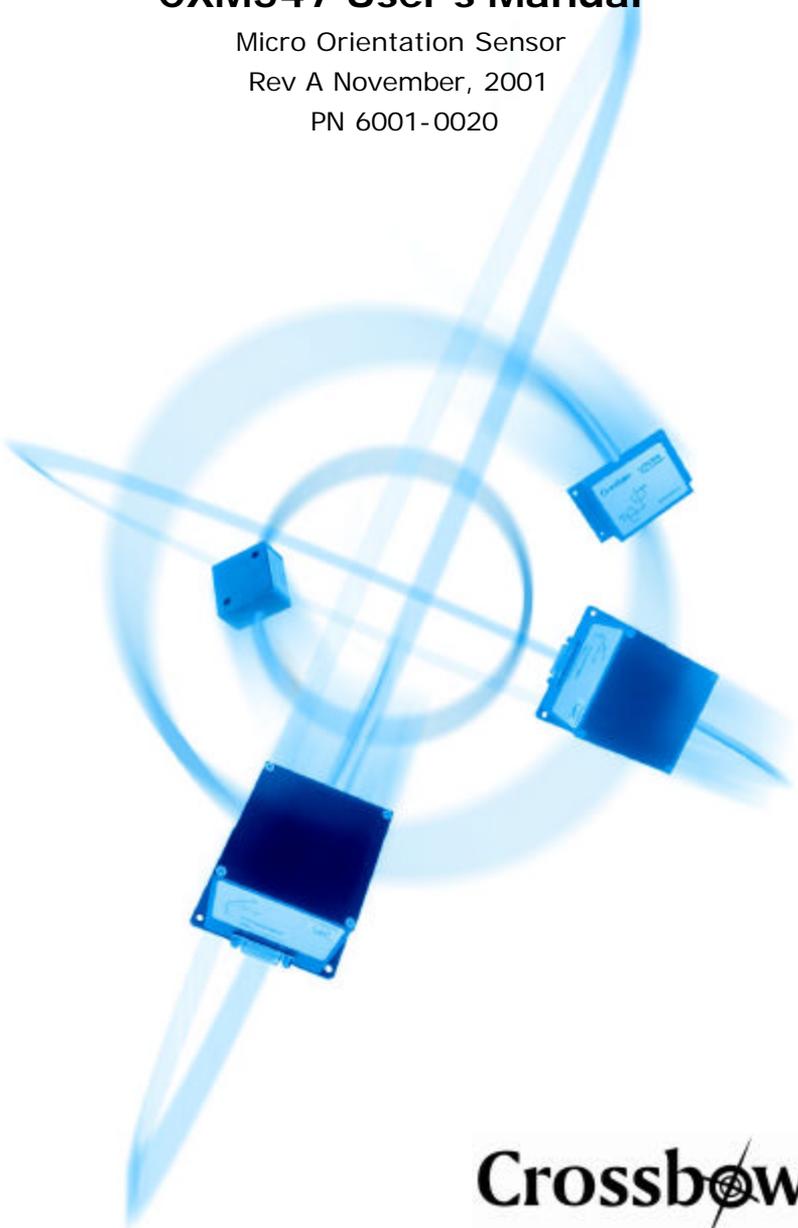
Crossbow

Table of Contents

About this Manual

The following annotations have been used to provide additional information.

◀ NOTE

Note provides additional information about the topic.

☑ EXAMPLE

Examples are given throughout the manual to help the reader understand the terminology.

☞ IMPORTANT

This symbol defines items that have significant meaning to the user

☀ WARNING

The user should pay particular attention to this symbol. It means there is a chance that physical harm could happen to either the person or the equipment.

The following paragraph heading formatting is used in this manual:

1 Heading 1

1.1 Heading 2

1.1.1 Heading 3

Normal

1 Description of the System

The CXM547 Micro Orientation Sensor system contains both a 3-axis fluxgate magnetometer and a 3-axis accelerometer. The combination of these two sensor systems enables the roll, pitch and azimuth angles of the CXM547 reference frame to be determined. Roll and pitch angles are determined from the accelerometer subsystem which measures the pull of gravity. After roll and pitch are known, the magnetometer subsystem is used to determine the azimuth angle referred to as declination or heading angle. Knowledge of the roll and pitch angles enable determination of the horizontal components of the earth's local magnetic field; this information defines the azimuth angle.

The CXM547 systems also contain a microprocessor (68HC11) and 7 channels of 16-bit analog to digital conversion. Six channels are assigned to the magnetometer and accelerometer outputs. One channel provides temperature data from an internal thermometer.

The CXM547 communicates with the outside world over a serial bi-directional TTL interface. An ASCII character command language has been created to facilitate communication with sensors. For instance, if the ASCII for O, S and D (followed by a carriage return) are sent in sequence, the CXM547 interprets this as a "send data" command and responds by sending over the serial interface an ASCII string representing the value of all magnetometer, accelerometer and temperature outputs. The leading zero in this sequence denotes the system serial number. The CXM547 can also be configured to send angle data instead of the accelerometer and magnetometer sensor data.

An auto send data mode is included in the CXM547 software. When this mode is active, data is repeatedly sent after the power up at a rate of two transmissions per second in angular output mode and three times per second in magnetometer/accelerometer mode.

The CXM547 system accelerometers are calibrated by placing the system in a precision rotation fixture and systematically changing the system orientation in the earth's gravitational field. The CXM547 system magnetometers are calibrated by placing the system in a precision 3-axis Helmholtz coil system, which enables the application of known magnetic

fields to the system. Both the rotation fixture and Helmholtz coil have alignment pins, which mate to the CXM547 cylindrical case and alignment pins. System calibration is performed over a temperature range (for example 0 to 100°C). When the system is calibrated over a temperature range, data is read from the system at temperature intervals between the minimum and maximum temperature specification. For instance, for calibration over the interval of 25 to 125°C, data is usually read at 25°C temperatures between 25 and 125°C. The data taken at each temperature includes scale offset and sensor alignment data. The recorded data is then used to create a look up table for scale offset and alignment corrections. This table is then downloaded into the CXM547 internal EEROM memory where it can be accessed by the system internal microprocessor. Corrections to the read sensor data can then be made by the internal microprocessor system before data is transmitted.

2   System Specifications

Angular Accuracy (0° to 125°C)

	Azimuth	±1.5°
	Inclination	±0.4°
	Roll	±0.4°

Axis alignment   ±0.2°

Alignment of axes with package reference frame

±0.2°

Temperature Range

Operating   0° to 125°C

Storage   -55° to +165°C

Noise level on Magnetometers 5×10^{-6} Gauss
Noise level on Accelerometer 2×10^{-4} G

Power   +7 VDC to +12VDC (unregulated) or
                         +5 VDC ±0.05 VDC (regulated)
                         75 mA @ 7 VDC

Size   1" OD x 10.4" long cylinder

Shock   1000 G 1ms half sine wave

Mass   250 grams

Vibration   20G RMS, 5-1000Hz

Digital Outputs

logic level	TTL/CMOS
baud	user programmable up to 9600
protocol	user selectable ASCII or binary
maximum throughput for all data	5 transmissions/sec

Connector   Cannon MDM9SH003P
Mating Connector   Cannon MDM9PH003L

3   Electrical and Mechanical Interface

3.1   Electrical Interface

Electrical connection to the CXM547 is made by means of a 9-pin MDM type connector. The functions of the pins on this connector are shown in Fig. 1.   Two powering options are available for the CXM547.   An unregulated voltage between +7V and +12V (connected to pin 3) or a regulated +5V voltage (connected to pin 4) can be used.   When the +7 to +12 volt option is used, internal regulators inside the CXM547 produce the internal voltages required by the system.   When a regulated +5V supply is used, the input regulators are bypassed. The tolerance of the +5V input is ±0.05V.

The serial communications interface to the CXM547 is provided by the serial in and serial out lines shown in Figure 1. An external computer communicates with the CXM547 on the serial in line and replies from the CXM547 are transmitted out on the serial out line. The serial in and serial out lines operate at TTL/CMOS levels and are normally set to operate at 9600 baud with one stop bit and no parity. The user however can change the baud rate by setting bits in the system EEROM.

Apply Power to either Pin 3 or Pin 4.

Pin	Function
1	Serial in (TTL)
2	Serial out (TTL)
3	+7 VDC to + 12 VDC
4	+5 VDC (± 0.05 V)
5	Ground
6	No Connect
7	No Connect
8	No Connect
9	No Connect
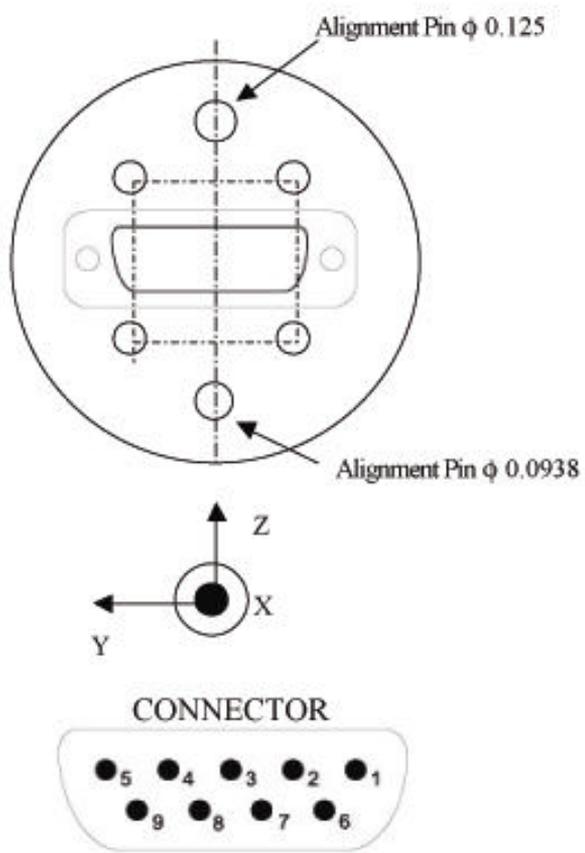
Figure 1. Connection diagram CXM547

Two communication protocols are available: 1) ASCII and 2) BINARY. These are both described in detail later in this manual in the "Heading Sensor Command Set Definition" (see Appendix). The ASCII protocol is based upon sending ASCII characters to the CXM547 to obtain data. The reflected data is also sent out as an ASCII data stream complete with carriage returns and line feeds so that it can be easily displayed on a video terminal (provided a TTL to RS232 conversion is made by the user). The binary protocol is used for high-speed computer-to-computer interchange. In this case, one byte is sent to request data. The CXM547 then responds with a multi-byte data packet containing the desired data plus header and checksum.

The system internal constants include a number of byte as well as floating constants. Generally, byte constants are used to control the system operating modes. Floating constants are employed to scale the output data and correct it for temperature variations. A detailed description of the serial command protocol and constants definition is included in the Appendix.

3.2   Mechanical Features

An outline drawing of the CXM547 system is shown in Figure 2. The system can be mounted by clamping the OD of the package and by using the threaded 4-40 tapped holes in the top and bottom plates of the sensor. The CXM547 coordinate system is defined by the direction of the outer circular housing and by the direction of the gauge plug mounted on the system top end.

The orientation of the X Y and Z axes is shown in Figure 1. The output polarity sense of the axes is such that a field increase in the direction of the arrows shown in Fig. 2 produces an increase in the voltage output for that axis. For example, if the X magnetometer is oriented so the X-axis arrow points north in the northern hemisphere, then the X-axis output voltage will be positive. If the X-axis accelerometer arrow is pointed down, the X-axis accelerometer will be positive.
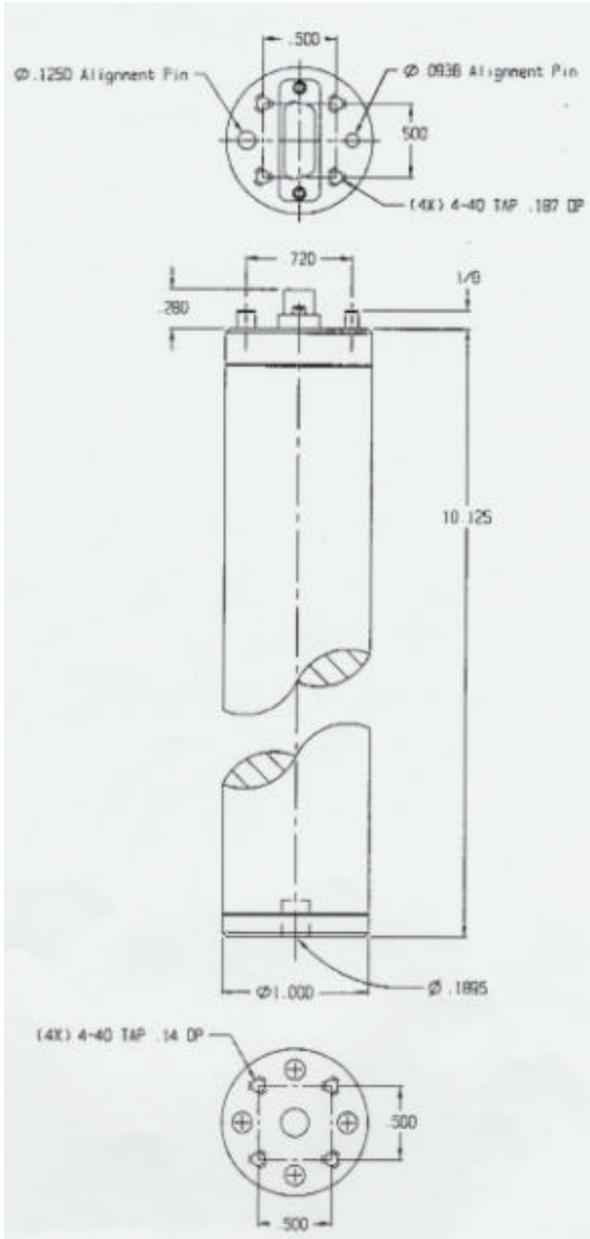
Figure 2. CXM547 Micro-Orientation Sensor

(Dimensions are in inches)

4   System Startup and Checkout

4.1   Initial Setup of the System

The computer interface of the CXM547 system is a TTL level serial interface. Data is transmitted to the CXM547 on pin 1 and transmitted out of the CXM547 on pin 2.   The default baud rate is 9600 with one stop bit and no parity. The baud rate can be changed by the user by making changes to internal constants 08 and 09 in the system EEROM.

In order to communicate with an external personal computer (PC), the TTL levels of the CXM547 must be shifted to the RS232 levels used by the serial ports on all PCs. TTL levels generally are +5V (mark) and 0v (space) whereas RS232 levels are generally -5 to -12V (mark) and +5 to +12V (space). Integrated circuits such as the Max 232 (Maxim Integrated Products) or the LT1019 (Linear Technology) are suitable for constructing an RS232 to TTL interface.

Communication with the CXM547 is initially controlled by an external computer. Communication is initiated when the external computer issues a command such as *0SD<CR>* (0 Send Data). The 0 in this sequence is the de-fault serial number of the unit.   When this command is issued, the CXM547 will respond with a formatted output similar to the following:

MX: 0.465	**AX: 0.986**
MY: 0.231	**AY: 0.234**
MZ: 0.100	**AZ: 0.111**

When internal binary constant 02=02, the CXM547 is in sensor output mode and the above numb ers represent the magnetometer X, Y and Z sensor outputs in Gauss and the accelerometer outputs in Gs. When binary byte 02=03, the CXM547 is in angle mode and the above numbers represent:

MX: Roll	**AX:** Magnetic roll
MY: Inclination	**AY:** Total magnetic field
MZ: Azimuth	**AZ:** Total gravitational field.

4.2   Descriptions of the System Internal Constants

The operating characteristics of the CXM547 are controlled by the value of internal binary constants. The most important constants are:

Byte	Function
00	enables echoing when non zero
01	enables autosend when = 5A
02	enables sensor A/D count output when =0, sensor output when =02 and angle output when =03
08	Sets power on mode (e.g., =10 enables autosend in ASCII mode on power on)
09	Baud rate lock (=5A if any baud rate other than 9600 is to be used)
10	Sets baud rate

Detailed definitions of all of the internal byte constants can be found in "Heading Sensor Command Set" in the Appendix.

In order to change the internal system binary constants, a write enable command must first be issued.

To accomplish this, type: *0L<CR>*
The unit will respond with the message: *enabled!*

To write byte constant 02=03, type the command: *0WC0203b03<CR>*
The unit will respond by sending the message: *done*

The reading of internal constants can be accomplished by issuing the command: *0SC02b<CR>*

(0 Send Constant 02 binary) when this command is sent the CXM547 will respond by sending the value of constant 02. Wildcards are also recognized. The command *OSC*b* will cause the CXM547 to send the value of all internal binary constants.

In addition to internal byte constants, the CXM547 also has a number of floating constants. These are used to store the calibration data in the CXM547 EEROM. These constants can be read by using the commands

> *0SC*f <CR>* for all constants
> *0SC06F <CR>* for constant 06

A list of all the floating constants and their definitions can be found in Appendix A of the manual. The most important floating constants are:

04	X Magnetometer Offset
05	Y Magnetometer Offset
06	Z Magnetometer Offset
07	X Accelerometer Offset
08	Y Accelerometer Offset
09	Z Accelerometer Offset
10	X Magnetometer scale
11	Y Magnetometer scale
12	Z Magnetometer scale
13	X Accelerometer scale
14	Y Accelerometer scale
15	Z Accelerometer scale

Floating constants 16 to 34 contain the system alignment coefficients. For example, floating constant 17 contains data on the magnetometer X sensor alignment in the Y direction.

In addition to an ASCII communication mode, the CXM547 also has a binary communications mode. This mode is more suitable for microprocessor communication because it is much shorter and faster than the ASCII communications mode. Binary communications are initiated by an external computer by the issuance of a single byte, e.g., ASCII 128. The CXM547, upon receiving this request, responds with a binary answer consisting of 22 bytes containing the requested data in binary form together with a header, footer and checksum.

The binary commands available in CXM547 are as follows:

Command	Command definition
128	send sensor data in binary format
129	send sensor data in IEEE 32 bit format
131	send angle data in binary format
132	send angle data in IEEE 32 bit format

The CXM547 response to these commands is of the following form:

\<number of data bytes to be sent> \<MX> \<AX> \<MY> \<AY> \<MZ>
\<AZ> \<MT> \<AT> \<0> \<data check sum>\<end>

Where \<MX> is the X magnetometer data, \<AX> is the X accelerometer
data, etc. The system temperature is contained in the \<MT> variable. The
\<AT> variable contains the system input voltage.

5   Description of the CXM547 Orientation Angles

This section describes how to calculate the roll, pitch and azimuth angles
for the CXM547 orientation sensor from accelerometer and magnetometer
output data.

5.1   Sensor Based Coordinate System

The coordinate system of the CXM547 system is defined in Figure 3. The
accelerometer and magnetometer coordinate systems are both aligned with
the physical package coordinate systems shown. For the magnetometer
sensors, a positive output voltage will result if the sensor is pointed north.
For the accelerometers, a positive voltage will result if the sensors are
pointed down.
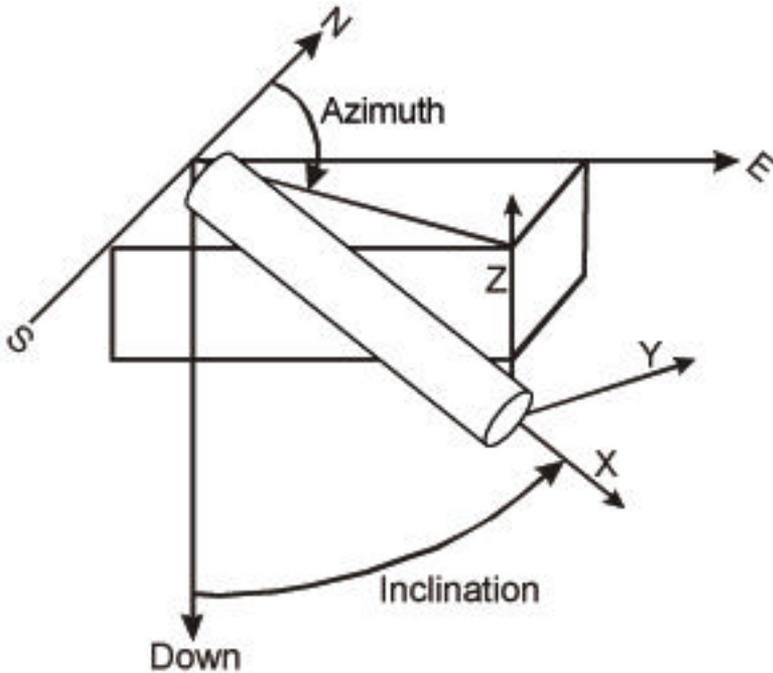
Figure 3. CXM547 Coordinate System

5.2 Definition of Orientation Angles

The orientation angles are defined in Figure 4. Azimuth or yaw is defined as the angle measured from magnetic north (clockwise from above) to the projection of the X-axis on the horizontal plane.

Inclination is the angle that the X-axis makes with the down direction and is 0° when the x-axis is down and 90° when the X-axis is horizontal.

Roll or gravity tool face is defined as the angle of counterclockwise rotation about the X-axis (looking in the positive X-axis direction) required to zero the Y-axis accelerometer output and position the Z-axis accelerometer so that its output polarity is positive.

Magnetic roll or magnetic toolphase is defined as the angle of counterclockwise rotation about the X-axis (looking in the positive X-axis direction) required to zero the Y-axis magnetometer output and position the Z-axis magnetometer so that its output polarity is negative.

Magnetic roll is useful in defining the CXM547 orientation when inclination is near vertical. In this situation, g_y and g_z are near zero and roll and azimuth calculations become less accurate.

5.3 Definitions

The following sections describe equations for determining the CXM547 orientation angles.

These equations make use of the following definitions:

g_x accelerometer X-axis output
g_y accelerometer Y-axis output
g_z accelerometer Z-axis output
H_x magnetometer X-axis output
H_y magnetometer Y-axis output
H_z magnetometer Z-axis output
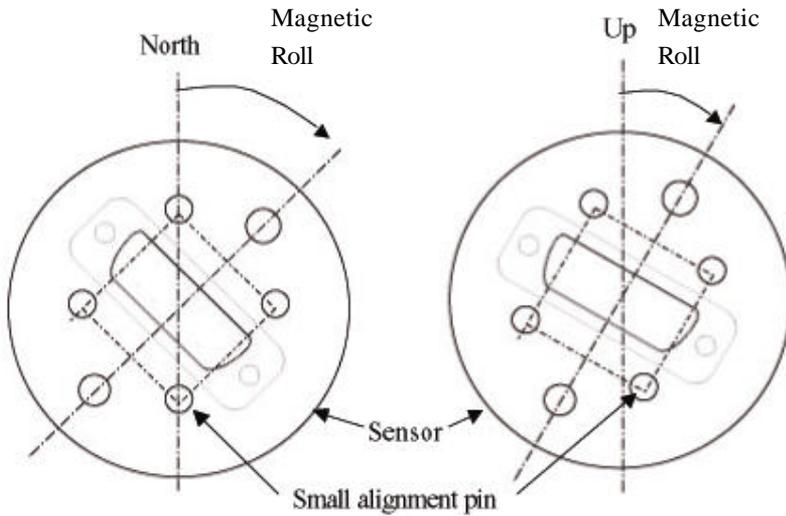
Figure 4. CXM547 Orientation Angles

5.4 Calculation of Roll and Magnetic Roll

The roll angle, ?, is determined by using the following equations
$(0 = \theta = 2p)$:

$$\cos\boldsymbol{q} = \frac{g_z}{\sqrt{g_y^2 + g_z^2}}$$

$$\sin\boldsymbol{q} = \frac{g_y}{\sqrt{g_y^2 + g_z^2}}$$

$$\tan\boldsymbol{q} = \frac{g_y}{g_z}$$

Roll is $0°$ when $g_y = 0$ and $g_z > 0$. Roll is 2π radians when $g_y = 0$ and $g_z < 0$.
When the X-axis is near vertical (pitch $< 5°$), the quantities g_y and g_z
become very small and the above expressions yield a less accurate value of
θ. In this situation, magnetic roll is often used to determine the angular
orientation of the CXM547 about the longitudinal (X) axis. Magnetic roll,
θ_m, is given by the following

$(0 = \theta_m = 2p)$:

$$\sin \boldsymbol{q}_m = \frac{-H_y}{\sqrt{H_y{}^2 + H_z{}^2}}$$

$$\cos \boldsymbol{q}_m = \frac{-H_z}{\sqrt{H_y{}^2 + H_z{}^2}}$$

$$\tan \boldsymbol{q}_m = \frac{H_y}{H_z}$$

5.5 Calculation of Inclination

Inclination, e, is determined from the following equations

$(0 = e = p)$:

$$\cos \boldsymbol{e} = \frac{g_x}{g}$$

$$\sin \boldsymbol{e} = \frac{\sqrt{g_y{}^2 + g_z{}^2}}{g}$$

$$\tan \boldsymbol{e} = \frac{\sqrt{g_y{}^2 + g_z{}^2}}{g_x}$$

where, $g = \sqrt{g_x{}^2 + g_y{}^2 + g_z{}^2}$

Inclination is 0 when the CXM547 X-axis is pointed down and p/2 radians when horizontal.

5.6 Magnetic Heading (Azimuth, Yaw)

We first give expressions for the magnetic field in a horizontal reference defined by x1, y1, z1 where x1 is aligned with the projection of the CXM547 x axis in the horizontal plane and z1 is down.

$$H_{x1} = \frac{H_x \left(g_y{}^2 + g_z{}^2\right) - H_y g_y g_x - H_z g_x g_z}{g\sqrt{g_y{}^2 + g_z{}^2}}$$

$$H_{y1} = \frac{H_y g_z - H_z g_y}{\sqrt{g_y{}^2 + g_z{}^2}}$$

$$H_{z1} = \frac{H_x g_x + H_y g_y + H_z g_z}{g}$$

Magnetic heading , ø, is then given by $(0 = \phi = 2p)$

$$\cos f = \frac{H_{x1}}{\sqrt{H_{x1}^2 + H_{y1}^2}}$$

$$\sin f = \frac{-H_{y1}}{\sqrt{H_{x1}^2 + H_{y1}^2}}$$

$$\tan f = \frac{-H_{y1}}{H_{x1}} = \frac{\left(H_z g_y - H_y g_z\right)g}{H_x\left(g_y^2 + g_z^2\right) - H_y g_x g_y - H_z g_x g_z}$$

Magnetic heading is 0 when the CXM547 X-axis points North and p/2 radians when it points East.

6 Appendix A. CXM547 Command and Data Format Specifications

6.1 Heading Sensor Command Set

CXM547
V3.47D 11/2000

Addressed Command is : <Address><Command><CR>

An Address is <0-9 | A-Z>
CR is a Carriage Return
Axis is <X - Z | T>
Constant Number is <00 - 99>
Constant Value is <[-]<number>[<number>]>
Constant String is <A-Z | 0-9><Constant String>
A number is <<0-9>[number]>

A command consists of one of the following:

Send All Data:
SD All Data is sent   MX: ±#.##### AX:±#.#####<CR><LF>
as requested,   MY: ±#.##### AY:±#.#####<CR><LF>
raw, vectors or   MZ: ±#.##### AZ:±#.#####<CR><LF>
angles as set by   MT: ±###.####<CR><LF>
byte constant 2   AT: ±###.####<CR><LF><EOT>

Send Ana1 Input: SA ANA: #.#####<CR><LF><EOT>

Read Float Constant: SC<Constant Number>F
     ±########±E##<CR><LF><EOT>

Read All Float Constants : SC*F   00: ±########±E##<CR><LF>
     01: ±########±E##<CR><LF>
     ...
     ##: ±########±E##<CR><LF><EOT>

Read Byte Constant: SC<Constant Number>B ##<CR><LF><EOT>
     (In Hex)

Read All Byte Constants: SC*B   00: ##<CR><LF> (In Hex)
     01: ##<CR><LF><EOT>
     ...
     ##: ##<CR><LF><EOT>

Read int Constant : SC<Constant Number>I #####<CR><LF><EOT>

Read all int Constants : SC*I   00: #####<CR><LF>
                                       01: #####<CR><LF><EOT>
                                               ...
                                       ##: ##<CR><LF><EOT>

Reset A/D :   RA Done<CR><LF><EOT>

Test ROM Checksum(*) : TC if low level writes are
                        enabled, the output is: STORED
                        HHHH<CR><LF><EOT>
                        If low level writes are not enabled and the
                        Checksum of the ROM is the same as the one
                        stored: HHHH<CR><LF><EOT>
                        else the Output is:
                                ERR HHHH<CR><LF><EOT>

Test EEROM Checksum(*) : TE if low level writes are
                        enabled, the output is: STORED
                        HHHH<CR><LF><EOT>
                        If low level writes are not enabled and the
                        Checksum of the ROM is the same as the one
                        stored: HHHH<CR><LF><EOT>
                        else the Output is:
                                ERR HHHH<CR><LF><EOT>

Test RAM: TR Either an OK or an error
                        OK<CR><LF><EOT>
                        The Error Addr. is in Hex
                        ERR AT: HHHH<CR><LF><EOT>

Test EEROM Read/Write*: TW (This takes up to 10 Seconds)
                        Either an OK or an error OK<CR><LF><EOT>
                        The Error Addr. is in Hex
                        ERR AT: HHHH<CR><LF><EOT>

Test Serial Port: TS Always Echoes OK<CR><LF><EOT>

Return Software Version: TV Ver: #.###<CR><LF><EOT>

Set Serial Number*:   TV=XXXXXXXXXX
        where, XXXXXXXXXX is the Serial Number

Enable Low Level Writes: L Enabled!<CR><LF><EOT>
Write Float To Constant*:   WC<Constant Number>F<Value>
                            Done<CR><LF><EOT>

Write Byte To Constant*:   WC<Constant Number>B<number>
                            Done<CR><LF><EOT>

Write Int To Constant*:   WC<Constant Number>B<number>
                            Done<CR><LF><EOT>

Read All Temp Table as Bytes: ST*B
Read All Temp Table as Int: ST*I
Read All Temp Table as Flaot: ST*F
Read Temp Table as Byte: ST####B
Read Temp Table as Int: ST####I
Read Temp Table as Float: ST####F

☀ **WARNING**

Any Writes to the temp table can introduce errors into data that will only
appear when the sensor is at a specific temperature or cause the unit to fail.
Unless you know what you are doing, DON'T.

Write Temp Table as Byte: WT####B##
Write Temp Table as Int: WT####I##
Write Temp Table as Float: WT####F##
Write All Temp Table to ##: WT*B##

* All Commands marked with an * require the Enable Low Level Writes
command to be executed as the previous command for successful execution.
All Commands Marked with a (*) execute differently when Enable Low
Level Writes was executed as the previous command.

----- Binary Commands -----

<128>   Send All Data as Vectors in a Binary Format.
<129>   Send All Data as Vectors in a IEEE Float Format.
<131>   Send All Data as Angles in a Binary Format
<132>   Send All Data as Angles in a IEEE Float Format
<128> and <131> Sends All Data in an encoded Binary Format in the
current format as selected by Byte constant #2.

If the correction mode is 0, 1 or 2 the data is returned as:

<<<<Sent First <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<Sent Last<<<<

If The Command is 128:
<NUM DATA BYTES>
<MX><AX><MY><AY><MZ><AZ><MT><AT><0>
<DATACHKSUM><END>
size 8b 16b 16b 16b 16b 16b 16b 16b 16b 8b 8b 16b in bits

<END>   = 0x7FFF (is unique in the data stream)
<NUM DATA BYTES>= 16
<Status>= a value of 0x80 unless there is a mule shoe sensor
            then it will have a value of 0 when seated in the
            mule shoe.
<DATA CHECK SUM>= The lower 8 bits of the sum of all the bytes in the
                            data area.
<MX>,<MY>,<MZ>   = The Magnetometer Data encoded as below.
<AX>,<AY>,<AZ>   = The Accelerometer Data encoded as below.
<MT>,<AT>   = The Temp Data encoded as below.

All Data is Sent most significant byte first.
The Magnetometer and Accelerometer Data is in a two byte signed integer
format encoded as the float value times 10000. (1.2345 = 12345)
The Temp Data is in a two byte signed integer format encoded as the float
value times 100. (123.45 = 12345)

If The Command is 131:
<NUM DATA
BYTES><Roll><MRoll><Pitch><Mag><Head><Grav><MT><AT><0><
DATA CHK SUM><END>
size 8b 16b 16b 16b 16b 16b 16b 16b 16b 8b 8b 16b in bits
<END>   = 0x7FFF (is unique in the data stream)
<NUM DATA BYTES>   = 16
<0>   = a constant 0 (to allow END to be unique)
<DATA CHECK SUM>   = The lower 8 bits of the sum of all the bytes in
                                   the data area.
<Roll>   = The Roll data encoded as below.
<Pitch>   = The Pitch data encoded as below.
<Head>   = The Heading data encoded as below.
<MRoll>   = The Magnetic Roll data encoded as below.
<Mag>   = The Total Magnetic Field data encoded as
                                   below.
<Grav>   = The Total Gravitational Field data encoded as
                                   below.

<MX>,<MY>,<MZ>	= The Magnetometer data encoded as below.
<AX>,<AY>,<AZ>	= The Accelerometer data encoded as below.
<MT>,<AT>	= The Temp data encoded as below.

All data is sent most significant byte first.

If the correction mode is 0, the Magnetometer and Accelerometer data is in a two byte signed integer format in counts.

If the correction mode is 1 or 2, the Magnetometer and Accelerometer data is in a two byte signed integer format encoded as the float value times 10000. (1.2345 = 12345)

If the correction mode is 3 the data is angles and is represented in a two byte signed integer format encoded as the float value times 10. (123.45 = 1234)

The Temp data is in a two byte signed integer format encoded as the float value times 100. (123.45 = 12345)

<129>and <132> Sends All Data in IEEE Floating point format in the current format as selected by Byte constant #2.

If the correction mode is 0, 1 or 2 the data is returned as:

<<<<Sent First<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<Sent Last<<<<
If The Command is 129:
<NUM DATA
BYTES><MX><AX><MY><AY><MZ><AZ><MT><AT><0><DATA
CHECK SUM>
size 8b 32b 32b 32b 32b 32b 32b 32b 32b 8b 8b in bits
If The Command is 132:
<NUM DATA
BYTES><Roll><MRoll><Pitch><Mag><Head><Grav><MT><AT><0><
DATA CHECKSUM>
size 8b 32b 32b 32b 32b 32b 32b 32b 32b 8b 8b in bits

<END>	= 0x7FFF (is unique in the data stream)
<NUM DATA BYTES>	= 16
<0>	= a constant 0 (to allow END to be unique)
<DATA CHECK SUM>	= The lower 8 bits of the sum of all the bytes in the data area.
<Roll>	= The Roll data.

<Pitch>	= The Pitch data.
<Head>	= The Heading data.
<MRoll>	= The Magnetic Roll data.
<Mag>	= The Total Magnetic Field data.
<Grav>	= The Total Gravitational Field data.
<MX>,<MY>,<MZ>	= The Magnetometer data.
<AX>,<AY>,<AZ>	= The Accelerometer data.
<MT>,<AT>	= The Temp data.

All data is sent mo st significant byte first.

Ortho Constants are designated Result Axis first, Component Axis Second.

----- Float Constant Numbers and Their Meanings -----

00 - Ana1 Scale
01 - Temp Scale
02 - Ana1 Offset
03 - Temp Offset
04 - Magnetometer Base Offset X
05 - Magnetometer Base Offset Y
06 - Magnetometer Base Offset Z
07 - Accelerometer Base Offset X
08 - Accelerometer Base Offset Y
09 - Accelerometer Base Offset Z
10 - Magnetometer Base Scale X
11 - Magnetometer Base Scale Y
12 - Magnetometer Base Scale Z
13 - Accelerometer Base Scale X
14 - Accelerometer Base Scale Y
15 - Accelerometer Base Scale Z
16 - Magnetometer Scale2 X
17 - Magnetometer Scale2 Y
18 - Magnetometer Scale2 Z
19 - Accelerometer Scale2 X
20 - Accelerometer Scale2 Y
21 - Accelerometer Scale2 Z
22 - Magnetometer Base Ortho(X,X)
23 - Magnetometer Base Ortho(X,Y)
24 - Magnetometer Base Ortho(X,Z)
25 - Magnetometer Base Ortho(Y,X)
26 - Magnetometer Base Ortho(Y,Y)
27 - Magnetometer Base Ortho(Y,Z)
28 - Magnetometer Base Ortho(Z,X)
29 - Magnetometer Base Ortho(Z,Y)
30 - Magnetometer Base Ortho(Z,Z)

31 - Accelerometer Base Ortho(X,X)
32 - Accelerometer Base Ortho(X,Y)
33 - Accelerometer Base Ortho(X,Z)
34 - Accelerometer Base Ortho(Y,X)
35 - Accelerometer Base Ortho(Y,Y)
36 - Accelerometer Base Ortho(Y,Z)
37 - Accelerometer Base Ortho(Z,X)
38 - Accelerometer Base Ortho(Z,Y)
39 - Accelerometer Base Ortho(Z,Z)
40 - Bow Correction X
41 - Bow Correction Y
42 - Bow Correction Z

----- Temp. Table Constant Numbers and Their Meanings -----

Base+00/01 - Magnetometer Base Offset X
Base+02/03 - Magnetometer Base Offset Y
Base+04/05 - Magnetometer Base Offset Z
Base+06/07 - Accelerometer Base Offset X
Base+08/09 - Accelerometer Base Offset Y
Base+10/11 - Accelerometer Base Offset Z
Base+12/13 - Magnetometer Base Scale X
Base+14/15 - Magnetometer Base Scale Y
Base+16/17 - Magnetometer Base Scale Z
Base+18/19 - Accelerometer Base Scale X
Base+20/21 - Accelerometer Base Scale Y
Base+22/23 - Accelerometer Base Scale Z
Base+24/25 - Magnetometer Base Ortho(X,X)
Base+26/27 - Magnetometer Base Ortho(X,Y)
Base+28/29 - Magnetometer Base Ortho(X,Z)
Base+30/31 - Magnetometer Base Ortho(Y,X)
Base+32/33 - Magnetometer Base Ortho(Y,Y)
Base+34/35 - Magnetometer Base Ortho(Y,Z)
Base+36/37 - Magnetometer Base Ortho(Z,X)
Base+38/39 - Magnetometer Base Ortho(Z,Y)
Base+40/41 - Magnetometer Base Ortho(Z,Z)
Base+42/43 - Accelerometer Base Ortho(X,X)
Base+44/45 - Accelerometer Base Ortho(X,Y)
Base+46/47 - Accelerometer Base Ortho(X,Z)
Base+48/49 - Accelerometer Base Ortho(Y,X)
Base+50/51 - Accelerometer Base Ortho(Y,Y)
Base+52/53 - Accelerometer Base Ortho(Y,Z)
Base+54/55 - Accelerometer Base Ortho(Z,X)

Base+56/57 - Accelerometer Base Ortho(Z,Y)
Base+58/59 - Accelerometer Base Ortho(Z,Z)

Temp #0 : Base = 0000	($0000)	Temp #69 : Base = 4140	($102C)
Temp #1 : Base = 0060	($003C)	Temp #70 : Base = 4200	($1068)
Temp #2 : Base = 0120	($0078)	Temp #71 : Base = 4260	($10A4)
Temp #3 : Base = 0180	($00B4)	Temp #72 : Base = 4320	($10E0)
Temp #4 : Base = 0240	($00F0)	Temp #73 : Base = 4380	($111C)
Temp #5 : Base = 0300	($012C)	Temp #74 : Base = 4440	($1158)
Temp #6 : Base = 0360	($0168)	Temp #75 : Base = 4500	($1194)
Temp #7 : Base = 0420	($01A4)	Temp #76 : Base = 4560	($11D0)
Temp #8 : Base = 0480	($01E0)	Temp #77 : Base = 4620	($120C)
Temp #9 : Base = 0540	($021C)	Temp #78 : Base = 4680	($1248)
Temp #10 : Base = 0600	($0258)	Temp #79 : Base = 4740	($1284)
Temp #11 : Base = 0660	($0294)	Temp #80 : Base = 4800	($12C0)
Temp #12 : Base = 0720	($02D0)	Temp #81 : Base = 4860	($12FC)
Temp #13 : Base = 0780	($030C)	Temp #82 : Base = 4920	($1338)
Temp #14 : Base = 0840	($0348)	Temp #83 : Base = 4980	($1374)
Temp #15 : Base = 0900	($0384)	Temp #84 : Base = 5040	($13B0)
Temp #16 : Base = 0960	($03C0)	Temp #85 : Base = 5100	($13EC)
Temp #17 : Base = 1020	($03FC)	Temp #86 : Base = 5160	($1428)
Temp #18 : Base = 1080	($0438)	Temp #87 : Base = 5220	($1464)
Temp #19 : Base = 1140	($0474)	Temp #88 : Base = 5280	($14A0)
Temp #20 : Base = 1200	($04B0)	Temp #89 : Base = 5340	($14DC)
Temp #21 : Base = 1260	($04EC)	Temp #90 : Base = 5400	($1518)
Temp #22 : Base = 1320	($0528)	Temp #91 : Base = 5460	($1554)
Temp #23 : Base = 1380	($0564)	Temp #92 : Base = 5520	($1590)
Temp #24 : Base = 1440	($05A0)	Temp #93 : Base = 5580	($15CC)
Temp #25 : Base = 1500	($05DC)	Temp #94 : Base = 5640	($1608)
Temp #26 : Base = 1560	($0618)	Temp #95 : Base = 5700	($1644)
Temp #27 : Base = 1620	($0654)	Temp #96 : Base = 5760	($1680)
Temp #28 : Base = 1680	($0690)	Temp #97 : Base = 5820	($16BC)
Temp #29 : Base = 1740	($06CC)	Temp #98 : Base = 5880	($16F8)
Temp #30 : Base = 1800	($0708)	Temp #99 : Base = 5940	($1734)
Temp #31 : Base = 1860	($0744)	Temp #100 : Base = 6000	($1770)
Temp #32 : Base = 1920	($0780)	Temp #101 : Base = 6060	($17AC)
Temp #33 : Base = 1980	($07BC)	Temp #102 : Base = 6120	($17E8)
Temp #34 : Base = 2040	($07F8)	Temp #103 : Base = 6180	($1824)
Temp #35 : Base = 2100	($0834)	Temp #104 : Base = 6240	($1860)
Temp #36 : Base = 2160	($0870)	Temp #105 : Base = 6300	($189C)
Temp #37 : Base = 2220	($08AC)	Temp #106 : Base = 6360	($18D8)
Temp #38 : Base = 2280	($08E8)	Temp #107 : Base = 6420	($1914)
Temp #39 : Base = 2340	($0924)	Temp #108 : Base = 6480	($1950)

Temp #40 : Base = 2400	($0960)	Temp #109 : Base = 6540 ($198C)
Temp #41 : Base = 2460	($099C)	Temp #110 : Base = 6600 ($19C8)
Temp #42 : Base = 2520	($09D8)	Temp #111 : Base = 6660 ($1A04)
Temp #43 : Base = 2580	($0A14)	Temp #112 : Base = 6720 ($1A40)
Temp #44 : Base = 2640	($0A50)	Temp #113 : Base = 6780 ($1A7C)
Temp #45 : Base = 2700	($0A8C)	Temp #114 : Base = 6840 ($1AB8)
Temp #46 : Base = 2760	($0AC8)	Temp #115 : Base = 6900 ($1AF4)
Temp #47 : Base = 2820	($0B04)	Temp #116 : Base = 6960 ($1B30)
Temp #48 : Base = 2880	($0B40)	Temp #117 : Base = 7020 ($1B6C)
Temp #49 : Base = 2940	($0B7C)	Temp #118 : Base = 7080 ($1BA8)
Temp #50 : Base = 3000	($0BB8)	Temp #119 : Base = 7140 ($1BE4)
Temp #51 : Base = 3060	($0BF4)	Temp #120 : Base = 7200 ($1C20)
Temp #52 : Base = 3120	($0C30)	Temp #121 : Base = 7260 ($1C5C)
Temp #53 : Base = 3180	($0C6C)	Temp #122 : Base = 7320 ($1C98)
Temp #54 : Base = 3240	($0CA8)	Temp #123 : Base = 7380 ($1CD4)
Temp #55 : Base = 3300	($0CE4)	Temp #124 : Base = 7440 ($1D10)
Temp #56 : Base = 3360	($0D20)	Temp #125 : Base = 7500 ($1D4C)
Temp #57 : Base = 3420	($0D5C)	Temp #126 : Base = 7560 ($1D88)
Temp #58 : Base = 3480	($0D98)	Temp #127 : Base = 7620 ($1DC4)
Temp #59 : Base = 3540	($0DD4)	Temp #128 : Base = 7680 ($1E00)
Temp #60 : Base = 3600	($0E10)	Temp #129 : Base = 7740 ($1E3C)
Temp #61 : Base = 3660	($0E4C)	Temp #130 : Base = 7800 ($1E78)
Temp #62 : Base = 3720	($0E88)	Temp #131 : Base = 7860 ($1EB4)
Temp #63 : Base = 3780	($0EC4)	
Temp #64 : Base = 3840	($0F00)	
Temp #65 : Base = 3900	($0F3C)	
Temp #66 : Base = 3960	($0F78)	
Temp #67 : Base = 4020	($0FB4)	
Temp #68 : Base = 4080	($0FF0)	

----- Byte Constant Numbers and Their Meanings -----

00 - Command Echo Flag. 0 is no command echo; anything else, echo commands

01 - Autostart Flag. If 0x5A executes the selected autostart option on powerup.

02 - Correction Level 0- Raw / 2- Vectors / 3- Angles

In Angles mode, Roll is labeled MX, Pitch is labeled MY, Heading is Labeled MZ, Magnetic. Roll is labeled AX, Total Magnetic Field is labeled AY,

Total Grav. Field is labeled AZ.

03 - Months since 1/90 when device was calibrated

04 - Version of the Calibration Software used

05 - Power on self test flag. If zero, a self test will be done on power up.

06 - Enable extended error messages

07 - This Sensor's Address Number 0-36 => 0-9,A-Z

08 - Auto Start Mode, on power up start accepting commands then:

       0x00: Send all data in text mode once.

       0x01: Send all data In Binary mode once.

       0x02: Send all data In IEEE mode once.

       0x10: Send all data in text mode a loop until AutoStart is not
0x5A.

       0x11: Send all data In Bin. mode in a loop until AutoStart is not
0x5A.

       0x12: Send All Data In IEEE mode in a loop until AutoStart is not
0x5A.

09 - User power on baud rate lock (if not 0x5A sensor will use 9600 Baud).

10 - User power on baud rate. --Use With Caution—

Target Baud	Const # 10	Target Baud	Const # 10
75	37	1200	33
150	36	2400	32
300	35	4800	31
600	34	9600	30

11 - Not Used

12 - Lowest Calibration Temp. in counts. LSB

13 - Lowest Calibration Temp. in counts. MSB

14 - Distance in counts between Calibration Points LSB

15 - Distance in counts between Calibration Points MSB

16 - Number of Temp. Points in the SEEROM.

17 - The Table Magnetometer Offset Scale Value.
      (Offset = TableOffset / $2^{<Byte\#18>}$)

18 - The Table Accelerometer Offset Scale Value.
      (Offset = TableOffset / $2^{<Byte\#19>}$)

19 - The Table Mag Scale Value. (Scale = TableScale / $2^{<Byte\#20>}$)

20 - The Table Acc Scale Value. (Scale = TableScale / $2^{<Byte\#21>}$)

21 - The Table Mag Ortho Scale Value. (Ortho = TableOrtho/$2^{<Byte\#22>}$)

22 - The Table Acc Ortho Scale Value. (Ortho = TableOrtho/ $2^{<Byte\#23>}$)

23 – Not Used

24 - Product I.D. String char #1

25 - Product I.D. String char #2

26 - Product I.D. String char #3

27 - Product I.D. String char #4

28 - Product I.D. String char #5

29 - Product I.D. String char #6

30 - Product I.D. String char #7

31 - Product I.D. String char #8
32 - Product I.D. String char #9
33 - Product I.D. String char #10
34 - Constant 0
35 - RTS Delay is the time in ~2 millisecond units that the RTS line is high
before data is sent. Any value but 0 will slow the data rate by the same
amount.
36 - Number of chars in Address minus 1(0:1char 1:2chars 2:3chars)
37 - Address char #1 <'0' - 30>
38 - Address char #2 <'B' - 42>
39 - Address char #3 <' ' - 20>

7 Appendix B. Warranty and Support Information

7.1 Customer Service

As a Crossbow Technology customer you have access to product support services, which include:

- Single-point return service
- Web-based support service
- Same day troubleshooting assistance
- Worldwide Crossbow representation
- Onsite and factory training available
- Preventative maintenance and repair programs
- Installation assistance available

7.2 Contact Directory

United States: Phone: 1-408-965-3300 (7 AM to 7 PM PST)

Fax: 1-408-324-4840 (24 hours)

Email: techsupport@xbow.com

Non-U.S.: refer to website www.xbow.com

7.3 Return Procedure

7.3.1 Authorization

Before returning any equipment, please contact Crossbow to obtain a Returned Material Authorization number (RMA).

Be ready to provide the following information when requesting a RMA:

- Name
- Address
- Telephone, Fax, Email
- Equipment Model Number
- Equipment Serial Number
- Installation Date
- Failure Date
- Fault Description

7.3.2   Identification and Protection

If the equipment is to be shipped to Crossbow for service or repair, please attach a tag TO THE EQUIPMENT, as well as the shipping container(s), identifying the owner. Also indicate the service or repair required, the problems encountered, and other information considered valuable to the service facility such as the list of information provided to request the RMA number.

Place the equipment in the original shipping container(s), making sure there is adequate packing around all sides of the equipment. If the original shipping containers were discarded, use heavy boxes with adequate padding and protection.

7.3.3   Sealing the Container

Seal the shipping container(s) with heavy tape or metal bands strong enough to handle the weight of the equipment and the container.

7.3.4   Marking

Please write the words, "***FRAGILE, DELICATE INSTRUMENT***" in several places on the outside of the shipping container(s). In all correspondence, please refer to the equipment by the model number, the serial number, and the RMA number.

7.3.5   Return Shipping Address

Use the following address for all returned products:

Crossbow Technology, Inc.
    41 E. Daggett Drive
    San Jose, CA 95134
    Attn: RMA Number (XXXXXX)

7.4   Warranty

The Crossbow product warranty is one year from date of shipment.

SUNSTAR 商斯达实业集团是集研发、生产、工程、销售、代理经销 、技术咨询、信息服务等为一体的高科技企业，是专业高科技电子产品生产厂家，是具有 10 多年历史的专业电子元器件供应商，是中国最早和最大的仓储式连锁规模经营大型综合电子零部件代理分销商之一,是一家专业代理和分销世界各大品牌 IC 芯片和電子元器件的连锁经营綜合性国际公司,专业经营进口、国产名厂名牌电子元件，型号、种类齐全。在香港、北京、深圳、上海、西安、成都等全国主要电子市场设有直属分公司和产品展示展销窗口门市部专卖店及代理分销商,已在全国范围内建成强大统一的供货和代理分销网络。 我们专业代理经销、开发生产电子元器件、集成电路、传感器、微波光电元器件、工控机/DOC/DOM 电子盘、专用电路、单片机开发、**MCU/DSP/ARM/FPGA** 软件硬件、二极管、三极管、模块等,是您可靠的一站式现货配套供应商、方案提供商、部件功能模块开发配套商。商斯达实业公司拥有庞大的资料库,有数位毕业于著名高校——有中国电子工业摇篮之称的西安电子科技大学（西军电）并长期从事国防尖端科技研究的高级工程师为您精挑细选、量身订做各种高科技电子元器件,并解决各种技术问题。

更多产品请看本公司产品专用销售网站:

商斯达中国传感器科技信息网：http://www.sensor-ic.com/

商斯达工控安防网：http://www.pc-ps.net/

商斯达电子元器件网：http://www.sunstare.com/

商斯达微波光电产品网:HTTP://www.rfoe.net/

商斯达消费电子产品网://www.icasic.com/

商斯达实业科技产品网://www.sunstars.cn/

传感器销售热线：

地址：深圳市福田区福华路福庆街鸿图大厦 1602 室

电话：0755-83370250 83376489 83376549 83607652 83370251　82500323

传真：0755-83376182　（0）13902971329　MSN：SUNS8888@hotmail.com

邮编：518033　E-mail:szss20@163.com　　QQ：195847376

深圳赛格展销部：深圳华强北路赛格电子市场 2583 号 电话：0755-83665529　25059422

技术支持：0755-83394033 13501568376

欢迎索取免费详细资料、设计指南和光盘 ；产品凡多，未能尽录，欢迎来电查询。

北京分公司：北京海淀区知春路 132 号中发电子大厦 3097 号

TEL：010-81159046 82615020 13501189838 FAX：010-62543996

上海分公司：上海市北京东路 668 号上海赛格电子市场 2B35 号

TEL：021-28311762 56703037 13701955389 FAX：021-56703037

西安分公司：西安高新开发区 20 所(中国电子科技集团导航技术研究所)

西安劳动南路 88 号电子商城二楼 D23 号

TEL：029-81022619 13072977981 FAX:029-88789382