

# Spaceball® SpaceOrb 360® Protocol

Revision: 1.00  
Created: May 7, 1998  
Revised: May 7, 1998  
Author: John Hilton  
Filename: hwdev\sw\reference\SoPrctl.doc

Copyright © 1995-1998, Spacetec IMC Corporation.

## CONFIDENTIAL

This document contains confidential information that is proprietary to Spacetec IMC Corporation. The information contained herein may not be copied in whole or part without the prior written permission of Spacetec IMC Corporation.

Spaceball, Spaceware, SpaceController and SpaceOrb 360 are registered trademarks of Spacetec IMC Corporation.

## Introduction

The SpaceOrb 360 Protocol is designed to provide a minimal communications protocol. Notable changes from the 2003 and the SpaceController protocol are the lower ball data resolution and the elimination of the beeper.

## Communications' Settings

The SpaceOrb 360 communicates using an RS232-C interface at 9600 baud, 8 data bits, no parity and one stop bit. There is no flow control. Five lines are used; transmit, receive, signal ground, RTS and DTR. Receive, RTS and DTR are used for power.

## Data Packets

Data generated by the SpaceOrb 360 consists of a header byte, zero or more data bytes and an xor byte. The header byte is a printable character while the data bytes and the xor byte have their top bit set.

The seven SpaceOrb 360 generated packet types are:

- the reset packet

  - R Spaceball (R) Vx.xx dd-mmm-yy Copyright (C) yyyy<xor>

  - E.g. R Spaceball (R) V4.34 19-Oct-96 Copyright (C) 1996<xor>

- the ball data packet

  - D<button status><D1><D2><D3><D4><D5><D6><D7><D8><D9><xor>

- the button data packet

  - K<period><button status><reserved><xor>

- the error packet

  - E<flags><reserved><xor>

- the null region packet

  - N<null region><xor>

- the terminator packet

  - <CR>

the information packets

!1 Spaceball (R) V4.34 19-Oct-96 Copyright (C) 1996<xor>

!2 11.52N 0.2557Nm 10bit<xor>

Commands generated by the host consist of a header byte (top bit clear) and zero or more data bytes (top bit set).

The six host generated packet types are:

the query packet

?

the pulse packet

P<Pulse hi><Pulse lo>

the ball data request packet

d

the null region request packet

n

the null region data packet

N<null region>

the terminator packet

<CR>

## Packet Termination

A packet is terminated by either the beginning of the next packet or the terminator packet. The terminator packet is generated only when there are no other packets ready to be sent.

## Power-up & Rezero Operation

When the SpaceOrb 360 is powered up it performs the following operations:

- CPU test
- RAM test
- program ROM checksum
- calibration EEPROM checksum
- circuit operation tests
- set the default settings
- rezero (set the ball's "rest" position as the current position)
- wait 1 second then send <CR> and the reset packet

If any faults occurred an error packet will be generated following the reset packet. The reset packet has the form:

R Spaceball (R) Vx.xx dd-mmm-yy Copyright (C) yyyy<xor>

where

x.xx is the firmware version number; eg. "4.02"

dd-mmm-yy is the date; eg. "03-Sep-94"

yyyy is the year; eg. "1994"

When the rezero button is activated the SpaceOrb 360 reloads the EEPROM values and rezeros the ball. The ball should be in its rest position (i.e. should not be touched) during the rezero.

Refer to the relevant packet descriptions in the *Packet Reference* section for detailed information.

## Ball Data Packet

Sensing of the force and torque is done *relative* to a rest, or zero, position which normally is just the weight of the ball. There are two situations when the rest position is set; when the unit is powered up and when the user presses the rezero button.

Force and torque values are passed as signed 10 bit numbers with a force and torque range of -16.4 to +16.4 N and -0.717 to +0.715 mNm. The standard orientation of the PowerSensor is with the stem vertical. Positive FrcX is to the right, +ve FrcY is up, +ve FrcZ is backwards. Positive torque values are clockwise as viewed from the center of the ball out along the positive force axes. The ball data packet has the form:

D<button status><D1><D2><D3><D4><D5><D6><D7><D8><D9><xor>

where

<button status> is a copy of the last button status returned in a button data packet

<D1> to <D9> are data bytes containing the force and torque components. These values are generated by first packing the six 10 bit values then xoring them with the characters

"SpaceWare". Packed data has the form:

1XXX XXXX	1XXX YYYY	1YYY YYYZ	1ZZZ ZZZZ
1ZZx xxxx	1xxx xxyy	1yyy yyyy	1yzz zzzz
1zzz z000			

where

XXXXXXXXXX, YYYYYYYYYY, ZZZZZZZZZZ are the components of force.

xxxxxxxxxx, yyyyyyyyyy, zzzzzzzzzz are the components of torque.

The force components are calibrated to a resolution of 32 mN and the torque components are calibrated to a resolution of 1.4 mNm. The button status is repeated in the ball data packet for robustness. Note the "at rest" packet, the packet with all force and torque components at zero, will contain the word "SpaceWare".

DIAGRAM TO BE COMPLETED

Ball data packets are generated at a rate of specified by the pules setting except when the ball is in its rest position. The default pulse setting is 33 msec which corresponds to 30 Hz. Releasing the ball and allowing it to return to its rest position results in the applied force and torque returning to zero. Only one packet containing zero force and torque values will be transmitted. No more ball data packets will be generated until the ball is once again activated or a ball data request packet is received. If a ball data request packet is received one more zero packet will be generated. Internally the SpaceOrb 360 will continue to sense the ball at the pulse rate.

Setting the pulse to 0 will stop the SpaceOrb 360 from taking ball readings, reducing the power consumption somewhat. When the pulse is at 0 a ball data request packet will result in the SpaceOrb 360 generating a ball data packet.

## Button Data Packet

The SpaceOrb 360 has seven buttons; A, B, C, D, E, F and rezero. The rezero button is a special button used to instigate a rezero operation. Its state is returned for diagnostic purposes and it is not normally used by applications.

A button data packet has the form:

K<period><status><reserved><xor>

where

<period> is the number of milliseconds divided by 10 since the last button data packet

<status> has the form:

1<rezero><F><E><D><C><B><A>

where a 1 indicates the button is pressed.

<reserved> is reserved for future use.

A button data packet is generated at a rate of 1 Hz or whenever any button changes state. Therefore a program can detect when the SpaceOrb 360 has been disconnected if no packets arrive within 1 second.

## Error Packet

An error packet is generated whenever one or more errors occur and has the form:

E<flags><reserved><xor>

where

<flags> is a bit mask of error flags of the form:

1<r><r><r><r><brown-out><EEPROM checksum error><hardware fault>

where

<r> are reserved flags

<brown-out> occurs when a power brown-out was detected

<eprom checksum error> occurs if the EEPROM checksum test failed

<hardware fault> occurs if a hardware fault was detected

a 1 indicates the error occurred.

<reserved> is reserved for future use.

## Query and Information Packets

Sending a query packet:

?<CR>

to the SpaceOrb 360 results in a response of two information packets:

!1 Spaceball (R) Vx.xx dd-mmm-yy Copyright (C) yyyy<xor>

!2 20.48N 0.5632Nm 8bit<xor>

The information packets identify the device, giving the firmware version and creation date, the maximum force and torque values and the internal resolution.

## Pulse Packet

To set the period between successive ball data packets send a pulse packet of the form:

P<Pulse hi><Pulse lo><CR>

where

<Pulse hi> and <Pulse lo> have the form:

**1000 pppp 1ppp pppp**

where

**pppppppppppp** is an 11 bit pulse value

The pulse value is the number of milliseconds between successive ball data packets, defaulting to 33 msec which corresponds to 30 Hz.

There is no way to query the pulse setting.

## Null Region Packet

The null region specifies the threshold below which forces and torques are set to zero. It can be considered to specify the minimum detectable force and torque. The default factory setting is normally adequate for all uses. A higher setting will make the unit less susceptible to drift (due to small forces and torques) but will also make the ball less responsive to light efforts. Lower settings will make drift more likely but will also make the ball more sensitive at the low end. The null region is an internal number that relates to the square of the lengths of the force and torque vectors. It is applied to each vector independently. The ball is considered to be at rest when both the force and the torque vectors are less than the null region.

To set the null region a packet of the form:

N<null region><CR>

where

<null region> is a seven bit value.

The SpaceOrb 360 will respond with

N<null region><xor>

To query the null region send:

n<CR>

The SpaceOrb 360 will respond with

N<null region><xor>

## Ball Data Request Packet

Sending a ball data request packet:

d<CR>

has no effect when the ball is not at rest and the pulse setting is not 0. If the ball is at rest the ball request data packet causes another zero packet to be generated. If the pulse setting is 0 a ball data packet is immediately generated.

## **Revision History**

7-MAY-98 JAH 1.0

Copied and edited from spptcol.doc