# *SiRF Binary Protocol Reference Manual*

SiRF Technology, Inc.
217 Devcon Drive
San Jose, CA 95112 U.S.A.
Phone: +1 (408) 467-0410
Fax: +1 (408) 467-0420
www.sirf.com

Part Number: 1050-0041
Revision 2.3, December 2007

# *SiRF Binary Protocol Reference Manual*



## *About This Document*

This document contains information on SiRF products. SiRF Technology, Inc. reserves the right to make changes in its products, specifications and other information at any time without notice. SiRF assumes no liability or responsibility for any claims or damages arising out of the use of this document, or from the use of integrated circuits based on this document, including, but not limited to claims or damages based on infringement of patents, copyrights or other intellectual property rights. SiRF makes no warranties, either express or implied with respect to the information and specifications contained in this document. Performance characteristics listed in this data sheet do not constitute a warranty or guarantee of product performance. All terms and conditions of sale are governed by the SiRF Terms and Conditions of Sale, a copy of which you may obtain from your authorized SiRF sales representative.

## *Getting Help*

If you have any problems, contact your SiRF representative or call or send an e-mail to the SiRF Technology support group:

| | |
|---|---|
| phone | +1 (408) 467-0410 |
| e-mail | support@sirf.com |

# *Contents*

# *Tables*

# Preface

The *SiRF Binary Protocol Reference Manual* provides detailed information about the SiRF Binary protocol – the standard protocol used by the SiRFstar family of products.

## Who Should Use This Guide

This manual was written assuming the user is familiar with serial communications interface protocols, including their definitions and use.

## How This Guide Is Organized

**Chapter 1, “Protocol Layers”** information about SiRF Binary protocol layers.

**Chapter 2, “Input Messages”** definitions and examples of each available SiRF Binary input messages.

**Chapter 3, “Output Messages”** definitions and examples of each available SiRF Binary output messages.

**Chapter 4, “Additional Information”** Other useful information pertaining to the SiRF Binary protocol.

## Related Manuals

You can also refer to the following literature for additional information:

- *SiRF NMEA Reference Manual*
- *ICD-GPS-200*
- *RTCM Recommended Standards for Differential GNSS*

## *Troubleshooting/Contacting SiRF Technical Support*

Address:

SiRF Technology Inc.
217 Devcon Drive
San Jose, CA 95112 U.S.A.

SiRF Technical Support:

Phone: +1 (408) 467-0410 (9 am to 5 pm Pacific Standard Time)
E-mail: support@sirf.com

General enquiries:

Phone: +1 (408) 467-0410 (9 am to 5 pm Pacific Standard Time)
E-mail: gps@sirf.com

## *Helpful Information When Contacting SiRF Technical Support*

Receiver Serial Number: ____________________

Receiver Software Version: _________________

SiRFDemo Version: _________________________

# Protocol Layers 1

SiRF Binary protocol is the standard interface protocol used by the SiRFstar family of products.

This serial communication protocol is designed to include:

- Reliable transport of messages
- Ease of implementation
- Efficient implementation
- Independence from payload

## Transport Message

| Start Sequence | Payload Length | Payload | Message Checksum | End Sequence |
|---|---|---|---|---|
| 0xA0[1], 0xA2 | Two-bytes (15-bits) | Up to $2^{10}$ -1(<1023) | Two-bytes (15-bits) | 0xB0, 0xB3 |

1. Characters preceded by “0x” denotes a hexadecimal value. 0xA0 equals 160.

## Transport

The transport layer of the protocol encapsulates a GPS message in two start-of-message characters and two end-of message characters. The values are chosen to be easily identifiable and unlikely to occur frequently in the data. In addition, the transport layer prefixes the message with a 2-byte (15-bit) message length, and adds a 2-byte (15-bit) checksum before the two stop characters. The values of the start and stop characters and the choice of a 15-bit value for length and checksum ensure message length and checksum cannot alias with either the stop or start code.

## Message Validation

The validation layer is of part of the transport, but operates independently. The byte count refers to the payload byte length. The checksum is a sum on the payload.

## Payload Length

The payload length is transmitted high order byte first followed by the low byte.

| High Byte | Low Byte |
|---|---|
| ≤ 0x7F | Any value |

Even though the protocol has a maximum length of ($2^{15}$-1) bytes, practical considerations require the SiRF GPS module implementation to limit this value to a smaller number. The SiRF receiving programs such as SiRFDemo, may limit the actual size to something less than this maximum.

## Payload Data

The payload data follows the payload length. It contains the number of bytes specified by the payload length. The payload data may contain any 8-bit value.

Where multi-byte values are in the payload data, neither the alignment nor the byte order are defined as part of the transport although SiRF payloads uses the big-endian order.

The Message ID tables in Chapter 2, “Input Messages” and Chapter 3, “Output Messages” describe the payload data, variable length, and variable data type. The Bytes column contains a number that specifies the number of bytes in each field of the message, and a letter that describes how to interpret the value. The letters and their description are shown in Table 1-1.

*Table 1-1* Data Types in Bytes Field of Message ID Tables

| Letter | Description |
|---|---|
| D | Discrete –The field consists of a bit mapped value, or subfields of groups of bits that are described in the Description field. Values should be considered unsigned |
| S | Signed – The field contains a signed integer value in two’s complement format |
| U | Unsigned – The field contains an unsigned integer value |
| Dbl | Double precision floating point – See Note after Table 3-68 on page 32 for a detailed description of this data type |
| Sgl | Single precision floating point – See Note after Table 3-68 on page 32 for a detailed description of this data type |

## Checksum

The checksum is transmitted high order byte first followed by the low byte. This is the so-called big-endian order.

| High Byte | Low Byte |
|---|---|
| ≤ 0x7F | Any value |

The checksum is 15-bit checksum of the bytes in the payload data. The following pseudo code defines the algorithm used.

Let message be the array of bytes to be sent by the transport.

Let msgLen be the number of bytes in the message array to be transmitted.

```
Index = first
checkSum = 0
while index < msgLen
    checkSum = checkSum + message[index]
checkSum = checkSum AND (2^15-1).
    increment index
```

# Input Messages 2

The following chapter provides full information about available SiRF Binary input messages. For each message, a full definition and example is provided.

**Note –** The input message buffer size limit is 912 bytes.

Table 2-1 describes the message list for the SiRF Binary input messages. Table 2-2 provides the Message Sub IDs for SiRFDRive Input Message ID 172 (0xAC). Table 2-3 provides information about which message is supported by which software.

*Table 2-1* SiRF Messages – Input Message List

| Hex | Decimal | Name | Description |
|---|---|---|---|
| 35 | 53 | Advanced Power Management | Power management scheme for SiRFLoc & SiRFXTrac |
| 80 | 128 | Initialize Data Source | Receiver initialization and associated parameters |
| 81 | 129 | Switch to NMEA Protocol | Enable NMEA messages, output rate, & bit rate |
| 82 | 130 | Set Almanac (upload) | Sends an existing almanac file to the receiver |
| 83 | 131 | Handle Formatted Dump Data | Outputs formatted data |
| 84 | 132 | Poll Software Version | Polls for the loaded software version |
| 85 | 133 | DGPS Source Control | DGPS correction source & beacon receiver information |
| 86 | 134 | Set Binary Serial Port | bit rate, data bits, stop bits, and parity |
| 87 | 135 | Set Protocol | Switches protocol |
| 88 | 136 | Mode Control | Navigation mode configuration |
| 89 | 137 | DOP Mask Control | DOP mask selection and parameters |
| 8A | 138 | DGPS Mode | DGPS mode selection and timeout value |
| 8B | 139 | Elevation Mask | Elevation tracking and navigation masks |
| 8C | 140 | Power Mask | Power tracking and navigation masks |
| 8F | 143 | Static Navigation | Configuration for static operation |
| 90 | 144 | Poll Clock Status | Polls the clock status |
| 91 | 145 | Set DGPS Serial Port | DGPS port bit rate, data bits, stop bits, & parity |
| 92 | 146 | Poll Almanac | Polls for almanac data |
| 93 | 147 | Poll Ephemeris | Polls for ephemeris data |
| 94 | 148 | Flash Update | On the fly software update |
| 95 | 149 | Set Ephemeris (upload) | Sends an existing ephemeris to the receiver |
| 96 | 150 | Switch Operating Mode | Test mode selection, SV ID, and period. |
| 97 | 151 | Set TricklePower Parameters | Push to fix mode, duty cycle, and on time |
| 98 | 152 | Poll Navigation Parameters | Polls for the current navigation parameters |

*Table 2-1* SiRF Messages – Input Message List *(Continued)*

| Hex | Decimal | Name | Description |
|---|---|---|---|
| A5 | 165 | Set UART Configuration | Protocol selection, bit rate, data bits, stop bits, & parity |
| A6 | 166 | Set Message Rate | SiRF Binary message output rate |
| A7 | 167 | Set Low Power Acquisition Parameters | Low power configuration parameters |
| A8 | 168 | Poll Command Parameters | Poll for parameters:<br>0x80: Receiver initialized & associated params<br>0x85: DGPS source and beacon receiver info<br>0x88: Navigation mode configuration<br>0x89: DOP mask selection and parameters<br>0x8A: DGPS mode selection and timeout values<br>0x8B: Elevation tracking and navigation masks<br>0x8C: Power tracking and navigation masks<br>0x8F: Static navigation configuration<br>0x97: Low power parameters |
| AA | 170 | Set SBAS Parameters | SBAS configuration parameters |
| AC | 172 | SiRF Dead Reckoning Class of Input Messages | The Message ID is partitioned into messages identified by Message Sub IDs. Refer to Table 2-2. |
| AF | 175 | User Input Command | User settable input command string and parser. |
| B4-C7 | 180-199 | MID_UserInputBegin – MID_UserInputEnd | Available for SDK user input messages only. |
| B4 | 180 | Preset Software Configuration | Selection of the Preset Software Configurations as defined in bits [3:2] of the GSC2xr chip configuration register |
| B6 | 182 | Set UART Configuration | Obsolete. |
| CD | 205 | Software Control | Generic Software Input Message |
| E4 | 228 | SiRF internal message | Reserved |
| E8 | 232 | Extended Ephemeris Proprietary | Extended Ephemeris and Debug Flag |

*Table 2-2* Message Sub IDs for SiRF Dead Reckoning Input Message ID 172 (0xAC)

| Sub ID | Message | Supports SiRFDRive | Supports SiRFDiRect |
|---|---|---|---|
| 1 | Initialize GPS/DR Navigation | Yes | Yes |
| 2 | Set GPS/DR Navigation Mode | Yes | Yes |
| 3 | Set DR Gyro Factory Calibration | Yes, (SiRFDRive 1 only) | No |
| 4 | Set DR Sensors’ Parameters | Yes, (SiRFDRive 1 only) | No |
| 5 | Poll DR Validity (not implemented) | No | No |
| 6 | Poll DR Gyro Factory Calibration | Yes, (SiRFDRive 1 only) | No |
| 7 | Poll DR Sensors’ Parameters | Yes, (SiRFDRive 1 only) | No |
| 9 | Input Car Bus Data | Yes, (SiRFDRive 1.5 and 2) | No |
| 10 | Car Bus Enabled | Yes, (SiRFDRive 2 only) | No |
| 11 | Car Bus Disabled | Yes, (SiRFDRive 2 only) | No |
| 14 | Input Car Bus Data 2[1] | No | Yes |

1. Output message only at this time.

SiRF Binary protocol is an evolving standard along with continued development of SiRF software and GPS solutions, not all SiRF Binary messages are supported by all SiRF GPS solutions.

Table 2-3 identifies the supported input messages for each SiRF architecture.

*Table 2-3* Supported Input Messages

| | SiRF Software Options | | | | | |
|---|---|---|---|---|---|---|
| **Message ID** | **GSW2** | **SiRFDRive** | **SiRFXTrac** | **SiRFLoc** | **GSW3 & GSWLT3** | **SiRFDiRect** |
| 53 | No | No | Yes | No | No | No |
| 128 | Yes | Yes | Yes | Yes | Yes | Yes |
| 129 | Yes | Yes | Yes | No | Yes | Yes |
| 130 | Yes | Yes | No | No | No | Yes |
| 131 | No | No | No | No | Yes | Yes |
| 132 | Yes | Yes | Yes | Yes | Yes | Yes |
| 133 | Yes | Yes | No | No | Yes | Yes |
| 134 | Yes | Yes | Yes | Yes | Yes | Yes |
| 135 | No | No | No | No | Yes | Yes |
| 136 | Yes | Yes | Yes | Yes | Yes | Yes |
| 137 | Yes | Yes | Yes | Yes | Yes | Yes |
| 138 | Yes | Yes | Yes | Yes | Yes | No |
| 139 | Yes | Yes | Yes | Yes | Yes | Yes |
| 140 | Yes | Yes | Yes | Yes | Yes | Yes |
| 143 | Yes | Yes | Yes | Yes | Yes | Yes |
| 144 | Yes | Yes | Yes | Yes | Yes | Yes |
| 145 | Yes | Yes | No | No | No | Yes |
| 146 | Yes | Yes | No | Yes | Yes | Yes |
| 147 | Yes | Yes | No | Yes | Yes | Yes |
| 148 | Yes | Yes | Yes | No | Yes | Yes |
| 149 | Yes | Yes | No | Yes | No | Yes |
| 150 | Yes | Yes | Yes | Yes | Yes | Yes |
| 151 | Yes | Yes | Yes | No | Yes | Yes |
| 152 | Yes | Yes | Yes | Yes | Yes | Yes |
| 165 | Yes | Yes | Yes | No | Yes | Yes |
| 166 | Yes | Yes | Yes | Yes | Yes | Yes |
| 167 | Yes | Yes | Yes | No | Yes | Yes |
| 168 | Yes | Yes | Yes | Yes | Yes | Yes |
| 170 | 2.3 or above | Yes | No | No | Yes | No |
| 172 | No | Yes[1] | No | No | No | Yes[1] |
| 175 | No | No | No | No | Yes | Yes |
| 180[2] | Yes | No | No | No | No | No |
| 180-199 | Yes | Yes | Yes | Yes | Yes | No |
| 205 | No | No | No | No | 3.2.5 or above | No |
| 228 | No | No | No | No | Yes (reserved) | No |
| 232 | 2.5 or above | No | 2.3 or above | No | 3.2.0 or above | Yes |

1. Not all Message Sub IDs supported

2. Only with GSC2xr chip

## Advanced Power Management – Message ID 53

Implements Advanced Power Management (APM). APM allows power savings while ensuring that the quality of the solution is maintained when signal levels drop. APM does not engage until all information is received.

Example:

The following example sets the receiver to operate in APM mode with 0 cycles before sleep (continuous operation), 20 seconds between fixes, 50% duty cycle, a time between fixes priority, and no preference for accuracy.

A0A2000C—Start Sequence and Payload Length

350100140003070000A0100—Payload

005FB0B3—Message Checksum and End Sequence

*Table 2-4* Advanced Power Management – Message ID 53

| Name | Bytes | Binary (Hex) Scale | Binary (Hex) Example | Unit | Description |
|---|---|---|---|---|---|
| Message ID | 1 | | 35 | | Decimal 53 |
| APM Enabled | 1 | | 01 | | 1 = True, 0 = False |
| Number Fixes | 1 | | 00 | | Number of requested APM cycles. Range 0-255[1] |
| Time Between Fixes | 1 | 1 | 14 | sec | Requested time between fixes. Range 0-255[2] |
| Spare Byte 1 | 1 | | 00 | | Reserved |
| Maximum Horizontal Error | 1 | | 03 | | Maximum requested horizontal error (See Table 2-5). |
| Maximum Vertical Error | 1 | | 07 | | Maximum requested vertical error (See Table 2-5) |
| Maximum Response Time | 1 | 1 | 00 | sec | Maximum response time. Not currently used |
| Time Acc Priority | 1 | | 00 | | 0x00 = No priority,<br>0x01 = Response Time Max has higher priority<br>0x02 = Horizontal Error Max has higher priority.<br>Not currently used. |
| Power Duty Cycle | 1 | 5 | 0A | % | Power Duty Cycle, defined as the time in full power to total operation time. 1->20; duty cycle (%) is this value *5.[3] |
| Time Duty Cycle | 1 | | 01 | | Time/Power Duty cycle priority.<br>0x01 = Time between two consecutive fixes has priority<br>0x02 = Power Duty cycle has higher priority. Bits 2..7 reserved for expansion. |
| Spare Byte 2 | 1 | | 00 | | Reserved |

Payload length: 12 bytes

1. A value of zero indicates that continuous APM cycles is requested.
2. It is bound from 10 to180 s.
3. If a duty-cycle of 0 is entered, it is rejected as out of range. If a duty cycle value of 20 is entered, the APM module is disabled and continuous power operation is resumed.

*Table 2-5* Horizontal/Vertical Error

| Value | Position Error |
|---|---|
| 0x00 | < 1 meter |
| 0x01 | < 5 meter |
| 0x02 | < 10 meter |
| 0x03 | < 20 meter |
| 0x04 | < 40 meter |
| 0x05 | < 80 meter |
| 0x06 | < 160 meter |
| 0x07 | No Maximum |
| 0x08 - 0xFF | Reserved |

## Initialize Data Source – Message ID 128

Causes the receiver to restart. Optionally, it can provide position, clock drift, and time data to initialize the receiver.

---

**Note –** Some software versions do not support use of the initializing data.

---

Table 2-6 contains the input values for the following example:

Command a Warm Start with the following initialization data: ECEF XYZ (-2686727 m, -4304282 m, 3851642 m), Clock Offset (75,000 Hz), Time of Week (86,400 sec), Week Number (924), and Channels (12). Raw track data enabled, Debug data enabled.

Example:

A0A20019—Start Sequence and Payload Length

80FFD700F9FFBE5266003AC57A000124F80083D600039C0C33—Payload

0A91B0B3—Message Checksum and End Sequence

*Table 2-6* Initialize Data Source – Message ID 128

| | | Binary (Hex) | | | |
|---|---|---|---|---|---|
| Name | Bytes | Scale | Example | Unit | Description |
| Message ID | 1 U | | 80 | | Decimal 128 |
| ECEF X | 4 S | | FFD700F9 | meters | |
| ECEF Y | 4 S | | FFBE5266 | meters | |
| ECEF Z | 4 S | | 003AC57A | meters | |
| Clock Drift | 4 S | | 000124F8 | Hz | |
| Time of Week | 4 U | *100 | 0083D600 | sec | |
| Week Number | 2 U | | 51F | | Extended week number (0 - no limit) |
| Channels | 1 U | | 0C | | Range 1-12 |
| Reset Configuration Bit Map | 1 D | | 33 | | See Table 2-7 |

Payload length: 25 bytes

*Table 2-7* Reset Configuration Bit Map

| Bit | Description |
|---|---|
| 0 | Data valid flag: 1 = Use data in ECEF X, Y, Z, Clock Offset, Time of Week and Week number to initialize the receiver; 0 = Ignore data fields |
| 1 | Clear ephemeris from memory: blocks Snap or Hot Start from occurring |
| 2 | Clear all history (except clock drift) from memory: blocks Snap, Hot, and Warm Starts |
| 3 | Factory Reset: clears all GPS memory including clock drift. Also clears almanac stored in flash memory |
| 4 | Enable Nav Lib data (YES = 1, NO = 0)[1] |
| 5 | Enable debug data (YES = 1, NO = 0) |
| 6 | Indicate that Real-Time Clock (RTC) is not precise: blocks Snap Start |
| 7 | SiRFstarII = clear user data in memory; SiRFstarIII = perform full system reset |

1. If Nav Lib data are enabled, the resulting messages are enabled: Clock Status (Message ID 7), 50BPS (Message ID 8), Raw DGPS (Message ID 17), NL Measurement Data (Message ID 28), DGPS Data (Message ID 29), SV State Data (Message ID 30), and NL Initialized Data (Message ID 31). All messages sent at 1 Hz. If SiRFDemo is used to enable Nav Lib data, the bit rate is automatically set to 57600 by SiRFDemo.

## *Switch To NMEA Protocol – Message ID 129*

Switches a serial port from binary to NMEA protocol and sets message output rates and bit rate on the port.

Table 2-8 contains the input values for the following example:

Request the following NMEA data at 9600 bits per second:
GGA – ON at 1 sec, GLL – OFF, GSA – ON at 1sec,
GSV – ON at 5 sec, RMC – ON at 1sec, VTG-OFF, MSS – OFF, ZDA-OFF.

Example:

A0A20018—Start Sequence and Payload Length

810201010001010105010101000100010001000100012580—Payload

013AB0B3—Message Checksum and End Sequence

*Table 2-8* Switch To NMEA Protocol – Message ID 129

| Name | Bytes | Example | Unit | Description |
|---|---|---|---|---|
| Message ID | 1 U | 0x81 | | Decimal 129 |
| Mode | 1 U | 0x02 | | See Table 2-9 |
| GGA Message[1] | 1 U | 0x01 | sec | See NMEA Protocol Reference Manual for format |
| Checksum[2] | 1 U | 0x01 | | Send checksum with GGA message |
| GLL Message | 1 U | 0x00 | sec | See NMEA Protocol Reference Manual for format |
| Checksum | 1 U | 0x01 | | |
| GSA Message | 1 U | 0x01 | sec | See NMEA Protocol Reference Manual for format |
| Checksum | 1 U | 0x01 | | |
| GSV Message | 1 U | 0x05 | sec | See NMEA Protocol Reference Manual for format |
| Checksum | 1 U | 0x01 | | |
| RMC Message | 1 U | 0x01 | sec | See NMEA Protocol Reference Manual for format |
| Checksum | 1 U | 0x01 | | |
| VTG Message | 1 U | 0x00 | sec | See NMEA Protocol Reference Manual for format |
| Checksum | 1 U | 0x01 | | |
| MSS Message | 1 U | 0x00 | sec | Output rate for MSS message |
| Checksum | 1 U | 0x01 | | |
| Unused Field[3] | 1 U | 0x00 | | |

*Table 2-8* Switch To NMEA Protocol – Message ID 129 *(Continued)*

| Name | Bytes | Example | Unit | Description |
|---|---|---|---|---|
| Unused Field[3] | 1 U | 0x00 | | |
| ZDA Message | 1 U | 0x00 | sec | See NMEA Protocol Reference Manual for format |
| Checksum | 1 U | 0x01 | | |
| Unused Field[3] | 1 U | 0x00 | | |
| Unused Field[3] | 1 U | 0x00 | | |
| Bit Rate | 2 U | 0x2580 | | 1200, 2400, 4800, 9600, 19200, 38400, and 57600 |

Payload length: 24 bytes

1. A value of 0x00 implies not to send message, otherwise data is sent at 1 message every X seconds requested (e.g., to request a message to be sent every 5 seconds, request the message using a value of 0x05). Maximum rate is 1/255 sec.
2. A value of 0x00 implies the checksum is not transmitted with the message (not recommended). A value of 0x01 has a checksum calculated and transmitted as part of the message (recommended).
3. These fields are available if additional messages have been implemented in the NMEA protocol.

*Table 2-9* Mode Values

| Value | Meaning |
|---|---|
| 0 | Enable NMEA debug messages |
| 1 | Disable NMEA debug messages |
| 2 | Do not change last-set value for NMEA debug messages |

In TricklePower mode, update rate is specified by the user. When switching to NMEA protocol, the message update rate is also required. The resulting update rate is the product of the TricklePower update rate and the NMEA update rate (e.g., TricklePower update rate = 2 seconds, NMEA update rate = 5 seconds, resulting update rate is every 10 seconds, (2 X 5 = 10)).

**Note –** To return to the SiRF Binary protocol, send a SiRF NMEA message to revert to SiRF binary mode. (See the *SiRF NMEA Reference Manual* for more information).

# *Set Almanac – Message ID 130*

Enables the user to upload an almanac file to the receiver.

**Note –** Some software versions do not support this command.

Example:

A0A20381 – Start Sequence and Payload Length

82xx………………….    – Payload

xxxxB0B3 – Message Checksum and End Sequence

*Table 2-10* Set Almanac – Message ID 130

| | | Binary (Hex) | | | |
|---|---|---|---|---|---|
| Name | Bytes | Scale | Example | Unit | Description |
| Message ID | 1 U | | 82 | | Decimal 130 |
| Almanac[448] | 2 S | | 00 | | Reserved |

Payload length: 897 bytes

The almanac data is stored in the code as a 448-element array of INT16 values. These elements are partitioned as a 32 x 14 two-dimensional array where the row represents the satellite ID minus 1 and the column represents the number of INT16 values associated with this satellite. The data is actually packed and the exact format of this representation and packing method can be extracted from the *ICD-GPS-200* document. The *ICD-GPS-200* document describes the data format of each GPS navigation sub-frame and is available on the web at http://www.arinc.com/gps.

## *Handle Formatted Dump Data – Message ID 131*

Requests the output of formatted data from anywhere within the receiver's memory map. It is designed to support software development and can handle complex data types up to an array of structures. Message ID 10 Error 255 is sent in response to this message.

**Note –** The buffer size limit is 912 bytes.

Table 2-11 contains the input values for the following example. This example shows how to output an array of elements. Each element structure appears as follows:

```
Typedef structure // structure size = 9 bytes
{
    UINT8 Element 1
    UINT16 Element 2
    UINT8 Element 3
    UINT8 Element 4
    UINT32 Element 5
} tmy_struct
tmy_struc my_struct [3]
```

Example:

A0A2002B—Start Sequence and Payload Length

8303600001050050102010104486 56C6C6F0025326420253264202532642025326420 25313

02E316C660000—Payload

0867B0B3—Message Checksum and End Sequence

*Table 2-11* Handle Formatted Dump Data – Message Parameters

| Name | Bytes | Binary (Hex) Example | Unit | Description |
|---|---|---|---|---|
| Message ID | 1 U | 83 | | Decimal 131 |
| Elements | 1 U | 03 | | Number of elements in array to dump (minimum 1) |
| Data address | 4 S | 60000150 | | Address of the data to be dumped |
| Members | 1 U | 05 | | Number of items in the structure to be dumped |
| Member Size | Elements S | 01 02 01 01 04 | bytes | List of element sizes in the structure. See Table 2-12 for definition of member size (total of 5 for this example) |
| Header | string length + 1 S | "Hello"0 | | String to print out before data dump (total of 8 bytes in this example) |

*Table 2-11* Handle Formatted Dump Data – Message Parameters *(Continued)*

| Name | Bytes | Binary (Hex) Example | Unit | Description |
|---|---|---|---|---|
| Format | string length + 1 S | “%2d %2d %2d %2d %10.1lf”0 | | Format string for one line of output (total of 26 bytes in this example) with 0 termination |
| Trailer | string length + 1 S | 00 | | Not used |

Payload length: Variable

Table 2-12 defines the values associated with the member size data type.

*Table 2-12* Member Size Data Type

| Data Type | Value for Member Size (Bytes) |
|---|---|
| char, INT8, UINT8 | 1 |
| short int, INT16, UINT16, SINT16, BOOL16 | 2 |
| long int, float, INT32, UINT32, SINT32, BOOL32, FLOAT32 | 4 |
| long long, double INT64, DOUBLE64 | 8 |

## *Poll Software Version – Message ID 132*

Requests the output of the software version string. Message ID 6 is sent in response.

Table 2-13 contains the input values for the following example:

Poll the software version

Example:

A0A20002—Start Sequence and Payload Length

8400—Payload

0084B0B3—Message Checksum and End Sequence

*Table 2-13* Software Version – Message ID 132

| Name | Bytes | Binary (Hex) Scale | Binary (Hex) Example | Unit | Description |
|---|---|---|---|---|---|
| Message ID | 1 U | | 84 | | Decimal 132 |
| Control | 1 U | | 00 | | Not used |

Payload length: 2 bytes

## *DGPS Source – Message ID 133*

Allows the user to select the source for Differential GPS (DGPS) corrections. The default source is external RTCM SC-104 data on the secondary serial port. Options available are:

External RTCM SC-104 Data (any serial port)

Satellite Based Augmentation System (SBAS) – subject to SBAS satellite availability

Internal DGPS beacon receiver (supported only on specific GPS receiver hardware)

Example 1: Set the DGPS source to External RTCM SC-104 Data

A0A200007—Start Sequence and Payload Length

85020000000000—Payload

0087B0B3—Checksum and End Sequence

*Table 2-14* DGPS Source Selection (Example 1)

| Name | Bytes | Scale | Hex | Unit | Decimal | Description |
|---|---|---|---|---|---|---|
| Message ID | 1 U | | 85 | | 133 | Message Identification |
| DGPS Source | 1 U | | 02 | | 2 | See Table 2-16 |
| Internal Beacon Frequency | 4 U | | 00000000 | | 0 | Not used |
| Internal Beacon Bit Rate | 1 U | | 0 | | 0 | Not used |

Payload length: 7 bytes

Example 2: Set the DGPS source to Internal DGPS Beacon Receiver

Search Frequency 310000, Bit Rate 200

A0A200007—Start Sequence and Payload Length

85030004BAF0C802—Payload

02FEB0B3—Checksum and End Sequence

*Table 2-15* DGPS Source Selection (Example 2)

| Name | Bytes | Scale | Hex | Unit | Decimal | Description |
|---|---|---|---|---|---|---|
| Message ID | 1 U | | 85 | | 133 | Message Identification |
| DGPS Source | 1 U | | 03 | | 3 | See Table 2-16 |
| Internal Beacon Frequency | 4 U | | 0004BAF0 | Hz | 310000 | See Note 1 |
| Internal Beacon Bit Rate | 1 U | | C8 | BPS | 200 | See Note 2 |

Payload length: 7 bytes

**Note –** 1 – Beacon frequency valid range is 283500 to 325000 Hz. A value of zero indicates the Beacon should be set to automatically scan all valid frequencies.

**Note –** 2 – Bit rates may be 25, 50, 100 or 200 BPS. A value of zero indicates the Beacon should be set to automatically scan all bit rates.

*Table 2-16* DGPS Source Selections

| Value | DGPS Source | Description |
|---|---|---|
| 0 | None | DGPS corrections are not used (even if available) |
| 1 | SBAS | Uses SBAS Satellite (subject to availability) |
| 2 | External RTCM Data | External RTCM input source (e.g., Coast Guard Beacon) |
| 3 | Internal DGPS Beacon Receiver | Internal DGPS beacon receiver |
| 4 | User Software | Corrections provided using a module interface routine in a custom user application |

## Set Binary Serial Port – Message ID 134

Sets the serial port values that are used whenever the binary protocol is activated on a port. It also sets the current values for the port currently using the binary protocol. The values that can be adjusted are: bit rate, parity, data bits per character, stop bit length.

Table 2-17 contains the input values for the following example:

Set Binary serial port to 9600,n,8,1.

Example:

A0A20009—Start Sequence and Payload Length

860000258008010000—Payload

0134B0B3—Message Checksum and End Sequence

*Table 2-17* Set Main Serial Port – Message ID 134

| | | Binary (Hex) | | | |
|---|---|---|---|---|---|
| **Name** | **Bytes** | **Scale** | **Example** | **Unit** | **Description** |
| Message ID | 1 U | | 86 | | Decimal 134 |
| Bit Rate | 4 U | | 00002580 | | 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200 |
| Data Bits | 1 U | | 08 | | 8 |
| Stop Bit | 1 U | | 01 | | 1 = 1 Stop Bit |
| Parity | 1 U | | 00 | | None = 0, Odd = 1, Even = 2 |
| Pad | 1 U | | 00 | | Reserved |

Payload length: 9 bytes

## Set Protocol – Message ID 135

Switches the protocol to another protocol. For most software, the default protocol is SiRF binary. For SiRFstarIII software, refer to tCtrl_ProtocolEnum in ctrl_sif.h.

Table 2-18 contains the input values for the following example:

Set protocol to NMEA

Example:

A0A20002—Start Sequence and Payload Length

8702—Payload

0089B0B3—Message Checksum and End Sequence.

*Table 2-18* Set Protocol – Message ID 135

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | 87 | | Decimal 135 |
| Protocol[1] | 1 U | | 02 | | Null = 0<br>SiRF Binary = 1<br>NMEA = 2<br>ASCII = 3<br>RTCM = 4<br>USER1 = 5 (note1)<br>SiRFLoc = 6<br>Statistic = 7 |

Payload length: 2 bytes

1. Use caution when switching to User1 protocol. Use it only when User1 protocol supports switching back to SiRF Binary protocol.

**Note –** In any system only some of these protocols are present. Switching to a protocol that is not implemented may cause unpredictable results.

## *Mode Control – Message ID 136*

Sets up the navigation operations. It controls use of fewer than four satellites, and enables or disables the track smoothing filter. Using fewer than four satellites results in what is commonly called a ‘2-D’ fix. Four or more satellites allow a ‘3-D’ fix.

Table 2-19 contains the input values for the following example:

Alt Constraining = Yes, Degraded Mode = clock then direction
Altitude = 0, Alt Hold Mode = Auto, Alt Source = Last Computed,
Degraded Time Out = 5, DR Time Out = 2, Track Smoothing = Yes

Example:

A0A2000E—Start Sequence and Payload Length

880000010000000000000050201—Payload

0091B0B3—Message Checksum and End Sequence

*Table 2-19* Mode Control – Message ID 136

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | 88 | | Decimal 136 |
| Reserved | 2 U | | 0000 | | Reserved |
| Degraded Mode[1] | 1 U | | 01 | | Controls use of 2-SV and 1-SV solutions. See Table 2-20 |
| Reserved | 2 U | | 0000 | | Reserved |
| Altitude | 2 S | | 0000 | meters | User specified altitude, range -1,000 to +10,000 |
| Alt Hold Mode | 1 U | | 00 | | Controls use of 3-SV solution.See Table 2-21 |
| Alt Hold Source | 1 U | | 00 | | 0 = Use last computed altitude,<br>1 = Use user-input altitude |
| Reserved | 1 U | | 00 | | Reserved |

*Table 2-19* Mode Control – Message ID 136 *(Continued)*

| Name | Bytes | Binary (Hex) Scale | Binary (Hex) Example | Unit | Description |
|---|---|---|---|---|---|
| Degraded Time Out | 1 U | | 05 | sec | 0 = disable degraded mode, 1-120 seconds degraded mode time limit |
| DR Time Out | 1 U | | 02 | sec | 0 = disable dead reckoning, 1-120 seconds dead reckoning mode time limit |
| Track Smoothing | 1 U | | 01 | | 0 = disable, 1 = enable |

Payload length: 14 bytes

1. Degraded Mode is not supported in GSW3.2.5 and later. This field should be set to four in these software versions.

*Table 2-20* Degraded Mode

| Byte Value | Description |
|---|---|
| 0 | Allow 1-SV navigation, freeze direction for 2-SV fix, then freeze clock drift for 1-SV fix |
| 1 | Allow 1-SV navigation, freeze clock drift for 2-SV fix, then freeze direction for 1-SV fix |
| 2 | Allow 2-SV navigation, freeze direction. Does not allow 1-SV solution |
| 3 | Allow 2-SV navigation, freeze clock drift. Does not allow 1-SV solution |
| 4 | Do not allow Degraded Modes (2-SV and 1-SV navigation) |

**Note –** Degraded Mode is not supported in GSW3.2.5 and later. This field should be set to four in these software versions.

*Table 2-21* Altitude Hold Mode

| Byte Value | Description |
|---|---|
| 0 | Automatically determine best available altitude to use |
| 1 | Always use user-input altitude |
| 2 | Do not use altitude hold – Forces all fixes to be 3-D fixes |

# *DOP Mask Control – Message ID 137*

Dilution of Precision (DOP) is a measure of how the geometry of the satellites affects the current solution's accuracy. This message provides a method to restrict use of solutions when the DOP is too high. When the DOP mask is enabled, solutions with a DOP higher than the set limit is marked invalid. Table 2-22 contains the input values for the following example:

Auto PDOP/HDOP, GDOP = 8 (default), PDOP = 8,HDOP = 8

Example:

A0A20005—Start Sequence and Payload Length

8900080808—Payload

00A1B0B3—Message Checksum and End Sequence

*Table 2-22* DOP Mask Control – Message ID 137

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | 89 | | Decimal 137 |
| DOP Selection | 1 U | | 00 | | See Table 2-23 |
| GDOP Value | 1 U | | 08 | | Range 1 to 50 |
| PDOP Value | 1 U | | 08 | | Range 1 to 50 |
| HDOP Value | 1 U | | 08 | | Range 1 to 50 |

Payload length: 5 bytes

*Table 2-23* DOP Selection

| Byte Value | Description |
|---|---|
| 0 | Auto: PDOP for 3-D fix; HDOP for 2-D fix |
| 1 | PDOP |
| 2 | HDOP |
| 3 | GDOP |
| 4 | Do Not Use |

## DGPS Control – Message ID 138

Enables users to control how the receiver uses differential GPS (DGPS) corrections.

Table 2-24 contains the input values for the following example:

Set DGPS to exclusive with a time out of 30 seconds.

Example:

A0A20003—Start Sequence and Payload Length

8A011E—Payload

00A9B0B3—Message Checksum and End Sequence

*Table 2-24* DGPS Control – Message ID 138

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | 8A | | Decimal 138 |
| DGPS Selection | 1 U | | 01 | | See Table 2-25 |
| DGPS Time Out: | 1 U | | 1E | sec | Range 0 to 255 |

Payload length: 3 bytes

*Table 2-25* DGPS Selection

| Byte Value | Description |
|---|---|
| 0 | Auto = use corrections when available |
| 1 | Exclusive = include in navigation solution only SVs with corrections |
| 2 | Never Use = ignore corrections |

**Note –** DGPS Timeout interpretation varies with DGPS correction source. For internal beacon receiver or RTCM SC-104 external source, a value of 0 means infinite timeout (use corrections until another one is available). A value of 1-255 means use the corrections for a maximum of this many seconds. For DGPS corrections from an SBAS source, the timeout value is ignored unless Message ID 170, Flag bit 0 is set to 1 (User Timeout). If Message ID 170 specifies User Timeout, a value of 1 to 255 here means that SBAS corrections may be used for the number of seconds specified. A value of 0 means to use the timeout specified in the SBAS satellite message (usually 18 seconds).

## *Elevation Mask – Message ID 139*

Elevation mask is an angle above the horizon. Unless a satellite's elevation is greater than the mask, it is not used in navigation solutions. This message permits the receiver to avoid using the low-elevation-angle satellites most likely to have multipath problems.

Table 2-26 contains the input values for the following example:

Set Navigation Mask to 15.5 degrees (Tracking Mask is defaulted to 5 degrees).

Example:

A0A20005—Start Sequence and Payload Length

8B0032009B—Payload

0158B0B3—Message Checksum and End Sequence

*Table 2-26* Elevation Mask – Message ID 139

| | | **Binary (Hex)** | | | |
|---|---|---|---|---|---|
| **Name** | **Bytes** | **Scale** | **Example** | **Unit** | **Description** |
| Message ID | 1 U | | 8B | | Decimal 139 |
| Tracking Mask | 2 S | *10 | 0032 | degrees | Not implemented |
| Navigation Mask | 2 S | *10 | 009B | degrees | Range -20.0 to 90.0 |

Payload length: 5 bytes

**Note –** A satellite with an elevation angle that is below the specified navigation mask angle is not used in the navigation solution.

## *Power Mask – Message ID 140*

The power mask is a limit on which satellites are used in navigation solutions. Satellites with signals lower than the mask are not used.

Table 2-27 contains the input values for the following example:

Navigation mask to 33 dB-Hz (tracking default value of 28)

Example:

A0A20003—Start Sequence and Payload Length

8C1C21—Payload

00C9B0B3—Message Checksum and End Sequence

*Table 2-27* Power Mask – Message ID 140

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | 8C | | Decimal 140 |
| Tracking Mask | 1 U | | 1C | dBHz | Not implemented |
| Navigation Mask | 1 U | | 21 | dBHz | Range 20[1] to 50 |

Payload length: 3 bytes

1. The range for GSW3 and GSWLT3 is 12 to 50.

**Note –** Satellites with received signal strength below the specified navigation mask signal level are used in the navigation solution.

# *Static Navigation – Message ID 143*

Allows the user to enable or disable static navigation to the receiver.

Example:

A0A20002 – Start Sequence and Payload Length

8F01 – Payload

0090B0B3 – Message Checksum and End Sequence

*Table 2-28* Static Navigation – Message ID 143

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | 8F | | Decimal 143 |
| Static Navigation Flag | 1 U | | 01 | | 1 = enable; 0 = disable |

Payload length: 2 bytes

**Note –** Static navigation is a position filter designed to be used with applications intended for motor vehicles. When the vehicle's speed falls below a threshold, the position and heading are frozen, and speed is set to zero. This condition continues until the computed speed rises above 1.2 times the threshold or until the computed position is at least a set distance from the frozen place. The threshold speed and set distance may vary with software versions.

# *Poll Clock Status – Message ID 144*

Causes the receiver to report the most recently computed clock status. The resulting clock status is reported in Message ID 7.

Table 2-29 contains the input values for the following example:

Poll the clock status.

Example:

A0A20002—Start Sequence and Payload Length

9000—Payload

0090B0B3—Message Checksum and End Sequence

*Table 2-29* Clock Status – Message ID 144

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | 90 | | Decimal 144 |
| Control | 1 U | | 00 | | Not used |

Payload length: 2 bytes

**Note –** Returned message is Message ID 7. See "Response: Clock Status Data – Message ID 7" on page 3-8.

## *Set DGPS Serial Port – Message ID 145*

Sets the serial port settings associated with the RTCM SC-104 protocol. If the RTCM SC-104 protocol is currently assigned to a port, it also changes that port's settings. The values entered are stored in battery-backed RAM (called NVRAM in this document) and are used whenever the RTCM protocol is assigned to a port. The settings control: serial bit rate, parity, bits per character, stop bit length.

Table 2-30 contains the input values for the following example:

Set DGPS Serial port to 9600,n,8,1.

Example:

A0A20009—Start Sequence and Payload Length

910000258008010000—Payload

013FB0B3—Message Checksum and End Sequence

*Table 2-30* Set DGPS Serial Port – Message ID 145

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | 91 | | Decimal 145 |
| Bit Rate | 4 U | | 00002580 | | 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200 |
| Data Bits | 1 U | | 08 | | 8,7 |
| Stop Bit | 1 U | | 01 | | 0,1 |
| Parity | 1 U | | 00 | | None = 0, Odd = 1, Even = 2 |
| Pad | 1 U | | 00 | | Reserved |

Payload length: 9 bytes

**Note –** Setting the DGPS serial port using Message ID 145 affects COM-B only regardless of the port being used to communicate with the Evaluation Receiver.

## Poll Almanac – Message ID 146

Causes the most recently stored almanacs to be reported by the receiver. Almanacs are reported in Message ID 14, with a total of 32 messages being sent in response.

**Note –** Some software versions do not support this command.

Table 2-31 contains the input values for the following example:

Poll for the almanac.

Example:

A0A20002—Start Sequence and Payload Length

9200—Payload

0092B0B3—Message Checksum and End Sequence

*Table 2-31* Almanac – Message ID 146

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | 92 | | Decimal 146 |
| Control | 1 U | | 00 | | Not used |

Payload length: 2 bytes

**Note –** Returned message is Message ID 14. See "Almanac Data – Message ID 14" on page 3-21.

## Poll Ephemeris – Message ID 147

Causes the receiver to respond with the ephemeris of the requested satellite. The ephemeris is sent using Message ID 15. It can also request all ephemerides, resulting in as many Message 15s as there are ephemerides currently stored in the receiver.

**Note –** Some software versions do not support this command.

Table 2-32 contains the input values for the following example:

Poll for Ephemeris Data for all satellites.

Example:

A0A20003—Start Sequence and Payload Length

930000—Payload

0092B0B3—Message Checksum and End Sequence

*Table 2-32* Poll Ephemeris – Message ID 147

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | 93 | | Decimal 147 |
| Sv ID[1] | 1 U | | 00 | | Range 0 to 32 |
| Control | 1 U | | 00 | | Not used |

Payload length: 3 bytes

1. A value of zero requests all available ephemeris records. This results in a maximum of twelve output messages. A value of 1 through 32 requests only the ephemeris of that SV.

---

**Note –** Returned message is Message ID 15. See “Ephemeris Data (Response to Poll) – Message ID 15” on page 3-22.

---

## *Flash Update – Message ID 148*

Allows the user to command the receiver to enter internal boot mode without setting the hardware bootstrap configuration input. Internal boot mode allows the user to re-flash the embedded code in the receiver.

---

**Note –** It is highly recommended that all hardware designs provide access to the hardware bootstrap configuration input pin(s) in the event of a failed flash upload.

---

Example:

A0A20001 – Start Sequence and Payload Length

94 – Payload

0094B0B3 – Message Checksum and End Sequence

*Table 2-33* Flash Update – Message ID 148

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | 94 | | Decimal 148 |

Payload length: 1 bytes

---

**Note –** Some software versions do not support this command

---

## *Set Ephemeris – Message ID 149*

Enables the user to upload an ephemeris file to the receiver.

Example:

A0A2005B – Start Sequence and Payload Length

95.................... – Payload

xxxxB0B3 – Message Checksum and End Sequence

*Table 2-34* Set Ephemeris – Message ID 149

| Name | Bytes | Binary (Hex) Scale | Example | Unit | Description |
|---|---|---|---|---|---|
| Message ID | 1 U | | 95 | | Decimal 149 |
| Ephemeris Data [45] | 2 U | | 00 | | Reserved |

Payload length: 91 bytes

The ephemeris data for each satellite is stored as a two dimensional array of [3][15] UNIT16 elements. The row represents three separate sub-frames. See Message ID 15 (“Ephemeris Data (Response to Poll) – Message ID 15” on page 3-22) for a detailed description of this data format.

**Note –** Some software versions do not support this command.

## *Switch Operating Modes – Message ID 150*

Places the receiver in production test or normal operating mode.

Table 2-35 contains the input values for the following example:

Sets the receiver to track SV ID 6 on all channels and to collect test mode performance statistics for 30 seconds.

Example:

A0A20007—Start Sequence and Payload Length

961E510006001E—Payload

0129B0B3—Message Checksum and End Sequence

*Table 2-35* Switch Operating Modes – Message ID 150

| Name | Bytes | Binary (Hex) Scale | Example | Unit | Description |
|---|---|---|---|---|---|
| Message ID | 1 U | | 96 | | Decimal 150 |
| Mode | 2 U | | 1E51 | | 0 = normal, 1E51 = Testmode1, 1E52 = Testmode2, 1E53 = Testmode3, 1E54 = Testmode4 |
| SvID | 2 U | | 0006 | | Satellite to Track |
| Period | 2 U | | 001E | sec | Duration of Track |

Payload length: 7 bytes

**Note –** In GSW3 and GSWLT3, processing this messages puts MaxOffTime and MaxAcqTime to default values. Requires Message ID 167 after this to restore those to non-default values.

## Set TricklePower Parameters – Message ID 151

Allows the user to set some of the power-saving modes of the receiver.

Table 2-36 contains the input values for the following example:

Sets the receiver to low power modes.

Example: Set receiver to TricklePower at 1 Hz update and 200 ms on-time.

A0A20009—Start Sequence and Payload Length

97000000C8000000C8—Payload

0227B0B3—Message Checksum and End Sequence

*Table 2-36* Set TricklePower Parameters – Message ID 151

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | 97 | | Decimal 151 |
| Push-to-Fix Mode | 2 S | | 0000 | | ON = 1, OFF = 0 |
| Duty Cycle | 2 S | *10 | 00C8 | % | % Time ON. A duty cycle of 1000 (100%) means continuous operation |
| On-Time[1] | 4 S | | 000000C8 | msec | range 200 - 900 msec |

Payload length: 9 bytes

1. On-time of 700, 800, or 900 ms is invalid if an update rate of 1 second is selected.

### Computation of Duty Cycle and On-Time

The Duty Cycle is the desired time to be spent tracking. The On-Time is the duration of each tracking period (range is 200 - 900 msec). To calculate the TricklePower update rate as a function of Duty Cycle and On Time, use the following formula:

$$\text{Update Rate} = \frac{\text{On-Time (in sec)}}{\text{Duty Cycle}}$$

**Note –** It is not possible to enter an on-time > 900 msec.

Following are some examples of selections:

*Table 2-37* Example of Selections for TricklePower Mode of Operation

| Mode | On Time (ms) | Duty Cycle (%) | Interval Between Updates (sec) |
|---|---|---|---|
| Continuous[1] | 200[2] | 100 | 1 |
| TricklePower | 200 | 20 | 1 |
| TricklePower | 200 | 10 | 2 |
| TricklePower | 300 | 10 | 3 |
| TricklePower | 500 | 5 | 10 |

1. when the duty cycle is set to 100 %, the on time has no effect. However, the command parser might still test the value against the 200-600 ms limits permitted for a 1-second cycle time. Therefore, we recommend that you set the on-time value to 200 ms.

2. When the duty cycle is set to 100%, the value in this field has no effect. Thus, any legal value (100 to 900) may be used.

*Table 2-38* Duty Cycles for Supported TricklePower Settings

| | Update Rates (sec) | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **On-Time (ms)** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| **200**[1] | 200 | 100 | 67 | 50 | 40 | 33 | 29 | 25 | 22 | 20 |
| **300** | 300 | 150 | 100 | 75 | 60 | 50 | 43 | 37 | 33 | 30 |
| **400** | 400 | 200 | 133 | 100 | 80 | 67 | 57 | 50 | 44 | 40 |
| **500** | 500 | 250 | 167 | 125 | 100 | 83 | 71 | 62 | 56 | 50 |
| **600** | 600 | 300 | 200 | 150 | 120 | 100 | 86 | 75 | 67 | 60 |
| **700** | Value not permitted | 350 | 233 | 175 | 140 | 117 | 100 | 88 | 78 | 70 |
| **800** | Value not permitted | 400 | 267 | 200 | 160 | 133 | 114 | 100 | 89 | 80 |
| **900** | Value not permitted | 450 | 300 | 225 | 180 | 150 | 129 | 112 | 100 | 90 |

1. When the duty cycle is set to 100%, the on time has no effect. However, the command parser may still test the value against the 200-600 ms limits permitted for a 1-second cycle time. Therefore, set the on-time value to 200 ms.

---

**Note –** Values are in % times 10 as needed for the duty cycle field. For 1 second update rate, on-times greater than 600 ms are not allowed.

---

### *Push-to-Fix*

In this mode the receiver turns on every cycle period to perform a system update consisting of an RTC calibration and satellite ephemeris data collection if required (i.e., a new satellite has become visible) as well as all software tasks to support Snap Start in the event of a Non-Maskable Interrupt (NMI). Ephemeris collection time in general takes 18 to 36 seconds. If ephemeris data is not required then the system re-calibrates and shuts down. In either case, the amount of time the receiver remains off is in proportion to how long it stayed on:

Off period = On Period*(1-Duty Cycle)
Duty Cycle

The off period has a possible range between 10 and 7200 seconds. The default is 1800 seconds. Push-to-Fix cycle period is set using Message ID 167.

---

**Note –** When Message ID 151 is issued in GSW3 software, the receiver resets both MaxOffTime and MaxSearchTime to default values. If different values are needed, Message ID 151 must be issued before Message ID 167.

---

## *Poll Navigation Parameters – Message ID 152*

Requests the receiver to report its current navigation parameter settings. The receiver responds to this message with Message ID 19. Table 2-39 contains the input values for the following example:

Example: Poll receiver for current navigation parameters.

A0A20002—Start Sequence and Payload Length

9800—Payload

0098B0B3—Message Checksum and End Sequence

*Table 2-39* Poll Receiver for Navigation Parameters – Message ID 152

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | 98 | | Decimal 152 |
| Reserved | 1 U | | 00 | | Reserved |

Payload length: 2 bytes

## *Set UART Configuration – Message ID 165*

Sets the protocol, bit rate, and port settings on any UART.

**Note –** This message supports setting up to four UARTs.

Table 2-40 contains the input values for the following example:

Example: Set port 0 to NMEA with 9600 bits per second, 8 data bits, 1 stop bit, no parity. Set port 1 to SiRF binary with 57600 bits per second, 8 data bits, 1 stop bit, no parity. Do not configure ports 2 and 3.

Example:

A0A20031—Start Sequence and Payload Length

A50001010000258008010000000100000000E1000801000000FF050500000000000000000000FF0505000000000000000000—Payload

0452B0B3—Message Checksum and End Sequence

*Table 2-40* Set UART Configuration – Message ID 165

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | A5 | | Decimal 165 |
| Port[1] | 1 U | | 00 | | For UART 0 |
| In Protocol[2] | 1 U | | 01 | | For UART 0 |
| Out Protocol | 1 U | | 01 | | For UART 0 (Set to in protocol) |
| Bit Rate[3] | 4 U | | 00002580 | | For UART 0 |
| Data Bits[4] | 1 U | | 08 | | For UART 0 |
| Stop Bits[5] | 1 U | | 01 | | For UART 0 |
| Parity[6] | 1 U | | 00 | | For UART 0 |
| Reserved | 1 U | | 00 | | For UART 0 |
| Reserved | 1 U | | 00 | | For UART 0 |
| Port | 1 U | | 01 | | For UART 1 |
| In Protocol | 1 U | | 00 | | For UART 1 |
| Out Protocol | 1 U | | 00 | | For UART 1 |
| Bit Rate | 4 U | | 0000E100 | | For UART 1 |
| Data Bits | 1 U | | 08 | | For UART 1 |
| Stop Bits | 1 U | | 01 | | For UART 1 |
| Parity | 1 U | | 00 | | For UART 1 |
| Reserved | 1 U | | 00 | | For UART 1 |
| Reserved | 1 U | | 00 | | For UART 1 |
| Port | 1 U | | FF | | For UART 2 |

*Table 2-40* Set UART Configuration – Message ID 165 *(Continued)*

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| In Protocol | 1 U | | 05 | | For UART 2 |
| Out Protocol | 1 U | | 05 | | For UART 2 |
| Bit Rate | 4 U | | 00000000 | | For UART 2 |
| Data Bits | 1 U | | 00 | | For UART 2 |
| Stop Bits | 1 U | | 00 | | For UART 2 |
| Parity | 1 U | | 00 | | For UART 2 |
| Reserved | 1 U | | 00 | | For UART 2 |
| Reserved | 1 U | | 00 | | For UART 2 |
| Port | 1 U | | FF | | For UART 3 |
| In Protocol | 1 U | | 05 | | For UART 3 |
| Out Protocol | 1 U | | 05 | | For UART 3 |
| Bit Rate | 4 U | | 00000000 | | For UART 3 |
| Data Bits | 1 U | | 00 | | For UART 3 |
| Stop Bits | 1 U | | 00 | | For UART 3 |
| Parity | 1 U | | 00 | | For UART 3 |
| Reserved | 1 U | | 00 | | For UART 3 |
| Reserved | 1 U | | 00 | | For UART 3 |

Payload length: 49 bytes

1. 0xFF means to ignore this port; otherwise, put the port number in this field (e.g., 0 or 1).
2. 0 = SiRF Binary, 1 = NMEA, 2 = ASCII, 3 = RTCM, 4 = User1, 5 = No Protocol. Any software version only supports some subset of these protocols. Selecting a protocol that is not supported by the software may cause unexpected results.
3. Valid values are 1200, 2400, 4800, 9600, 19200, 38400, 57600, and 115200.
4. Valid values are 7 and 8.
5. Valid values are 1 and 2.
6. 0 = None, 1 = Odd, 2 = Even.

---

**Note –** While this message supports four UARTs, the specific baseband chip in use may contain fewer.

---

## *Set Message Rate – Message ID 166*

Controls the output rate of binary messages. Table 2-41 contains the input values for the following example:

Set Message ID 2 to output every five seconds starting immediately.

Example:

A0A20008—Start Sequence and Payload Length

A600020500000000—Payload

00ADB0B3—Message Checksum and End Sequence

*Table 2-41* Set Message Rate – Message ID 166

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | A6 | | decimal 166 |
| Mode[1] | 1 U | | 00 | | 00: enable/disable one message<br>01: poll one message instantly<br>02: enable/disable all messages<br>03: enable/disable default navigation messages (Message ID 2 and 4)<br>04: enable/disable default debug messages (Message ID 9 and 255)<br>05: enable/disable navigation debug messages (Message ID 7, 28, 29, 30, and 31 |
| Message ID to be set | 1 U | | 02 | | |
| Update Rate[2] | 1 U | | 05 | sec | Range = 0 - 30 |
| Reserved | 1 U | | 00 | | Not used, set to zero |
| Reserved | 1 U | | 00 | | No used, set to zero |
| Reserved | 1 U | | 00 | | Not used, set to zero |
| Reserved | 1 U | | 00 | | Not used, set to zero |

Payload Length: 8 bytes

1. Values 02 - 05 are available for GSW3 and SLC3 software only.
2. A value of 0 means to stop sending the message. A value in the range of 1 - 30 specifies the cycle period.

## *Set Low Power Acquisition Parameters – Message ID 167*

Provides tools to set MaxOffTime, MaxSearchTime, Push-to-Fix period and Adaptive TricklePower. These settings affect low-power modes as follows:

MaxOffTime: when the receiver is unable to acquire satellites for a TricklePower or Push-to-Fix cycle, it returns to sleep mode for this period of time before it tries again.

MaxSearchTime: in TricklePower and Push-to-Fix modes, when the receiver is unable to reacquire at the start of a cycle, this parameter sets how long it tries. After this time expires, the unit returns to sleep mode for MaxOffTime (if in TricklePower or ATP mode) or Push-to-Fix cycle time (in Push-to-Fix mode).

Table 2-42 contains the input values for the following example:

Set maximum time for sleep mode and maximum satellite search time to default values. Also set Push-to-Fix cycle time to 60 seconds and disable Adaptive TricklePower.

Example:

A0A2000F—Start Sequence and Payload Length

A7000075300001D4C00000003C0000—Payload

031DB0B3—Message Checksum and End Sequence

*Table 2-42* Set Low Power Acquisition Parameters – Message ID 167

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | A7 | | Decimal 167 |
| Max Off Time | 4 U | | 00007530 | msec | Maximum time for sleep mode. Default value: 30 seconds |
| Max Search Time | 4 U | | 0001D4C0 | msec | Max. satellite search time. Default value: 120 seconds |
| Push-to-Fix Period | 4 U | | 0000003C | sec | Push-to-Fix cycle period |
| Adaptive TricklePower | 2 U | | 0001 | | To enable Adaptive TricklePower 0 = off; 1 = on |

Payload length: 15 bytes

**Note –** When Message ID 151 is issued in GSW3 software, the receiver resets both MaxOffTime and MaxSearchTime to default values. If different values are needed, Message ID 151 must be issued before Message ID 167.

## Poll Command Parameters – Message ID 168

Queries the receiver to send specific response messages for one of the following messages: 128, 133, 136, 137, 138, 139, 140, 143 and 151. In response to this message, the receiver sends Message ID 43.

Table 2-43 contains the input values for the following example:

Query the receiver for current low power parameter settings set by Message ID 0x97.

Example:

A0A20002–Start Sequence and Payload Length

A897-Payload

013FB0B3-Message Checksum and End Sequence

*Table 2-43* Poll Command Parameters – Message ID 168

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | A8 | | Decimal 168 |
| Poll Msg ID | 1 U | | 97 | | Requesting Msg ID 0x97[1] |

Payload length: 2 bytes

1. Valid Message IDs are 0x80, 0x85, 0x88, 0x89, 0x8A, 0x8B, 0x8C, 0x8F, and 0x97.

## Set SBAS Parameters – Message ID 170

Allows the user to set the SBAS parameters.

Table 2-44 contains the input values for the following example:

Set automatic SBAS search and testing operating mode.

Example:

A0A20006—Start Sequence and Payload Length

AA0000010000—Payload

01B8B0B3—Message Checksum and End Sequence

*Table 2-44* Set SBAS Parameters – Message ID 170

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 U | | AA | | Decimal 170 |
| SBAS PRN | 1 U | | 00 | | 0 = Auto mode<br>PRN 120-138 = Exclusive |
| SBAS Mode | 1 U | | 00 | | 0 = Testing, 1 = Integrity<br>Integrity mode rejects SBAS corrections if the SBAS satellite is transmitting in a test mode<br>Testing mode accepts/uses SBAS corrections even if satellite is transmitting in a test mode |
| Flag Bits[1] | 1 D | | 01 | | Bit 0: Timeout; 0 = Default 1 = User<br>Bit 1: Health; Reserved<br>Bit 2: Correction; Reserved<br>Bit 3: SBAS PRN; 0 = Default 1 = User |
| Spare | 2 | | 0000 | | |

Payload length: 6 bytes

1. If Bit 0 = 1, user-specified timeout from Message ID 138 is used. If Bit 0 = 0, timeout specified by the SBAS satellite is used (this is usually 18 seconds). If Bit 3 = 1, the SBAS PRN specified in the SBAS PRN field is used. If Bit 3 = 0, the system searches for any SBAS PRN.

# *Initialize GPS/DR Navigation – Message ID 172 (Sub ID 1)*

Sets the navigation initialization parameters and commands a software reset based on these parameters.

*Table 2-45* Navigation Initialization Parameters

| Name | Bytes | Scale | Unit | Description |
|---|---|---|---|---|
| Message ID | 1 | | | = 0xAC |
| Message Sub ID | 1 | | | = 0x01 |
| Latitude | 4 | | deg | for Warm Start with user input |
| Longitude | 4 | | deg | for Warm Start with user input |
| Altitude (ellipsoid) | 4 | | m | for Warm Start with user input |
| True heading | 2 | | deg | for Warm Start with user input |
| Clock drift | 4 | | Hz | for Warm Start with user input |
| GPS time of week | 4 | 100 | sec | for Warm Start with user input |
| GPS week number | 2 | | | for Warm Start with user input |
| Channel count | 1 | | | for Warm Start with user input |

*Table 2-45* Navigation Initialization Parameters *(Continued)*

| Name | Bytes | Scale | Unit | Description |
|---|---|---|---|---|
| Reset configuration bits[1] | 1 | | | Bit 0: use initial data provided in this message for start-up<br>Bit 1: clear ephemeris in memory<br>Bit 2: clear all memory<br>Bit 3: perform Factory Reset<br>Bit 4: enable SiRF Binary output messages for raw track data, navigation library, 50 bps info, RTCM data, clock status, and DR status<br>Bit 5: enable debug output messages<br>Bit 6: Reserved<br>Bit 7: Reserved |

Payload length: 28 bytes

1. Bits 0 - 3 determine the reset mode: 0000 = Hot; 0010 = Warm; 0011 = Warm with user input; 0100 = Cold; 1000 = Factory.

## Set GPS/DR Navigation Mode – Message ID 172 (Sub ID 2)

Sets the GPS/DR navigation mode control parameters.

*Table 2-46* GPS/DR Navigation Mode Control Parameters – Message ID 172 (Sub ID 2)

| Name | Bytes | Description |
|---|---|---|
| Message ID | 1 | = AC |
| Message Sub ID | 1 | = 0x02 |
| Mode | 1 | Bit 0 : GPS-only navigation<br>Bit 1 : DR nav acceptable with stored/default calibration<br>Bit 2 : DR nav acceptable with current GPS calibration<br>Bit 3 : DR-only navigation |
| Reserved | 1 | |

## Set DR Gyro Factory Calibration – Message ID 172 (Sub ID 3)

Sets DR gyro factory calibration parameters.

*Table 2-47* DR Gyro Factory Calibration Parameters – Message ID 172 (Message Sub ID 3)

| Name | Bytes | Scale | Unit | Description |
|---|---|---|---|---|
| Message ID | 1 | | | = 0xAC |
| Message Sub ID | 1 | | | = 0x03 |
| Calibration | 1 | | | Bit 0 : Start gyro bias calibration<br>Bit 1 : Start gyro scale factor calibration<br>Bits 2 - 7 : Reserved |
| Reserved | 1 | | | |

Payload length: 4 bytes

## Set DR Sensors' Parameters – Message ID 172 (Sub ID 4)

Sets DR sensors parameters.

*Table 2-48* DR Sensors Parameters – Message ID 172 (Message Sub ID 4)

| Name | Bytes | Scale | Unit | Description |
|---|---|---|---|---|
| Message ID | 1 | | | = 0xAC |
| Message Sub ID | 1 | | | = 0x04 |
| Base speed scale factor | 1 | | ticks/m | |
| Base gyro bias | 2 | $10^4$ | mV | |
| Base gyro scale factor | 2 | $10^3$ | mV/deg/s | |

Payload length: 7 bytes

## Poll DR Gyro Factory Calibration – Message ID 172 (Sub ID 6)

Polls the DR gyro factory calibration status.

*Table 2-49* DR Gyro Factory Calibration Status – Message ID 172 (Message Sub ID 6)

| Name | Bytes | Description |
|---|---|---|
| Message ID | 1 | = AC |
| Message Sub ID | 1 | = 0x06 |

Payload length: 2 bytes

## Poll DR Sensors' Parameters – Message ID 172 (Sub ID 7)

Message 172 Sub IDs apply to SiRFDiRect only

Polls the DR sensors parameters.

*Table 2-50* DR Sensors Parameters – Message ID 172 (Message Sub ID 7)

| Name | Bytes | Description |
|---|---|---|
| Message ID | 1 | = AC |
| Message Sub ID | 1 | = 0x07 |

Payload length: 2 bytes

# Input Car Bus Data to NAV – Message ID 172 (Sub ID 9)

Sensor data output converted into engineering units.

*Table 2-51* Input Car Bus Data – Message ID 172 (Message Sub ID 9)

| Byte | Field | Data Type | Bytes | Unit | Range | Res |
|---|---|---|---|---|---|---|
| 1 | Message ID | UINT8 | 1 | N/A | 0xAC | N/A |
| 2 | Message Sub-ID | UINT8 | 1 | N/A | 0x09 | N/A |
| 3 | Sensor Data Type (depends on sensor) | UINT8 | 1 | N/A | 0-127<br>1: Gyro, Speed Data, and Reverse<br>2. 4 Wheel Pulses, and Reverse<br>3: 4 Wheel Speed, and Reverse<br>4: 4 Wheel Angular Speed, and Reverse<br>5: Gyro, Speed Data, NO Reverse<br>6: 4 Wheel Pulses, NO Reverse<br>7: 4 Wheel Speed, NO Reverse<br>8: 4 Wheel Angular Speed, NO Reverse<br>9: Gyro, Speed Data, Reverse, Steering Wheel Angle, Longitudinal Acceleration, Lateral Acceleration<br>10: Yaw Rate Gyro,<br>Vertical Acceleration (Up)(Z),<br>Longitudinal Acceleration (Front)(X),<br>Lateral Acceleration (Left)(Y)<br>11-127: Reserved | N/A |
| 4 | Number of Valid data sets | UINT8 | 1 | N/A | 0-11 | N/A |
| 5 | Reverse Bit Map<br>N/A for SDT = 10 | UINT16 | 2 | N/A | Bit-mapped indication of REVERSE status corresponding to each sensor data set, i.e. bit 0 corresponds to the first data set, bit 1 corresponds to the second data set, etc. | N/A |
| 7+(N-1)* 16[1] | Valid Sensor Indication | UINT8 | 1 | N/A | Valid/Not Valid indication for each one of the four possible sensor inputs in a individual data set; when a particular bit is set to 1 the corresponding data is Valid, when the bit is set to 0 the corresponding data is NOT valid.<br><br>Bit 0 corresponds to Data Set Time Tag<br>Bit 1 corresponds to Odometer Speed<br>Bit 2 corresponds to Data 1<br>Bit 3 corresponds to Data 2<br>Bit 4 corresponds to Data 3<br>Bit 5 corresponds to Data 4<br><br>Bits 6-7 : Reserved | N/A |
| 8+(N-1)* 16[1] | Data Set Time Tag | UINT32 | 4 | msec | 0-4294967295 | 1 |
| 12+ (N-1)*16[1] | Odometer Speed (also known as VSS) N/A for SDT = 10 | UINT16 | 2 | m/sec | 0 to 100 | 0.01 |
| 14+(N-1)* 16[1] | **Data 1** Depends on SDT | INT16 | 2 | Depends on SDT | Depends on SDT | Depends on SDT |
| | SDT = 1, 5, 9, 10: gyro rate | | | Deg/sec | -120 to 120 | 0.01 |
| | SDT = 2, 6: right front wheel pulses | | | N/A | 4000 | 1 |
| | SDT = 3, 7: right front wheel speed | | | m/sec | 0 to 100 | 0.01 |
| | SDT = 4, 8: right front wheel angular speed | | | rad/sec | -327.67 to 327.67 | 0.01 |
| 16+(N-1)* 16[1] | **Data 2** Depends on SDT | INT16 | 2 | Depends on SDT | Depends on SDT | Depends on SDT |

*Table 2-51* Input Car Bus Data – Message ID 172 (Message Sub ID 9) *(Continued)*

| Byte | Field | Data Type | Bytes | Unit | Range | Res |
|---|---|---|---|---|---|---|
| | SDT = 1: N/A | | | N/A | N/A | N/A |
| | SDT = 2, 6: left front wheel pulses | | | N/A | 4000 | 1 |
| | SDT = 3, 7: left front wheel speed | | | m/sec | 0 to 100 | 0.01 |
| | SDT = 4, 8: left front wheel angular speed | | | rad/sec | -327.67 to 327.67 | 0.01 |
| | SDT = 9: steering wheel angle | | | deg | -720 to 720 | 0.05 |
| | SDT = 10: downward acceleration | | | $m/sec^2$ | -15 to 15 | 0.001 |
| 18+(N-1)* 16[1] | **Data 3** Depends on SDT | INT16 | 2 | Depends on SDT | Depends on SDT | Depends on SDT |
| | SDT = 1: N/A | | | N/A | N/A | N/A |
| | SDT = 2, 6: right rear wheel pulses | | | N/A | 4000 | 1 |
| | SDT = 3, 7: right rear wheel speed | | | m/sec | 0 to 100 | 0.01 |
| | SDT = 4, 8: right rear wheel speed | | | rad/sec | -327.67 to 327.67 | 0.01 |
| | SDT = 9, 10: longitudinal acceleration | | | $m/sec^2$ | -15 to 15 | 0.001 |
| 20+(N-1)* 16[1] | **Data 4** Depends on SDT | INT16 | 2 | Depends on SDT | Depends on SDT | Depends on SDT |
| | SDT = 1: N/A | | | N/A | N/A | N/A |
| | SDT = 2, 6: left rear wheel pulses | | | N/A | 4000 | 1 |
| | SDT = 3, 7: left rear wheel speed | | | m/sec | 0 to 100 | 0.01 |
| | SDT = 4, 8: left rear wheel speed | | | rad/sec | -327.67 to 327.67 | 0.01 |
| | SDT = 9, 10: lateral acceleration | | | $m/sec^2$ | -15 to 15 | 0.001 |
| 22+(N-1)* 16[1] | Reserved | UINT8 | 1 | N/A | N/A | N/A |

Payload length: 22 to 182 bytes

Note 1: N indicates the number of valid data sets in the message

## *Car Bus Enabled – Message ID 172 (Sub ID 10)*

Sending the message enables the car bus. Mode is reserved for future use.

*Table 2-52* Bus Enabled – Message ID 172 (Message Sub ID 10)

| Name | Bytes | Description |
|---|---|---|
| Message ID | 1 | 0xAC |
| Message Sub ID | 1 | 0xA |
| Mode | 4 | Undefined/not used |

Payload length: 6 bytes

## *Car Bus Disabled – Message ID 172 (Sub ID 11)*

Sending the message disables the car bus. Mode is reserved for future use.

*Table 2-53* Bus Disabled – Message ID 172 (Message Sub ID 11)

| Name | Bytes | Description |
|---|---|---|
| Message ID | 1 | 0xAC |
| Message Sub ID | 1 | 0xB |
| Mode | 4 | Undefined/not used |

Payload length: 6 bytes

# Input Car Bus Data 2 – Message ID 172 (Sub ID 14)

Message applies to SiRFDiRect only

Sensor data output converted into engineering units.

*Table 2-54* Binary Message Definition – Message ID 172 (Message Sub ID 14)

| Byte | Field | Data Type | Bytes | Unit | Range | Resolution |
|---|---|---|---|---|---|---|
| 1 | Message ID | UINT8 | 1 | N/A | 0xAC | N/A |
| 2 | Sub-ID | UINT8 | 1 | N/A | 0x0E | N/A |
| 3 | SensorDataType | UINT8 | 1 | N/A | Fixed at 10 | N/A |
| 4 | NumValidDataSets | UINT8 | 1 | N/A | 0 to 10 valid data sets in message | N/A |
| 5 | DataFrequency | UINT8 | 1 | N/A | Fixed at 10 | N/A |
| 6 | ValidSensorIndication[0] | UINT16 | 2 | N/A | Bit 0x1: Time tag valid<br>Bit 0x2: Reserved<br>Bit 0x4: Data[0] valid<br>Bit 0x8: Data[1] valid<br>Bit 0x10: Data[2] valid<br>Bit 0x20: Data[3] valid<br>Bit 0x40: Data[4] valid<br>Bit 0xFF80: Reserved | N/A |
| 8 | DataSetTimeTag[0] | UINT32 | 4 | N/A | 0 to 0xFFFFFFFF | N/A |
| 12 | Heading Gyro[0] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 14 | Z-Axis[0] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 16 | X-Axis[0]] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 18 | Y-Axis[0] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 20 | Pitch Gyro[0] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 22 | Reserved[0] | UINT8 | 1 | N/A | 0 to 0xff | 1 |
| 23 | ValidSensorIndication[1] | UINT16 | 2 | N/A | Bit 0x1: Time tag valid<br>Bit 0x2: Reserved<br>Bit 0x4: Data[0] valid<br>Bit 0x8: Data[1] valid<br>Bit 0x10: Data[2] valid<br>Bit 0x20: Data[3] valid<br>Bit 0x40: Data[4] valid | N/A |
| 25 | DataSetTimeTag[1] | UINT32 | 4 | N/A | 0 to 0xFFFFFFFF | N/A |
| 29 | Heading Gyro[1] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 31 | Z-Axis[1] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 33 | X-Axis[1]] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 35 | Y-Axis[1] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 37 | Pitch Gyro[1] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 39 | Reserved[1] | UINT8 | 1 | N/A | 0 to 0xff | 1 |
| 40 | ValidSensorIndication[2] | UINT16 | 2 | N/A | Bit 0x1: Time tag valid<br>Bit 0x2: Reserved<br>Bit 0x4: Data[0] valid<br>Bit 0x8: Data[1] valid<br>Bit 0x10: Data[2] valid<br>Bit 0x20: Data[3] valid<br>Bit 0x40: Data[4] valid | N/A |
| 42 | DataSetTimeTag[2] | UINT32 | 4 | N/A | 0 to 0xFFFFFFFF | N/A |
| 46 | Heading Gyro[2] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 48 | Z-Axis[2] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 50 | X-Axis[2] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 52 | Y-Axis[2] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 54 | Pitch Gyro[2] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |

*Table 2-54* Binary Message Definition – Message ID 172 (Message Sub ID 14) *(Continued)*

| Byte | Field | Data Type | Bytes | Unit | Range | Resolution |
|---|---|---|---|---|---|---|
| 56 | Reserved[2] | UINT8 | 1 | N/A | 0 to 0xff | 1 |
| 57 | ValidSensorIndication[3] | UINT16 | 2 | N/A | Bit 0x1: Time tag valid<br>Bit 0x2: Reserved<br>Bit 0x4: Data[0] valid<br>Bit 0x8: Data[1] valid<br>Bit 0x10: Data[2] valid<br>Bit 0x20: Data[3] valid<br>Bit 0x40: Data[4] valid | N/A |
| 59 | DataSetTimeTag[3] | UINT32 | 4 | N/A | 0 to 0xFFFFFFFF | N/A |
| 63 | Heading Gyro[3] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 65 | Z-Axis[3] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 67 | X-Axis[3] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 69 | Y-Axis[3] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 71 | Pitch Gyro[3] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 73 | Reserved[3] | UINT8 | 1 | N/A | 0 to 0xff | 1 |
| 74 | ValidSensorIndication[4] | UINT16 | 2 | N/A | Bit 0x1: Time tag valid<br>Bit 0x2: Reserved<br>Bit 0x4: Data[0] valid<br>Bit 0x8: Data[1] valid<br>Bit 0x10: Data[2] valid<br>Bit 0x20: Data[3] valid<br>Bit 0x40: Data[4] valid | N/A |
| 76 | DataSetTimeTag[4] | UINT32 | 4 | N/A | 0 to 0xFFFFFFFF | N/A |
| 80 | Heading Gyro[4] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 82 | Z-Axis[4] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 84 | X-Axis[4]] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 86 | Y-Axis[4] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 88 | Pitch Gyro[4] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 90 | Reserved[4] | UINT8 | 1 | N/A | 0 to 0xff | 1 |
| 91 | ValidSensorIndication[5] | UINT16 | 2 | N/A | Bit 0x1: Time tag valid<br>Bit 0x2: Reserved<br>Bit 0x4: Data[0] valid<br>Bit 0x8: Data[1] valid<br>Bit 0x10: Data[2] valid<br>Bit 0x20: Data[3] valid<br>Bit 0x40: Data[4] valid | N/A |
| 93 | DataSetTimeTag[5] | UINT32 | 4 | N/A | 0 to 0xFFFFFFFF | N/A |
| 97 | Heading Gyro[5] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 99 | Z-Axis[5] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 101 | X-Axis[5]] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 103 | Y-Axis[5] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 105 | Pitch Gyro[5] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 107 | Reserved[5] | UINT8 | 1 | N/A | 0 to 0xff | 1 |
| 108 | ValidSensorIndication[6] | UINT16 | 2 | N/A | Bit 0x1: Time tag valid<br>Bit 0x2: Reserved<br>Bit 0x4: Data[0] valid<br>Bit 0x8: Data[1] valid<br>Bit 0x10: Data[2] valid<br>Bit 0x20: Data[3] valid<br>Bit 0x40: Data[4] valid | N/A |
| 110 | DataSetTimeTag[6] | UINT32 | 4 | N/A | 0 to 0xFFFFFFFF | N/A |
| 114 | Heading Gyro[6] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 116 | Z-Axis[6] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 118 | X-Axis[6]] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |

*Table 2-54* Binary Message Definition – Message ID 172 (Message Sub ID 14) *(Continued)*

| Byte | Field | Data Type | Bytes | Unit | Range | Resolution |
|---|---|---|---|---|---|---|
| 120 | Y-Axis[6] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 122 | Pitch Gyro[6] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 124 | Reserved[6] | UINT8 | 1 | N/A | 0 to 0xff | 1 |
| 125 | ValidSensorIndication[7] | UINT16 | 2 | N/A | Bit 0x1: Time tag valid<br>Bit 0x2: Reserved<br>Bit 0x4: Data[0] valid<br>Bit 0x8: Data[1] valid<br>Bit 0x10: Data[2] valid<br>Bit 0x20: Data[3] valid<br>Bit 0x40: Data[4] valid | N/A |
| 127 | DataSetTimeTag[7] | UINT32 | 4 | N/A | 0 to 0xFFFFFFFF | N/A |
| 131 | Heading Gyro[7] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 133 | Z-Axis[7] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 135 | X-Axis[7]] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 137 | Y-Axis[7] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 139 | Pitch Gyro[7] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 141 | Reserved[7] | UINT8 | 1 | N/A | 0 to 0xff | 1 |
| 142 | ValidSensorIndication[8] | UINT16 | 2 | N/A | Bit 0x1: Time tag valid<br>Bit 0x2: Reserved<br>Bit 0x4: Data[0] valid<br>Bit 0x8: Data[1] valid<br>Bit 0x10: Data[2] valid<br>Bit 0x20: Data[3] valid<br>Bit 0x40: Data[4] valid | N/A |
| 144 | DataSetTimeTag[8] | UINT32 | 4 | N/A | 0 to 0xFFFFFFFF | N/A |
| 148 | Heading Gyro[8] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 150 | Z-Axis[8] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 152 | X-Axis[8] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 154 | Y-Axis[8] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 156 | Pitch Gyro[8] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 158 | Reserved[8] | UINT8 | 1 | N/A | 0 to 0xff | 1 |
| 159 | ValidSensorIndication[9] | UINT16 | 2 | N/A | Bit 0x1: Time tag valid<br>Bit 0x2: Reserved<br>Bit 0x4: Data[0] valid<br>Bit 0x8: Data[1] valid<br>Bit 0x10: Data[2] valid<br>Bit 0x20: Data[3] valid<br>Bit 0x40: Data[4] valid | N/A |
| 161 | DataSetTimeTag[9] | UINT32 | 4 | N/A | 0 to 0xFFFFFFFF | N/A |
| 165 | Heading Gyro[9] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 167 | Z-Axis[9] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 169 | X-Axis[9]] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 171 | Y-Axis[9] | INT16 | 2 | M/sec^2 | ±2 Gs | 1/1668.0 |
| 173 | Pitch Gyro[9] | INT16 | 2 | deg/sec | ±60 degrees per second | 1/1e2 |
| 175 | Reserved[9] | UINT8 | 1 | N/A | 0 to 0xff | 1 |

Payload length: 175 bytes

## User Set Command – Message ID 175

Allows user to send an input command string and parse the associated functions.

Table 2-55 describes the message content.

*Table 2-55* User Set Command – Message ID 175

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 | | AF | | Decimal 175 |
| User Set Command | Variable | | | | Depends on user's input |

Payload length: Variable bytes

**Note –** This message can only be used by SDK customers.

## Preset Operating Configuration – Message ID 180

**Note –** This Message ID 180 is used only with GSC2xr chip.

Overrides the Preset Operating Configuration as defined in bits [3:2] of the GSC2xr chip configuration register. The valid input values mapped to the Preset Operating Configuration are described in Table 2-56.

*Table 2-56* Valid Input Values

| Mapping | |
|---|---|
| Input Values | Preset Configuration |
| 0 | 1 |
| 1 | 2 |
| 2 | 3 |
| 3 | 4 |
| 4 | Standard GSW2 and GSW2x software default configuration[1] |

1. The default configuration is SiRF Binary at 38400 bps using UART A and RTCM at 9600 bps using UART B.

Table 2-57 contains the input values for the following example:

Set receiver to Standard GSW2 Default Configuration.

Example:

A0A20002—Start Sequence and Payload Length

B404—Payload

00B8B0B3—Message Checksum and End Sequence

*Table 2-57* GSC2xr Preset Operating Configuration – Message ID 180

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 | | B4 | | Decimal 180 |
| Input[1] | 1 | | 04 | | Valid input value from 0 to 4 |

Payload length: 2bytes

1. Invalid input value yields a Rejected MID_UserInputBegin while a valid input value yields a Acknowledged MID_UserInputBegin response in the SiRFDemo response view.

*Table 2-58* GSC2xr Preset Operating Configurations

| New Config | Nav Status | Config 4 | Config 3 | Config 2 | Config 1 |
|---|---|---|---|---|---|
| UARTA | | NMEA v2.2 | NMEA v2.2 | SiRF Binary | NMEA v2.2 |
| UARTB | | RTCM | RTCM | NMEA v2.2 | SiRF Binary |
| Build | | GSWx2.4.0 and greater | GSWx2.4.0 and greater | GSWx2.4.0 and greater | GSWx2.4.0 and greater, Adaptive TricklePower @ 300,1 |
| UARTA bit rate | | 4800 n, 8, 1 | 19200 n, 8, 1 | 57600 n, 8, 1 | 4800 n, 8, 1 |
| UARTB bit rate | | 9600 n, 8, 1 | 9600 n, 8, 1 | 115200 n, 8, 1 | 38400 n, 8, 1 |
| SiRF Binary Output Messages[1] | | 2, 4, 9, 13, 18, 27, 41, 52 | 2, 4, 9, 13, 18, 27, 41, 52 | 2, 4, 9, 13, 18, 27, 41, 52 | 2, 4, 9, 13, 18, 27, 41, 52 |
| NMEA Messages | | RMC, GGA, VTG, GSA (GSV@ 1/5 Hz), ZDA | GGA, GLL, GSA, GSV, RMC, VTG, ZDA | GGA, GLL, GSA, GSV, RMC, VTG, ZDA | GGA, GLL, GSA, GSV, RMC, VTG, ZDA |
| GPIO A (GPIO 1) | No Nav | On | On | On | On |
| | Nav | 100 ms on, 1 Hz | 100 ms on, 1 Hz | 100 ms on, 1 Hz | 100 ms on, 1 Hz |
| GPIO B (GPIO 3) | No Nav | Off | Off | Off | Off |
| | Nav | 100 ms on, 1 Hz | 100 ms on, 1 Hz | 100 ms on, 1 Hz | 100 ms on, 1 Hz |
| GPIO C (GPIO 13) | No Nav | On | On | On | On |
| | Nav | 1s on, 1s off | 1s on, 1s off | 1s on, 1s off | 1s on, 1s off |
| GPIO D (GPIO 2) | No Nav | Off | Off | Off | Off |
| | Nav | On | On | On | On |
| Static Filter | | Off | Off | Off | Off |
| Track Smoothing | | On | On | On | On |
| WAAS | | Disabled | Enabled | Enabled | Disabled |
| DR | | Off | Off | Off | Off |

1. SiRF Binary Messages: 2 – Measured Nav Data, 4 – Measured Track Data, 9 – Through Put, 13 – Visible List, 18 – OK to Send, 27 – DGPS Status, 41 – Geodetic Nav Data, 52 – 1 PPS Time Message.

# *Software Control – Message ID 205*

Used by GSW3 and GSWLT3 software (versions 3.2.5 or above) for generic input. Based on the Message Sub ID, there are different interpretations.

*Table 2-59* Software Control – Message ID 205

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 | | CD | | Decimal 205 |
| Message Sub ID | 1 | | 10 | | Message Sub ID |
| Data | | | | | Varies with Message Sub ID |

Payload length: Variable

# Software Commanded Off – Message ID 205 (Sub ID 16)

Shuts down the chip.

*Table 2-60* Software Commanded Off – Message ID 205 (Message Sub ID 16)

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 | | CD | | Decimal 205 |
| Message Sub ID | 1 | | 10 | | Message Sub ID for software commanded off |

Payload length: 0 bytes

# Reserved – Message ID 228

SiRF proprietary

# Extended Ephemeris – Message ID 232

Used by GSW2 (2.5 or above), SiRFXTrac (2.3 or above), and GSW3 (3.2.0 or above), and GSWLT3 software. This message has two Message Sub IDs.

*Table 2-61* Extended Ephemeris – Message ID 232

| Name | Bytes | Binary (Hex) | | Unit | Description |
|---|---|---|---|---|---|
| | | Scale | Example | | |
| Message ID | 1 | | E8 | | Decimal 232 |
| Message Sub ID | 1 | | 01 | | Message Sub ID |
| Data | | | | | Varies with Message Sub ID |

Payload length: variable (2 bytes + Message Sub ID payload bytes)

# Extended Ephemeris Proprietary – Message ID 232 (Sub ID 1)

Output Rate: Depending on the Client Location Manager (CLM)

Example:

A0A201F6—Start Sequence and Payload Length

*Table 2-62* Extended Ephemeris – Message ID 232 (Message Sub ID 1)

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 | | E8 | | | 232 |
| Message Sub ID | 1 | | 01 | | | Ephemeris input |
| SiRF Proprietary Ephemeris Format | 500 | | | | | Content proprietary |

Payload length: variable

## Format – Message ID 232 (Sub ID 2)

This message polls ephemeris status on up to 12 satellite PRNs. In response to this message, the receiver sends Message ID 56, Message Sub ID 3.

*Table 2-63* Format – Message ID 232 (Message Sub ID 2)

| Name | Bytes | Description |
|---|---|---|
| Message ID | 1 | Hex 0xE8, Decimal 232 |
| Message Sub ID | 1 | 2-Poll Ephemeris Status |
| SVID Mask | 4 | Bitmapped Satellite PRN[1] |

Payload length: 6 bytes

1. SVID Mask is a 32-bit value with a 1 set in each location for which ephemeris status is requested. Bit 0 represents PRN 1, …, Bit 31 represents PRN 32. If more than 12 bits are set, the response message responds with data on only the 12 lowest PRNs requested.

## Extended Ephemeris Debug – Message ID 232 (Sub ID 255)

Example:

A0A20006—Start Sequence and Payload Length

E8FF01000000 – Payload

01E8B0B3—Message Checksum and End Sequence

*Table 2-64* Extended Ephemeris – Message ID 232 (Message Sub ID 255)

| | | Binary (Hex) | | | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| Name | Bytes | Scale | Example | Unit | Scale | Example |
| Message ID | 1 | | E8 | | | 232 |
| Message Sub ID | 1 | | FF | | | 255-EE Debug |
| DEBUG_FLAG | 4 | | | | | Proprietary |

Payload length: 6 bytes

# Output Messages 3

This chapter provides information about available SiRF Binary output messages. For each message, a full definition and example is provided.

*Table 3-1* SiRF Binary Messages – Output Message List

| Hex | Decimal | Name | Description |
|---|---|---|---|
| 01 | 1 | Reference Navigation Data | Not Implemented |
| 02 | 2 | Measured Navigation Data | Position, velocity, and time |
| 03 | 3 | True Tracker Data | Not Implemented |
| 04 | 4 | Measured Tracking Data | Satellite and C/N0 information |
| 05 | 5 | Raw Track Data | Not supported by SiRFstarII |
| 06 | 6 | SW Version | Receiver software |
| 07 | 7 | Clock Status | Current clock status |
| 08 | 8 | 50 BPS Subframe Data | Standard ICD format |
| 09 | 9 | Throughput | Navigation complete data |
| 0A | 10 | Error ID | Error coding for message failure |
| 0B | 11 | Command Acknowledgment | Successful request |
| 0C | 12 | Command NAcknowledgment | Unsuccessful request |
| 0D | 13 | Visible List | Auto Output |
| 0E | 14 | Almanac Data | Response to poll |
| 0F | 15 | Ephemeris Data | Response to poll |
| 10 | 16 | Test Mode 1 | For use with SiRFtest (Test Mode 1) |
| 11 | 17 | Differential Corrections | Received from DGPS broadcast |
| 12 | 18 | OkToSend | CPU ON / OFF (TricklePower) |
| 13 | 19 | Navigation Parameters | Response to Poll |
| 14 | 20 | Test Mode 2/3/4 | Test Mode 2, 3, or 4 test data |
| 1B | 27 | DGPS Status | Differential GPS status information |
| 1C | 28 | Nav. Lib. Measurement Data | Measurement data |
| 1D | 29 | Nav. Lib. DGPS Data | Differential GPS data |
| 1E | 30 | Nav. Lib. SV State Data | Satellite state data |
| 1F | 31 | Nav. Lib. Initialization Data | Initialization data |
| 29 | 41 | Geodetic Navigation Data | Geodetic navigation information |
| 2B | 43 | Queue Command Parameters | Command parameters |
| 2D | 45 | Raw DR Data | Raw DR data from ADC |
| 2E | 46 | Test Mode 3 & 4 (GSW3 & SLC3) | Test data (Test Mode 3 and 4) |
| 30 | 48[1] | Test Mode 4 for SiRFLoc v2.x only | Test data (Test Mode 4) |

*Table 3-1* SiRF Binary Messages – Output Message List *(Continued)*

| Hex | Decimal | Name | Description |
|---|---|---|---|
| 30 | 48 | SiRF Dead Reckoning Class of Output Messages | The Message ID is partitioned into messages identified by Message Sub IDs, refer to Table 3-2 |
| 31 | 49 | Test Mode 4 for SiRFLoc v2.x only | Additional test data (Test Mode 4) |
| 32 | 50 | SBAS Parameters | SBAS operating parameters |
| 34 | 52 | 1 PPS Time Message | Time message for 1 PPS |
| 37 | 55 | Test Mode 4 | Track Data |
| 38 | 56 | Extended Ephemeris Data | Extended Ephemeris Mask & Integrity Information |
| E1 | 225 | SiRF internal message | Reserved |
| FF | 255 | Development Data | Various status messages |

1. This Message ID 48 for Test Mode 4 is not to be confused with Message ID 48 for DR Navigation. SiRFLoc v2 Message ID 48 will be transferred to a different Message ID in the near future.

*Table 3-2* Message Sub IDs for SiRFDRive and SiRFDiRect Output – Message ID 48 (0x30)

| Sub ID | Message ID | SiRFDRive 1 | SiRFDRive 2 | SiRFDiRect |
|---|---|---|---|---|
| 1 | DR Navigation Status | Yes | Yes | Yes |
| 2 | DR Navigation State | Yes | Yes | Yes |
| 3 | Navigation Subsystem | Yes | Yes | Yes |
| 4 | Raw DRData (not implemented) | No | No | No |
| 5 | DR Validity | No | No | No |
| 6 | DR Gyro Factory Calibration | Yes | No | No |
| 7 | DR Sensors Parameters | Yes | No | No |
| 8 | DR Data Block | Yes | No | No |
| 9 | Generic Sensor Parameters (not implemented) | No | No | No |

Since the SiRF Binary protocol is evolving along with continued development of SiRF software and GPS solutions, not all SiRF Binary messages are supported by all SiRF GPS solutions.

Table 3-3 identifies the supported output messages for each SiRF architecture.

*Table 3-3* Supported Output Messages

| Message ID | SiRF Software Options | | | | | |
|---|---|---|---|---|---|---|
| | GSW2 | SiRFDRive | SiRFXTrac | SiRFLoc | GSW3 & GSWLT3 | SiRFDiRect |
| 1 | No | No | No | No | No | No |
| 2 | Yes | Yes | Yes | Yes | Yes | Yes |
| 3 | No | No | No | No | No | No |
| 4 | Yes | Yes | Yes | Yes | Yes | Yes |
| 5 | No | No | No | No | No | No |
| 6 | Yes | Yes | Yes | Yes | Yes | Yes |
| 7 | Yes | Yes | Yes | Yes | Yes | Yes |
| 8 | Yes | Yes | Yes | Yes | Yes | No |
| 9 | Yes | Yes | Yes | Yes | Yes GSW3; No GSWLT3 | No |
| 10 | Yes | Yes | Yes | Yes | Yes | Yes |
| 11 | Yes | Yes | Yes | Yes | Yes | Yes |
| 12 | Yes | Yes | Yes | Yes | Yes | Yes |

*Table 3-3* Supported Output Messages *(Continued)*

| Message ID | SiRF Software Options | | | | | |
|---|---|---|---|---|---|---|
| | GSW2 | SiRFDRive | SiRFXTrac | SiRFLoc | GSW3 & GSWLT3 | SiRFDiRect |
| 13 | Yes | Yes | Yes | Yes | Yes | No |
| 14 | Yes | Yes | No | Yes | Yes | Yes |
| 15 | Yes | Yes | No | Yes | Yes | Yes |
| 16 | Yes | Yes | No | No | No | No |
| 17 | Yes | Yes | No | No | No | No |
| 18 | Yes | Yes | Yes | Yes | Yes | Yes |
| 19 | Yes | Yes | Yes | Yes | Yes | Yes |
| 20 | Test Mode 2 only | Test Mode 2 only | Test Modes 2/3/4 | Test Mode 4 (2.x only) | No | No |
| 27 | Yes | Yes | No | No | Yes | No |
| 28 | Yes | Yes | No | No | Yes | Yes |
| 29 | Yes | Yes | No | No | No | No |
| 30 | Yes | Yes | No | No | Yes | Yes |
| 31 | Yes | Yes | No | No | Yes | Yes |
| 41 | 2.3 & above | Yes | 2.0 & above | No | Yes | Yes |
| 43 | No | No | No | No | Yes | Yes |
| 45 | No | Yes | No | No | No | No |
| 46 | Yes | Yes | No | 3.x & above | Yes | Yes |
| 48[1] (Test Mode 4) | No | No | No | 2.x only | No | No |
| 48 (DR) | No | Yes[2] | No | No | No | Yes[2] |
| 49 | No | No | No | 2.x only | No | No |
| 50 | 2.3 & above | Yes | No | No | 3.2.5 & above | No |
| 52 | 2.3.2 & above | No | No | No | No | No |
| 55 | No | No | No | 3.x & above | Yes | Yes |
| 56 | 2.5 & above | No | 2.3 & above | No | 3.2.5 & above | No |
| 56 (Sub ID 4) | No | Yes | No | No | 3.2.5 & above | No |
| 225 | No | No | No | No | Yes (reserved) | No |
| 232 | No | No | No | No | Yes | Yes |
| 255 | Yes | Yes | Yes | Yes | Yes | No |

1. This Message ID 48 for Test Mode 4 is not to be confused with Message ID 48 for DR Navigation. Message ID 48 for SiRFLoc will be transferred to a different Message ID in the near future.

2. Not all Message Sub IDs supported.

## *Reference Navigation Data – Message ID 1*

This message is defined as Reference Navigation data but has not been implemented.

## *Measure Navigation Data Out – Message ID 2*

Output Rate: 1 Hz

Table 3-4 lists the message data format for the measured navigation data.

Example:

A0A20029—Start Sequence and Payload Length

02FFD6F78CFFBE536E003AC004000000030001040A00036B039780E3
0612190E160F04000000000000—Payload

09BBB0B3—Message Checksum and End Sequence

*Table 3-4* Measured Navigation Data Out – Message ID 2

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | **Scale** | **Example** | | **Scale** | **Example** |
| Message ID | 1 U | | 02 | | | 2 |
| X-position | 4 S | | FFD6F78C | m | | -2689140 |
| Y-position | 4 S | | FFBE536E | m | | -4304018 |
| Z-position | 4 S | | 003AC004 | m | | 3850244 |
| X-velocity | 2 S | *8 | 0000 | m/sec | Vx÷8 | 0 |
| Y-velocity | 2 S | *8 | 0003 | m/sec | Vy÷8 | 0.375 |
| Z-velocity | 2 S | *8 | 0001 | m/sec | Vz÷8 | 0.125 |
| Mode 1 | 1 D | | 04 | Bitmap[1] | | 4 |
| HDOP[2] | 1 U | *5 | 0A | | $^{3}5$ | 2.0 |
| Mode 2 | 1 D | | 00 | Bitmap[3] | | 0 |
| GPS Week[4] | 2 U | | 036B | | | 875 |
| GPS TOW | 4 U | *100 | 039780E3 | sec | ÷100 | 602605.79 |
| SVs in Fix | 1 U | | 06 | | | 6 |
| CH 1 PRN[5] | 1 U | | 12 | | | 18 |
| CH 2 PRN[5] | 1 U | | 19 | | | 25 |
| CH 3 PRN[5] | 1 U | | 0E | | | 14 |
| CH 4 PRN[5] | 1 U | | 16 | | | 22 |
| CH 5 PRN[5] | 1 U | | 0F | | | 15 |
| CH 6 PRN[5] | 1 U | | 04 | | | 4 |
| CH 7 PRN[5] | 1 U | | 00 | | | 0 |
| CH 8 PRN[5] | 1 U | | 00 | | | 0 |
| CH 9 PRN[5] | 1 U | | 00 | | | 0 |
| CH 10 PRN[5] | 1 U | | 00 | | | 0 |
| CH 11 PRN[5] | 1 U | | 00 | | | 0 |
| CH 12 PRN[5] | 1 U | | 00 | | | 0 |

Payload length: 41 bytes

1. For further information see Table 3-5 and Table 3-6. Note that the Degraded Mode positioning mode is not supported in GSW3.2.5 and newer
2. HDOP value reported has a maximum value of 50.
3. For further information see Table 3-7.
4. GPS week reports only the ten LSBs of the actual week number.
5. PRN values are reported only for satellites used in the navigation solution.

---

**Note –** Binary units scaled to integer values must be divided by the scale value to receive true decimal value (i.e., decimal $X_{vel}$ = binary $X_{vel}$ ÷ 8).

---

Mode 1 of Message ID 2 is a bit-mapped byte with five sub-values. Table 3-5 shows the location of the sub-values and Table 3-6 shows the interpretation of each sub-value.

*Table 3-5* Mode 1

| **Bit** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
|---|---|---|---|---|---|---|---|---|
| **Bit(s) Name** | DGPS | DOP-Mask | ALTMODE | | TPMODE | PMODE | | |

*Table 3-6* Mode 1 Bitmap Information

| Bit(s) Name | Name | Value | Description |
|---|---|---|---|
| PMODE | Position mode | 0 | No navigation solution |
| | | 1 | 1-SV solution (Kalman filter) |
| | | 2 | 2-SV solution (Kalman filter) |
| | | 3 | 3-SV solution (Kalman filter) |
| | | 4 | > 3-SV solution (Kalman filter) |
| | | 5 | 2-D point solution (least squares) |
| | | 6 | 3-D point solution (least squares) |
| | | 7 | Dead-Reckoning[1]solution (no satellites) |
| TPMODE | TricklePower mode | 0 | Full power position |
| | | 1 | TricklePower position |
| ALTMODE | Altitude mode | 0 | No altitude hold applied |
| | | 1 | Holding of altitude from KF |
| | | 2 | Holding of altitude from user input |
| | | 3 | Always hold altitude (from user input) |
| DOPMASK | DOP mask status | 0 | DOP mask not exceeded |
| | | 1 | DOP mask exceeded |
| DGPS | DGPS status | 0 | No differential corrections applied |
| | | 1 | Differential corrections applied |

1. In standard software, Dead Reckoning solution is computed by taking the last valid position and velocity and projecting the position using the velocity and elapsed time.

Mode 2 of Message ID bit-mapped byte information is described in Table 3-7.

*Table 3-7* Mode 2 Bitmap

| Bit | Description |
|---|---|
| 0[1] | 1 = sensor DR in use<br>0 = velocity DR if PMODE sub-value in Mode<br>1 = 7;else check Bits 6 & 7 for DR error status |
| 1[2] | If set, solution is validated (5 or more SVs used)[3] |
| 2 | If set, velocity DR timeout |
| 3 | If set, solution edited by UI (e.g., DOP Mask exceeded) |
| 4[4] | If set, velocity is invalid |
| 5 | Altitude hold mode:<br>0 = enabled<br>1 = disabled (3-D fix only) |
| 7,6[5] | Sensor DR error status:<br>00 = GPS-only navigation<br>01 = DR in calibration<br>10 = DR sensor errors<br>11 = DR in test mode |

1. Bit 0 is controlled by the acquisition hardware. The rest of the bits are controlled by the tracking hardware, except that in SiRFstarIII receivers, bit 2 is also controlled by the acquisition hardware.
2. Bit 1 set means that the phase relationship between the I and Q samples is being tracked.
3. From an unvalidated state, a 5-SV fix must be achieved to become a validated position. If the receiver continues to navigate in a degraded mode (less than 4 SVs), the validated status remains. If navigation is lost completely, an unvalidated status results.
4. Bit 4 set means that the Doppler corrections have been made so that the phase between the I and Q samples is stable.
5. Generally, bit 6 cannot be set at the same time other bits are set. However, some firmware versions use the special case of setting

**Note –** Mode 2 of Message ID 2 is used to define the Fix field of the Measured Navigation Message View. It should be used only as an indication of the current fix status of the navigation solution and not as a measurement of TTFF.

## True Tracker Data – Message ID 3

Defined as True Tracker data, but not yet implemented.

## Measured Tracker Data Out – Message ID 4

Output Rate: 1 Hz

Table 3-8 lists the message data format for the measured tracker data.

Example:

A0A200BC—Start Sequence and Payload Length

04036C0000937F0C0EAB46003F1A1E1D1D191D1A1A1D1F1D59423F1A1A...—Payload

....B0B3—Message Checksum and End Sequence

*Table 3-8* Measured Tracker Data Out – Message ID 4

| | | Binary (Hex) | | | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| **Name** | **Bytes** | **Scale** | **Example** | **Unit** | **Scale** | **Example** |
| Message ID | 1 U | | 04 | | | 4 |
| GPS Week[1] | 2 S | | 036C | | | 876 |
| GPS TOW | 4 U | s*100 | 0000937F | sec | s÷100 | 37759 |
| Chans | 1 U | | 0C | | | 12 |
| 1st SVid | 1 U | | 0E | | | 14 |
| Azimuth | 1 U | Az*[2/3] | AB | deg | $^{3}$[2/3] | 256.5 |
| Elev | 1 U | El*2 | 46 | deg | $^{3}$2 | 35 |
| State | 2 D | | 003F | Bitmap[2] | | 63 |
| C/N0 1 | 1 U | | 1A | dB-Hz | | 26 |
| C/N0 2 | 1 U | | 1E | dB-Hz | | 30 |
| C/N0 3 | 1 U | | 1D | dB-Hz | | 29 |
| C/N0 4 | 1 U | | 1D | dB-Hz | | 29 |
| C/N0 5 | 1 U | | 19 | dB-Hz | | 25 |
| C/N0 6 | 1 U | | 1D | dB-Hz | | 29 |
| C/N0 7 | 1 U | | 1A | dB-Hz | | 26 |
| C/N0 8 | 1 U | | 1A | dB-Hz | | 26 |
| C/N0 9 | 1 U | | 1D | dB-Hz | | 29 |
| C/N0 10 | 1 U | | 1F | dB-Hz | | 31 |
| 2nd SVid | 1 U | | 1D | | | 29 |
| Azimuth | 1 U | Az*[2/3] | 59 | deg | $^{3}$[2/3] | 89 |
| Elev | 1 U | El*2 | 42 | deg | $^{3}$2 | 66 |
| State | 2 D | | 003F | Bitmap[2] | | 63 |
| C/N0 1 | 1 U | | 1A | dB-Hz | | 26 |
| C/N0 2 | 1 U | | 1A | dB-Hz | | 63 |
| ... | | | | | | |
| SVid, Azimuth, Elevation, State, and C/N0 1-10 values are repeated for each of the 12 channels | | | | | | |

Payload length: 188 bytes

1. GPS week number is reported modulo 1024 (ten LSBs only).

2. For further information, see Table 3-9 for state values for each channel.

*Table 3-9* State Values for Each Channel

| Bit | Description When Bit is Set to 1 |
|---|---|
| 0[1] | Acquisition/re-acquisition has been completed successfully |
| 1[2] | The integrated carrier phase is valid – delta range in Message ID 28 is also valid |
| 2 | Bit synchronization has been completed |
| 3 | Subframe synchronization has been completed |
| 4[3] | Carrier pullin has been completed (Costas lock) |
| 5 | Code has been locked |
| 6[4,5] | Satellite acquisition has failed |
| 7 | Ephemeris data is available |
| 8-15 | Reserved |

1. Bit 0 is controlled by the acquisition hardware. The rest of the bits are controlled by the tracking hardware except in SiRFstarIII receivers, where bit 2 is also controlled by the acquisition hardware.
2. Bit 1 set means that the phase relationship between the I and Q samples is being tracked.
3. Bit 4 set means that the Doppler corrections have been made so that the phase between the I and Q samples is stable.
4. Generally, bit 6 cannot be set at the same time other bits are set. However, some firmware versions use the special case of setting all bits 0-7 to 1 (0xFF) to indicate that this channel is being used to test the indicated PRN for an auto or cross correlation.
5. Bit 6 is typically set to1 only when other bits are turned off. However, a special situation exists: when all bits are on (value 0xFF) there is a special meaning: this channel is being used to test for auto- and cross-correlations rather than tracking a satellite for use in the solution. When a 0xFF state exists in a channel, there often will be another channel that is actually tracking the SV PRN value shown.

## *Raw Tracker Data Out – Message ID 5*

This message is not supported by the SiRFstarII or SiRFstarIII architecture.

## *Software Version String (Response to Poll) – Message ID 6*

This message has a variable length from 1 to 81 bytes.

Output Rate: Response to polling message

Example:

A0A2001F—Start Sequence and Payload Length

06322E332E322D475358322D322E30352E3032342D4331464C4558312E32 —Payload

0631B0B3—Message Checksum and End Sequence

*Table 3-10* Software Version String – Message ID 6

| | | Binary (Hex) | | | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| Name | Bytes | Scale | Example | Unit | Scale | Example |
| Message ID | 1 U | | 06 | | | 6 |
| Character [80] | 1 U | | [1] | | | [2] |

Payload Length:1-81 bytes

1. Payload example is shown above.
2. 2.3.2-GSW2-2.05.024-C1FLEX1.2

**Note –** Convert ASCII to symbol to assemble message (i.e., 0x4E is 'N'). Effective with version GSW 2.3.2, message length was increased from 21 to 81 bytes to allow for up to an 80-character version string.

## Response: Clock Status Data – Message ID 7

This message is output as part of each navigation solution. It tells the actual time of the measurement (in GPS time), and gives the computed clock bias and drift information computed by the navigation software.

Control of this message is unique. In addition to being able to control it using the message rate commands, it also acts as part of the "Navigation Library" messages controlled by bit 4 of the Reset Configuration Bit Map field of message ID 128. When navigation library messages are enabled or disabled, this message is enabled or disabled. It is also enabled by default whenever a system reset occurs.

Output Rate: 1 Hz or response to polling message

Example:

A0A20014—Start Sequence and Payload Length

0703BD021549240800012231000047281 4D4DAEF—Payload

0598B0B3—Message Checksum and End Sequence

*Table 3-11* Clock Status Data – Message ID 7

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 07 | | | 7 |
| Extended GPS Week | 2 U | | 03BD | | | 957 |
| GPS TOW | 4 U | *100 | 02154924 | sec | ÷100 | 349494.12 |
| SVs | 1 U | | 08 | | | 8 |
| Clock Drift | 4 U | | 00012231 | Hz | | 74289 |
| Clock Bias | 4 U | | 00004728 | ns | | 18216 |
| Estimated GPS Time | 4 U | | 14D4DAEF | ms | | 349493999 |

Payload length: 20 bytes

*Table 3-12* Detailed Description of Message ID 7 Fields

| Field | Description |
|---|---|
| Extended GPS Week | GPS week number is reported by the satellites with only 10 bits. The receiver extends that number with any higher bits and reports the full resolved week number in this message. |
| GPS TOW | Seconds into the current week, accounting for clock bias, when the current measurement was made. This is the true GPS time of the solution. |
| SVs | Total number of satellites used to compute this solution. |
| Clock Drift[1] | Rate of change of the Clock Bias. Clock Drift is a direct result of the GPS crystal frequency, so it is reported in Hz. |
| Clock Bias | This is the difference in nanoseconds between GPS time and the receiver's internal clock. In different SiRF receivers this value has different ranges, and as the computed bias approaches the limit of the range, the next measurement interval will be adjusted to be longer or shorter so that the bias remains in the selected range. |
| Estimated GPS Time[2] | This is the GPS time of the measurement, estimated before the navigation solution is computed. Due to variations in clock drift and other factors, this will normally not equal GPS TOW, which is the true GPS time of measurement computed as part of the navigation solution. |

1. Clock Drift in SiRF receivers is directly related to the frequency of the GPS clock, derived from the GPS crystal. From the reported frequency, you can compute the GPS clock frequency, and you can predict the next clock bias. Clock drift also appears as a Doppler bias in Carrier Frequency reported in Message ID 28.

2. Estimated GPS time is the time estimated when the measurements were made. Once the measurements were made, the GPS navigation solution was computed, and true GPS time was computed. Variations in clock drift and measurement intervals generally make the estimate slightly wrong, which is why GPS TOW and Estimated GPS time typically disagree at the microsecond level.

For detailed information about computing GPS clock frequency, see “Computing GPS Clock Frequency” in Chapter 4.

## 50 BPS Data – Message ID 8

Output Rate: Approximately every six seconds for each channel

Example:

A0A2002B—Start Sequence and Payload Length

08001900C0342A9B688AB0113FDE2D714FA0A7FFFACC5540157EFFEEDFFF A80365A867FC67708BEB5860F4—Payload

15AAB0B3—Message Checksum and End Sequence

*Table 3-13* 50 BPS Data – Message ID 8

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 08 | | | 8 |
| Channel | 1 U | | 00 | | | 0 |
| SV ID | 1 U | | 19 | | | 25 |
| Word[10] | 4 U | | | | | |

Payload length: 43 bytes

## CPU Throughput – Message ID 9

Output Rate: 1 Hz

Example:

A0A20009—Start Sequence and Payload Length

09003B0011001601E5—Payload

0151B0B3—Message Checksum and End Sequence

*Table 3-14* CPU Throughput – Message ID 9

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 09 | | | 9 |
| SegStatMax | 2 U | *186 | 003B | ms | $^3$186 | 0.3172 |
| SegStatLat | 2 U | *186 | 0011 | ms | ÷186 | 0.0914 |
| AveTrkTime | 2 U | *186 | 0016 | ms | ÷186 | 0.1183 |
| Last Millisecond | 2 U | | 01E5 | ms | | 485 |

Payload length: 9 bytes

## Error ID Data – Message ID 10

Output Rate: As errors occur

Message ID 10 messages have a different format from other messages. Rather than one fixed format, there are several formats, each designated by an error ID. However, the format is standardized as indicated in Table 3-15. The specific format of each error ID message follows

*Table 3-15* Message ID 10 Overall Format

| Name | Bytes | Description |
|---|---|---|
| Message ID | 1 U | Message ID number - 10 |
| Error ID | 2 U | Sub-message type |
| Count | 2 U | Count of number of 4-byte values that follow |
| Data[n] | 4 U | Actual data for the message, *n* is equal to Count |

## *Error ID: 2*

Code Define Name:ErrId_CS_SVParity

Error ID Description:Satellite subframe # failed parity check.

Example:

A0A2000D – Start Sequence and Payload Length

0A0002000200000001000000002 – Payload

0011B0B3 – Message Checksum and End Sequence

*Table 3-16* Error ID

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 0A | | | 10 |
| Error ID | 2 U | | 0002 | | | 2 |
| Count | 2 U | | 0002 | | | 2 |
| Satellite ID | 4 U | | 00000001 | | | 1 |
| Subframe No | 4 U | | 00000002 | | | 2 |

Payload Length: 13 bytes

*Table 3-17* Error ID 2 Message Description

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Error ID | Error ID (see Error ID description above) |
| Count | Number of 32 bit data in message |
| Satellite ID | Satellite pseudo-random noise (PRN) number |
| Subframe No | The associated subframe number that failed the parity check. Valid subframe number is 1 through 5. |

## *Error ID: 9*

Code Define Name:ErrId_RMC_GettingPosition

Error ID Description:Failed to obtain a position for acquired satellite ID.

Example:

A0A20009 – Start Sequence and Payload Length

0A0009000100000001 – Payload

0015B0B3 – Message Checksum and End Sequence

*Table 3-18* Error ID 9 Message

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 0A | | | 10 |
| Error ID | 2 U | | 0009 | | | 9 |
| Count | 2 U | | 0002 | | | 2 |
| Satellite ID | 4 U | | 00000001 | | | 1 |

Payload Length: 9 bytes

*Table 3-19* Error ID 9 Message Description

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Error ID | Error ID (see Error ID description above) |
| Count | Number of 32 bit data in message |
| Satellite ID | Satellite pseudo-random noise code number |

## *Error ID: 10*

Code Define Name:ErrId_RXM_TimeExceeded

Error ID Description:Conversion of Nav Pseudo Range to Time of Week (TOW) for tracker exceeds limits: Nav Pseudo Range > 6.912e5 (1 week in seconds) || Nav Pseudo Range < -8.64e4.

Example:

A0A20009 – Start Sequence and Payload Length

0A000A000100001234 – Payload

005BB0B3 – Message Checksum and End Sequence

*Table 3-20* Error ID 10 Message

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 0A | | | 10 |
| Error ID | 2 U | | 000A | | | 10 |
| Count | 2 U | | 0001 | | | 1 |
| Pseudorange | 4 U | | 00001234 | | | 4660 |

Payload length: 9 bytes

*Table 3-21* Error ID 10 Message Description

| Name | Description |
|---|---|
| Message ID | Message ID number. |
| Error ID | Error ID (see Error ID description above). |
| Count | Number of 32 bit data in message. |
| Pseudorange | Pseudo range. |

## Error ID: 11

Code Define Name:ErrId_RXM_TDOPOverflow

Error ID Description:Convert pseudorange rate to Doppler frequency exceeds limit.

Example:

A0A20009 – Start Sequence and Payload Length

0A000B0001xxxxxxxx – Payload

xxxxB0B3 – Message Checksum and End Sequence

*Table 3-22* Error ID 11 Message

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 0A | | | 10 |
| Error ID | 2 U | | 000B | | | 11 |
| Count | 2 U | | 0001 | | | 1 |
| Doppler Frequency | 4 U | | xxxxxxxx | | | xxxxxxxx |

Payload length: 9 bytes

*Table 3-23* Error ID 11 Message Description

| Name | Description |
|---|---|
| Message ID | Message ID number. |
| Error ID | Error ID (see Error ID description above). |
| Count | Number of 32 bit data in message. |
| Doppler Frequency | Doppler frequency. |

## Error ID: 12

Code Define Name:ErrId_RXM_ValidDurationExceeded

Error ID Description:Satellite ephemeris age has exceeded 2 hours (7200 s).

Example:

A0A2000D – Start Sequence and Payload Length

0A000C0002xxxxxxxxaaaaaaaa – Payload

xxxxB0B3 – Message Checksum and End Sequence

*Table 3-24* Error ID 12 Message

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 0A | | | 10 |
| Error ID | 2 U | | 000C | | | 12 |
| Count | 2 U | | 0002 | | | 2 |
| Satellite ID | 4 U | | xxxxxxxx | | | xxxxxxxx |
| Age Of Ephemeris | 4 U | | aaaaaaaa | sec | | aaaaaaaa |

Payload Length: 13 bytes

*Table 3-25* Error ID 12 Message Description

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Error ID | Error ID (see Error ID description above) |
| Count | Number of 32 bit data in message |
| Satellite ID | Satellite pseudo-random noise number |
| Age of Ephemeris | The satellite ephemeris age in seconds |

## *Error ID: 13*

Code Define Name:ErrId_STRTP_BadPostion

Error ID Description:SRAM position is bad during a cold start.

Example:

A0A20011 – Start Sequence and Payload Length

0A000D0003xxxxxxxxaaaaaaaabbbbbbbb – Payload

xxxxB0B3 – Message Checksum and End Sequence0

*Table 3-26* Error ID 13 Message

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 0A | | | 10 |
| Error ID | 2 U | | 000D | | | 13 |
| Count | 2 U | | 0003 | | | 3 |
| X | 4 U | | xxxxxxxx | | | xxxxxxxx |
| Y | 4 U | | aaaaaaaa | | | aaaaaaaa |
| Z | 4 U | | bbbbbbbb | | | bbbbbbbb |

Payload length: 17 bytes

*Table 3-27* Error ID 13 Message Description

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Error ID | Error ID (see Error ID description above) |
| Count | Number of 32 bit data in message |
| X | X position in ECEF |
| Y | Y position in ECEF |
| Z | Z position in ECEF |

## *Error ID: 4097 (0x1001)*

Code Define Name:ErrId_MI_VCOClockLost

Error ID Description:VCO lost lock indicator.

Example:

A0A20009 – Start Sequence and Payload Length

0A1001000100000001 – Payload

001DB0B3 – Message Checksum and End Sequence

*Table 3-28* Error ID 4097 Message

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 0A | | | 10 |
| Error ID | 2 U | | 1001 | | | 4097 |
| Count | 2 U | | 0001 | | | 1 |
| VCOLost | 4 U | | 00000001 | | | 1 |

Payload length: 9 bytes

*Table 3-29* Error ID 4097 Message Description

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Error ID | Error ID (see Error ID description above) |
| Count | Number of 32 bit data in message |
| VCOLost | VCO lock lost indicator. If VCOLost ! = 0, then send failure message |

## *Error ID: 4099 (0x1003)*

Code Define Name:ErrId_MI_FalseAcqReceiverReset

Error ID Description:Nav detect false acquisition, reset receiver by calling NavForceReset routine.

Example:

A0A20009 – Start Sequence and Payload Length

0A1003000100000001 – Payload

001FB0B3 – Message Checksum and End Sequence

*Table 3-30* Error ID 4099 Message

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 0A | | | 10 |
| Error ID | 2 U | | 1003 | | | 4099 |
| Count | 2 U | | 0001 | | | 1 |
| InTrkCount | 4 U | | 00000001 | | | 1 |

Payload Length: 9 bytes

*Table 3-31* Error ID 4099 Message Description

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Error ID | Error ID (see Error ID description above) |
| Count | Number of 32 bit data in message |
| InTrkCount | False acquisition indicator. If InTrkCount < = 1, then send failure message and reset receiver |

## Error ID: 4104 (0x1008)

Code Define Name:ErrId_STRTP_SRAMCksum

Error ID Description:Failed SRAM checksum during startup.

- Four field message indicates receiver control flags had checksum failures.
- Three field message indicates clock offset checksum failure or clock offset value is out of range.
- Two field message indicates position and time checksum failure forces a cold start.

Example:

A0A2xxxx – Start Sequence and Payload Length

0A10080004xxxxxxxaaaaaaaa00000000cccccccc – Payload

xxxxB0B3 – Message Checksum and End Sequence

*Table 3-32* Error ID 4104 Message

| | | Binary (Hex) | | | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| **Name** | **Bytes** | **Scale** | **Example** | **Unit** | **Scale** | **Example** |
| Message ID | 1 U | | 0A | | | 10 |
| Error ID | 2 U | | 1008 | | | 4104 |
| Count | 2 U | | 0004 or 0003 or 0002 | | | 4 or 3 or 2 |
| Computed Receiver Control Checksum | 4 U | | xxxxxxxx | | | xxxx |
| NVRAM Receiver Control Checksum | 4 U | | aaaaaaaa | | | aaaa |
| NVRAM Receiver Control OpMode | 4 U | | 00000000 | | | 0 |
| NVRAM Receiver Control Channel Count | 4 U | | cccccccc | | | cccc |
| Compute Clock Offset Checksum | 4 U | | xxxxxxxx | | | xxxx |
| NVRAM Clock Offset Checksum | 4 U | | aaaaaaaa | | | aaaa |
| NVRAM Clock Offset | 4 U | | bbbbbbbb | | | bbbb |
| Computed Position Time Checksum | 4 U | | xxxxxxxx | | | xxxx |
| NVRAM Position Time Checksum | 4 U | | aaaaaaaa | | | aaaa |

Payload length: 21, 17, or 11 bytes

*Table 3-33* Error ID 4104 Message Description

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Error ID | Error ID (see Error ID description above) |
| Count | Number of 32 bit data in message |
| Computed Receiver Control Checksum | Computed receiver control checksum of SRAM.Data.Control structure |
| NVRAM Receiver Control Checksum | NVRAM receiver control checksum stored in SRAM.Data.DataBuffer. CntrlChkSum. |
| NVRAM Receiver Control OpMode | NVRAM receiver control checksum stored in SRAM.Data.Control.OpMode. Valid OpMode values are as follows:<br>OP_MODE_NORMAL = 0<br>OP_MODE_TESTING = 0x1E51<br>OP_MODE_TESTING2 = 0x1E52<br>OP_MODE_TESTING3 = 0x1E53 |
| NVRAM Receiver Control Channel Count | NVRAM receiver control channel count in SRAM.Data.Control.ChannelCnt Valid channel count values are 0-12 |

*Table 3-33* Error ID 4104 Message Description *(Continued)*

| Name | Description |
|---|---|
| Compute Clock Offset Checksum | Computed clock offset checksum of SRAM.Data.DataBuffer.clkOffset. |
| NVRAM Clock Offset Checksum | NVRAM clock offset checksum of SRAM.Data.DataBuffer.clkChkSum |
| NVRAM Clock Offset | NVRAM clock offset value stored in SRAM.Data.DataBuffer,clkOffset |
| Computed Position Time Checksum | Computed position time checksum of SRAM.Data.DataBuffer.postime[1] |
| NVRAM Position Time Checksum | NVRAM position time checksum of SRAM.Data.DataBuffer.postimeChkSum[1] |

## *Error ID: 4105 (0x1009)*

Code Define Name:ErrId_STRTP_RTCTimeInvalid

Error ID Description:Failed RTC SRAM checksum during startup. If one of the double buffered SRAM.Data.LastRTC elements is valid and RTC days is not 255 days, the GPS time and week number computed from the RTC is valid. If not, this RTC time is invalid.

Example:

A0A2000D – Start Sequence and Payload Length

0A10090002xxxxxxxxaaaaaaaa – Payload

xxxxB0B3 – Message Checksum and End Sequence

*Table 3-34* Error ID 4105 Message

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 0A | | | 10 |
| Error ID | 2 U | | 1009 | | | 4105 |
| Count | 2 U | | 0002 | | | 2 |
| TOW | 4 U | | xxxxxxxx | sec | | xxxx |
| Week Number | 4 U | | aaaaaaaa | | | aaaa |

Payload length: 13 bytes

*Table 3-35* Error ID 4105 Message Description

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Error ID | Error ID (see Error ID description above) |
| Count | Number of 32 bit data in message |
| TOW | GPS time of week in seconds. Range 0 to 604800 seconds |
| Week Number | GPS week number |

## *Error ID: 4106 (0x100A)*

Code Define Name:ErrId_KFC_BackupFailed_Velocity

Error ID Description: Failed saving position to NVRAM because the ECEF velocity sum was greater than 3600.

Example:

A0A20005 – Start Sequence and Payload Length

0A100A0000 – Payload

0024B0B3 – Message Checksum and End Sequence

*Table 3-36* Error ID 4106 Message

| | | Binary (Hex) | | | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| **Name** | **Bytes** | **Scale** | **Example** | **Unit** | **Scale** | **Example** |
| Message ID | 1 U | | 0A | | | 10 |
| Error ID | 2 U | | 100A | | | 4106 |
| Count | 2 U | | 0000 | | | 0 |

Payload length: 5 bytes

*Table 3-37* Error ID 4106 Message Description

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Error ID | Error ID (see Error ID description above) |
| Count | Number of 32 bit data in message |

## *Error ID: 4107 (0x100B)*

Code Define Name:ErrId_KFC_BackupFailed_NumSV

Error ID Description: Failed saving position to NVRAM because current navigation mode is not KFNav and not LSQFix.

Example:

A0A20005 – Start Sequence and Payload Length

0A100B0000 – Payload

0025B0B3 – Message Checksum and End Sequence

*Table 3-38* Error ID 4107 Message

| | | Binary (Hex) | | | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| **Name** | **Bytes** | **Scale** | **Example** | **Unit** | **Scale** | **Example** |
| Message ID | 1 U | | 0A | | | 10 |
| Error ID | 2 U | | 100B | | | 4107 |
| Count | 2 U | | 0000 | | | 0 |

Payload length: 5 bytes

*Table 3-39* Error ID 4107 Message Description

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Error ID | Error ID (see Error ID description above) |
| Count | Number of 32 bit data in message |

## Error ID: 8193 (0x2001)

Code Define Name:ErrId_MI_BufferAllocFailure

Error ID Description:Buffer allocation error occurred. Does not appear to be active because uartAllocError variable never gets set to a non-zero value in the code.

Example:

A0A20009 – Start Sequence and Payload Length

0A2001000100000001 – Payload

002DB0B3 – Message Checksum and End Sequence

*Table 3-40* Error ID 8193 Message

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | **Scale** | **Example** | | **Scale** | **Example** |
| Message ID | 1 U | | 0A | | | 10 |
| Error ID | 2 U | | 2001 | | | 8193 |
| Count | 2 U | | 0001 | | | 1 |
| uartAllocError | 4 U | | 00000001 | | | 1 |

Payload length: 9 bytes

*Table 3-41* Error ID 8193 Message Description

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Error ID | Error ID (see Error ID description above) |
| Count | Number of 32 bit data in message |
| uartAllocError | Contents of variable used to signal UART buffer allocation error |

## Error ID: 8194 (0x2002)

Code Define Name:ErrId_MI_UpdateTimeFailure

Error ID Description:PROCESS_1SEC task was unable to complete upon entry. Overruns are occurring.

Example:

A0A2000D – Start Sequence and Payload Length

0A200200020000000100000064 – Payload

0093B0B3 – Message Checksum and End Sequence

*Table 3-42* Error ID 8194 Message

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | **Scale** | **Example** | | **Scale** | **Example** |
| Message ID | 1 U | | 0A | | | 10 |
| Error ID | 2 U | | 2002 | | | 8194 |
| Count | 2 U | | 0002 | | | 2 |
| Number of in process errors. | 4 U | | 00000001 | | | 1 |
| Millisecond errors | 4 U | | 00000064 | | | 100 |

Payload length: 13 bytes

*Table 3-43* Error ID 8194 Message Description

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Error ID | Error ID (see Error ID description above) |
| Count | Number of 32 bit data in message |
| Number of in process errors | Number of one second updates not complete on entry |
| Millisecond errors | Millisecond errors caused by overruns |

### *Error ID: 8195 (0x2003)*

Code Define Name:ErrId_MI_MemoryTestFailed

Error ID Description:Failure of hardware memory test.

Example:

A0A20005 – Start Sequence and Payload Length

0A20030000 – Payload

002DB0B3 – Message Checksum and End Sequence

*Table 3-44* Error ID 8195 Message

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 0A | | | 10 |
| Error ID | 2 U | | 2003 | | | 8195 |
| Count | 2 U | | 0000 | | | 0 |

Payload length: 5 bytes

*Table 3-45* Error ID 8195 Message Description

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Error ID | Error ID (see Error ID description above) |
| Count | Number of 32 bit data in message |

## *Command Acknowledgment – Message ID 11*

This reply is sent in response to messages accepted by the receiver. If the message being acknowledged requests data from the receiver, the data is sent first, then this acknowledgment.

Output Rate: Response to successful input message

This is a successful almanac request (Message ID 0x92) example:

A0A20002—Start Sequence and Payload Length

0B92—Payload

009DB0B3—Message Checksum and End Sequence

*Table 3-46* Command Acknowledgment – Message ID 11

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 0x0B | | | 11 |
| ACK ID | 1 U | | 0x92 | | | 146 |

Payload length: 2 bytes

## *Command Negative Acknowledgment – Message ID 12*

This reply is sent when an input command to the receiver is rejected. Possible causes are: the input message failed checksum, contained an argument that was out of the acceptable range, or that the receiver was unable to comply with the message for some technical reason.

Output Rate: Response to rejected input message

This is an unsuccessful almanac request (Message ID 0x92) example:

A0A20002—Start Sequence and Payload Length

0C92—Payload

009EB0B3—Message Checksum and End Sequence

*Table 3-47* Command Negative Acknowledgment – Message ID 12

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 0x0C | | | 12 |
| N'Ack ID | 1 U | | 0x92 | | | 146 |

Payload length: 2 bytes

**Note –** Commands can be Nack'd for several reasons including: failed checksum, invalid arguments, unknown command, or failure to execute command.

## *Visible List – Message ID 13*

This message reports the satellites that are currently above the local horizon. Generally there are from 6 to 13 satellites visible at any one time.

Output Rate: Updated approximately every 2 minutes

**Note –** This is a variable length message. Only the number of visible satellites are reported (as defined by Visible SVs in Table 3-48).

Example:

A0A2002A—Start Sequence and Payload Length

0D081D002A00320F009C0032....—Payload

....B0B3—Message Checksum and End Sequence

*Table 3-48* Visible List – Message ID 13

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 0D | | | 13 |
| Visible SVs | 1 U | | 08 | | | 8 |
| Ch 1 – SV ID | 1 U | | 10 | | | 16 |
| Ch 1 – SV Azimuth | 2 S | | 002A | degrees | | 42 |
| Ch 1 – SV Elevation | 2 S | | 0032 | degrees | | 50 |
| Ch 2 – SV ID | 1 U | | 0F | | | 15 |
| Ch 2 – SV Azimuth | 2 S | | 009C | degrees | | 156 |
| Ch 2 – SV Elevation | 2 S | | 0032 | degrees | | 50 |
| ... | | | | | | |

Payload length: variable (2 + 5 times number of visible SVs).

# *Almanac Data – Message ID 14*

This message is sent in response to the Poll Almanac command, Message ID 146. When Message ID 146 is sent, the receiver responds with 32 individual Message ID 14 messages, one for each of the possible satellite PRNs. If no almanac exists for a given PRN, the data in that message is all zeros.

Output Rate: Response to poll

*Table 3-49* Contents of Message ID 14

| Name | Bytes | Description |
|---|---|---|
| Message ID | 1 U | Hex 0x0E (decimal 14) |
| SV ID | 1 U | SV PRN code, hex 0x01..0x02, decimal 1..32 |
| Almanac Week & Status | 2 S | 10-bit week number in 10 MSBs, status in 6 LSBs (1 = good; 0 = bad) |
| Data[1] [12] | 2 S | UINT16[12] array with sub-frame data |
| Checksum | 2 S | |

Payload length: 30 bytes

1. The data area consists of an array of 12 16-bit words consisting of the data bytes from the navigation message sub-frame. Table 3-50 shows how the actual bytes in the navigation message correspond to the bytes in this data array. Note that these are the raw navigation message data bits with any inversion removed and the parity bits removed.

*Table 3-50* Byte Positions Between Navigation Message and Data Array

| Navigation Message | | Data Array | | Navigation Message | | Data Array | |
|---|---|---|---|---|---|---|---|
| Word | Byte | Word | Byte | Word | Byte | Word | Byte |
| 3 | MSB | [0] | LSB | 7 | MSB | [6] | MSB |
| 3 | Middle | [0] | MSB | 7 | Middle | [6] | LSB |
| 3 | LSB | [1] | LSB | 7 | LSB | [7] | MSB |
| 4 | MSB | [1] | MSB | 8 | MSB | [7] | LSB |
| 4 | Middle | [2] | LSB | 8 | Middle | [8] | MSB |
| 4 | LSB | [2] | MSB | 8 | LSB | [8] | LSB |
| 5 | MSB | [3] | LSB | 9 | MSB | [9] | MSB |
| 5 | Middle | [3] | MSB | 9 | Middle | [9] | LSB |

*Table 3-50* Byte Positions Between Navigation Message and Data Array

| Navigation Message | | Data Array | | Navigation Message | | Data Array | |
|---|---|---|---|---|---|---|---|
| **Word** | **Byte** | **Word** | **Byte** | **Word** | **Byte** | **Word** | **Byte** |
| 5 | LSB | [4] | LSB | 9 | LSB | [10] | MSB |
| 6 | MSB | [4] | MSB | 10 | MSB | [10] | LSB |
| 6 | Middle | [5] | LSB | 10 | Middle | [11] | MSB |
| 6 | LSB | [5] | MSB | 10 | LSB | [11] | LSB |

**Note –** Message ID 130 uses a similar format, but sends an array of 14 16-bit words for each SV and a total of 32 SVs in the message (almanac for SVs 1..32, in ascending order). For that message, a total of 448 words constitutes the data area. For each of 32 SVs, that corresponds to 14 words per SV. Those 14 words consist of one word containing the week number and status bit (described in Table 3-49 above as Almanac Week & Status), 12 words of the same data as described for the data area above, then a single 16-bit checksum of the previous 13 words. The SV PRN code is not included in the message 130 because the SV ID is inferred from the location in the array.

## *Ephemeris Data (Response to Poll) – Message ID 15*

This message is output in response to the Poll Ephemeris command, Message ID 147. If Message ID 147 specifies a satellite PRN, 1-32, a single Message ID 15 containing the ephemeris for that satellite PRN will be output. If Message ID 147 specifies satellite PRN 0, then the receiver sends as many Message ID 15 messages as it has available ephemerides.

The ephemeris data that is polled from the receiver is in a special SiRF format based on the ICD-GPS-200 format for ephemeris data.

Output Rate: Response to poll

*Table 3-51* Contents of Message ID 15

| Name | Bytes | Description |
|---|---|---|
| Message ID | 1 U | Hex 0x0E (decimal 14) |
| SV ID | 1 U | SV PRN code, hex 0x01..0x02, decimal 1..32 |
| Data[1] [45] | 2 U | UINT16 [3][15] array with sub-frames 1..3 data |

Payload length: 92 bytes

1. The data area consists of a 3x15 array of unsigned integers, 16 bits long. The first word of each row in the array ([0][0], [1][0], and [2][0]) contain the SV ID. The remaining words in the row contain the data from the navigation message sub-frame, with row [0] containing sub-frame 1, row [1] containing sub-frame 2, and row [2] containing sub-frame 3. Data from the sub-frame is stored in a packed format, meaning that the 6 parity bits of each 30-bit navigation message word have been removed, and the remaining 3 bytes are stored in 1.5 16-bit words. Since the first word of the sub-frame, the telemetry word (TLM), does not contain any data needed by the receiver, it is not saved. Thus, there are 9 remaining words, with 3 bytes in each sub-frame. This total of 27 bytes is stored in 14 16-bit words. The second word of the sub-frame, the handover word (HOW), has its high byte (MSB) stored as the low byte (LSB) of the first of the 16-bit words. Each following byte is stored in the next available byte of the array. Table 3-52 shows where each byte of the sub-frame is stored in the row of 16-bit words.

*Table 3-52* Byte Positions Between Navigation Message and Data Array

| Navigation Message | | Data Array | | Navigation Message | | Data Array | |
|---|---|---|---|---|---|---|---|
| **Word** | **Byte** | **Word** | **Byte** | **Word** | **Byte** | **Word** | **Byte** |
| 2 (HOW) | MSB | [][1] | LSB | 7 | MSB | [][9] | MSB |
| 2 | Middle | [][2] | MSB | 7 | Middle | [][9] | LSB |
| 2 | LSB | [][2] | LSB | 7 | LSB | [][10] | MSB |
| 3 | MSB | [][3] | MSB | 8 | MSB | [][10] | LSB |
| 3 | Middle | [][3] | LSB | 8 | Middle | [][11] | MSB |
| 3 | LSB | [][4] | MSB | 8 | LSB | [][11] | LSB |
| 4 | MSB | [][4] | LSB | 9 | MSB | [][12] | MSB |
| 4 | Middle | [][5] | MSB | 9 | Middle | [][12] | LSB |
| 4 | LSB | [][5] | LSB | 9 | LSB | [][13] | MSB |
| 5 | MSB | [][6] | MSB | 10 | MSB | [][13] | LSB |
| 5 | Middle | [][6] | LSB | 10 | Middle | [][14] | MSB |
| 5 | LSB | [][7] | MSB | 10 | LSB | [][14] | LSB |
| 6 | MSB | [][7] | LSB | | | | |
| 6 | Middle | [][8] | MSB | | | | |
| 6 | LSB | [][8] | LSB | | | | |

**Note –** Message ID 149 uses the same format, except the SV ID (the second byte in Message ID 15) is omitted. Message ID 149 is thus a 91-byte message. The SV ID is still embedded in elements [0][0], [1][0], and [2][0] of the data array.

## *Test Mode 1 – Message ID 16*

This message is output when the receiver is in test mode 1. It is sent at the end of each test period as set by Message ID 150.

Output Rate: Variable – set by the period as specified in Message ID 150

Example:

A0A20011—Start Sequence and Payload Length

100015001E000588B800C81B5800040001—Payload

02D8B0B3—Message Checksum and End Sequence

*Table 3-53* Test Mode 1 Data – Message ID 16

| | | Binary (Hex) | | | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| **Name** | **Bytes** | **Scale** | **Example** | **Unit** | **Scale** | **Example** |
| Message ID | 1 U | | 10 | | | 16 |
| SV ID | 2 U | | 0015 | | | 21 |
| Period | 2 U | | 001E | sec | | 30 |
| Bit Sync Time | 2 U | | 0005 | sec | | 5 |
| Bit Count | 2 U | | 88B8 | | | 35000 |
| Poor Status | 2 U | | 00C8 | | | 200 |
| Good Status | 2 U | | 1B58 | | | 7000 |
| Parity Error Count | 2 U | | 0004 | | | 4 |
| Lost VCO Count | 2 U | | 0001 | | | 1 |

Payload length: 17 bytes

*Table 3-54* Detailed Description of Test Mode 1 Data

| Name | Description |
|---|---|
| Message ID | Message ID number |
| SV ID | The number of the satellite being tracked |
| Period | The total duration of time (in seconds) that the satellite is tracked |
| Bit Sync Time | The time it takes for channel 0 to achieve the status of 37 |
| Bit Count | The total number of data bits that the receiver is able to demodulate during the test period. As an example, for a 20 second test period, the total number of bits that can be demodulated by the receiver is 12000 (50BPS x 20sec x 12 channels). |
| Poor Status | This value is derived from phase accumulation time. Phase accumulation is the amount of time a receiver maintains phase lock. Every 100 msec of loss of phase lock equates to 1 poor status count. As an example, the total number of status counts for a 60 second period is 7200 (12 channels x 60 sec x 10 / sec). |
| Good Status | This value is derived from phase accumulation time. Phase accumulation is the amount of time a receiver maintains phase lock. Every 100 msec of phase lock equates to 1 good status count. |
| Parity Error Count | The number of word parity errors. This occurs when the parity of the transmitted word does not match the receiver's computed parity. |
| Lost VCO Count | The number of 1 msec VCO lost lock was detected. This occurs when the PLL in the RFIC loses lock. A significant jump in crystal frequency and/or phase causes a VCO lost lock. |

## *Differential Corrections – Message ID 17*

Message ID 17 provides the RTCM data received from a DGPS source. The data is sent as a SiRF Binary message and is based on the RTCM SC-104 format. To interpret the data, see *RTCM Recommended Standards for Differential GNSS* by the Radio Technical Commission for Maritime Services. Data length and message output rate vary based on received data.

*Table 3-55* RTCM message – Message ID 17

| Name | Bytes | Example (Hex) | Example (Decimal) |
|---|---|---|---|
| Message ID | 1 U | 11 | 17 |
| Data length | 2 S | 002D | 45 |
| Data[1] | variable U | | |

Payload length: variable

1. Data length and message output rate vary based on received data. Data consists of a sequence of bytes that are "Data length" long.

## *OkToSend – Message ID 18*

The OkToSend message is sent by a receiver that is in power-saving mode such as TricklePower or Push-to-Fix. It is sent immediately upon powering up, with an argument indicating it is OK to send messages to the receiver, and it is sent just before turning off power with an argument that indicates no more messages should be sent.

Output Rate: Two messages per power-saving cycle

Example:

A0A20002—Start Sequence and Payload Length

1200—Payload

0012B0B3—Message Checksum and End Sequence

*Table 3-56* Almanac Data – Message ID 18

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 12 | | | 18 |
| Send Indicator[1] | 1 U | | 00 | | | 00 |

Payload length: 2 bytes

1. 0 implies that CPU is about to go OFF, OkToSend==NO, 1 implies CPU has just come ON, OkToSend==YES

## *Navigation Parameters (Response to Poll) – Message ID 19*

This message is sent in response to Message ID 152, Poll Navigation Parameters. It reports the current settings of various parameters in the receiver.

Output Rate: Response to Poll (See Message ID 152)

Example:

A0 A2 00 41 —Start Sequence and Payload Length

13 00 00 00 00 00 00 00 00 01 1E 0F 01 00 01 00 00 00 00 04 00 4B 1C 00 00 00 00 02 00 1E 00 00 00 00 00 00 00 03 E8 00 00 03 E8 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00—Payload

02 A4 B0 B3—Message Checksum and End Sequence

*Table 3-57* Navigation Parameters – Message ID 19

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 13 | | | 19 |
| Message Sub ID[1] | 1 U | | 00 | | | |
| Reserved | 3 U | | 00 | | | |
| Altitude Hold Mode[2] | 1 U | | 00 | | | |
| Altitude Hold Source[2] | 1 U | | 00 | | | |
| Altitude Source Input[2] | 2 S | | 0000 | m | | |
| Degraded Mode[2] | 1 U | | 00 | | | |
| Degraded Timeout[2] | 1 U | | 00 | sec | | |
| DR Timeout[2] | 1 U | | 01 | sec | | |
| Track Smooth Mode[2] | 1 U | | 1E | | | |
| Static Navigation[3] | 1 U | | 0F | | | |
| 3SV Least Squares[4] | 1 U | | 01 | | | |
| Reserved | 4 U | | 00000000 | | | |
| DOP Mask Mode[5] | 1 U | | 04 | | | |
| Navigation Elevation Mask[6] | 2 S | | 004B | | | |
| Navigation Power Mask[7] | 1 U | | 1C | | | |
| Reserved | 4 U | | 00000000 | | | |
| DGPS Source[8] | 1 U | | 02 | | | |
| DGPS Mode[9] | 1 U | | 00 | | | |
| DGPS Timeout[9] | 1 U | | 1E | sec | | |
| Reserved | 4 U | | 00000000 | | | |
| LP Push-to-Fix[10] | 1 U | | 00 | | | |
| LP On-time[10] | 4 S | | 000003E8 | | | |

*Table 3-57* Navigation Parameters – Message ID 19 *(Continued)*

| | | Binary (Hex) | | | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| **Name** | **Bytes** | **Scale** | **Example** | **Unit** | **Scale** | **Example** |
| LP Interval[10] | 4 S | | 000003E8 | | | |
| User Tasks Enabled[4] | 1 U | | 00 | | | |
| User Task Interval[4] | 4 S | | 00000000 | | | |
| LP Power Cycling Enabled[11] | 1 U | | 00 | | | |
| LP Max. Acq. Search Time[12] | 4 U | | 00000000 | sec | | |
| LP Max. Off Time[12] | 4 U | | 00000000 | sec | | |
| APM Enabled/Power Duty Cycle[13,14] | 1 U | | 00 | | | |
| Number of Fixes[14] | 2 U | | 0000 | | | |
| Time Between Fixes[14] | 2 U | | 0000 | sec | | |
| Horizontal/Vertical Error Max[15] | 1 U | | 00 | m | | |
| Response Time Max[14] | 1 U | | 00 | sec | | |
| Time/Accu & Time/Duty Cycle Priority[16] | 1 U | | 00 | | | |

Payload length: 65 bytes

1. 00 = GSW2 definition; 01 = SiRF Binary APM definition; other values reserved.
2. These values are set by Message ID 136. See description of values in Table 2-19. Note that Degraded Mode is not supported in GSW3.2.5 and newer.
3. These values are set by Message ID 143. See description of values in Table 2-28.
4. These parameters are set in the software and are not modifiable via the User Interface.
5. These values are set by Message ID 137. See description of values in Table 2-22.
6. These values are set by Message ID 139. See description of values in Table 2-26.
7. These values are set by Message ID 140. See description of values in Table 2-27.
8. These values are set by Message ID 133. See description of values in Table 2-14.
9. These values are set by Message ID 138. See description of values in Table 2-24.
10. These values are set by Message ID 151. See description of values in Table 2-36.
11. This setting is derived from the LP on-time and LP interval.
12. These values are set by Message ID 167. See description of values in Table 2-42.
13. Bit 7: APM Enabled, 1 = enabled, 0 = disabled; Bits 0-4: Power Duty Cycle, range: 1-20 scaled to 5%, 1 = 5%, 2 = 10%
14. Only used in SiRFLoc software.
15. These values are set by Message ID 53. See description of values in Table 2-4
16. Bits 2-3: Time Accuracy, 0x00 = no priority imposed, 0x01 = RESP_TIME_MAX has higher priority, 0x02 = HORI_ERR_MAX has higher priority, Bits 0-1: Time Duty Cycle, 0x00 = no priority imposed, 0x01 = time between two consecutive fixes has priority, 0x02 = power duty cycle has higher priority.

*Table 3-58* Horizontal/Vertical Error

| Value | Position Error |
|---|---|
| 0x00 | < 1 meter |
| 0x01 | < 5 meter |
| 0x02 | < 10 meter |
| 0x03 | < 20 meter |
| 0x04 | < 40 meter |
| 0x05 | < 80 meter |
| 0x06 | < 160 meter |
| 0x07 | No Maximum (disabled) |
| 0x08 - 0xFF | Reserved |

# Test Mode 2/3/4 – Message ID 20, 46, 48 (SiRFLoc v2.x), 49, and 55

Table 3-59 describes the SiRF software and test mode 2/3/4 with respect to their respective Message ID.

*Table 3-59* SiRF Software and Test Mode in Relation with – Message ID 20, 46, 48, 49, and 55

| Software | Test Mode | Message ID |
|---|---|---|
| GSW2 | 2 | 20 |
| | 3/4 | 46 |
| SiRFDRive | 2 | 20 |
| | 3/4 | 46 |
| SiRFXTrac | 2/3/4 | 20 |
| SiRFLoc (version 2.x) | 4 | 20, 48[1], and 49 |
| SiRFLoc (version 3.x) | 3 | 46 |
| | 4 | 46, 55 |
| GSW3, GSWLT3 | 3 | 46 |
| | 4 | 46, 55 |

1. This Message ID 48 for Test Mode 4 is not to be confused with Message ID 48 for DR Navigation. Message ID 48 for SiRFLoc will be transferred to a different Message ID in a near future.

Refer to each specific Message ID for more details.

# Test Mode 2/3/4 – Message ID 20

## Test Mode 2

This is supported by either GSW2, SiRFDRive, and SiRFXTrac. Test Mode 2 requires approximately 1.5 minutes of data collection before sufficient data is available.

The definition of Message ID 20 is different depending on the version and type of software being used.

Example:

A0A20033—Start Sequence and Payload Length

140001001E00023F70001F0D2900000000000601C600051B0E000EB41A0000000000000 000000000000000000000000000000—Payload

0316B0B3—Message Checksum and End Sequence

*Table 3-60* Test Mode 2 – Message ID 20

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 14 | | | 20 |
| SV ID | 2 U | | 0001 | | | 1 |
| Period | 2 U | | 001E | sec | | 30 |
| Bit Sync Time | 2 U | | 0002 | sec | | 2 |
| Bit Count | 2 U | | 3F70 | | | 13680 |
| Poor Status | 2 U | | 001F | | | 31 |
| Good Status | 2 U | | 0D29 | | | 3369 |
| Parity Error Count | 2 U | | 0000 | | | 0 |

*Table 3-60* Test Mode 2 – Message *(Continued)*ID 20 *(Continued)*

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Lost VCO Count | 2 U | | 0000 | | | 0 |
| Frame Sync Time | 2 U | | 0006 | sec | | 6 |
| C/N0 Mean | 2 S | *10 | 01C6 | | ÷10 | 45.4 |
| C/N0 Sigma | 2 S | *10 | 0005 | | ÷10 | 0.5 |
| Clock Drift Change | 2 S | *10 | 1B0E | Hz | ÷10 | 692.6 |
| Clock Drift | 4 S | *10 | 000EB41A | Hz | ÷10 | 96361.0 |
| Reserved | 2 S | | 0000 | | | |
| Reserved | 4 S | | 00000000 | | | |
| Reserved | 4 S | | 00000000 | | | |
| Reserved | 4 S | | 00000000 | | | |
| Reserved | 4 S | | 00000000 | | | |
| Reserved | 4 S | | 00000000 | | | |

Payload length: 51 bytes

*Table 3-61* Detailed Description of Test Mode 2 Message ID 20

| Name | Description |
|---|---|
| Message ID | Message ID number |
| SV ID | The number of the satellite being tracked |
| Period | The total duration of time (in seconds) that the satellite is tracked |
| Bit Sync Time | The time it takes for channel 0 to achieve the status of 37 |
| Bit Count | The total number of data bits that the receiver is able to demodulate during the test period. As an example, for a 20 second test period, the total number of bits that can be demodulated by the receiver is 12000 (50 bps x 20 sec x 12 channels). |
| Poor Status | This value is derived from phase accumulation time. Phase accumulation is the amount of time a receiver maintains phase lock. Every 100 msec of loss of phase lock equates to 1 poor status count. As an example, the total number of status counts for a 60 second period is 7200 (12 channels x 60 sec x 10 sec) |
| Good Status | This value is derived from phase accumulation time. Phase accumulation is the amount of time a receiver maintains phase lock. Every 100 msec of phase lock equates to 1 good status count. |
| Parity Error Count | The number of word parity errors. This occurs when the transmitted parity word does not match the receivers parity check. |
| Lost VCO Count | The number of 1 msec VCO lost lock was detected. This occurs when the PLL in the RFIC loses lock. A significant jump in crystal frequency and / or phase causes a VCO lost lock. |
| Frame Sync | The time it takes for channel 0 to reach a 3F status. |
| C/N0 Mean | Calculated average of reported C/N0 by all 12 channels during the test period. |
| C/N0 Sigma | Calculated sigma of reported C/N0 by all 12 channels during the test period. |
| Clock Drift Change | Difference in clock frequency from start and end of the test period. |
| Clock Drift | Rate of change in clock bias. |

## *Test Mode 3*

This is supported by SiRFXTrac only as Message ID 20. Test Mode 3 requires approximately 10 seconds of measurement data collection before sufficient summary information is available.

Example:

A0A20033—Start Sequence and Payload Length

140001001E00023F70001F0D2900000000000601C600051B0E000EB41A000000000000000
00000000000000000000000000000—Payload

0316B0B3—Message Checksum and End Sequence

*Table 3-62* Test Mode 3 – Message ID 20

| | | Binary (Hex) | | | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| **Name** | **Bytes** | **Scale** | **Example** | **Unit** | **Scale** | **Example** |
| Message ID | 1 U | | 14 | | | 20 |
| SV ID | 2 U | | 0001 | | | 1 |
| Period | 2 U | | 001E | sec | | 30 |
| Bit Sync Time | 2 U | | 0002 | sec | | 2 |
| Bit Count | 2 U | | 3F70 | | | 13680 |
| Poor Status | 2 U | | 001F | | | 31 |
| Good Status | 2 U | | 0D29 | | | 3369 |
| Parity Error Count | 2 U | | 0000 | | | 0 |
| Lost VCO Count | 2 U | | 0000 | | | 0 |
| Frame Sync Time | 2 U | | 0006 | sec | | 6 |
| C/N0 Mean | 2 S | *10 | 01C6 | | ÷10 | 45.4 |
| C/N0 Sigma | 2 S | *10 | 0005 | | ÷10 | 0.5 |
| Clock Drift Change | 2 S | *10 | 1B0E | Hz | ÷10 | 692.6 |
| Clock Drift | 4 S | *10 | 000EB41A | Hz | ÷10 | 96361.0 |
| Bad 1 kHz Bit Count | 2 S | | 0000 | | | |
| Abs I20 ms | 4 S | | 00000000 | | | |
| Abs Q1 ms | 4 S | | 00000000 | | | |
| Reserved | 4 S | | 00000000 | | | |
| Reserved[1] | 4 S | | 00000000 | | | |
| Reserved | 4 S | | 00000000 | | | |

Payload length: 51 bytes

1. In some later versions of GSW3 (3.2.4 or later) this field is split into two new fields: RTC Frequency 2 U (in Hz) and Reserved 2 U.

*Table 3-63* Detailed Description of Test Mode 3 Message ID 20

| Name | Description |
|---|---|
| Message ID | Message ID number |
| SV ID | The number of the satellite being tracked |
| Period | The total duration of time (in seconds) that the satellite is tracked |
| Bit Sync Time | The time it takes for channel 0 to achieve the status of 37 |
| Bit Count | The total number of data bits that the receiver is able to demodulate during the test period. As an example, for a 20 second test period, the total number of bits that can be demodulated by the receiver is 12000 (50 bps x 20sec x 12 channels). |
| Poor Status | This value is derived from phase accumulation time. Phase accumulation is the amount of time a receiver maintains phase lock. Every 100 msec of loss of phase lock equates to 1 poor status count. As an example, the total number of status counts for a 60 second period is 7200 (12 channels x 60 sec x 10 sec) |
| Good Status | This value is derived from phase accumulation time. Phase accumulation is the amount of time a receiver maintains phase lock. Every 100 msec of phase lock equates to 1 good status count. |
| Parity Error Count | The number of word parity errors. This occurs when the transmitted parity word does not match the receivers parity check. |
| Lost VCO Count | The number of 1 msec VCO lost lock was detected. This occurs when the PLL in the RFIC loses lock. A significant jump in crystal frequency and / or phase causes a VCO lost lock. |

*Table 3-63* Detailed Description of Test Mode 3 Message ID 20 *(Continued)*

| Name | Description |
|---|---|
| Frame Sync | The time it takes for channel 0 to reach a 3F status. |
| C/N0 Mean | Calculated average of reported C/N0 by all 12 channels during the test period |
| C/N0 Sigma | Calculated sigma of reported C/N0 by all 12 channels during the test period |
| Clock Drift Change | Difference in clock frequency from start and end of the test period |
| Clock Drift | Rate of change of clock bias |
| Bad 1 kHz Bit Count | Errors in 1 ms post correlation I count values |
| Abs I20 ms | Absolute value of the 20 ms coherent sums of the I count over the duration of the test period |
| Abs Q20 ms | Absolute value of the 20 ms Q count over the duration of the test period |
| RTC Frequency | The measured frequency of the RTC crystal oscillator, reported in Hertz |

## Test Mode 4

Supported by SiRFXTrac only. For other Test Mode 4 outputs, refer to MID 46.

*Table 3-64* Test Mode 4 – Message ID 20

| | | Binary (Hex) | | | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| Name | Bytes | Scale | Example | Unit | Scale | Example |
| Message ID | 1 U | | 14 | | | 20 |
| Test Mode | 1 U | | 04 | | | 4 |
| Message Variant | 1 U | | 01 | | | 1 |
| SV ID | 2 U | | 0001 | | | 1 |
| Period | 2 U | | 001E | sec | | 30 |
| Bit Sync Time | 2 U | | 0002 | sec | | 2 |
| C/N0 Mean | 2 S | *10 | 01C6 | | ÷10 | 45.4 |
| C/N0 Sigma | 2 S | *10 | 0005 | | ÷10 | 0.5 |
| Clock Drift Change | 2 S | *10 | 1B0E | Hz | ÷10 | 692.6 |
| Clock Drift | 4 S | *10 | 000EB41A | Hz | ÷10 | 96361.0 |
| I Count Errors | 2 S | | 0003 | | | 3 |
| Abs I20ms | 4 S | | 0003AB88 | | | 240520 |
| Abs Q1ms | 4 S | | 0000AFF0 | | | 45040 |

Payload length: 29 bytes

*Table 3-65* Detailed Description of Test Mode 4 Message ID 20

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Test Mode | 3 = Testmode 3, 4 = Testmode 4 |
| Message Variant | The variant # of the message (variant change indicates possible change in number of fields or field description) |
| SV ID | The number of the satellite being tracked |
| Period | The total duration of time (in seconds) that the satellite is tracked |
| Bit Sync Time | The time it takes for channel 0 to achieve the status of 37 |
| C/N0 Mean | Calculated average of reported C/N0 by all 12 channels during the test period |
| C/N0 Sigma | Calculated sigma of reported C/N0 by all 12 channels during the test period |
| Clock Drift Change | Difference in clock frequency from start and end of the test period |
| Clock Drift | The internal clock offset |
| I Count Errors | Errors in 1 ms post correlation I count values |
| Abs I20 ms | Absolute value of the 20 ms coherent sums of the I count over the duration of the test period |
| Abs Q 1 ms | Absolute value of the 1 ms Q count over the duration of the test period |

## DGPS Status Format – Message ID 27

Reports on the current DGPS status, including the source of the corrections and which satellites have corrections available.

Output Rate: Every measurement cycle (full power / continuous: 1 Hz)

Example (with SBAS):

A0A20034—Start Sequence and Payload Length

1B14444444444007252864A2EC .... —Payload

1533B0B3—Message Checksum and End Sequence

The above example looks as follows in ASCII format:

27, 1, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 0, 0, 7, 594, 8, 100, 10, 748

*Table 3-66* DGPS Status Format – Message ID 27

| | | **Binary (Hex)** | | | **ASCII (Decimal)** | |
|---|---|---|---|---|---|---|
| **Name** | **Bytes** | **Scale** | **Example** | **Unit** | **Scale** | **Example** |
| Message I.D. | 1 U | | 1B | | | 27 |
| DGPS source[1] | 1 U | | 1 | | | 1 = SBAS |
| If the DGPS source is Beacon, next 14 bytes are interpreted as follows: | | | | | | |
| Beacon Frequency | 4 S | 100 | 0 = 0xFFF<br>0 = 190K, 0xFFF = 599.5K<br>Frequency = (190000)+(100*value) | Hz | | |
| Beacon Bit Rate | 1 U | | Bits 2 - 0 :<br>000 25 bits/sec<br>001 50 bits/sec<br>010 100 bits/sec<br>011 110 bits/sec<br>100 150 bits/sec<br>101 200 bits/sec<br>110 250 bits/sec<br>111 300 bits/sec<br>Bit 4 : modulation<br>(0 = MSK, 1 = FSK)<br>Bit 5 : SYNC type<br>(0 = async, 1 = sync)<br>Bit 6 : broadcast coding<br>(0 = No Coding,<br>1 = FEC coding) | BPS | | |
| Status | 1 U | | Bitmapped<br>0x01: signal valid<br>0x02: auto frequency used<br>0x04: auto bit rate used | | | Bitmapped<br>0x |
| Signal Magnitude | 4 S | | | internal counts | | |
| Signal Strength | 2 S | | | dB | | |
| SNR | 2 S | | | dB | | |
| If the DGPS source is not Beacon, next 14 bytes are interpreted as follows: | | | | | | |
| Correction Age[2] [12] | 1 x 12 | | 4 | sec | | 4 |
| Reserved | 2 | | | | | |
| Remainder of the table applies to all messages, and reports on available corrections | | | | | | |

*Table 3-66* DGPS Status Format – Message ID 27 *(Continued)*

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Satellite PRN Code | 1 U | | 18 | | | SV = 24 |
| DGPS Correction | 2 S | | 24E | meters | 100 | 5.90 |
| The above 3 bytes are repeated a total of 12 times. If less than 12 satellite corrections are available, the unused entries have values of 0. | | | | | | |

Payload length: 52 bytes

1. Possible values for this field are given in Table 3-67. If the GSPS source is set to none, three messages are being sent and then the message is disabled.

2. Correction age is reported in the same order as satellites are listed in the satellite PRN code fields that follow.

*Table 3-67* DGPS Correction Types

| DGPS Correction Types | Value | Description |
|---|---|---|
| None | 0 | No DGPS correction type have been selected |
| SBAS | 1 | SBAS |
| Serial Port | 2 | RTCM corrections |
| Internal Beacon | 3 | Beacon corrections (available only for GSW2 software) |
| Software | 4 | Software Application Program Interface (API) corrections |

**Note –** This message differs from others in that it has multiple formats. Further, not all SiRF software versions implement all of the features. All versions implement the first 2 bytes and the last 3 x 12 bytes (3 bytes per satellite times 12 satellites) the same. The 14 bytes in between these two sections vary depending on the source of the DGPS information. If the source is an internal beacon, the 14 bytes are used to display information about the beacon itself (frequency, bit rate, etc.). If the source is something other than an internal beacon, some software versions display the age of the corrections while other versions only fill this area with zeroes.

## *Navigation Library Measurement Data – Message ID 28*

Output Rate: Every measurement cycle (full power / continuous: 1 Hz)

Example:

A0A20038—Start Sequence and Payload Length

1C00000660D015F143F62C4113F42F417B235CF3FBE95E468C6964B8FBC582415 CF1C375301734.....03E801F400000000—Payload

1533B0B3—Message Checksum and End Sequence

*Table 3-68* Navigation Library Measurement Data – Message ID 28

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 1C | | | 28 |
| Channel | 1 U | | 00 | | | 0 |
| Time Tag[1] | 4 U | | 000660D0 | ms | | 135000 |
| Satellite ID | 1 U | | 15 | | | 20 |
| GPS Software Time[2] | 8 Dbl | | 41740B0B48353F7D | sec | | 2.4921113696e+005 |
| Pseudorange[3] | 8 Dbl | | 7D3F354A0B0B7441 | m | | 2.1016756638e+007 |

*Table 3-68* Navigation Library Measurement Data – Message ID 28 *(Continued)*

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Carrier Frequency | 4 Sgl | | 89E98246 | m/s | | 1.6756767578e+004 |
| Carrier Phase[4] | 8 Dbl | | A4703D4A0B0B7441 | m | | 2.1016756640e+007 |
| Time in Track | 2 U | | 7530 | ms | | 10600 |
| Sync Flags | 1 D | | 17 | | | 23 |
| C/N0 1 | 1 U | | 34 | dB-Hz | | 43 |
| C/N0 2 | 1 U | | | dB-Hz | | 43 |
| C/N0 3 | 1 U | | | dB-Hz | | 43 |
| C/N0 4 | 1 U | | | dB-Hz | | 43 |
| C/N0 5 | 1 U | | | dB-Hz | | 43 |
| C/N0 6 | 1 U | | | dB-Hz | | 43 |
| C/N0 7 | 1 U | | | dB-Hz | | 43 |
| C/N0 8 | 1 U | | | dB-Hz | | 43 |
| C/N0 9 | 1 U | | | dB-Hz | | 43 |
| C/N0 10 | 1 U | | | dB-Hz | | 43 |
| Delta Range Interval | 2 U | | 03E801F4 | ms | | 1000 |
| Mean Delta Range Time | 2 U | | 01F4 | ms | | 500 |
| Extrapolation Time[5] | 2 S | | 0000 | ms | | |
| Phase Error Count | 1 U | | 00 | | | 0 |
| Low Power Count | 1 U | | 00 | | | 0 |

Payload length: 56 bytes

1. Internal time for relative measure only.
2. GPS software time minus clock bias = GPS time of measurement.
3. Pseudorange does not contain ionospheric, tropospheric or clock corrections
4. GSW3 and GSWLT3 software does not report the Carrier Phase.
5. Reserved for SiRF use with GSW3, GSWLT3, GSW2.0 or above.

**Note –** For GPS Software Time, Pseudorange, Carrier Frequency, and Carrier Phase, the fields are floating point (4-byte fields) or double-precision floating point (8-byte fields), per IEEE-754 format. The byte order may have to be changed to be properly interpreted on some computers. Also, GSW3.x and GSWLT3 use the same byte ordering method as the GSW 2.2.0. Therefore, GSW 2.2.0 (and older) and GSW 3.0 (and newer) use the original byte ordering method; GSW 2.3.0 through 2.9.9 use an alternate byte ordering method.

To convert the data to be properly interpreted on a PC-compatible computer, do the following: For double-precision (8-byte) values: Assume the bytes are transmitted in the order of B0, B1, ... , B7. For version 2.2.0 and earlier software, rearrange them to B3, B2, B1, B0, B7, B6, B5, B4. For version 2.3.0 and later software, rearrange them to B7, B6, B5, ... , B0. For single-precision (4-byte) values: Assume bytes are transmitted in the order of B0, B1, B2, B3. Rearrange them to B3, B2, B1, B0 (that is, byte B3 goes into the lowest memory address, B0 into the highest).

With these remappings, the values should be correct. To verify, compare the same field from several satellites tracked at the same time. The reported exponent should be similar (within 1 power of 10) among all satellites. The reported Time of Measurement, Pseudorange and Carrier Phase are all uncorrected values.

Message ID 7 contains the clock bias that must be considered. Adjust the GPS

Software time by subtracting clock bias, adjust pseudorange by subtracting clock bias times the speed of light, and adjust carrier phase by subtracting clock bias times speed of light/GPS L1 frequency. To adjust the reported carrier frequency do the following: Corrected Carrier Frequency (m/s) = Reported Carrier Frequency (m/s) – Clock Drift (Hz)*C / 1575420000 Hz. For a nominal clock drift value of 96.25 kHz (equal to a GPS Clock frequency of 24.5535 MHz), the correction value is 18315.766 m/s.

**Note –** GPS Software Time – Clock Bias = Time of Receipt = GPS Time. GPS Software Time – Pseudorange (sec) = Time of Transmission = GPS Time. Adjust SV position in Message ID 30 by (GPS Time $_{MID\ 30}$ – Time of Transmission) * $V_{sat}$.

*Table 3-69* Sync Flag Fields (for GSW2 software ONLY)

| Bit Fields | Description |
|---|---|
| [0] | Coherent Integration Time<br>0 = 2 ms<br>1 = 10 ms |
| [2:1] | Synch State<br>00 = Not aligned<br>01 = Consistent code epoch alignment<br>10 = Consistent data bit alignment<br>11 = No millisecond errors |
| [4:3] | Autocorrelation Detection State<br>00 = Verified not an autocorrelation<br>01 = Testing in progress<br>10 = Strong signal, autocorrelation detection not run<br>11 = Not used |

*Table 3-70* Detailed Description of the Measurement Data

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Channel | Receiver channel number for a given satellite being searched or tracked. Range of 0-11 for channels 1-12, respectively |
| Time Tag | This is the Time Tag in milliseconds of the measurement block in the receiver software time. Time tag is an internal millisecond counter which has no direct relationship to GPS time, but is started as the receiver is turned on or reset. |
| Satellite ID | Pseudo-Random Noise (PRN) number. |
| GPS Software Time | This is GPS Time of Week (TOW) estimated by the software in millisecond |
| Pseudorange | This is the generated pseudorange measurement for a particular SV. When carrier phase is locked, this data is smoothed by carrier phase. |
| Carrier Frequency | This can be interpreted in two ways:<br>1. The delta pseudorange normalized by the reciprocal of the delta pseudorange measurement interval.<br>2. The frequency from the AFC loop. If, for example, the delta pseudorange interval computation for a particular channel is zero, it can be the AFC measurement, otherwise it is a delta pseudorange computation.[1] |
| Carrier Phase | For GSW2 software, the integrated carrier phase (meters), which initially is made equal to pseudorange, is integrated as long as carrier lock is retained. Discontinuity in this value generally means a cycle slip and renormalization to pseudorange. |
| Time in Track | The Time in Track counts how long a particular SV has been in track. For any count greater than zero (0), a generated pseudorange is present for a particular channel. The length of time in track is a measure of how large the pull-in error may be. |

*Table 3-70* Detailed Description of the Measurement Data *(Continued)*

| Name | Description |
|---|---|
| Sync Flags | For GSW2, this byte contains two 2-bit fields and one 1-bit field that describe the Autocorrelation Detection State, Synch State and Coherent Integration Time. Refer to Table 3-69 for more details. For GSW3, this field contains a duplicate of the state field of Message ID 4. See Table 3-9 for details. |
| C/N0 1 | This array of Carrier To Noise Ratios is the average signal power in dB-Hz for each of the 100-millisecond intervals in the previous second or last epoch for each particular SV being track in a channel. First 100 millisecond measurement |
| C/N0 2 | Second 100 millisecond measurement |
| C/N0 3 | Third 100 millisecond measurement |
| C/N0 4 | Fourth 100 millisecond measurement |
| C/N0 5 | Fifth 100 millisecond measurement |
| C/N0 6 | Sixth 100 millisecond measurement |
| C/N0 7 | Seventh 100 millisecond measurement |
| C/N0 8 | Eighth 100 millisecond measurement |
| C/N0 9 | Ninth 100 millisecond measurement |
| C/N0 10 | Tenth 100 millisecond measurement |
| Delta Range Interval | This is the delta-pseudorange measurement interval for the preceding second. A value of zero indicated that the receiver has an AFC measurement or no measurement in the Carrier Frequency field for a particular channel. |
| Mean Delta Range Time | This is the mean calculated time of the delta-pseudorange interval in milliseconds measured from the end of the interval backwards |
| Extrapolation Time | In GSW2, this is the pseudorange extrapolation time, in milliseconds, to reach the common Time tag value. Reserved for SiRF use in GSW3 and GSWLT3. |
| Phase Error Count | This is the count of the phase errors greater than 60 degrees measured in the preceding second as defined for a particular channel |
| Low Power Count | This is the low power measurements for signals less than 28 dB-Hz in the preceding second as defined for a particular channel. Similar for GSW3 and GSWLT3 but does not use 28 dB-Hz; it uses a filter with time constant (t) that equals approximately 1 sec |

1. Carrier frequency may be interpreted as the measured Doppler on the received signal. The value is reported in metres per second but can be converted to hertz using the Doppler equation:

   Doppler frequency / Carrier frequency = Velocity / Speed of light, where Doppler frequency is in Hz; Carrier frequency = 1,575,420,000 Hz; Velocity is in m/s; Speed of light = 299,792,458 m/s.

   Note that the computed Doppler frequency contains a bias equal to the current clock drift as reported in Message ID 7. This bias, nominally 96.250 kHz, is equivalent to over 18 km/s.

## *Navigation Library DGPS Data – Message ID 29*

Output Rate: Every measurement cycle (full power / continuous: 1 Hz)

Example:

A0A2001A—Start Sequence and Payload Length

1D000F00B501BFC97C673CAAAAAB3FBFFE1240A0000040A00000—Payload

0956B0B3—Message Checksum and End Sequence

*Table 3-71* Navigation Library DGPS Data – Message ID 29

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 1D | | | 29 |
| Satellite ID | 2 S | | 000F | | | 15 |
| IOD | 2 S | | 00B5 | | | 181 |
| Source[1] | 1 U | | 01 | | | 1 |
| Pseudorange Correction | 4 Sgl | | BFC97C67 | m | | -1.574109 |
| Pseudorange rate Correction | 4 Sgl | | 3CAAAAAB | m/sec | | 0.020833 |
| Correction Age | 4 Sgl | | 3FBFFE12 | sec | | 1.499941 |
| Reserved | 4 Sgl | | | | | |
| Reserved | 4 Sgl | | | | | |

Payload length: 26 bytes

1. 0 = Use no corrections, 1 = SBAS channel, 2 = External source, 3 = Internal Beacon, 4 = Set Corrections via software

---

**Note –** The fields Pseudorange Correction, Pseudorange Rate Correction, and Correction Age are floating point values per IEEE-754. To properly interpret these in a computer, the bytes must be rearranged in reverse order.

---

## *Navigation Library SV State Data – Message ID 30*

The data in Message ID 30 reports the computed satellite position and velocity at the specified GPS time.

---

**Note –** When using Message ID 30 SV position, adjust for difference between GPS Time $_{\text{MID 30}}$ and Time of Transmission (see note in Message ID 28). Iono delay is not included in pseudorange in Message ID 28.

---

Output Rate: Every measurement cycle (full power / continuous: 1 Hz)

Example:

A0A20053—Start Sequence and Payload Length

1E15....2C64E99D01....408906C8—Payload

2360B0B3—Message Checksum and End Sequence

*Table 3-72* Navigation Library SV State Data – Message ID 30

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 1E | | | 30 |
| Satellite ID | 1 U | | 15 | | | 21 |
| GPS Time | 8 Dbl | | | sec | | |
| Position X | 8 Dbl | | | m | | |
| Position Y | 8 Dbl | | | m | | |
| Position Z | 8 Dbl | | | m | | |
| Velocity X | 8 Dbl | | | m/sec | | |
| Velocity Y | 8 Dbl | | | m/sec | | |

*Table 3-72* Navigation Library SV State Data – Message ID 30 *(Continued)*

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Velocity Z | 8 Dbl | | | m/sec | | |
| Clock Bias | 8 Dbl | | | sec | | |
| Clock Drift | 4 Sgl | | 2C64E99D | s/s | | 744810909 |
| Ephemeris Flag (see details in Table 3-73) | 1 D | | 01 | | | 1 |
| Reserved | 4 Sgl | | | | | |
| Reserved | 4 Sgl | | | | | |
| Ionospheric Delay | 4 Sgl | | 408906C8 | m | | 1082721992 |

Payload length: 83 bytes

**Note –** Each of the 8-byte fields as well as Clock Drift and Ionospheric Delay fields are floating point values per IEEE-754. To properly interpret these in a computer, the bytes must be rearranged. See Note in “Navigation Library Measurement Data – Message ID 28” on page 32 for byte orders.

*Table 3-73* Ephemeris Flag Definition

| Ephemeris Flag Value | Definition |
|---|---|
| 0x00 | No Valid SV state |
| 0x01 | SV state calculated from broadcast ephemeris |
| 0x02 | SV state calculated from almanac at least 0.5 week old |
| 0x03 | Assist data used to calculate SV state |
| 0x04 | SV state calculated from almanac less than 0.5 weeks old |
| 0x11 | SV state calculated from server-based synthesized ephemeris with age of 1 day |
| 0x12 | SV state calculated from server-based synthesized ephemeris with age of 2 day |
| 0x13 | SV state calculated from server-based synthesized ephemeris with age of 3 day |
| 0x14 | SV state calculated from server-based synthesized ephemeris with age of 4 day |
| 0x15 | SV state calculated from server-based synthesized ephemeris with age of 5 day |
| 0x16 | SV state calculated from server-based synthesized ephemeris with age of 6 day |
| 0x17 | SV state calculated from server-based synthesized ephemeris with age of 7 day |
| 0x21 | SV state calculated from client-based synthesized ephemeris with age of 1 day |
| 0x22 | SV state calculated from client-based synthesized ephemeris with age of 2 day |
| 0x23 | SV state calculated from client-based synthesized ephemeris with age of 3 day |
| 0x24 | SV state calculated from client-based synthesized ephemeris with age of 4 day |
| 0x25 | SV state calculated from client-based synthesized ephemeris with age of 5 day |
| 0x26 | SV state calculated from client-based synthesized ephemeris with age of 6 day |
| 0x27 | SV state calculated from client-based synthesized ephemeris with age of 7 day |

## *Navigation Library Initialization Data – Message ID 31*

Output Rate: Every measurement cycle (full power / continuous: 1 Hz)

Example:

A0A20054—Start Sequence and Payload Length

1F....00000000000001001E000F....00....000000000F....00....02....043402....

....02—Payload

0E27B0B3—Message Checksum and End Sequence

*Table 3-74* Navigation Library Initialization Data – Message ID 31

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 1F | | | 31 |
| Reserved | 1 U | | | | | |
| Altitude Mode[1] | 1 U | | 00 | | | 0 |
| Altitude Source | 1 U | | 00 | | | 0 |
| Altitude | 4 Sgl | | 00000000 | m | | 0 |
| Degraded Mode[2] | 1 U | | 01 | | | 1 |
| Degraded Timeout | 2 S | | 001E | sec | | 30 |
| Dead-reckoning Timeout | 2 S | | 000F | sec | | 15 |
| Reserved | 2 S | | | | | |
| Track Smoothing Mode[3] | 1 U | | 00 | | | 0 |
| Reserved | 1 U | | | | | |
| Reserved | 2 S | | | | | |
| Reserved | 2 S | | | | | |
| Reserved | 2 S | | | | | |
| DGPS Selection[4] | 1 U | | 00 | | | 0 |
| DGPS Timeout | 2 S | | 0000 | sec | | 0 |
| Elevation Nav. Mask | 2 S | 2 | 000F | deg | | 15 |
| Reserved | 2 S | | | | | |
| Reserved | 1 U | | | | | |
| Reserved | 2 S | | | | | |
| Reserved | 1 U | | | | | |
| Reserved | 2 S | | | | | |
| Static Nav. Mode[5] | 1 U | | 00 | | | 0 |
| Reserved | 2 S | | | | | |
| Position X | 8 Dbl | | | m | | |
| Position Y | 8 Dbl | | | m | | |
| Position Z | 8 Dbl | | | m | | |
| Position Init. Source[6] | 1 U | | 02 | | | 2 |
| GPS Time | 8 Dbl | | | sec | | |
| GPS Week | 2 S | | 0434 | | | 1076 |
| Time Init. Source[7] | 1 U | | 02 | sec | | 2 |
| Drift | 8 Dbl | | | Hz | | |
| Drift Init. Source[8] | 1 U | | 02 | sec | | 2 |

Payload length: 84 bytes

1. 0 = Use last know altitude, 1 = Use user input altitude, 2 = Use dynamic input from external source
2. 0 = Use direction hold and then time hold, 1 = Use time hold and then direction hold, 2 = Only use direction hold, 3 = Only use time hold, 4 = Degraded mode is disabled. Note that Degraded Mode is not supported in GSW3.2.5 and newer.
3. 0 = True, 1 = False
4. 0 = Use DGPS if available, 1 = Only navigate if DGPS corrections are available, 2 = Never use DGPS corrections
5. 0 = True, 1 = False
6. 0 = ROM position, 1 = User position, 2 = SRAM position, 3 = Network assisted position
7. 0 = ROM time, 1 = User time, 2 = SRAM time, 3 = RTC time, 4 = Network assisted time
8. 0 = ROM clock, 1 = User clock, 2 = SRAM clock, 3 = Calibration clock, 4 = Network assisted clock

**Note –** Altitude is a single-precision floating point value while position XYZ, GPS time, and drift are double-precision floating point values per IEEE-754. To properly interpret these values in a computer, the bytes must be rearranged. See note in Message ID 28 for byte orders.

## Geodetic Navigation Data – Message ID 41

Output Rate: Every measurement cycle (full power / continuous: 1 Hz)

Example:

A0 A2 00 5B—Start Sequence and Payload Length

29 00 00 02 04 04 E8 1D 97 A7 62 07 D4 02 06 11 36 61 DA 1A 80 01 58 16 47 03 DF B7 55 48 8F FF FF FA C8 00 00 04 C6 15 00 00 00 00 00 00 00 00 00 00 00 00 00 BB 00 00 01 38 00 00 00 00 00 00 6B 0A F8 61 00 00 00 00 00 1C 13 14 00 00 00 00 00 00 00 00 00 00 00 00 08 05 00—Payload

11 03 B0 B3—Message Checksum and End Sequence

*Table 3-75* Geodetic Navigation Data – Message ID 41

| Name | Bytes | Description |
|---|---|---|
| Message ID | 1 U | Hex 0x29 (decimal 41) |
| Nav Valid | 2 D | 0x0000 = valid navigation (any bit set implies navigation solution is not optimal);<br>Bit 0 ON: solution not yet overdetermined[1] (< 5 SVs),<br>OFF: solution overdetermined[1] (> = 5 SV)<br>Bits 1 – 2 : Reserved<br>Bits 8 – 14 : Reserved<br>(The following are for SiRFDRive only)<br>Bit 3 ON : invalid DR sensor data<br>Bit 4 ON : invalid DR calibration<br>Bit 5 ON : unavailable DR GPS-based calibration<br>Bit 6 ON : invalid DR position fix<br>Bit 7 ON : invalid heading<br>(The following is for SiRFNav only)<br>Bit 15 ON : no tracker data available |

*Table 3-75* Geodetic Navigation Data – Message ID 41 *(Continued)*

| Name | Bytes | Description |
|---|---|---|
| NAV Type | 2 D | Bits 2 – 0 : GPS position fix type<br>000 = no navigation fix<br>001 = 1-SV KF solution<br>010 = 2-SV KF solution<br>011 = 3-SV KF solution<br>100 = 4 or more SV KF solution<br>101 = 2-D least-squares solution<br>110 = 3-D least-squares solution<br>111 = DR solution (see bits 8, 14-15)<br>Bit 3 : TricklePower in use<br>Bits 5 – 4 : altitude hold status<br>00 = no altitude hold applied<br>01 = holding of altitude from KF<br>10 = holding of altitude from user input<br>11 = always hold altitude (from user input)<br>Bit 6 ON : DOP limits exceeded<br>Bit 7 ON : DGPS corrections applied<br>Bit 8 : Sensor DR solution type (SiRFDRive only)<br>1 = sensor DR<br>0 = velocity DR[2] if Bits 0 – 2 = 111;<br>else check Bits 14-15 for DR error status<br>Bit 9 ON : navigation solution overdetermined[1]<br>Bit 10 ON : velocity DR[2] timeout exceeded<br>Bit 11 ON : fix has been edited by MI functions<br>Bit 12 ON : invalid velocity<br>Bit 13 ON : altitude hold disabled<br>Bits 15 – 14 : sensor DR error status (SiRFDRive only)<br>00 = GPS-only navigation<br>01 = DR calibration from GPS<br>10 = DR sensor error<br>11 = DR in test |
| Extended Week Number | 2 U | GPS week number; week 0 started January 6 1980. This value is extended beyond the 10-bit value reported by the SVs. |
| TOW | 4 U | GPS time of week in seconds x $10^3$ |
| UTC Year | 2 U | UTC time and date. Seconds reported as integer milliseconds only |
| UTC Month | 1 U | |
| UTC Day | 1 U | |
| UTC Hour | 1 U | |
| UTC Minute | 1 U | |
| UTC Second | 2 U | |
| Satellite ID List | 4 D | Bit map of SVs used in solution. Bit 0 = SV 1, Bit 31 = SV 32. A bit set ON means the corresponding SV was used in the solution |
| Latitude | 4 S | In degrees (+ = North) x $10^7$ |
| Longitude | 4 S | In degrees (+ = East) x $10^7$ |
| Altitude from Ellipsoid | 4 S | In meters x $10^2$ |
| Altitude from MSL | 4 S | In meters x $10^2$ |
| Map Datum[3] | 1 S | See footnote |
| Speed Over Ground (SOG) | 2 U | In m/s x $10^2$ |
| Course Over Ground (COG, True) | 2 U | In degrees clockwise from true north x $10^2$ |
| Magnetic Variation | 2 S | Not implemented |
| Climb Rate | 2 S | In m/s x $10^2$ |
| Heading Rate | 2 S | deg/s x $10^2$ (SiRFDRive only) |

*Table 3-75* Geodetic Navigation Data – Message ID 41 *(Continued)*

| Name | Bytes | Description |
|---|---|---|
| Estimated Horizontal Position Error | 4 U | EHPE in meters x $10^2$ |
| Estimated Vertical Position Error | 4 U | EVPE in meters x $10^2$ |
| Estimated Time Error | 4 U | ETE in seconds x $10^2$ (SiRFDRive only) |
| Estimated Horizontal Velocity Error | 2 U | EHVE in m/s x $10^2$ (SiRFDRive only) |
| Clock Bias | 4 S | In m x $10^2$ |
| Clock Bias Error | 4 U | In meters x $10^2$ (SiRFDRive only) |
| Clock Drift[4] | 4 S | In m/s x $10^2$ |
| Clock Drift Error | 4 U | In m/s x $10^2$ (SiRFDRive only) |
| Distance | 4 U | Distance traveled since reset in meters (SiRFDRive only) |
| Distance error | 2 U | In meters (SiRFDRive only) |
| Heading Error | 2 U | In degrees x $10^2$ (SiRFDRive only) |
| Number of SVs in Fix | 1 U | Count of SVs indicated by SV ID list |
| HDOP | 1 U | Horizontal Dilution of Precision x 5 (0.2 resolution) |
| AdditionalModeInfo | 1 D | Additional mode information:<br>Bit 0: map matching mode for Map Matching only<br>0 = map matching feedback input is disabled<br>1 = map matching feedback input is enabled<br>Bit 1: map matching feedback received for Map Matching only<br>0 = map matching feedback was not received<br>1 = map matching feedback was received<br>Bit 2: map matching in use for Map Matching only<br>0 = map matching feedback was not used to calculate position<br>1 = map matching feedback was used to calculate position<br>Bit 3-6: reserved<br>Bit 7: DR direction for SiRFDRive only<br>0 = forward<br>1 = reserve |

Payload length: 91 bytes

1. An overdetermined solution (see bit 0 from Nav Valid and bit 9 of Nav Type) is one where at least one additional satellite has been used to confirm the 4-satellite position solution. Once a solution has been overdetermined, it remains so even if several satellites are lost, until the system drops to no-navigation status (Nav Type bits 0-2 = 000).
2. Velocity Dead Reckoning (DR) is a method by which the last solution computed from satellite measurements is updated using the last computed velocity and time elapsed to project the position forward in time. It assumes heading and speed are unchanged, and is thus reliable for only a limited time. Sensor DR is a position update method based on external sensors (e.g., rate gyroscope, vehicle speed pulses, accelerometers) to supplement the GPS measurements. Sensor DR is only applicable to SiRFDRive products.
3. Map Datum indicates the datum to which latitude, longitude, and altitude relate. 21 = WGS-84, by default. Other values are defined as other datums are implemented. Available datums include: 21 = WGS-84, 178 = Tokyo Mean, 179 = Tokyo Japan, 180 = Tokyo Korea, 181 = Tokyo Okinawa.
4. To convert Drift m/s to Hz: Drift (m/s) *L1(Hz)/c = Drift (Hz).

**Note –** Values are transmitted as integer values. When scaling is indicated in the description, the decimal value has been multiplied by the indicated amount and then converted to an integer. Example: Value transmitted: 2345; indicated scaling: $10^2$; actual value: 23.45.

## Queue Command Parameters – Message ID 43

This message is output in response to Message ID 168, Poll Command Parameters. The response message will contain the requested parameters in the form of the requested message. In the example shown below, in response to a request to poll the static navigation parameters, this message has been sent with the payload of Message ID 143 (0x8F) contained in it. Since the payload of Message ID 143 is two bytes long, this message is sent with a payload 3 bytes long (Message ID 43, then the 2-byte payload of message 143).

Output Rate: Response to poll

This message outputs Packet/Send command parameters under SiRF Binary Protocol.

Example with MID_SET_STAT_NAV message:

A0A20003—Start Sequence and Payload Length

438F00—Payload

00D2B0B3—Message Checksum and End Sequence

*Table 3-76* Queue Command Parameters – Message ID 43

| Name | Bytes | Scale | Unit | Description |
|---|---|---|---|---|
| Message ID | 1 U | | | = 0x2B |
| Polled Msg ID[1] | 1 U | | | = 0x8F (example) |
| Data[2] | Variable[3] | | | Depends on the polled Message ID length |

Payload length: Variable length bytes (3 bytes in the example)

1. Valid Message IDs are 0x80, 0x85, 0x88, 0x89, 0x8A, 0x8B, 0x8C, 0x8F, 0x97, and 0xAA.
2. The data area is the payload of the message whose Message ID is listed in the Polled Msg ID field. For the specific details of the possible payloads, see the description of that message in Chapter 2
3. Data type follows the type defined for the Polled Message ID. For example, if the Polled Message ID is 128, see Message ID 128 payload definition in Table 2-6 on page 5 in Chapter 2, “Input Messages”.

## DR Raw Data – Message ID 45

*Table 3-77* 1-Hz DR Raw Data from ADC (Output After Collection of Data) – Message ID 45

| Name | Bytes | Scale | Unit | Description |
|---|---|---|---|---|
| Message ID | 1 | | | = 0x2D |
| 1st 100-ms time-tag | 4 | | ms | |
| 1st 100-ms ADC2 average measurement | 2 | | | |
| Reserved | 2 | | | |
| 1st 100-ms odometer count | 2 | | | |
| 1st 100-ms GPIO input states | 1 | | | Bit 0: reverse |
| 2nd 100-ms time-tag | 4 | | ms | |
| 2nd 100-ms ADC2 average measurement | 2 | | | |
| Reserved | 2 | | | |
| 2nd 100-ms odometer count | 2 | | | |
| 2nd 100-ms GPIO input states | 1 | | | Bit 0: reverse |
| ... | | | | |
| 10th 100-ms time-tag | 4 | | ms | |
| 10th 100-ms ADC2 average measurement | 2 | | | |
| Reserved | 2 | | | |
| 10th 100-ms odometer count | 2 | | | |
| 10th 100-ms GPIO input states | 1 | | | Bit 0: reverse |

Payload length: 111 bytes

## Test Mode 3/4 – Message ID 46

Message ID 46 is used by GSW2, SiRFDRive, SiRFLoc v3.x, GSW3, GSWLT3, and SLCLT3 software.

Output Rate: Variable – set by the period as defined in Message ID 150.

Example for GSW2, SiRFDRive, SiRFLoc v3.x, and GSW3 software output:

A0A20033—Start Sequence and Payload Length

2E0001001E00023F70001F0D2900000000000601C600051B0E000EB41A0000000 000000000000000000000000000000000000—Payload

0316B0B3—Message Checksum and End Sequence

Example for GSWLT3 and SLCLT3 software output:

A0A20033—Start Sequence and Payload Length

2E0001001E00023F70001F0D2900000000000601C600051B0E000EB41A0000000 000000000000000000000800000002F000000—Payload

0316B0B3—Message Checksum and End Sequence

*Table 3-78* Test Mode 3/4 – Message ID 46

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | **Scale** | **Example** | | **Scale** | **Example** |
| Message ID | 1 U | | 2E | | | 46 |
| SV ID | 2 U | | 0001 | | | 1 |
| Period | 2 U | | 001E | sec | | 30 |
| Bit Sync Time[1] | 2 U | | 0002 | sec | | 2 |
| Bit Count[1] | 2 U | | 3F70 | | | 16420 |
| Poor Status[1] | 2 U | | 001F | | | 31 |
| Good Status[1] | 2 U | | 0D29 | | | 3369 |
| Parity Error Count[1] | 2 U | | 0000 | | | 0 |
| Lost VCO Count[1] | 2 U | | 0000 | | | 0 |
| Frame Sync Time[1] | 2 U | | 0006 | sec | | 6 |
| C/N0 Mean | 2 S | *10 | 01C6 | dB/Hz | ÷10 | 45.4 |
| C/N0 Sigma | 2 S | *10 | 0005 | dB/Hz | ÷10 | 0.5 |
| Δ Clock Drift | 2 S | *10 | 1B0E | Hz | ÷10 | 692.6 |
| Clock Drift | 4 S | *10 | 000EB41A | Hz | ÷10 | 96361.0 |
| Bad 1 kHz Bit Count[1] | 2 S | | 0000 | | | 0 |
| Abs I20 ms[2] | 4 S | | 000202D5 | Counts | | 131797 |
| Abs Q1 ms[2] | 4 S | | 000049E1 | Counts | | 18913 |
| Phase Lock Indicator[3] | 4 S | | 00000000 | | 0.001 | 0 |
| RTC Frequency[4] | 2 S | | 8000 | Hz | | 32768 |
| ECLK Ratio[3] | 2 S | | 0000 | | 3*Value/ 65535 | 0 (no ECLK input) |
| Timer Synch input[3] (bit 7) AGC[3] (bit 0 - 6) | 1 D | | 2F | Timer Synch = True/False AGC = ~0.8 dB per step | | TS 0 = no activity and 47 for AGC |
| Reserved | 3 U | | | | | |

Payload length: 51 bytes

1. Field not filled for GSW3 and GSWLT3 software in Test Mode 3/4.
2. Phase error = (Q20 ms)/(I20 ms).
3. A value of 0.9 to 1.0 generally indicates phase lock
4. Only for GSWLT3 and SLCLT3 software

*Table 3-79* Detailed Description of Test Mode 3/4 Message ID 46

| Name | Description |
|---|---|
| Message ID | Message ID number |
| SV ID | The number of the satellite being tracked |
| Period | The total duration of time (in seconds) that the satellite is tracked. This field is not filled for GSW3 and GSWLT3 software in Test Mode 3/4. |
| Bit Sync Time | The time it takes for channel 0 to achieve the status of 0x37. This field is not filled for GSW3 and GSWLT3 software in Test Mode 3/4. |
| Bit Count | The total number of data bits that the receiver is able to demodulate during the test period. As an example, for a 20 second test period, the total number of bits that can be demodulated by the receiver is 12000 (50 bps x 20 sec x 12 channels). This field is not filled for GSW3 and GSWLT3 software in Test Mode 3/4. |

*Table 3-79* Detailed Description of Test Mode 3/4 Message ID 46 *(Continued)*

| Name | Description |
|---|---|
| Poor Status | This value is derived from phase accumulation time. Phase accumulation is the amount of time a receiver maintains phase lock. Every 100 msec of loss of phase lock equates to 1 poor status count. As an example, the total number of status counts for a 60 second period is 7200 (12 channels x 60 sec x 10 100-ms intervals). This field is not filled for GSW3 and GSWLT3 software in Test Mode 3/4. |
| Good Status | This value is derived from phase accumulation time. Phase accumulation is the amount of time a receiver maintains phase lock. Every 100 msec of phase lock equates to 1 good status count. This field is not filled for GSW3 and GSWLT3 software in Test Mode 3/4. |
| Parity Error Count | The number of word parity errors. This occurs when the transmitted parity word does not match the receivers parity check. This field is not filled for GSW3 and GSWLT3 software in Test Mode 3/4. |
| Lost VCO Count | The number of 1 msec VCO lost lock was detected. This occurs when the PLL in the RFIC loses lock. A significant jump in crystal frequency and / or phase causes a VCO lost lock. This field is not filled for GSW3 and GSWLT3 software in Test Mode 3/4. |
| Frame Sync | The time it takes for channel 0 to reach a 0x3F status. This field is not filled for GSW3 and GSWLT3 software in Test Mode 3/4. |
| C/N0 Mean | Calculated average of reported C/N0 by all 12 channels during the test period. |
| C/N0 Sigma | Calculated sigma of reported C/N0 by all 12 channels during the test period. |
| Clock Drift Change | Difference in clock drift from start and end of the test period. |
| Clock Drift | The measured internal clock drift. |
| Bad 1 kHz Bit Count | Errors in 1 ms post correlation I count values. This field is not filled for GSW3 and GSWLT3 software in Test Mode 3/4. |
| Abs I20 ms | Absolute value of the 20 ms coherent sums of the I count over the duration of the test period. |
| Abs Q1 ms | Absolute value of the 20 ms Q count over the duration of the test period. |
| Phase Lock Indicator | Quality of the received signal with 1 being perfect and decreasing as noise level increases. A value of 0.9 to 1.0 generally indicates phase lock. |
| RTC Frequency[1] | F(RTC counts/CLCKACQ counts over test interval). 16-bit unsigned integer value of RTC frequency in Hz.<br>Value = 0, no RTC<br>Value = 1 to 65534, 32678±1 = good RTC frequency<br>Value = 65535, RTC frequency = 65535 Hz of higher |
| ECLK Ratio[1] | F(ECLK counts/CLCKACQ counts over test interval). 16-bit unsigned integer value of scaled value of ratio.<br>Value = 0, no ECLK input<br>0< Value <3, Ratio = 3*Value/65535<br>Value >3, Ratio = 65535 |
| Timer Synch[1] | Timer Synch input activity bit<br>Value = 0, no Timer Synch input activity<br>Value = 1, activity |
| AGC[1] | Automatic Gain Control value<br>Value = 0, gain set to maximum saturated<br>1< Value > 62, active gain range<br>Value = 63, gain set to minimum saturated |

1. Supported only by GSWLT3 and SLCLT3 software. When test mode command is issued, test report interval time value and PRN are specified. Reports every interval whether SV signals or not and data is accumulated every interval period. Continuous output until software is reset or unit is restarted.

## Test Mode 4 – Message ID 48 (SiRFLoc v2.x only)

SiRFLoc results from Test Mode 4 are output by Message IDs 48 and 49. Message ID 48 for Test Mode 4 used by SiRFLoc version 2.x only is not to be confused with SiRFDRive Message ID 48.

*Table 3-80* Test Mode 4 – Message ID 48

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 | | 30 | | | 48 |
| nChannel | 1 | | 01 | | | 1 |
| Reserved | 4 | | 00000000 | | | 0 |
| Channel | 1 | | 00 | | | 0 |
| Satellite ID | 1 | | 18 | | | 24 |
| Receiver Time Tag | 4 | | 000660D0 | ms | | 30995 |
| Pseudo-range | 4 | A | 0 | m | 10 | 0 |
| Carrier Frequency | 4 | 64 | 174ADC | m/sec | 100 | 1526492 |

Payload length: 20 bytes

*Table 3-81* Detailed Description of Test Mode 4 Message ID 48

| Name | Description |
|---|---|
| Message ID | Message ID number |
| nChannel | Number of channels reporting |
| Reserved | Reserved |
| Channel | Receiver channel number for a given satellite being searched or tracked |
| Satellite ID | Satellite or Space Vehicle (SV ID number or Pseudo-Random Noise (PRN) number |
| Receiver Time Tag | Count of ms interrupts from the start of the receiver (power on) until measurement sample is taken. Millisecond interrupts are generated by the receiver clock |
| Pseudorange | Generated pseudorange measurement for a particular SV |
| Carrier Frequency | Can be interpreted in two ways:<br>1. Delta pseudorange normalized by the reciprocal of the delta pseudorange measurement interval<br>2. Frequency from the AFC loop. If, for example, the delta pseudorange interval computation for a particular channel is zero, it can be the AFC measurement, otherwise it is a delta pseudorange computation |

## DR Navigation Status – Message ID 48 (Sub ID 1)

DR navigation status information (output on every navigation cycle).

*Table 3-82* DR Navigation Status – Message ID 48 (Sub ID 1)

| Name | Bytes | Description |
|---|---|---|
| Message ID | 1 | = 0x30 |
| Message Sub ID | 1 | = 0x01 |
| DR navigation | 1 | 0x00 = valid DR navigation; else<br>Bit 0 ON : GPS-only navigation required<br>Bit 1 ON : speed not zero at start-up<br>Bit 2 ON : invalid DR position<br>Bit 3 ON : invalid DR heading<br>Bit 4 ON : invalid DR calibration<br>Bit 5 ON : invalid DR data<br>Bit 6 ON : system in Cold Start<br>Bit 7 : Reserved |

*Table 3-82* DR Navigation Status – Message ID 48 (Sub ID 1) *(Continued)*

| Name | Bytes | Description |
|---|---|---|
| DR data | 2 | 0x0000 = valid DR data; else<br>Bit 0 ON : DR gyro subsystem not operational<br>Bit 1 ON : DR speed subsystem not operational<br>Bit 2 ON : DR measurement time < 80 ms<br>Bit 3 ON : invalid serial DR message checksum<br>Bit 4 ON : no DR data for > 2 sec<br>Bit 5 ON : DR data timestamp did not advance<br>Bit 6 ON : DR data byte stream all 0x00 or 0xFF<br>Bit 7 ON : composite wheel-tick count jumped > 255 between successive DR messages<br>Bit 8 ON : input gyro data bits (15) of 0x0000 or 0x3FFF<br>Bit 9 ON : > 10 DR messages received in 1 sec<br>Bit 10 ON : time difference between two consecutive measurements is < = 0<br>Bits 11 - 15 : Reserved. |
| DR calibration and DR gyro bias calibration | 1 | Bits 0 - 3 : 0000 = valid DR calibration; else<br>Bit 0 ON : invalid DR gyro bias calibration<br>Bit 1 ON : invalid DR scale factor calibration<br>Bit 2 ON : invalid DR speed scale factor calibration<br>Bit 3 ON : GPS calibration required but not ready<br>Bits 4 - 6 : 000 = valid DR gyro bias calibration; else<br>Bit 4 ON : invalid DR data<br>Bit 5 ON : zero-speed gyro bias calibration not updated<br>Bit 6 ON : heading rate scale factor < = -1<br>Bit 7 : Reserved |
| DR gyro scale factor calibration and DR speed scale factor calibration | 1 | Bits 0 - 3 : 0000 = valid DR gyro scale factor calibration; else<br>Bit 0 ON : invalid DR heading<br>Bit 1 ON : invalid DR data<br>Bit 2 ON : invalid DR position<br>Bit 3 ON : heading rate scale factor < = -1<br>Bits 4 - 7 : 0000 = valid DR speed scale factor calibration; else<br>Bit 4 ON : invalid DR data<br>Bit 5 ON : invalid DR position<br>Bit 6 ON : invalid GPS velocity for DR<br>Bit 7 ON : DR speed scale factor < = -1 |
| DR Nav across reset and DR position | 1 | Bits 0 - 1 : 00 = valid DR nav across reset; else<br>Bit 0 ON : invalid DR navigation<br>Bit 1 ON : speed > 0.01 m/s<br>Bit 2 : Reserved<br>Bits 3 - 6 : 0000 = valid DR position; else<br>Bit 3 ON : speed not zero at start-up<br>Bit 4 ON : invalid GPS position<br>Bit 5 ON : system in Cold Start<br>Bit 6 ON : invalid DR data<br>Bit 7 : Reserved |
| DR heading | 1 | Bits 0 - 6 : 0000000 = valid DR heading; else<br>Bit 0 ON : speed not zero at start-up<br>Bit 1 ON : invalid GPS position<br>Bit 2 ON : invalid GPS speed<br>Bit 3 ON : GPS did not update heading<br>Bit 4 ON : delta GPS time < 0 and > 2<br>Bit 5 ON : system in Cold Start<br>Bit 6 ON : invalid DR data<br>Bit 7 : Reserved |

*Table 3-82* DR Navigation Status – Message ID 48 (Sub ID 1) *(Continued)*

| Name | Bytes | Description |
|---|---|---|
| DR gyro subsystem and DR speed subsystem | 1 | Bits 0 - 3 : 0000 = updated DR gyro bias and scale factor calibration; else<br>Bit 0 ON : invalid DR data<br>Bit 1 ON : invalid DR position<br>Bit 2 ON : invalid GPS velocity for DR<br>Bit 3 ON : GPS did not update heading<br>Bits 4 - 6 : 000 = updated DR speed calibration; else<br>Bit 4 ON : invalid DR data<br>Bit 5 ON : invalid DR position<br>Bit 6 ON : invalid GPS velocity for DR<br>Bit 7 : 0 = updated DR navigation state |
| DR Nav state integration ran and zero-speed gyro bias calibration updated | 1 | Bits 0 - 7 : 00000000 = GPS updated position; else<br>Bit 0 ON : update mode != KF<br>Bit 1 ON : EHPE > 50<br>Bit 2 ON : no previous GPS KF update<br>Bit 3 ON : GPS EHPE < DR EHPE<br>Bit 4 ON : DR EHPE < 50<br>Bit 5 ON : less than 4 SVs in GPS navigation<br>Bit 6 ON : no SVs in GPS navigation<br>Bit 7 ON : DR-only navigation required |
| Updated DR gyro bias/scale factor calibration, updated DR speed calibration, and updated DR Nav state | 1 | Bits 0 - 3 : 0000 = updated DR gyro bias and scale factor calibration; else<br>Bit 0 ON : invalid DR data<br>Bit 1 ON : invalid DR position<br>Bit 2 ON : invalid GPS velocity for DR<br>Bit 3 ON : GPS did not update heading<br>Bits 4 - 6 : 000 = updated DR speed calibration; else<br>Bit 4 ON : invalid DR data<br>Bit 5 ON : invalid DR position<br>Bit 6 ON : invalid GPS velocity for DR<br>Bit 7 : 0 = updated DR navigation state |
| GPS updated position | 1 | Bits 0 - 7 : 00000000 = GPS updated position; else<br>Bit 0 ON : update mode != KF<br>Bit 1 ON : EHPE > 50<br>Bit 2 ON : no previous GPS KF update<br>Bit 3 ON : GPS EHPE < DR EHPE<br>Bit 4 ON : DR EHPE < 50<br>Bit 5 ON : less than four SVs in GPS navigation<br>Bit 6 ON : no SVs in GPS navigation<br>Bit 7 ON : DR-only navigation required |
| GPS updated heading | 1 | Bits 0 - 6 : 0000000 = GPS updated heading; else<br>Bit 0 ON : update mode != KF<br>Bit 1 ON : GPS speed < = 5 m/s<br>Bit 2 ON : less than 4 SVs in GPS navigation<br>Bit 3 ON : horizontal velocity variance > 1 $m^2/s^2$<br>Bit 4 ON : GPS heading error > = DR heading error<br>Bit 5 ON : GPS KF not updated<br>Bit 6 ON : incomplete initial speed transient<br>Bit 7 : Reserved |

*Table 3-82* DR Navigation Status – Message ID 48 (Sub ID 1) *(Continued)*

| Name | Bytes | Description |
|---|---|---|
| GPS position & GPS velocity | 1 | Bits 0 - 2 : 000 = valid GPS position for DR; else<br>Bit 0 ON : less than 4 SVs in GPS navigation<br>Bit 1 ON : EHPE > 30<br>Bit 2 ON : GPS KF not updated<br>Bit 3 : Reserved<br>Bits 4 - 7 : 0000 = valid GPS velocity for DR; else<br>Bit 4 ON : invalid GPS position for DR<br>Bit 5 ON : EHVE > 3<br>Bit 6 ON : GPS speed < 2 m/s<br>Bit 7 ON : GPS did not update heading. |
| Reserved | 2 | Reserved |

Payload length: 17 bytes

## DR Navigation State – Message ID 48 (Sub ID 2)

DR speed, gyro bias, navigation mode, direction, and heading (output on every navigation cycle).

*Table 3-83* DR Navigation State – Message ID 48 (Sub ID 2)

| Name | Bytes | Scale | Unit | Description |
|---|---|---|---|---|
| Message ID | 1 | | | = 0x30 |
| Message Sub ID | 1 | | | = 0x02 |
| DR speed | 2 | $10^2$ | m/s | |
| DR speed error | 2 | $10^4$ | m/s | |
| DR speed scale factor | 2 | $10^4$ | | |
| DR speed scale factor error | 2 | $10^4$ | | |
| DR heading rate | 2 | $10^2$ | deg/s | |
| DR heading rate error | 2 | $10^2$ | deg/s | |
| DR gyro bias | 2 | $10^2$ | deg/s | |
| DR gyro bias error | 2 | $10^2$ | deg/s | |
| DR gyro scale factor | 2 | $10^4$ | | |
| DR gyro scale factor error | 2 | $10^4$ | | |
| Total DR position error | 4 | $10^2$ | m | |
| Total DR heading error | 2 | $10^2$ | deg | |
| DR Nav mode control | 1 | | | 1 = GPS-only nav required (no DR nav allowed)<br>2 = GPS + DR nav using default/stored calibration<br>3 = GPS + DR nav using current GPS calibration<br>4 = DR-only nav (no GPS nav allowed) |
| Reverse | 1 | | | DR direction: 0 = forward; 1 = reverse. |
| DR heading | 2 | $10^2$ | deg/s | |

Payload length: 32 bytes

## Navigation Subsystem – Message ID 48 (Sub ID 3)

Heading, heading rate, speed, and position of both GPS and DR (output on every navigation cycle).

*Table 3-84* Navigation Subsystem – Message ID 48 (Sub ID 3)

| Name | Bytes | Scale | Unit | Description |
|---|---|---|---|---|
| Message ID | 1 | | | = 0x30 |
| Message Sub ID | 1 | | | = 0x03 |
| GPS heading rate | 2 | $10^2$ | deg/s | |
| GPS heading rate error | 2 | $10^2$ | deg/s | |
| GPS heading | 2 | $10^2$ | deg | |
| GPS heading error | 2 | $10^2$ | deg | |
| GPS speed | 2 | $10^2$ | m/s | |
| GPS speed error | 2 | $10^2$ | m/s | |
| GPS position error | 4 | $10^2$ | m | |
| DR heading rate | 2 | $10^2$ | deg/s | |
| DR heading rate error | 2 | $10^2$ | deg/s | |
| DR heading | 2 | $10^2$ | deg | |
| DR heading error | 2 | $10^2$ | deg | |
| DR speed | 2 | $10^2$ | m/s | |
| DR speed error | 2 | $10^2$ | m/s | |
| DR position error | 4 | $10^2$ | m | |
| Reserved | 2 | | | |

Payload length: 36 bytes

## DR Gyro Factory Calibration – Message ID 48 (Sub ID 6)

DR gyro factory calibration parameters (response to poll).

*Table 3-85* DR Gyro Factory Calibration – Message ID 48 (Sub ID 6)

| Name | Bytes | Scale | Unit | Description |
|---|---|---|---|---|
| Message ID | 1 | | | = 0x30 |
| Message Sub ID | 1 | | | = 0x06 |
| Calibration | 1 | | | Bit 0 : Start gyro bias calibration<br>Bit 1 : Start gyro scale factor calibration<br>Bits 2 - 7 : Reserved |
| Reserved | 1 | | | |

Payload length: 4 bytes

## DR Sensors Parameters – Message ID 48 (Sub ID 7)

DR sensors parameters (response to poll).

*Table 3-86* DR Sensors Parameters – Message ID 48 (Sub ID 7)

| Name | Bytes | Scale | Unit | Description |
|---|---|---|---|---|
| Message ID | 1 | | | = 0x30 |
| Message Sub ID | 1 | | | = 0x07 |
| Base speed scale factor | 1 | | ticks/m | |
| Base gyro bias | 2 | $10^4$ | mV | |
| Base gyro scale factor | 2 | $10^3$ | mV/deg/s | |

Payload length: 7 bytes

## DR Data Block – Message ID 48 (Sub ID 8)

1-Hz DR data block (output on every navigation cycle).

*Table 3-87* DR Data Block – Message ID 48 (Sub ID 8)

| Name | Bytes | Scale | Unit | Description |
|---|---|---|---|---|
| Message ID | 1 | | | = 0x30 |
| Message Sub ID | 1 | | | = 0x08 |
| Measurement type | 1 | | | 0 = odometer and gyroscope (always);<br>1 .. 255 = Reserved |
| Valid count | 1 | | | Count (1 .. 10) of valid DR measurements |
| Reverse indicator | 2 | | | Bits 0 .. 9, each bit:<br>ON = reverse, OFF = forward |
| 1st 100-ms time-tag | 4 | | ms | |
| 1st 100-ms DR speed | 2 | $10^2$ | m/s | |
| 1st 100-ms gyro heading rate | 2 | $10^2$ | deg/s | |
| 2nd 100-ms time-tag | 4 | | ms | |
| 2nd 100-ms DR speed | 2 | $10^2$ | m/s | |
| 2nd 100-ms gyro heading rate | 2 | $10^2$ | deg/s | |
| ... | | | | |
| 10th 100-ms time-tag | 4 | | ms | |
| 10th 100-ms DR speed | 2 | $10^2$ | m/s | |
| 10th 100-ms gyro heading rate | 2 | $10^2$ | deg/s | |

Payload length: 86 bytes

## SID_GenericSensorParam – Message ID 48 (Sub ID 9)

Output message of Sensor Package parameters

**Note –** This message is not Supported by SiRFDemoPPC

The user can enable a one time transmission of this message via the SiRFDemo Poll command for SiRFDRive. In the SiRFDRive menu, select *Poll Sensors Parameters*.

*Table 3-88* DR Package Sensor Parameters – Message ID 48 (Sub ID 9)

| Byte | Name | Data Type | Bytes | Unit | Description | Res |
|---|---|---|---|---|---|---|
| 1 | Message ID | UINT8 | 1 | N/A | 0x30 | N/A |
| 2 | Sub-ID | UINT8 | 1 | N/A | 0x09 | N/A |
| 3 | Sensors[0] SensorType | UINT8 | 1 | N/A | GYRO_SENSOR = 0x1<br>ACCELERATION_SENSOR = 0x2 | N/A |
| 4 | Sensors[0] ZeroRateVolts | UINT32 | 4 | volts | 0 to 5.0[1] | 0.0001 |
| 8 | Sensors[0] MilliVoltsPer | UINT32 | 4 | millivolts | 0 to 1000[2] | 0.0001 |
| 12 | Sensors[0] ReferenceVoltage | UINT32 | 4 | volts | 0 to 5.0 | 0.0001 |
| 16 | Sensors[1] SensorType | UINT8 | 1 | N/A | GYRO_SENSOR = 0x1<br>ACCELERATION_SENSOR = 0x2 | N/A |
| 17 | Sensors[1] ZeroRateVolts | UINT32 | 4 | volts | 0 to 5.0 | 0.0001 |
| 21 | Sensors[1] MilliVoltsPer | UINT32 | 4 | millivolts | 0 to 1000 | 0.0001 |

*Table 3-88* DR Package Sensor Parameters – Message ID 48 (Sub ID 9) *(Continued)*

| | | | | | | |
|---|---|---|---|---|---|---|
| 25 | Sensors[1] ReferenceVoltage | UINT32 | 4 | volts | 0 to 5.0 | 0.0001 |
| 29 | Sensors[2] SensorType | UINT8 | 1 | N/A | GYRO_SENSOR = 0x1 ACCELERATION_SENSOR = 0x2 | N/A |
| 30 | Sensors[2] ZeroRateVolts | UINT32 | 4 | volts | 0 to 5.0 | 0.0001 |
| 34 | Sensors[2] MilliVoltsPer | UINT32 | 4 | millivolts | 0 to 1000 | 0.0001 |
| 38 | Sensors[2] ReferenceVoltage | UINT32 | 4 | volts | 0 to 5.0 | 0.0001 |
| 39 | Sensors[3] SensorType | UINT8 | 1 | N/A | GYRO_SENSOR = 0x1 ACCELERATION_SENSOR = 0x2 | N/A |
| 43 | Sensors[3] ZeroRateVolts | UINT32 | 4 | volts | 0 to 5.0 | 0.0001 |
| 47 | Sensors[3] MilliVoltsPer | UINT32 | 4 | millivolts | 0 to 1000 | 0.0001 |
| 51 | Sensors[3] ReferenceVoltage | UINT32 | 4 | volts | 0 to 5.0 | 0.0001 |

Payload length: 54 bytes

1. To restore ROM defaults for ALL sensors, enter the value 0xdeadabba here. You must still include the remainder of the message, but these values will be ignored.

2. For gyro this is millivolts per degree per second. For the acceleration sensor it is millivolts per metre per second ^ 2

# *Test Mode 4 – Message ID 49*

SiRFLoc results from Test Mode 4 are output by Message IDs 48 and 49. Message ID 48 for Test Mode 4 used by SiRFLoc version 2.x only is not to be confused with SiRFDRive Message ID 48.

*Table 3-89* Test Mode 4 – Message ID 49

| | | Binary (Hex) | | | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| **Name** | **Bytes** | **Scale** | **Example** | **Unit** | **Scale** | **Example** |
| Message ID | 1 | | 31 | | | 49 |
| nChannel | 1 | | 01 | | | 1 |
| Reserved | 4 | | 00000000 | | | 0 |
| Channel | 1 | | 00 | | | 0 |
| Satellite ID | 1 | | 18 | | | 24 |
| Receiver Time Tag | 4 | | 000660D0 | ms | | 31085 |
| Carrier Doppler Rate | 4 | 100000 | 796D | carrier cycles/2 ms/10 ms | 1048576 | 271 |
| Carrier Doppler | 4 | 100000 | 10F | carrier cycles/2 ms | 1048576 | 168229578 |
| Carrier Phase | 4 | 400 | | carrier cycles | 1024 | 94319770 |
| Code Offset | 4 | 181000 | FFFFFFFFFFF C925C | chip | 1576960 | -224676 |

Payload length: 28 bytes

*Table 3-90* Detailed Description of Test Mode 4 Message ID 49

| Name | Description |
|---|---|
| Message ID | Message ID number |
| nChannel | Number of channels reporting |
| Channel | Receiver channel number for a given satellite being searched or tracked |
| Satellite ID | Satellite or Space Vehicle (SV ID number or Pseudo-Random Noise (PRN) number |
| Receiver Time Tag | Count of ms interrupts from the start of the receiver (power on) until measurement sample is taken. Millisecond interrupts are generated by the receiver clock |
| Carrier Doppler Rate | Carrier Doppler Rate value from the Costas tracking loop for the satellite ID on channel 0 |
| Carrier Doppler | Frequency from the Costas tracking loop for the satellite ID on channel 0 |
| Carrier Phase | Carrier phase value from the Costas tracking loop for the satellite ID on channel 0 |
| Code Offset | Code offset from the Code tracking loop for the satellite ID on channel 0 |

## *SBAS Parameters – Message ID 50*

Outputs SBAS operating parameter information including SBAS PRN, mode, timeout, timeout source, and SBAS health status.

Output Rate: Every measurement cycle (full power / continuous: 1Hz)

Example:

A0A2000D—Start Sequence and Payload Length

327A0012080000000000000000—Payload

00C6B0B3—Message Checksum and End Sequence

*Table 3-91* SBAS Parameters – Message ID 50

| | | Binary (Hex) | | | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| **Name** | **Bytes** | **Scale** | **Example** | **Unit** | **Scale** | **Example** |
| Message ID | 1 U | | 32 | | | 50 |
| SBAS PRN | 1 U | | 7A | | | 122 |
| SBAS Mode | 1 U | | 00 | | | 0 |
| DGPS Timeout | 1 U | | 12 | sec | | 18 |
| Flag bits | 1 D | | 08 | | | 00001000 |
| Spare | 8 U | | 0000000000000000 | | | |

Payload length: 13 bytes

*Table 3-92* Detailed Description of SBAS Parameters

| Name | Description |
|---|---|
| Message ID | Message ID number |
| SBAS PRN | This is the PRN code of the SBAS either selected by the user, the default PRN, or that currently in use<br>0 = Auto mod<br>SBAS PRN 120-138 = Exclusive (set by user) |
| SBAS Mode | 0 = Testing, 1 = Integrity<br>Integrity mode does not accept SBAS corrections if the SBAS satellite is transmitting in a test mode<br>Testing mode accepts and use SBAS corrections even if the SBAS satellite is transmitting in a test mode |

*Table 3-92* Detailed Description of SBAS Parameters *(Continued)*

| Name | Description |
|---|---|
| DGPS Timeout | Range 0-255 seconds. 0 returns to default timeout. 1-255 is value set by user.<br>The default value is initially 18 seconds. However, the SBAS data messages may specify a different value.<br>The last received corrections continue to be applied to the navigation solution for the timeout period. If the timeout period is exceeded before a new correction is received, no corrections are applied. |
| Flag bits | Bit 0: Timeout; 0 = Default 1 = User<br>Bit 1: Health; 0 = SBAS is healthy 1 = SBAS reported unhealthy and can't be used<br>Bit 2: Correction; 0 = Corrections are being received and used 1 = Corrections are not being used because: the SBAS is unhealthy, they have not yet been received, or SBAS is currently disabled in the receiver<br>Bit 3: SBAS PRN; 0 = Default 1 = User<br><br>Note: Bits 1 and 2 are only implemented in GSW3 and GSWLT3, versions 3.3 and later |
| Spare | These bytes are currently unused and should be ignored |

# 1 PPS Time – Message ID 52

Output time associated with current 1 PPS pulse. Each message is output within a few hundred ms after the 1 PPS pulse is output and tells the time of the pulse that just occurred. The Message ID 52 reports the UTC time of the 1 PPS pulse when it has a current status message from the satellites. If it does not have a valid status message, it reports time in GPS time, and so indicates by means of the status field.

This message may not be supported by all SiRF Evaluation receivers

Output Rate: 1 Hz (Synchronized to PPS)

Example:

A0A20013—Start Sequence and Payload Length

3415122A0E0A07D3000D000000050700000000—Payload

0190B0B3—Message Checksum and End Sequence

*Table 3-93* Timing Message Data – Message ID 52

| | | Binary (Hex) | | | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| Name | Bytes | Scale | Example | Unit | Scale | Example |
| Message ID | 1 U | | 34 | | | 52 |
| Hour | 1 U | | 15 | | | 21 |
| Minute | 1 U | | 12 | | | 18 |
| Second | 1 U | | 2A | | | 42 |
| Day | 1 U | | 0E | | | 15 |
| Month | 1 U | | 0A | | | 10 |
| Year | 2 U | | 07D3 | | | 2003 |
| UTCOffsetInt[1] | 2 S | | 000D | | | 13 |
| UTCOffsetFrac[1] | 4 U | $10^9$ | 00000005 | sec | $10^9$ | 0.000000005 |
| Status (see Table 3-94) | 1 D | | 7 | | | 7 |
| Reserved | 4 U | | 00000000 | | | 00000000 |

Payload length: 19 bytes

1. Difference between UTC and GPS time, integer, and fractional parts. GPS time = UTC time + UTCOffsetInt+UTCOffsetFrac x $10^{-9}$.

*Table 3-94* Status Byte Field in Timing Message

| Bit Fields | Meaning |
|---|---|
| 0 | When set, bit indicates that time is valid |
| 1 | When set, bit indicates that UTC time is reported in this message. Otherwise, GPS time |
| 2 | When set, bit indicates that UTC to GPS time information is current, (i.e., IONO/UTC time is less than 2 weeks old) |
| 3-7 | Reserved |

# *Test Mode 4 Track Data – Message ID 55*

Message ID 55 is used by GSW3, GSWLT3, and SiRFLoc (v3.0 and above) software.

*Table 3-95* Test Mode 4 – Message ID 55

| | | Binary (Hex) | | | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| **Name** | **Bytes** | **Scale** | **Example** | **Unit** | **Scale** | **Example** |
| Message ID | 1 U | | 37 | | | 55 |
| SV ID | 2 U | | 0001 | | | 1 |
| Acqclk Lsq | 4 U | | 12345678 | | | 12345678 |
| Code Phase | 4 U | $2^{-11}$ | 0000 | Chips | | 0 |
| Carrier Phase | 4 S | $2^{-32}$ | 0000 | Cycles | | 0 |
| Carrier Frequency | 4 S | 0.000476 | 0000 | Hz | 0.000476 | 0 |
| Carrier Acceleration | 2 S | 0.476 | 0000 | Hz/sec | 0.476 | 0 |
| Code Corrections | 4 S | | 0000 | | | 0 |
| Code Offset | 4 S | $2^{-11}$ | 0000 | Chips | $2^{-11}$ | 0 |
| MSec Number[1] | 2 S | ms | 0006 | ms | 0.001 | 0.006 |
| Bit Number[1] | 4 S | 20 ms | 01C6 | 20 ms | 0.02 | 9.08 |
| Reserved | 4 U | | 0000 | | | |
| Reserved | 4 U | | 0000 | | | |
| Reserved | 4 U | | 0000 | | | |
| Reserved | 4 U | | 0000 | | | |

Payload length: 51 bytes

1. SiRFLocDemo combines MSec Number and Bit Number for this message output which gives the GPS time stamp.

# *Extended Ephemeris Data – Message ID 56*

Message ID 56 is used by GSW2 (2.5 or above), SiRFXTrac (2.3 or above), and GSW3 (3.2.0 or above), and GSWLT3 software. This message has three Sub IDs.

*Table 3-96* Extended Ephemeris – Message ID 56

| | | Binary (Hex) | | | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| **Name** | **Bytes** | **Scale** | **Example** | **Unit** | **Scale** | **Example** |
| Message ID | 1 U | | 38 | | | 56 |
| Message Sub ID | 1 U | | 01 | | | 1 |

Payload length: variable (2 bytes + Sub ID payload bytes)

# *GPS Data and Ephemeris Mask – Message ID 56 (Sub ID 1)*

Output Rate: Six seconds until extended ephemeris is received

Example:

A0A2000D—Start Sequence and Payload Length

380101091E00000E7402000001 – Payload (Message ID, Message Sub ID, time valid; GPS week = 2334; GPS TOW = 37000 seconds; request flag for satellite 30 and 1)

00E6B0B3—Message Checksum and End Sequence

*Table 3-97* GPS Data and Ephemeris Mask – Message ID 56 (Message Sub ID 1)

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 38 | | | 56 |
| Message Sub ID | 1 U | | 01 | | | 1 |
| GPS_TIME_VALID_FLAG | 1 U | | 01 | | | 1 |
| GPS Week | 2 U | 1 | 091E | | | 2334 |
| GPS TOW | 4 U | 10 | 00000E74 | sec | | 3700 |
| EPH_REQ_MASK | 4 D | | 02000001 | | | SVs 30 and 1 |

Payload length: 13 bytes

*Table 3-98* Detailed Description of GPS Data and Ephemeris Mask Parameters

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Message Sub ID | Message Sub ID number |
| GPS_TIME_VALID_FLAG | LSB bit 0 = 1, GPS week is valid<br>LSB bit 0 = 0, GPS week is not valid<br>LSB bit 1 = 1, GPS TOW is valid<br>LSB bit 1 = 0, GPS TOW is not valid |
| GPS Week | Extended week number. Range from 0 to no limit |
| GPS TOW | GPS Time Of Week. Multiply by 10 to get the time in seconds. Range 0 to 604800 seconds. |
| EPH_REQ_MASK | Mask to indicate the satellites for which new ephemeris is needed<br>MSB is used for satellite 32, and LSB is for satellite 1 |

## *Extended Ephemeris Integrity – Message ID 56 (Sub ID 2)*

Output Rate: Upon host's request

Example:

A0A2000E—Start Sequence and Payload Length

3802000000400000004000000040 – Payload (Message ID, Message Sub ID, invalid position and clocks for SVID 7, and unhealthy bit for SVID 7)

00FAB0B3—Message Checksum and End Sequence

*Table 3-99* Extended Ephemeris Integrity Parameters – Message 56 (Message Sub ID 2)

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | 38 | | | 56 |
| Message Sub ID | 1 U | | 02 | | | 2 |
| SAT_POS_VALIDITY_FLAG | 4 D | | 00000040 | | | flag = 1, SV = 7 |
| SAT_CLK_VALIDITY_FLAG | 4 D | | 00000040 | | | flag = 1, SV = 7 |
| SAT_HEALTH_FLAG | 4 D | | 00000040 | | | flag = 1, SV = 7 |

Payload length: 14 bytes

*Table 3-100* Detailed Description of Extended Ephemeris Integrity Parameters

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Message Sub ID | Message Sub ID number |
| SAT_POS_VALIDITY_FLAG | 1 = invalid position found, 0 = valid position<br>SVID 1 validity flag is in LSB and subsequent bits have validity flags for SVIDs in increasing order up to SVID 32 whose validity flag are in MSB |
| SAT_CLK_VALIDITY_FLAG | 1 = invalid clock found, 0 = valid clock<br>SVID 1 validity flag is in LSB and subsequent bits have validity flags for SVIDs in increasing order up to SVID 32 whose validity flag are in MSB |
| SAT_HEALTH_FLAG | 1 = unhealthy satellite, 0 = healthy satellite<br>SVID 1 health flag is in the LSB and subsequent bits have health flags for SVIDs in increasing order up to SVID 32 whose validity flag are in MSB |

## *Extended Ephemeris Integrity – Message ID 56 (Sub ID 3)*

This is the ephemeris status response message. It is output in response to Poll Ephemeris Status message, Message ID 232, Message Sub ID 2.

*Table 3-101* Contents of Message ID 56 Message (Message Sub ID 3)

| Name | Bytes | Description |
|---|---|---|
| Message ID | 1 | Hex 0x38, Decimal 56 |
| Message Sub ID | 1 | Message Sub ID, 3 |
| The following data are repeated 12 times: | | |
| SVID | 1 | Satellite PRN, range 0-32 |
| Source | 1 | Source for this ephemeris[1] |
| Week # | 2 | Week number for ephemeris |
| Time of ephemeris | 2 | toe: effective time of week for ephemeris (seconds / 16, range 0 to 37800) |
| Integrity | 1 | Not used |
| Age | 1 | Age of ephemeris (days).<br>Bit 0 to 3 contain the age of the ephemeris.<br>Bit 4 and bit 5 are bit-mapped to indicate the source of ephemeris.<br>* When bit 4 is set, the source is server-generated.<br>* When bit 5 is set, the source is client-generated. |

Payload length: 98 bytes

1. Source for ephemeris: 0 = none; 1 = from network aiding; 2 = from SV; 3 = from extended ephemeris aiding

The Poll Ephemeris Status input message includes a satellite ID mask that specifies the satellite PRN codes to output. This message reports on the ephemeris of the requested satellites, up to a maximum of 12. If more than 12 PRN codes are requested, this message reports on the 12 with the lowest PRN codes. If the receiver does not have data for a requested PRN, the corresponding fields are set to 0. If fewer than 12 satellites are requested, the unused fields in the message are set to 0.

# EE Provide Synthesized Ephemeris Clock Bias Adjustment Message – Message ID 56 (Sub ID 4)

Output Rate: Variable

Example:

A0A20056 – Start Sequence and Payload Length

3804 0170801E000000 00000000000000 00000000000000 00000000000000 00000000000000 00000000000000 00000000000000 00000000000000 00000000000000 00000000000000 00000000000000 00000000000000 (Payload, message id, sub-id, sv_id, se_TOE and clock_bias_adjust for 12 satellites).

3992B0B3 – Message Checksum and End Sequence

*Table 3-102*EE Provide Synthesized Ephemeris Clock Bias Adjustment Message – Message 56 (Message Sub ID 4)

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) |
|---|---|---|---|---|---|
| | | Scale | Example | | Scale |
| Message ID | 1 | | 38 | | Decimal 56 |
| Message Sub-ID | 1 | | 04 | | Message Sub-ID for the Ephemeris Extension Message |
| The following 3 fields are repeated 12 times | | | | | |
| SV_ID | 1 | 1 | | Dimensionless | SV_ID = 0 means fields SE_TOE and Clock_Bias_Adjust are invalid |
| SE_TOE | 2 | 2^4 | | Seconds | The TOE of the Synthesized Ephemeris for which the clock bias adjustment is being reported |
| Clock_Bias_Adjust | 4 | 2^-31 | | Second | Clock bias adjustment (for af0) |

Payload length: 84 bytes

# Ephemeris Extension Messages – Message ID 56 (Sub ID 38)

Used for the ephemeris extension feature. Four sub-messages are created with the same Message ID.

*Table 3-103* General Structure for the Ephemeris Extension Messages – Message ID 56 (Message Sub ID 38)

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) |
|---|---|---|---|---|---|
| | | Scale | Example | | Scale |
| Message ID | 1 | | 38 | | Decimal 56 |
| Message Sub-ID | 1 | | 01 | | Message Sub-ID for the Ephemeris Extension Message |
| EE Payload | Variable | | | | Payload length depends on Sub-ID |

Payload length: 2 + EE Payload

# Extended Ephemeris ACK – Message ID 56 (Sub ID 255)

Output Rate: Variable.

This message is returned when input Message ID 232 Message Sub ID 255 is received. Refer to Chapter 2, “Input Messages” for more details on Message ID 232.

Example:

A0A20004—Start Sequence and Payload Length

E8FFE8FF – Payload (ACK for message 232 Message Sub ID 255)

03CEB0B3—Message Checksum and End Sequence

*Table 3-104*Extended Ephemeris Ack – Message 56 (Message Sub ID 255)

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | E8 | | | 232 |
| Message Sub ID | 1 U | | FF | | | 255 |
| ACK ID | 1 U | | E8 | | | 232 |
| ACK Sub ID | 1 U | | FF | | | 255 |

Payload length: 4 bytes

*Table 3-105*Detailed Description of Extended Ephemeris Ack Parameters

| Name | Description |
|---|---|
| Message ID | Message ID number |
| Message Sub ID | Message Sub ID number |
| ACK ID | Message ID of the message to ACK |
| ACK Sub ID | Message Sub ID of the message to ACK |

## *Reserved – Message ID 225*

This output message is SiRF proprietary except for Message Sub ID 6.

## *Statistics Channel – Message ID 225 (Sub ID 6)*

The message is only used by GSW3, GSWLT3, and SiRFLoc v3.x software and outputs the TTFF, aiding accuracy information and navigation status.

Output Rate: Once after every reset.

**Note –** Message ID 225 (Message Sub ID 6) only comes out when the debug messages are enabled. The debug message feature is enabled by either setting the output rate of message 225 using Message ID 166 or by setting bit 5 (enable debug data bit) in the configuration bit map of Message ID 128.

**Note –** Message ID 225 (Message Sub ID 6) may not be output when the system is not able to compute a navigation solution. This message is not supported by APM.

Example:

A0A20027—Start Sequence and Payload Length

E106—Message ID and Message Sub ID

010000000000000000000000000000000000000000000000000000000010100000 00—Payload

0107B0B3—Message Checksum and End Sequence

*Table 3-106*Statistic Channel – Message ID 225 (Message Sub ID 6)

| | | | Binary (Hex) | | | ASCII (Decimal) | |
|---|---|---|---|---|---|---|---|
| **Name** | **Sub Field** | **Bytes** | **Scale** | **Example** | **Unit** | **Scale** | **Example** |
| Message ID | | 1 U | | E1 | | | 225 |
| Message Sub ID | | 1 U | | 06 | | | 6 |
| TTFF | Since reset | 2 U | | | sec | 0.1 | range from 0 .0 to 6553.5 |
| | Since all aiding received[1] | 2 U | | | | | 0 |
| | First nav since reset[1] | 2 U | | | | | 0 |
| Position Aiding Error | North[1] | 4 S | | | | | 0 |
| | East[1] | 4 S | | | | | 0 |
| | Down[1] | 4 S | | | | | 0 |
| Time Aiding Error[1] | | 4 S | | | | | 0 |
| Frequency Aiding Error[1] | | 2 S | | | | | 0 |
| Position Uncertainty | Horizontal[1] | 1 U | | | | | 0 |
| | Vertical[1] | 2 U | | | | | 0 |
| Time Uncertainty[1] | | 1 U | | | | | 0 |
| Frequency Uncertainty[1] | | 1 U | | | | | 0 |
| Number of Aided Ephemeris[1] | | 1 U | | | | | 0 |
| Number of Aided Acquisition Assistance[1] | | 1 U | | | | | 0 |
| Navigation and Position Status | Navigation Mode | 1 D | | | | | see Table 3-107 |
| | Position Mode | 1 D | | | | | see Table 3-108 |
| | Status | 2 D | | | | | see Table 3-109 and Table 3-110 |
| Start Mode | | 1 D | | | | | see Table 3-111 |
| Reserved[1] | | 1 U | | | | | |

Payload length: 39 bytes

1. Valid with SiRFLoc only

*Table 3-107*Description of the Navigation Mode Parameters

| Bit Fields | Description |
|---|---|
| 0 | No Nav |
| 1 | Approximate from SV records |
| 2 | Time transfer |
| 3 | Stationary mode |
| 4 | LSQ fix |
| 5 | KF nav |
| 6 | SiRFDRive |
| 7 | DGPS base |

*Table 3-108* Description of the Position Mode Parameters

| Bit Fields | Description |
|---|---|
| 0 | Least Square (LSQ) mode 0 – no bit sync, approximate GPS time |
| 1 | LSQ mode 1 – no bit sync, accurate GPS time |
| 2 | LSQ mode 2 – bit sync, no frame sync, approximate GPS time |
| 3 | LSQ mode 3 – bit sync, no frame sync, accurate GPS time |
| 4 | LSQ mode 4 – bit and frame sync, user time (without aiding) See Table 3-109 |
| 5 | KF mode – Kalman Filtering |
| 6 | No position |
| 7 | Not used |

*Table 3-109* Description of the Status for Navigation LSQ Fix Mode

| Value | Status |
|---|---|
| 0x00 | Good solution |
| 0x01 | Uncertainty exceeded maximum (UNCER_EXCEED) |
| 0x02 | Input information to navigation had error (INPUT_ERR) |
| 0x04 | Not sufficient information to have a fix position (UNDER_DETERM) |
| 0x08 | Matrix inversion failed (MATR_INVT) |
| 0x010 | LSQ iteration exceeds predefined maximum (ITER_OUT) |
| 0x020 | Altitude check failed (ALT_OUT) |
| 0x040 | GPS time check failed (TIME_OFF) |
| 0x080 | Failure found in measurements (FDI_FAIL) |
| 0x100 | DOP exceeded threshold (DOP_FAIL) |
| 0x200 | Velocity check failed (VEL_FAIL) |

*Table 3-110* Description of the Status for Navigation KF Mode

| Value | Status |
|---|---|
| 0 | Solution is good |
| 1 | No solution |
| 2 | Altitude is out of range |
| 3 | Velocity is out of range |

*Table 3-111* Description of the Start Mode

| Value | Description |
|---|---|
| 0x00 | Cold |
| 0x01 | Warm |
| 0x02 | Hot |
| 0x03 | Fast |

## *Development Data – Message ID 255*

Output Rate: Receiver generated.

Example:

A0A2....—Start Sequence and Payload Length

FF....—Payload

....B0B3—Message Checksum and End Sequence

*Table 3-112*Development Data – Message ID 255

| Name | Bytes | Binary (Hex) | | Unit | ASCII (Decimal) | |
|---|---|---|---|---|---|---|
| | | Scale | Example | | Scale | Example |
| Message ID | 1 U | | FF | | | 255 |
| Data[1] | variable U | | | | | |

Payload length: variable

1. Data area consists of at least 1 byte of ASCII text information.

---

**Note –** Message ID 255 is output when SiRF Binary is selected and development data is enabled. It can also be enabled by setting its output rate to 1 using Message ID 166. The data output using Message ID 255 is essential for SiRF-assisted troubleshooting support.

---

# Additional Information 4

## TricklePower Operation in DGPS Mode

When in TricklePower mode, serial port DGPS corrections are supported if the firmware supports them in full-power mode. If the CPU can be awakened from sleep mode by the UART receiving data (this feature exists in SiRFstarII receivers, not in SiRFstarIII), then the incoming corrections awaken the receiver, and it stores the incoming data in a buffer and applies them when it awakens. If the receiver cannot be awakened by UART interrupts, messages should only be sent when the receiver has indicated OK to send, or they will be lost.

When in TricklePower mode, the use of SBAS corrections is not supported in any receiver.

## GPS Week Reporting

The GPS week number represents the number of weeks that have elapsed since the week of January 6, 1980. Per ICD-GPS-200, the satellites only transmit the 10 LSBs of the week number. On August 22, 1999, the week number became 1024, which was reported by the satellites as week 0. SiRF receivers resolve the reported week number internally. When messages report the week number, that value is either truncated to the 10 LSBs or is called an extended week number (see messages 7 and 41 for examples).

## Computing GPS Clock Frequency

To compute GPS clock frequency, you must know the receiver architecture. For receivers which use a GPS clock frequency of 16.369 MHz (newer SiRFstarII, most SiRFstarIII receivers), Crystal Factor in the below formula is 16. For receivers which use a GPS clock frequency of 24.5535 MHz (older SiRFstarII receivers such as those using GSP2e/LP), the Crystal Factor is 24. Refer to your receiver's data sheet to determine the GPS clock frequency for your receiver.

Clock Frequency = (GPS L1 Frequency + Clock Drift) * Crystal Factor / 1540

For example, in a SiRFstarIII receiver (Crystal Factor = 16), Clock Drift is reported to be 94.315 kHz. Clock Frequency is:

Clock Frequency = (1575.42 MHz + 94.315 kHz) * 16 / 1540 = 16.3689799 MHz

If this is used in a receiver where the GPS TCXO is nominally 16.369 MHz, then this frequency is the actual frequency of the crystal. If another frequency crystal is used, you must account for the frequency conversion factors in the synthesizer to compute the crystal frequency.

To predict clock bias, use the relationships between frequency and velocity. The reported clock drift value can be converted to a velocity using the Doppler formula, since in the SiRF architecture the clock drift value is a bias to the computed Doppler frequency:

Doppler Frequency / Carrier Frequency = Velocity / speed of light

Or:

Velocity = Doppler Frequency / Carrier Frequency * c

Next, the velocity can be converted to a time factor by dividing by the speed of light:

Change in Clock Bias = Velocity / c

Combining the above 2 formulae,

Change in Clock Bias = Doppler Frequency / Carrier Frequency

For a Clock Drift of 94.315 kHz as used above,

Change in Clock Bias = 94315 Hz / 1575.42 MHz = 59.867 μs

**Note –** Reported clock bias and clock bias computed using the above formula will likely agree only to within a few nanoseconds because the actual measurement interval may be slightly more or less than an exact second, and the clock drift is only reported to a (truncated) 1 Hz resolution.

## ADDITIONAL AVAILABLE PRODUCT INFORMATION

| Part Number | Description |
|---|---|
| 1050-0042 | NMEA Reference Manual |


**North America**
**Corporate HQ**
(1) (408) 467-0410
Sales@sirf.com

**Europe**
**United Kingdom**
(44) (1344) 668390
SalesUK@sirf.com

**Germany**
(49) (81) 529932-90
SalesGermany@sirf.com

**Belgium**
(32) (496) 152969
SalesBelgium@sirf.com

**Asia Pacific**
**China**
(86) (21) 5854-7153
SalesChina@sirf.com

**Taiwan**
(886) (2) 8174-8966
SalesTaiwan@sirf.com

**Japan**
(81) (44) 829-2186
SalesJapan@sirf.com

**India**
(91) (80) 41966000
SalesIndia@sirf.com

**South Korea**
(82) (2) 3424-3150
SalesKorea@sirf.com





December 2007