µ-blox ag
Bertastrasse 1
CH-8003 Zürich
Switzerland
http://www.u-blox.ch

Protocol Specification

– µ-blox GPS-MS1 and GPS-PS1 –

4th April 2000

Focus

This specification applies to the following GPS receivers:

- GPS-PS1
- GPS-MS1

These products are based on the GPS chipset manufactured by SiRF Technology. Unless otherwise specified, these receivers default to SiRF Binary Protocol, at 19200 baud, 8 databit, no parity, 1 stop bit. This specification covers the SiRF Binary Protocol and NMEA Protocol as implemented on these receivers.

About This Document

This document contains information on a product under development at µ-blox AG. This information is intended to help you evaluate this product. µ-blox reserves the right to change or discontinue work on this product without notice.
Part Number: GPSE-PR-01/00
Revision: 1.32

Revision History		
Date	**Revision**	**Changes**
Jan. 8, 2000	1.32	Added new 1.3.2 messages
July 8, 1999	1.311	Changed default baud rate to 19200
June, 1999	1.31	Added OkToSend, NMEA with TPM Notes, TPM Recommendations
Jan. 6, 1999	1.30	Added Messages and Changes for firmware release 1.3. Hyperlink Support
Nov. 10, 1998	1.04	Corrected typo (Tables for Messagelength and Checksum)
Sept. 9, 1998	1.03	Corrected error in Pseudo Code for SiRF Binary
Jul. 8, 1998	1.02	Updated Message tables
Jun. 23, 1998	1.01	Updated to Firmware Release 1.21
Jun. 1, 1998	1.00	First Public Release
Jan 1998	0.10	Initial Version

Contents

Chapter 1

SiRF Binary Protocol Specification

The serial communication protocol is designed to include:

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

Contents

1.1 Protocol Layers

1.1.1 Transport Message

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

1.1.2 Transport

The transport layer of the protocol encapsulates a GPS message in two start characters and two stop characters. The values are chosen to be easily identifiable and such that they are unlikely to occur frequently in the data. In addition, the transport layer prefixes the message with a two-byte (15-bit) message length and a two-byte (15-bit) check sum. The values of the start and stop characters and the choice of a 15-bit values for length and check sum are designed such that both message length and check sum can not alias with either the stop or start code.

1.1.3 Message Validation

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

1.1.4 Payload Length

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

High Byte	Low Byte
0x00 …0x7F	0x00 …0xFF

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. Likewise, the SiRF receiving programs (e.g., SiRFdemo) may limit the actual size to something less than this maximum.

1.1.5 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 will use the big-endian order. It should be noted that some processors do not allow arbitrary byte alignment of multi-byte data and therefore care should be used when reading data delivered as payload data.

1.1.6 Checksum

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

High Byte	Low Byte
0x00 …0x7F	0x00 …0xFF

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

```
Let message to 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]
    index = index + 1
checkSum = checkSum AND (2^15-1).
```

1.2 Input Messages for SiRF Binary Protocol

***Note** – All input messages are sent in BINARY format.*

The following table lists the message list for the SiRF input messages.

Hex	ASCII	Name
0 x 80	128	Initialize Data Source
0 x 81	129	Switch to NMEA Protocol
0 x 82	130	Set Almanac (upload)
0 x 84	132	Software Version (Poll)
0 x 86	134	Set Main Serial Port
0 x 88	136	Mode Control
0 x 89	137	DOP Mask Control
0 x 8A	138	DGPS Control
0 x 8B	139	Elevation Mask
0 x 8C	140	Power Mask
0 x 8D	141	Editing Residual
0 x 8E	142	Steady-State Detection
0 x 8F	143	Static Navigation
0 x 90	144	Clock Status (Poll)
0 x 91	145	Set DGPS Serial Port
0 x 92	146	Almanac (Poll)
0 x 93	147	Ephemeris (Poll)
0 x 95	149	Set Ephemeris (Upload)
0 x 96	150	Switch Operating Mode
0 x 97	151	Set Trickle Power Mode
0 x 98	152	Navigation Parameters (Poll)
0 x A5	165	Set Uart Configuration

1.2.1 Initialize Data Source - Message I.D. 128

Table 1.1 contains the input values for the following example:
Warm start the receiver with the following initialization data: ECEF XYZ (-2686727 m, -4304282 m, 3851642 m), Clock Offset (75,000Hz), Time of Week (86,400s), 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
```

			Binary (Hex)		
Name	Bytes	Scale	Example	Units	Description
Message ID	1		80		ASCII 128
ECEF X	4		FFD700F9	meters	
ECEF Y	4		FFBE5266	meters	
ECEF Z	4		003AC57A	meters	
Clock Offset	4		000124F8	Hz	
Time of Week	4	*100	0083D600	seconds	
Week Number	2		039C		
Channels	1		0C		Range 1-12
Reset Config.	1		33		See table 1.2
Payload Length	25 bytes				

Table 1.1: Initialize Data Source

Bit	Description
0	Data valid flag – set warm/hot start
1	Clear ephemeris – set warm start
2	Clear memory – set cold start
3	Reserved (must be 0)
4	Enable raw track data (YES=1, NO=0)
5	Enable debug data (YES=1, NO=0)
6	Reserved (must be 0)
7	Reserved (must be 0)

Table 1.2: Reset Configuration Bitmap

Note – *If Raw Track Data is ENABLED then additionally message I.D. 0x05 (ASCII 5 - Raw Track Data), message I.D. 0x08 (ASCII 8 - 50BPS data), and message I.D. 0x07 (ASCII 7 - Clock Status) are transmitted. All messages are sent at 1 Hz.*

1.2.2 Switch To NMEA Protocol - Message I.D. 129

Table 1.3 contains the input values for the following example:
Request the following NMEA data at 4800 baud:

```
GGA - ON at 1 sec, GLL - OFF, GSA - ON at 5 sec,
GSV - ON at 5 sec, RMC-OFF, VTG-OFF
```

Example:

```
A0A20018  -- Start Sequence and Payload Length
810201010001050105010001000100010001000112C0  -- Payload
016AB0B3  -- Message Checksum and End Sequence
```

Name	Bytes	Scale	Binary (Hex) Example	Units	Description
Message ID	1		81		ASCII 129
Mode	1		02		
GGA Message [a]	1		01	1/s	See Chapter 2 for format.
Checksum [b]	1		01		
GLL Message	1		00	1/s	See Chapter 2 for format.
Checksum	1		01		
GSA Message	1		05	1/s	See Chapter 2 for format.
Checksum	1		01		
GSV Message	1		05	1/s	See Chapter 2 for format.
Checksum	1		01		
RMC Message	1		00	1/s	See Chapter 2 for format.
Checksum:	1		01		
VTG Message	1		00	1/s	See Chapter 2 for format.
Checksum	1		01		
Unused Field	1		00		Recommended value.
Unused Field	1		01		Recommended value.
Unused Field	1		00		Recommended value.
Unused Field	1		01		Recommended value.
Unused Field	1		00		Recommended value.
Unused Field	1		01		Recommended value.
Unused Field	1		00		Recommended value.
Unused Field	1		01		Recommended value.
Baud Rate	2		12C0		38400,19200,9600,4800
Payload Length	24 bytes				

Table 1.3: Switch To NMEA Protocol

[a] A value of 0x00 implies NOT to send message, otherwise data is sent at 1 message every X seconds requested (i.e., to request a message to be sent every 5 seconds, request the message using a value of 0x05.) Maximum rate is 1/255s.

[b] A value of 0x00 implies the checksum is NOT transmitted with the message (not recommended). A value of 0x01 will have a checksum calculated and transmitted as part of the message (recommended).

***Note** – In Trickle Power mode, update rate is specified by the user. When you switch to NMEA protocol, message update rate is also required. The resulting update rate is the product of the Trickle Power Update rate AND the NMEA update rate (i.e. Trickle Power update rate = 2 seconds, NMEA update rate = 5 seconds, resulting update rate is every 10 seconds, (2 X 5 = 10)).*

1.2.3 Set Almanac - Message I.D. 130

This command enables the user to upload an almanac to the GPS receiver.

Note – *This feature is not documented in this manual.*

1.2.4 Software Version - Message I.D. 132

Table 1.4 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

		Binary (Hex)			
Name	Bytes	Scale	Example	Units	Description
Message ID	1		84		ASCII 132
TBD	1		00		Reserved
Payload Length	2 bytes				

Table 1.4: Software Version

1.2.5 Set Main Serial Port - Message I.D. 134

Table 1.5 contains the input values for the following example:
Set Main Serial port to 9600,n,8,1.
Example:

`A0A20009` -- Start Sequence and Payload Length
`860000258008010000` -- Payload
`0134B0B3` -- Message Checksum and End Sequence

		Binary (Hex)			
Name	Bytes	Scale	Example	Units	Description
Message ID	1		86		decimal 134
Baud	4		00002580		38400,19200,9600,4800,2400,1200
Data Bits	1		08		8,7
Stop Bit	1		01		0,1
Parity	1		00		None=0, Odd=1, Even=2
Pad	1		00		Reserved
Payload Length	9 bytes				

Table 1.5: Set Main Serial Port

1.2.6 Mode Control - Message I.D. 136

Table 1.6 contains the input values for the following example:

3D Mode	= Always	Alt Constraining	= Yes
Degraded Mode	= clock then direction	TBD	=1
DR Mode	= Yes	Altitude	= 0
Alt Hold Mode	= Auto	Alt Source	=Last Computed
Coast Time Out	= 20	Degraded Time Out	=5
DR Time Out	= 2	Track Smoothing	= Yes

The meaning of these parameters and their effect on the navigation calculation process is documented in the Evaluation Kit Manual, Section "Navigation".
Example:

```
A0A2000E -- Start Sequence and Payload Length
880101010101000000002140501 -- Payload
00A9B0B3 -- Message Checksum and End Sequence
```

Name	Bytes	Binary (Hex) Example	Units	Description
Message ID	1	88		ASCII 136
3D Mode	1	01		Always 1 (unused)
Alt Constraint	1	01		YES=1, NO=0
Degraded Mode	1	01		See Table 1.7
TBD	1	01		Reserved
DR Mode	1	01		YES=1, NO=0
Altitude	2	0000	meters	range -1,000 to 10,000
Alt Hold Mode	1	00		Auto=0, Always=1,Disable=2
Alt Source	1	02		Last Computed=0,Fixed to=1, Dynamic=2[a]
Coast Time Out	1	14	seconds	0 to 120
Degraded Time Out	1	05	seconds	0 to 120
DR Time Out	1	01	seconds	0 to 120
Track Smoothing	1	01		YES=1, NO=0
Payload Length	14 bytes			

Table 1.6: Mode Control

[a]Currently not implemented

Byte Value	Description
0	Use Direction then Clock Hold
1	Use Clock then Direction Hold
2	Direction (Curb) Hold Only
3	Clock (Time) Hold Only
4	Disable Degraded Modes

Table 1.7: Degraded Mode Byte Value

1.2.7 DOP Mask Control - Message I.D. 137

Table 1.8 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
```

			Binary (Hex)		
Name	Bytes	Scale	Example	Units	Description
Message ID	1		89		ASCII 137
DOP Selection	1		00		See Table 1.9
GDOP Value	1		08		Range 1 to 50
PDOP Value	1		08		Range 1 to 50
HDOP Value	1		08		Range 1 to 50
Payload Length	5 bytes				

Table 1.8: DOP Mask Control

Byte Value	Description
0	Auto PDOP/HDOP
1	PDOP
2	HDOP
3	GDOP
4	Do Not Use

Table 1.9: DOP Selection

1.2.8 DGPS Control - Message I.D. 138

Table 1.10 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
```

			Binary (Hex)		
Name	Bytes	Scale	Example	Units	Description
Message ID	1		8A		ASCII 138
DGPS Selection	1		01		See Table 1.11
DGPS Time Out:	1		1E	seconds	Range 1 to 120
Payload Length	3 bytes				

Table 1.10: DGPS Control

Byte Value	Description
0	Auto
1	Exclusive
2	Never

Table 1.11: DGPS Selection

1.2.9 Elevation Mask - Message I.D. 139

Table 1.12 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
```

Name	Bytes	Scale	Binary (Hex) Example	Units	Description
Message ID	1		8B		ASCII 139
Tracking Mask	2	*10	0032	degrees	Not currently used
Navigation Mask	2	*10	009B	degrees	Range -20.0 to 90.0
Payload Length	5 bytes				

Table 1.12: Elevation Mask

1.2.10 Power Mask - Message I.D. 140

Table 1.13 contains the input values for the following example:
Navigation mask to 33 dBHz (tracking default value of 28)
Example:

```
A0A20003    -- Start Sequence and Payload Length
8C1C21      -- Payload
00C9B0B3    -- Message Checksum and End Sequence
```

Name	Bytes	Scale	Binary (Hex) Example	Units	Description
Message ID	1		8C		ASCII 140
Tracking Mask	1		1C	dBHz	Not currently implemented
Navigation Mask	1		21	dBHz	Range 28 to 50
Payload Length	3 bytes				

Table 1.13: Power Mask

1.2.11 Editing Residual- Message I.D. 141

***Note** – Not currently implemented.*

1.2.12 Steady State Detection - Message I.D. 142

Table 1.14 contains the input values for the following example:
Set Steady State Threshold to 1.5 m/sec2
Example:

`A0A20002` -- Start Sequence and Payload Length
`8E0F` -- Payload
`009DB0B3` -- Message Checksum and End Sequence

			Binary (Hex)		
Name	Bytes	Scale	Example	Units	Description
Message ID	1		8E		ASCII 142
Threshold	1	*10	0F	m/sec2	Range 0 to 20
Payload Length	2 bytes				

Table 1.14: Steady State Detection

1.2.13 Static Navigation- Message I.D. 143

***Note** – Not currently implemented.*

1.2.14 Clock Status - Message I.D. 144

Table 1.15 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

Name	Bytes	Scale	Binary (Hex) Example	Units	Description
Message ID	1		90		ACSII 144
TBD	1		00		Reserved
Payload Length	2 bytes				

Table 1.15: Clock Status

1.2.15 Set DGPS Serial Port - Message I.D. 145

Table 1.16 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

Name	Bytes	Scale	Binary (Hex) Example	Units	Description
Message ID	1		91		ASCII 145
Baud	4		00002580		38400,19200,9600,4800,2400,1200
Data Bits	1		08		8,7
Stop Bit	1		01		0,1
Parity	1		00		None=0, Odd=1, Even=2
Pad	1		00		Reserved
Payload Length	9 bytes				

Table 1.16: Set DGPS Serial Port

1.2.16 Almanac - Message I.D. 146

Table 1.17 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

Name	Bytes	Scale	Binary (Hex) Example	Units	Description
Message ID	1		92		ASCII 146
TBD	1		00		Reserved
Payload Length	2 bytes				

Table 1.17: Almanac

To this request, the receiver replies with a MID 14.

1.2.17 Ephemeris Message I.D. 147

Table 1.18 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

			Binary (Hex)		
Name	Bytes	Scale	Example	Units	Description
Message ID	1		93		ASCII 147
Sv I.D.[a]	1		00		Range 0 to 32
TBD	1		00		Reserved
Payload Length	3 bytes				

Table 1.18: Ephemeris Message I.D. 147

[a] A value of 0 requests all available ephemeris records, otherwise the ephemeris of the Sv I.D. is requested.

To this request, the receiver replies with a MID 15.

1.2.18 Switch To SiRF Protocol

Note – *To switch to SiRF protocol you must send a SiRF NMEA message to revert to SiRF binary mode. (See Section 2 NMEA Input Messages for more information.)*

1.2.19 Switch Operating Modes - Message I.D. 150

Table 1.19 contains the input values for the following example:
Sets the receiver to track a single satellite on all channels.
Example:

`A0A20007` -- Start Sequence and Payload Length
`961E510006001E` --Payload
`0129B0B3` -- Message Checksum and End Sequence

			Binary (Hex)		
Name	Bytes	Scale	Example	Units	Description
Message ID	1		96		ASCII 150
Mode	2		1E51		1E51=test, 0=normal
SvID	2		0006		Satellite to Track
Period	2		001E	seconds	Duration of Track
Payload Length	7 bytes				

Table 1.19: Switch Operating Mode

1.2.20 Set Trickle Power Parameters - Message I.D. 151

Table 1.20 contains the input values for the following example:
Sets the receiver into low power Modes.
Example:
Set receiver into Trickle Power at 1 hz update and 200 ms On Time.

```
A0A20009          -- Start Sequence and Payload Length
97000000C8000000C8    Payload
0227B0B3          -- Message Checksum and End Sequence
```

			Binary (Hex)		
Name	Bytes	Scale	Example	Units	Description
Message ID	1		97		ASCII 151
Push To Fix Mode	2		0000		ON = 1, OFF = 0
Duty Cycle	2	*10	00C8	%	% Time ON
Milli Seconds On Time	4		000000C8	ms	range 200 - 500 ms
Payload Length	9 bytes				

Table 1.20: Trickle Power Parameters

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 ms). To calculate the TricklePower update rate as a function of Duty cycle and On Time, use the following formula:

$$OffTime = \frac{OnTime - (DutyCycle * OnTime)}{DutyCycle}$$

$$Updaterate = OffTime + OnTime$$

Note – *On Time inputs of > 900 ms will default to 1000 ms*

Table 1.21 lists some examples of selections:

Mode	On Time (ms)	Duty Cycle (%)	Update Rate(1/Hz)
Continuous	1000	100	1
Trickle Power	300	30	1
Trickle Power	400	20	2
Trickle Power	400	10	4
Trickle Power	500	5	10

Table 1.21: Example of Selections for Trickle Power Mode of Operation

Note – *To confirm the receiver is performing at the specified duty cycle and ms On Time, open the 12-Channel Signal Level View Screen in SiRFDemo. The C/No data bins will be fully populated at 100% duty and only a single C/No data bin populated at 20% duty cycle. Your position should be updated at the computed update rate.*

In Release 131 144 there are Trickle Power limitations. See Table 1.22 for supported/unsupported settings.

	1 second	2 - 8 seconds
200	no	no
300	yes	no
400	yes	yes
>400	yes	yes

Table 1.22: Trickle Power Mode Settings

Push-to-Fix

In this mode, the receiver will turn on periodically to check whether ephemeris collection is required (i.e., if a new satellite has become visible). If it is required, the receiver will collect ephemeris at that time. In general this takes on the order of 18 to 30 seconds. If it is not required, the receiver will turn itself off again.
The user specifies the DutyCycle parameter, ranging up to 10%. In either case, the amount of time the receiver remains off will be in proportion to how long it stayed on:

$$Offperiod = \frac{OnPeriod * (1 - DutyCycle)}{DutyCycle}$$

Off Period is limited to not more than 30 minutes, which means that in practice the duty cycle will not be less than approximately On Period/1800, or about 1%. Because Push-to-Fix keeps the ephemeris for all visible satellites up to date, a position/velocity fix can generally be computed relatively quickly when requested by the user: on the order of 3 seconds versus 46 seconds if Push-to-Fix were not available and the receiver warm-started.

Note – *The 3 second TTFF figure increases to 6 seconds if the off period exceeds 30 minutes. Frame synchronization is commanded in this case.*
For more information on Trickle power mode, see the Trickle Power Mode application note.

1.2.21 Poll Navigation Parameters - Message I.D. 152

Table 1.23 contains the input values for the following example:
Poll receiver for current navigation parameters. The receiver replies with a Message ID 19.
Example:

```
A0A20002    -- Start Sequence and Payload Length
9800    --Payload
0098B0B3    -- Message Checksum and End Sequence
```

Name	Bytes	Binary (Hex) Scale	Example	Units	Description
Message ID	1		98		ASCII 152
Reserved	1		00		Reserved
Payload Length	2 bytes				

Table 1.23: Poll Navigation Parameters

1.2.22 Set UART Configuration - Message I.D. 165

This command will configure the receiver for Data I/O with unique protocols at specified communication parameters. On GPS-MS1, all 4 ports are configured at the same time. On GPS-PS1, only the configuraitons for Port 0 and Port 1 are valid.
Example:
The following sets a GPS-MS1 to:

Serial Port	Protocol	Baud Rate	Databits/Parity/Stop Bits
Port 0	SiRF Binary	19200	8/N/1
Port 1	NMEA	4800	8/N/1
Port 2	RTCM-DGPS	9600	8/N/1
Port 3	No Input/Output		

```
A0A20031   -- Start Sequence and Payload Length
A50000000004B00080100000001010100012C0080100000002030300
0025800801000000FF0505000000000000000000   -- Payload
0396B0B3   -- Message Checksum and End Sequence
```

Name	Bytes	Scale	Binary (Hex) Example	Units	Description
Message ID	1		A5		ASCII 165
Serial Port Number	1		00		Fixed. See table 1.25
Input Protocol	1		00		See table 1.26
Output Protocol	1		00		See table 1.26
Baud Rate	4		00004B00	BPS	See table 1.27
Data Bits	1		08		Number of Data Bits (7 or 8)
Stop Bits	1		01		Number of Stop Bits (0 or 1)
Parity Bits	1		00		0=None, 1=Odd, 2=Even
reserved	2		0000		reserved for future use
Serial Port Number	1		01		Fixed. See table 1.25
Input Protocol	1		01		See table 1.26
Output Protocol	1		01		See table 1.26
Baud Rate	4		000012C0	BPS	See table 1.27
Data Bits	1		08		Number of Data Bits (7 or 8)
Stop Bits	1		01		Number of Stop Bits (0 or 1)
Parity Bits	1		00		0=None, 1=Odd, 2=Even
reserved	2		0000		reserved for future use
Serial Port Number	1		02		Fixed. See table 1.25
Input Protocol	1		03		See table 1.26
Output Protocol	1		03		See table 1.26
Baud Rate	4		00002580	BPS	See table 1.27
Data Bits	1		08		Number of Data Bits (7 or 8)
Stop Bits	1		01		Number of Stop Bits (0 or 1)
Parity Bits	1		00		0=None, 1=Odd, 2=Even
reserved	2		0000		reserved for future use
Serial Port Number	1		03		Fixed. See table 1.25
Input Protocol	1		05		See table 1.26
Output Protocol	1		05		See table 1.26
Baud Rate	4		00000000	BPS	See table 1.27
Data Bits	1		00		Number of Data Bits (7 or 8)
Stop Bits	1		00		Number of Stop Bits (0 or 1)
Parity Bits	1		00		0=None, 1=Odd, 2=Even
reserved	2		0000		reserved for future use
Payload Length	49 bytes				

Table 1.24: Set UART Configuration

The structure of this Set UART Configuration message are four identical blocks, each block representing the settings for the 4 Serial I/O ports. Every block starts with the Serial port number. If the serial port should keep its current setting, the Serial Port Number should be set to 0xff (Decimal 255) in order to maintain the current settings.

Serial Port	GPS-MS1	GPS-PS1
0	Port 0 (Pins 30/31)	Port A (Pins 3/5)
1	Port 1 (Pins 28/29)	Port B (Pins 1/7)
2	Port 2 (Pins 16/17)	n/a
3	Port 3 (Pins 14/15)	n/a
255	Ignore	Ignore

Table 1.25: Serial Port Assignment

The following restrictions apply to the current implementation of the Multi-Channel I/O Capability:

- One protocol can only be configured for one output port. For example, it is not possible to have NMEA out on two or more ports at the same time.
- Input- and Output protocols can not be mixed. For example, one can not configure a port to have RTCM input and NMEA output on the same port. When sending the Configuration command, always make sure that the Input Protocol and the Output Protocol Settings are identical.
- When Ports 2 and 3 are used, trickle power mode will not be available.

Protocol Name	Value	Description
SiRFBinary	00	Binary Navigation Data I/O Protocol
NMEA	01	ASCII-based Navigation Data I/O Protocol
ASCII	02	for GPS-SCK users
RTCM	03	Differential Correction Input Data
User 1	04	User 1 Protocol for GPS-SCK users
No I/O	05	No Input/Output
User 2	06	User 2 Protocol for GPS-SCK users
User 3	07	User 3 Protocol for GPS-SCK users
User 4	08	User 4 Protocol for GPS-SCK users

Table 1.26: Valid Protocols

1200
2400
4800
9600
19200
38400

Table 1.27: Valid Baud Rate Settings

1.3 Output Messages for SiRF Binary Protocol

Note – *All output messages are received in BINARY format. SiRFdemo interprets the binary data and saves it to the log file in ASCII format.*

Table 1.28 lists the message list for the SiRF output messages.

Hex	ASCII	Name	Description
0 x 02	2	Measured Navigation Data	Position, velocity, and time
0 x 04	4	Measured Tracking Data	Signal to noise information
0 x 05	5	Raw Track Data	Measurement information
0 x 06	6	SW Version	Receiver software
0 x 07	7	Clock Status	
0 x 08	8	50 BPS Subframe Data	Standard ICD format
0 x 09	9	Throughput	CPU load
0 x 0B	11	Command Acknowledgment	Successful request
0 x 0C	12	Command NAcknowledgment	Unsuccessful request
0 x 0D	13	Visible List	
0 x 0E	14	Almanac Data	
0 x 0F	15	Ephemeris Data	
0 x 12	18	OkToSend	CPU ON / OFF (Trickle Power)
0 x 13	19	Navigation Parameters	Response to Poll
0 x FF	255	Development Data	Various data messages

Table 1.28: SiRF Messages - Output Message List

1.3.1 Measure Navigation Data Out - Message I.D. 2

Output Rate: 1 Hz

Table 1.29 lists the binary and ASCII message data format for the measured navigation data

Example:

```
A0A20029   -- Start Sequence and Payload Length
02FFD6F78CFFBE536E003AC00400030104A00036B039780E3
0612190E160F04000000000000   -- Payload
09BBB0B3   -- Message Checksum and End Sequence
```

		Binary (Hex)			ASCII (Decimal)	
Name	Bytes	Scale	Example	Units	Scale	Example
Message ID	1		02			2
X-position	4		FFD6F78C	m		-2689140
Y-position	4		FFBE536E	m		-4304018
Z-position	4		003AC004	m		3850244
X-velocity	2	*8	0000	m/s	Vx/8	0
Y-velocity	2	*8	0003	m/s	Vy/8	0.375
Z-velocity	2	*8	0001	m/s	Vz/8	0.125
Mode 1	1		04	Bitmap[a]		4
DOP[b]	1	*5	A		/5	2.0
Mode 2	1		00	Bitmap[c]		0
GPS Week	2		036B			875
GPS TOW	4	*100	039780E3	seconds	/100	602605.79
SVs in Fix	1		06			6
CH 1	1		12			18
CH 2	1		19			25
CH 3	1		0E			14
CH 4	1		16			22
CH 5	1		0F			15
CH 6	1		04			4
CH 7	1		00			0
CH 8	1		00			0
CH 9	1		00			0
CH 10	1		00			0
CH 11	1		00			0
CH 12	1		00			0
Payload Length	41 bytes					

Table 1.29: Measured Navigation Data Out

[a] Table 1.30 lists the meaning of this field.

[b] Dilution of precision (DOP) field contains value of PDOP when position is obtained using 3D solution and HDOP in all other cases.

[c] Table 1.31 lists the meaning of the individual bits.

Note – *Binary units scaled to integer values need to be* divided *by the scale value to receive true decimal value (i.e., decimal Xvel = (binary Xvel)/8).*

Mode 1	Hex	ASCII	Description
Bit 0..2	0x00	0	No Navigation Solution
	0x01	1	1 Satellite Solution[a]
	0x02	2	2 Satellite Solution[b]
	0x03	3	3 Satellite Solution (2D) [c]
	0x04	4	$\geq$ 4 Satellite Solution (3D)
	0x05	5	2D Point Solution (Krause)
	0x06	6	3D Point Solution (Krause)
	0x07	7	Dead Reckoning
Bit 3	0x08	8	Reserved
Bit 4	0x10	16	Reserved
Bit 5	0x20	32	Reserved
Bit 6	0x40	64	DOP Mask Exceeded
Bit 7	0x80	128	DGPS Position

Table 1.30: Mode 1

[a] Altitude hold, direction hold and time hold
[b] Altitude hold and direction or time hold
[c] Altitude hold

Example: A value of 0x84 (132) is a DGPS fix with $\geq$ 4 Satellites (3D)

Mode 2	Hex	Ascii	Description
Bit 0	0x01	1	DR Sensor Data
Bit 1	0x02	2	1=Validated Fix/0=Unvalidated
Bit 2	0x04	4	if set, Dead Reckoning (Time Out)
Bit 3	0x08	8	if set, Output Edited by UI (i.e., DOP Mask exceeded)
Bit 4	0x10	16	Reserved
Bit 5	0x20	32	Reserved
Bit 6	0x40	64	Reserved
Bit 7	0x80	128	Reserved

Table 1.31: Mode 2

1.3.2 Measured Tracker Data Out - Message I.D. 4

Output Rate: 1 Hz
Table 1.32 lists the binary and ASCII message data format for the measured tracker data.
Example:

A0A200BC -- Start Sequence and Payload Length
04036C0000937F0C0EAB46003F1A1E1D1D191D1A1A1D1F1D59423F1A1A... --Payload
****B0B3 -- Message Checksum and End Sequence

		Binary (Hex)			ASCII (Decimal)	
Name	Bytes	Scale	Example	Units	Scale	Example
Message ID	1		04	None		4
GPS Week	2		036C			876
GPS TOW	4	s*100	0000937F	s	s/100	37759
Chans	1		0C			12
1st SVid	1		0E			14
Azimuth	1	Az*[2/3]	AB	deg	/[2/3]	256.5
Elev	1	El*2	46	deg	/2	35
State	2		003F	Bitmap (Table 1.33)		63
C/No 1	1		1A			26
C/No 2	1		1E			30
C/No 3	1		1D			29
C/No 4	1		1D			29
C/No 5	1		19			25
C/No 6	1		1D			29
C/No 7	1		1A			26
C/No 8	1		1A			26
C/No 9	1		1D			29
C/No 10	1		1F			31
2nd SVid	1		1D			29
Azimuth	1	Az*[2/3]	59	deg	/[2/3]	89
Elev	1	El*2	42	deg	/2	66
State	2		3F	Bitmap (Table 1.33)		63
C/No 1	1		1A			26
C/No 2	1		1A			63
....						
Payload Length	188 bytes					

Table 1.32: Measured Tracker Data Out

Note – *Message length is fixed to 188 bytes with nontracking channels reporting zero values.*

Note – *When a channel is fully locked and all data is valid, the status shown is 0 x BF.*

Field Definition	Hex Value	Description
ACQ_SUCCESS	0x0001	Set if acq/reacq is done successfully
DELTA_CARPHASE_VALID	0x0002	Integrated carrier phase is valid
BIT_SYNC_DONE	0x0004	Bit sync completed flag
SUBFRAME_SYNC_DONE	0x0008	Subframe sync has been done
CARRIER_PULLIN_DONE	0x0010	Carrier pullin done
CODE_LOCKED	0x0020	Code locked
ACQ_FAILED	0x0040	Failed to acquire S/V
GOT_EPHEMERIS	0x0080	Ephemeris data available

Table 1.33: TrktoNAVStruct.trk_status Field Definition

Note – *The status is reflected by the value of all bits as the receiver goes through each stage of satellite acquisition. The status will have a 0xBF value when a channel is fully locked and all data is valid.*

1.3.3 Raw Tracker Data Out - Message I.D. 5

GPS Pseudo-Range and Integrated Carrier Phase Computations Using SiRF Binary Protocol

This section describes the necessary steps to compute the GPS pseudo-range, pseudo-range rate, and integrated carrier phase data that can be used for post processing applications such as alternative navigation filters. This data enables the use of third party software to calculate and apply differential corrections based on the SiRF binary protocol. Additionally, description and example code is supplied to calculate the measurement data and decode the broadcast ephemeris required for post processing applications.

Output Rate: 1 Hz

Table 1.34 lists the binary and ASCII message data format for the raw tracker data.

Example:

```
a0a20033 -- Start Sequence and Payload Length
05000000010007003f00e2facf000a00830000dbb0
0012bc0900004a40fa2d2a6a0000 -- Payload
2e2e2d2e2e2e2e2e2e2e000003e80005
0ae6b0b3 -- Message Checksum and End Sequence
```

		Binary (Hex)			ASCII (Decimal)	
Name	Bytes	Scale	Example	Units	Scale	Example
Message ID	1		05			5
Channel	4		00000001			1
SVID	2		0007			07
State	2		003F	Bitmap (Table 1.33)		63
Bits	4		00e2facf	bit		14875343
ms	2		000a	ms		10
Chips	2		0083	chip		131
Code Phase	4	2^{16}	0000dbb0	chip	2^{-16}	0.858154296875
Carrier Doppler	4	2^{10}	0012bc09	rad/2ms	2^{-10}	1199.00878906
Time Tag	4		00004a40	ms		19008
Delta Carrier[a]	4	2^{10}	fa2d2a6a	cycles	2^{-10}	- 95413.3964844
Search Count	2		0000			0
C/No 1	1		2e	dBHz		46
C/No 2	1		2e	dBHz		46
C/No 3	1		2d	dBHz		45
C/No 4	1		2e	dBHz		46
C/No 5	1		2e	dBHz		46
C/No 6	1		2e	dBHz		46
C/No 7	1		2e	dBHz		46
C/No 8	1		2e	dBHz		46
C/No 9	1		2e	dBHz		46
C/No 10	1		2e	dBHz		46
Power Bad Cnt	1		00			0
Phase Bad Cnt	1		00			0
Delta Car Int	2		03e8	ms		1000
Correl Int	2		0005			5
Payload Length	51 bytes per satellite tracked (up to 12)					

Table 1.34: Raw Tracker Data Out

[a]Multiply by $\frac{1000}{4\pi}2^{-16}$ to convert to Hz.

***Note –** The status is reflected by the value of all bits as the receiver goes through each stage of satellite acquisition. The status will have a 0xBF value when a channel is fully locked and all data is valid.*

Message ID	Each SiRF binary message is defined based on the ID. Channel: Receiver channel where data was measured (range 1-12).
SVID	PRN number of the satellite on current channel.
State	Current channel tracking state (see Table 1.33).
Bit Number	Number of GPS bits transmitted since Sat-Sun midnight (in Greenwich) at a 50 bps rate.

Millisecond Number	Number of milliseconds of elapsed time since the last received bit (20 ms between bits).
Chip Number	Current C/A code symbol being transmitted (range 0 to 1023 chips; 1023 chips = 1 ms).
Code Phase	Fractional chip of the C/A code symbol at the time of sampling (scaled by 2 -16 , = 1/65536).
Carrier Doppler	The current value of the carrier frequency as maintained by the tracking loops.

Note – *The Bit Number, Millisecond Number, Chip Number, Code Phase, and Carrier Doppler are all sampled at the same receiver time.*

Receiver Time Tag	This is the count of the millisecond interrupts from the start of the receiver (power on) until the measurement sample is taken. The ms interrupts are generated by the receiver clock.
Delta Carrier Phase	The difference between the carrier phase (current) and the carrier phase (previous). Units are in carrier cycles with the LSB = 0.00185 carrier cycles. The delta time for the accumulation must be known.

Note – *Carrier phase measurements are not necessarily in sync with code phase measurement for each measurement epoch.*

Search Count	This is the number of times the tracking software has completed full satellite signal searches.
C/No	Ten measurements of carrier to noise ratio (C/No) values in dBHz at input to the receiver. Each value represents 100 ms of tracker data and its sampling time is not necessarily in sync with the code phase measurement.
Power Loss Count	The number of times the power detectors fell below the threshold between the present code phase sample and the previous code phase sample. This task is performed every 20 ms (max count is 50).
Phase Loss Count	The number of times the phase lock fell below the threshold between the present code phase sample and the previous code phase sample. This task is performed every 20 ms (max count is 50).
Integration Interval	The time in ms for carrier phase accumulation. This is the time difference (as calculated by the user clock) between the Carrier Phase (current) and the Carrier Phase (previous).

Track Loop Iteration	The tracking Loops are run at 2 ms and 10 ms intervals. Extrapolation values for each interval is 1 ms and 5 ms for range computations.

Calculation of Pseudo-Range Measurements

The pseudo-range measurement in meters can be determined from the raw track data by solving the following equation:

$$Pseudorange(PR) \quad = \quad [ReceivedTime(RT) - TransmitTime(TT)] * C$$

where C = speed of light.
The following variables from the raw track data are required for each satellite:

- Bit Number (BN) - 50 bits per second
- Millisecond Number (MSN)
- Chip Number (CN)
- Code Phase (CP)
- Receiver Time Tag (RTTag)
- Delta Carrier Phase (DCP)

The following steps are taken to get the psr data and carrier data for each measurement epoch.

Note – *See source code to* `calcpsr` *(part of the Toolkit).*

---

1. Computation of initial Receiver Time (RT) in seconds.

   Note – *The initial arbitrary value is chosen at start up to make the PR reasonable (i.e., set equal to TT + 70 ms) and then incriminated by one second for each measurement epoch.*

   ---

2. Computation of Transmit Time (TT) in seconds.
3. Calculate Pseudo-range at a common receiver time of the first channel of the measurement data set.

***Note** – All channel measurements are NOT taken at the same time. Therefore, all ranges must be extrapolated to a common measurement epoch. For simplicity, the first channel of each measurement set is used as the reference to which all other measurements are extrapolated.*

4. Extrapolate the pseudo-range based on the correlation interval to improve precision.
5. Compute the delta range.

If the accumulation time of the Delta Carrier Phase is 1000 ms then the measurement is valid and can be added to the previous Delta Carrier Phase to get Accumulated Carrier Phase data. If the accumulation time of the Delta Carrier Phase is not equal to 1000 ms then the measurement is not valid and the accumulation time must be restarted to get Accumulated Carrier Phase data.

1.3.4 Response: Software Version String - Message I.D. 6

Output Rate: Response to polling message
Example:

`A0A20015` -- Start Sequence and Payload Length
`0606312E322E30444B495431313920534D0000000000` -- Payload
`0382B0B3` -- Message Checksum and End Sequence

Name	Bytes	Scale	Binary (Hex) Example	Units	Scale	ASCII (Decimal) Example
Message ID	1		06			6
Character	20		06312E32 2E30444B 49543131 3920534D 00000000			1.2.0DKit119 SM
Payload Length	21 bytes					

Table 1.35: Software Version String

***Note** – Convert to symbol to assemble message (i.e., 0x4E is 'N'). These are low priority task and are not necessarily output at constant intervals.*

1.3.5 Response: Clock Status Data - Message I.D. 7

Output Rate: 1 Hz or response to polling message

Example:

`A0A20014` -- Start Sequence and Payload Length
`0703BD021549240822317923DAEF` -- Payload
`0598B0B3` -- Message Checksum and End Sequence

Name	Bytes	Binary (Hex) Scale	Example	Units	ASCII (Decimal) Scale	Example
Message ID	1		07			7
GPS Week	2		03BD			957
GPS TOW	4	s*100	02154924	s	s/100	349494.12
Svs	1		08			8
Clock Drift	4		2231	Hz		74289
Clock Bias	4		7923	nano s		128743715
Estimated GPS Time	4		DAEF	milli s		349493999
Payload Length	20 bytes					

Table 1.36: Clock Status Data Message

1.3.6 50 BPS Data - Message I.D. 8

Output Rate: As available (12.5 minute download time)
Example:

`A0A2002B` -- Start Sequence and Payload Length
`08********` --Payload
`****B0B3` -- Message Checksum and End Sequence

Name	Bytes	Binary (Hex) Scale	Example	Units	ASCII (Decimal) Scale	Example
Message ID	1		08			8
Channel	1					
Sv I.D	1					
Word[10]	40					
Payload Length	43 bytes per subframe (6 subframes per page, 25 pages Almanac)					

Table 1.37: 50 BPS Data

Note – *Data is logged in ICD format (available from* `www.navcen.uscg.mil`*).*

1.3.7 CPU Throughput - Message I.D. 9

Output Rate: 1 Hz
Example:

`A0A20009` -- Start Sequence and Payload Length
`09003B0011001601E5` -- Payload
`0151B0B3` -- Message Checksum and End Sequence

		Binary (Hex)			ASCII (Decimal)	
Name	Bytes	Scale	Example	Units	Scale	Example
Message ID	1		09			9
SegStatMax	2	*186	003B	milli s	/186	.3172
SegStatLat	2	*186	0011	milli s	/186	.0914
AveTrkTime	2	*186	0016	milli s	/186	.1183
Last MS	2		01E5	milli s		485
Payload Length	9 bytes					

Table 1.38: CPU Throughput

1.3.8 Command Acknowledgment - Message I.D. 11

Output Rate: Response to successful input message
This is successful almanac (message ID 0x92) request example:

`A0A20002` -- Start Sequence and Payload Length
`0B92` -- Payload
`009DB0B3` -- Message Checksum and End Sequence

		Binary (Hex)			ASCII (Decimal)	
Name	Bytes	Scale	Example	Units	Scale	Example
Message ID	1		0B			11
Ack. I.D.	1		92			146
Payload Length	2 bytes					

Table 1.39: Command Acknowledgment

1.3.9 Command NAcknowledgment - Message I.D. 12

Output Rate: Response to rejected input message
This is unsuccessful almanac (message ID 0x92) request example:

`A0A20002` -- Start Sequence and Payload Length
`0C92` -- Payload
`009EB0B3` -- Message Checksum and End Sequence

		Binary (Hex)			ASCII (Decimal)	
Name	Bytes	Scale	Example	Units	Scale	Example
Message ID	1		0C			12
NAck. I.D.	1		92			146
Payload Length	2 bytes					

Table 1.40: Command NAcknowledgment

1.3.10 Visible List - Message I.D. 13

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 1.41). Maximum is 12 satellites.*

Example:

```
A0A2002A -- Start Sequence and Payload Length
0D080700290038090133002C****************** -- Payload
****B0B3 -- Message Checksum and End Sequence
```

Name	Bytes	Binary (Hex) Scale	Binary (Hex) Example	Units	ASCII (Decimal) Scale	ASCII (Decimal) Example
Message ID	1		0D			13
Visible Svs	1		08			8
CH 1 - Sv I.D.	1		07			7
CH 1 - Sv Azimuth	2		0029	degrees		41
CH 1 - Sv Elevation	2		0038	degrees		56
CH 2 - Sv I.D.	1		09			9
CH 2 - Sv Azimuth	2		0133	degrees		307
CH 2 - Sv Elevation	2		002C	degrees		44
.....						
Payload Length	62 bytes (maximum)					

Table 1.41: Visible List

1.3.11 Almanac Data - Message I.D. 14

Output Rate: Response to poll
Example:

```
A0A203A1 -- Start Sequence and Payload Length
0E01************* -- Payload
****B0B3 -- Message Checksum and End Sequence
```

Name	Bytes	Binary (Hex) Scale	Binary (Hex) Example	Units	ASCII (Decimal) Scale	ASCII (Decimal) Example
Message I.D.	1		0E			14
Sv I.D. (1)	1		01			1
AlmanacData[14][2]	28					
....						
Sv I.D. (32)	1		20			32
AlmanacData[14][2]	28					
Payload Length	929 bytes					

Table 1.42: Almanac Data

1.3.12 Ephemeris Data - Message I.D. 15

This message consists of the ephemeris data for a particular SVID. Data consisting of 45 16-bit unsigned integers which make up 3 subframes of data each consisting of 15 unsigned 16 bit integers. This data is the ephemeris subframe data collected from the 50BPS data stream, and compressed by packing the each subframe from 10 subframe words (32 bits/word) into 15 words (16 bits/word) with the tlm and parity words stripped off. Successful reception will be indicated by ACK message output.
Output Rate: Response to poll

		Binary (Hex)			ASCII (Decimal)	
Name	Bytes	Scale	Example	Units	Scale	Example
Message I.D.	1		0F			15
Sv I.D. (1)	1		01			1
EphemerisData[45][2]	90					
Payload Length	92 bytes					

Table 1.43: Ephemeris Data

1.3.13 OkToSend - Message I.D. 18

Output Rate: Trickle Power CPU on/off indicator
Example:

```
A0A20002    -- Start Sequence and Payload Length
1200        -- Payload
0012B0B3    -- Message Checksum and End Sequence
```

		Binary (Hex)			ASCII (Decimal)	
Name	Bytes	Scale	Example	Units	Scale	Example
Message I.D.	1		12			18
Send Indicator	1		00[a]			00
Payload Length	2 bytes					

Table 1.44: OkToSend

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

1.3.14 Navigation Parameters (Response to Poll) - Message I.D. 19

Output Rate: Response to Poll
Example:

```
A0A20018    -- Start Sequence and Payload Length
130100000000011E3C0104001E004B1E00000500016400C8    Payload
022DB0B3    -- Message Checksum and End Sequence
```

		Binary (Hex)			ASCII (Decimal)	
Name	Bytes	Scale	Example	Units	Scale	Example
Message I.D.	1		13			19
Altitude Constraint	1		01			1
Altitude Hold Mode	1		00			0
Altitude Hold Source	1		00			0
Altitude Source Input	2		0000	meters		0
Degraded Mode(see table 1.6)	1		01			1
Degraded Timeout	1		1E	seconds		30
DR Timeout	1		3C	seconds		60
Track Smooth Mode	1		01			1
DOP Mask Mode (see table 1.8)	1		04			4
DGPS Mode (see table 1.10)	1		00			0
DGPS Timeout	1		1E	seconds		30
Elevation Mask	2	*10	004B	degrees	/10	7.5
Power Mask	1		1E	dBHz		30
Editing Residual	2		0000			0
Steady-State Detection	1	*10	05	m/s^2	/10	0.5
Static Navigation	1	*10	00		/10	0
Low Power Mode (see 1.20)	1		01			1
Low Power Duty Cycle	1		64	percent		100
Low Power On-Time	2		00C8	ms		200
Payload Length	24 bytes					

Table 1.45: Navigation Parameters

1.3.15 Development Data - Message I.D. 255

Output Rate: Receiver generated

Example:

```
A0A2****  --Start Sequence and Payload Length
FF**************  --Payload
****B0B3  -- Message Checksum and End Sequence
```

		Binary (Hex)			ASCII (Decimal)	
Name	Bytes	Scale	Example	Units	Scale	Example
Message ID	1		FF			255
Payload Length	variable					

Table 1.46: Development Data

Note *– Messages are output to give the user information of receiver activity. Convert to symbol to assemble message (i.e.,0 x 4E is 'N'). These are low priority task and are not necessarily output at constant intervals.*

Chapter 2

NMEA Input/Output Messages

The GPS receiver provided by µ-blox AG may also output data in NMEA-0183 format as defined by the National Marine Electronics Association (NMEA), Standard For Interfacing Marine Electronic Devices, Version 2.20, January 1, 1997.

Contents

2.1 NMEA Checksum

All NMEA sentences have an optional checksum. The Checksum can be enabled/disabled when setting up the NMEA Protocol. The optional checksum field consists of a "`*`" and two hex digits representing the exclusive OR of all characters between, but not including, the "`$`" and "`*`". The following pseudo code calculates a checksum over an array of characters "`line`". The first character in the array is "`line[0]`":

```
line = getline()
index = 1
```

```
checksum = 0

while line[index] <> '*' do
    checksum = checksum EXOR line[index]
    index = index+1

end while

```

2.2 NMEA Output Messages

The GPS receiver outputs the following messages as shown in table 2.1:

NMEA Record	Description
GGA	Global positioning system fixed data
GLL............	Geographic position - latitude/longitude
GSA...........	GNSS DOP and active satellites
GSV...........	GNSS satellites in view
RMC	Recommended minimum specific GNSS data
VTG...........	Course over ground and ground speed

Table 2.1: NMEA-0183 Output Messages

2.2.1 GGA – Global Positioning System Fixed Data

Table 2.2 contains the values for the following example:

```
$GPGGA,161229.487,3723.2475,N,12158.3416,W,1,07,1.0,9.0,M, , , ,0000*18
```

Name	Example	Units	Description
Message ID	$GPGGA		GGA protocol header
UTC Time	161229.487		hhmmss.sss
Latitude	3723.2475		ddmm.mmmm
N/S Indicator	N		N=north or S=south
Longitude	12158.3416		dddmm.mmmm
E/W Indicator	W		E=east or W=west
Position Fix Indicator[a]	1		
Satellites Used	07		Range 0 to 12
HDOP	1.0		Horizontal Dilution of Precision
MSL Altitude[b]	9.0	meters	
Units	M	meters	
Geoid Separation[b]		meters	
Units	M	meters	
Age of Diff. Corr.		second	Null fields when DGPS is not used
Diff. Ref. Station ID	0000		
Checksum	*18		
CR LF			End of message termination

Table 2.2: GGA Data Format

[a]See Table 2.3
[b]This implementation does not support geoid corrections. Values are WGS84 ellipsoid heights

Value	Description
0	Fix not available or invalid
1	GPS SPS Mode, fix valid
2	Differential GPS, SPS Mode, fix valid
3	GPS PPS Mode, fix valid

Table 2.3: Position Fix Indicator

2.2.2 GLL – Geographic Position - Latitude/Longitude

Table 2.4 contains the values for the following example:

`$GPGLL,3723.2475,N,12158.3416,W,161229.487,A*2C`

Name	Example	Units	Description
Message ID	$GPGLL		GLL protocol header
Latitude	3723.2475		ddmm.mmmm
N/S Indicator	N		N=north or S=south
Longitude	12158.3416		dddmm.mmmm
E/W Indicator	W		E=east or W=west
UTC Time	161229.487		hhmmss.sss
Status	A		A=data valid or V=data not valid
Checksum	*2C		
CR LF			End of message termination

Table 2.4: GLL Data Format

2.2.3 GSA – GNSS DOP and Active Satellites

Table 2.5 contains the values for the following example:

```
$GPGSA,A,3,07,02,26,27,09,04,15, , , , , ,1.8,1.0,1.5*33
```

Name	Example	Units	Description
Message ID	$GPGSA		GSA protocol header
Mode 1	A		See Table 2.7
Mode 2	3		See Table 2.6
Satellite Used[a]	07		Sv on Channel 1
Satellite Used[a]	02		Sv on Channel 2
....			....
Satellite Used[a]			Sv on Channel 12
PDOP	1.8		Position Dilution of Precision
HDOP	1.0		Horizontal Dilution of Precision
VDOP	1.5		Vertical Dilution of Precision
Checksum	*33		
CR LF			End of message termination

Table 2.5: GSA Data Format

[a] Used in solution

Value	Description
1	Fix not available
2	2D
3	3D

Table 2.6: Mode 1

Value	Description
M	Manual–forced to operate in 2D or 3D mode
A	Automatic–allowed to automatically switch 2D/3D

Table 2.7: Mode 2

2.2.4 GSV – GNSS Satellites in View

Table 2.8 contains the values for the following example:

```
$GPGSV,2,1,07,07,79,048,42,02,51,062,43,26,36,256,42,27,27,138,42*71
$GPGSV,2,2,07,09,23,313,42,04,19,159,41,15,12,041,42*41
```

Name	Example	Units	Description
Message ID	$GPGSV		GSV protocol header
Number of Messages[a]	2		Range 1 to 3
Message Number[a]	1		Range 1 to 3
Satellites in View	07		
Satellite ID	07		Channel 1 (Range 1 to 32)
Elevation	79	degrees	Channel 1 (Maximum 90)
Azimuth	048	degrees	Channel 1 (True, Range 0 to 359)
SNR (C/No)	42	dBHz	Range 0 to 99, null when not tracking
....			....
Satellite ID	27		Channel 4 (Range 1 to 32)
Elevation	27	degrees	Channel 4 (Maximum 90)
Azimuth	138	degrees	Channel 4 (True, Range 0 to 359)
SNR (C/No)	42	dBHz	Range 0 to 99, null when not tracking
Checksum	*71		
CR LF			End of message termination

Table 2.8: GSV Data Format

[a] Depending on the number of satellites tracked multiple messages if GSV data may be required. This is reported in the "Number of Messages" and "Message Number" fields.

2.2.5 RMC – Recommended Minimum Specific GNSS Data

Table 2.9 contains the values for the following example:

```
$GPRMC,161229.487,A,3723.2475,N,12158.3416,W,0.13,309.62,120598, ,*10
```

Name	Example	Units	Description
Message ID	$GPRMC		RMC protocol header
UTC Time	161229.487		hhmmss.sss
Status	A		A=data valid or V=data not valid
Latitude	3723.2475		ddmm.mmmm
N/S Indicator	N		N=north or S=south
Longitude	12158.3416		dddmm.mmmm
E/W Indicator	W		E=east or W=west
Speed Over Ground	0.13	knots	
Course Over Ground	309.62	degrees	True
Date	120598		ddmmyy
Magnetic Variation[a]		degrees	E=east or W=west
Checksum	*10		
CR LF			End of message termination

Table 2.9: RMC Data Format

[a]Magnetic Declination is not supported. All "course over ground" data are geodetic WGS84 directions.

2.2.6 VTG – Course Over Ground and Ground Speed

Table 2.10 contains the values for the following example:

```
$GPVTG,309.62,T, ,M,0.13,N,0.2,K*6E
```

Name	Example	Units	Description
Message ID	$GPVTG		VTG protocol header
Course	309.62	degrees	Measured heading
Reference	T		True
Course		degrees	Measured heading
Reference[a]	M		Magnetic
Speed	0.13	knots	Measured horizontal speed
Units	N		Knots
Speed	0.2	km/hr	Measured horizontal speed
Units	K		Kilometer per hour
Checksum	*6E		
CR LF			End of message termination

Table 2.10: VTG Data Format

[a] Magnetic Declination is not supported. All "course over ground" data are geodetic WGS84 directions.

2.3 SiRF Proprietary NMEA Input Messages

NMEA input messages are provided to allow you to control the GPS receiver while in NMEA protocol mode. The receiver may be put into NMEA mode by sending the SiRF Binary protocol message (Section 1.2.2) using a user program or using `Sirfdemo.exe` and selecting Switch to NMEA Protocol from the Action menu. If the receiver is in SiRF Binary mode, all NMEA input messages are ignored. Once the receiver is put into NMEA mode, the following messages may be used to command the module.

2.3.1 Transport Message

Start Sequence	Payload	Checksum	End Sequence
$PSRF<MID>[a]	Data[b]	*CKSUM[c]	<CR> <LF>[d]

[a] Message Identifier consisting of three numeric characters. Input messages begin at MID 100.
[b] Message specific data. Refer to a specific message section for <data>...<data> definition.
[c] CKSUM is a two-hex character checksum as defined in the NMEA specification. Use of checksums is required on all input messages.
[d] Each message is terminated using Carriage Return (CR) Line Feed (LF) which is \r \n which is hex 0D 0A. Because \r \n are not printable ASCII characters, they are omitted from the example strings, but must be sent to terminate the message and cause the receiver to process that input message.

Note – *All fields in all proprietary NMEA messages are required, none are optional. All NMEA messages are comma delimited.*

2.3.2 SiRF NMEA Input Messages

Message	Message Identifier (MID)	Description
SetSerialPort	100	Set PORT A parameters and protocol
Navigation Initialization	101	Parameters required for start using X/Y/Z
SetDGPSPort	102	Set PORT B parameters for DGPS input
Query/Rate Control	103	Query standard NMEA message and/or set output rate
LLANavigationInitialization	104	Parameters required for start using Lat/Lon/Alt (WGS84)
Development Data On/Off	105	Development Data messages On/Off

2.3.3 SetSerialPort

This command message is used to set the protocol (SiRF Binary or NMEA) and/or the communication parameters (baud, data bits, stop bits, parity). Generally, this command is used to switch the module back to SiRF Binary protocol mode where a more extensive command message set is available. When a valid message is received, the parameters are stored in battery-backed SRAM and then the receiver restarts using the saved parameters.
Table 2.11 contains the input values for the following example:
Switch to SiRF Binary protocol at 9600,8,N,1

```
$PSRF100,0,9600,8,1,0*0C
```

Name	Example	Units	Description
Message ID	$PSRF100		PSRF100 protocol header
Protocol	0		0=SiRF Binary, 1=NMEA
Baud	9600		4800, 9600, 19200, 38400
DataBits	8		8,7[a]
StopBits	1		0,1
Parity	0		0=None, 1=Odd, 2=Even
Checksum	*0C		
CR LF			End of message termination

Table 2.11: Set Serial Port Data Format

[a]SiRF protocol is only valid for 8 data bits, 1 stop bit and no parity.

2.3.4 NaviagtionInitialization

This command is used to initialize the module for a warm start, by providing current position (in X, Y, Z coordinates), clock offset, and time. This enables the GPS receiver to search for the correct satellite signals at the correct signal parameters. Correct initialization parameters enable the receiver to acquire signals quickly.
Table 2.12 contains the input values for the following example:

Start using known position and time.

```
$PSRF101,-2686700,-4304200,3851624,95000,497260,921,12,3*22
```

Name	Example	Units	Description
Message ID	$PSRF101		PSRF101 protocol header
ECEF X	-2686700	meters	X coordinate position
ECEF Y	-4304200	meters	Y coordinate position
ECEF Z	3851624	meters	Z coordinate position
ClkOffset	95000	Hz	Clock Offset of the Unit[a]
TimeOfWeek	497260	seconds	GPS Time Of Week
WeekNo	921		GPS Week Number
ChannelCount	12		Range 1 to 12
ResetCfg	3		See 2.13
Checksum	*22		
CR LF			End of message termination

Table 2.12: Navigation Initialization Data Format

[a] Use 0 for last stored value. If this is not available, a factory programmed default will be used.

Hex	Description
0x01	Data Valid–Warm/Hot Starts=1
0x02	Clear Ephemeris–Warm Start=1
0x04	Clear Memory–Cold Start=1

Table 2.13: Reset Configuration

2.3.5 SetDGPSPort

This command is used to control Serial Port B which is an input-only serial port used to receive RTCM differential corrections. Differential receivers may output corrections using different communication parameters. The default communication parameters for PORT B are 9600 baud, 8 data bits, stop bit, and no parity. If a DGPS receiver is used which has different communication parameters, use this command to allow the receiver to correctly decode the data. When a valid message is received, the parameters are stored in battery-backed SRAM and then the receiver restarts using the saved parameters.

Table 2.14 contains the input values for the following example:

Set DGPS Port to be 9600,8,N,1.

```
$PSRF102,9600,8,1,0*3C
```

Name	Example	Units	Description
Message ID	$PSRF102		PSRF102 protocol header
Baud	9600		4800, 9600, 19200, 38400
DataBits	8		8,7
StopBits	1		0,1
Parity	0		0=None, 1=Odd, 2=Even
Checksum	*3C		
CR LF			End of message termination

Table 2.14: Set DGPS Port Data Format

2.3.6 Query/Rate Control

This command is used to control the output of standard NMEA messages GGA, GLL, GSA, GSV, RMC, and VTG. Using this command message, standard NMEA messages may be polled once, or setup for periodic output. Checksums may also be enabled or disabled depending on the needs of the receiving program. NMEA message settings are saved in battery-backed memory for each entry when the message is accepted.
Table 2.15 contains the input values for the following examples:
Query the GGA message with checksum enabled

```
$PSRF103,00,01,00,01*25
```

Enable VTG message for a 1 Hz constant output with checksum enabled

```
$PSRF103,05,00,01,01*20
```

Disable VTG message

```
$PSRF103,05,00,00,01*21
```

Name	Example	Units	Description
Message ID	$PSRF103		PSRF103 protocol header
Msg	00		See 2.16
Mode	01		0=SetRate, 1=Query
Rate	00	seconds	Output–off=0, max=255
CksumEnable	01		0=Disable Checksum, 1=Enable Checksum
Checksum	*25		
CR LF			End of message termination

Table 2.15: Query/Rate Control Data Format (See example 1.)

Value	Description
0	GGA
1	GLL
2	GSA
3	GSV
4	RMC
5	VTG

Table 2.16: Messages Query/Rate

Note – *In Trickle Power mode, update rate is specified by the user. When you switch to NMEA protocol, message update rate is also required. The resulting update rate is the product of the Trickle Power Update rate AND the NMEA update rate (i.e. Trickle Power update rate = 2 seconds, NMEA update rate = 5 seconds, resulting update rate is every 10 seconds, (2 X 5 = 10)).*

2.3.7 LLANavigationInitialization

This command is used to initialize the module for a warm start, by providing current position (in latitude, longitude, and altitude coordinates), clock offset, and time. This enables the receiver to search for the correct satellite signals at the correct signal parameters. Correct initialization parameters enable the receiver to acquire signals quickly.

Table 2.17 contains the input values for the following example:

Start using known position and time.

```
$PSRF104,37.3875111,-121.97232,0,95000,237759,922,12,3*3A
```

Name	Example	Units	Description
Message ID	$PSRF104		PSRF104 protocol header
Lat	37.3875111	degrees	Latitude position (Range 90 to -90)
Lon	-121.97232	degrees	Longitude position (Range 180 to -180)
Alt	0	meters	Altitude position
ClkOffset	95000	Hz	Clock Offset of the GPS receiver[a]
TimeOfWeek	237759	seconds	GPS Time Of Week
WeekNo	922		GPS Week Number
ChannelCount	12		Range 1 to 12
ResetCfg	3		See 2.18
Checksum	*3A		
CR LF			End of message termination

Table 2.17: LLA Navigation Initialization Data Format

[a] Use 0 for last stored value. If this is not available, a factory programmed default will be used.

Hex	Description
0x01	Data Valid–Warm/Hot Starts=1
0x02	Clear Ephemeris–Warm Start=1
0x04	Clear Memory–Cold Start=1

Table 2.18: Reset Configuration

2.3.8 Development Data On/Off

Use this command to enable development data information if you are having trouble getting commands accepted. Invalid commands generate debug information that enables the user to determine the source of the command rejection. Common reasons for input command rejection are invalid checksum or parameter out of specified range.
Table 2.19 contains the input values for the following examples:
Debug On

```
$PSRF105,1*3E
```

Debug Off

```
$PSRF105,0*3F
```

Name	Example	Units	Description
Message ID	$PSRF105		PSRF105 protocol header
Debug	1		0=Off, 1=On
Checksum	*3E		
CR LF			End of message termination

Table 2.19: Development Data On/Off Data Format

List of Tables