

## DECODING OF SiRF BINARY PROTOCOL

Bartłomiej Oszczak, Krzysztof Serżysko  
University of Warmia and Mazury in Olsztyn  
Chair of Satellite Geodesy and Navigation  
e-mails: bartlomiej.oszczak@kgsin.pl, krzysztof.serzysko@gmail.com

**ABSTRACT.** In this article the SiRF binary format decoding algorithm was presented and parameters contained in it and methods of their use were described. Due to the current lack of practical software which would be able to decode the SiRF format the authors' application called SiRF Decoder was created. Algorithm used in the application allowing for fast reading and decoding of recorded binary SiRF protocol data was presented. For the analysis of decoded data the authors' program was used, on which the parameters contained in SiRF messages recorded by the GNSS receiver along with the possibilities of their use in the GNSS research were evaluated by the authors.

**Keywords:** SiRF, GPS, pseudorange, protocol.

### 1. INTRODUCTION

Among the many formats used for the GNSS data transmission the binary and text solutions with different levels of complexity can be found. In the international standards of the GNSS data exchange such as RINEX, NMEA and RTCM the binary or text coding is used. (Gurtner 2007, Klaus 2001, Mihai 2004, Standard 2011). One of the binary formats is SiRF Protocol. It is the data transmission protocol developed by SiRF Technology Inc. for the use in code GNSS receivers (Manual 2008). In the protocol, in addition to a large set of basic information about position and present timing in any GNSS data transmission standard, numerous diagnostic data used in the process of GNSS receiver testing were also implemented. The occurrence of dedicated test messages makes the SiRF format stand out from the above mentioned formats, because it is possible to use the additional data from the GNSS receiver for research purposes. There are many types of messages in the SiRF protocol that contain data of various types. Just as in the RTCM standard, the data is in binary format, but this is the only similarity, because the protocols have very different purposes. The information contained in the SiRF binary protocol include, among others:

- pseudoranges, expressed in meters (without taking into account the clock, ionosphere and troposphere corrections, for example: 29337863.355 m),
- pseudorange corrections, expressed in meters (in cases where the DGPS measurement method was used),
- value of satellite clock correction, expressed in seconds (for example: -0.000154071323924 s),

- value of receiver clock correction, expressed in nanoseconds (for example: 13336281 ns),
- value of the measurement error due to the ionospheric refraction, expressed in meters (for example: 3.2853796482 m – value decoded directly without using the Bradis-Krylov rule),
- satellite positions and velocity, expressed in meters and meters per second, given in Cartesian XYZ coordinate system,
- signal to noise ratios, expressed in dB-Hz,
- other parameters defining the current state of the receiver presented as Boolean values.

Authors drew attention to the fact, that in the opposition to the SiRF binary format, in any world standard for GNSS data exchange there is no direct information about the satellites' coordinates in Cartesian XYZ coordinate system at the moment of observation. There is also no format that contains the velocities of individual satellites. Parameters decoded from the SiRF protocol listed above open many possibilities for more detailed analysis of the functioning of the GNSS systems and receivers. They also facilitate teaching the basics of GNSS position computing by simplifying the algorithms for calculating the coordinates of navigation receiver due to the lack of need for decoding the navigation message to determine the approximate coordinates of the satellites. Also, there can be rarely found the GNSS receivers that are able to record the signal to noise ratios for a tracked satellite, where in SiRF format this information is given in a standard message.

The market lacks ready-made algorithms enabling the decoding of SiRF binary data in a simple and effective way. To take the advantage of the potential of the protocol properly the individual creation of an appropriate algorithm was necessary. In order to accomplish this task the authors of the publication created a program which is able to decode the SiRF binary format. Existing public program called SiRFDemo developed by SiRF Technology Inc. decodes the data in an incomplete way, which forces the user to get the appropriate manual and perform the rest of the decoding process manually. People who used the SiRFDemo application often asked about the program which will be able to perform the complete decoding of SiRF binary data. In case of decoding of the measurement data from the larger campaigns, the original program also proves to be too slow – decoding of five percent of the full data set from a half-hour measurement takes about 4 minutes. The algorithm written by the authors is able to decode the entire data set from a half-hour measurement in about 15 seconds. In the further section of this article the structure of the SiRF binary format was described and decoding algorithm was presented that allows its easy implementation in any programming language.

## **2. SIRF BINARY PROTOCOL DATA STRUCTURE**

The SiRF Binary protocol defines the rules for sending and reading the data used in GNSS modules. There are defined about 37 types of input messages and about 40 output messages, which allow controlling and configuring the GNSS receiver. The protocol has a layered structure: each layer fulfills a specific task. We can define three main layers of the protocol:

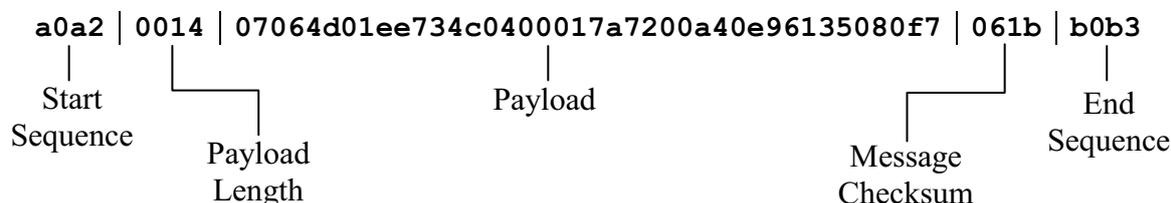
- transport layer – determines the beginning and end of the message frame by using the two sequences – start and end,
- control layer – contains information about the payload length and a message checksum used for data validation,
- payload layer – contains the actual block of data representing the information.

The layer structure of the SiRF binary protocol was carried out in the form of frames containing the appropriate message. Every frame, containing input or output message, is arranged in the same way. In the Fig. 1, the frame structure explaining individual layers of the protocol is shown. Sample values are presented in hexadecimal numbers.

Frame component	Value	Description
Start Sequence	a0a2	Permanent part of each frame specifying its start
Payload Length	two bytes	Two-byte number specifying the number of bytes in the message
Payload	up to 1023 bytes	The actual data block containing the information
Message Checksum	two bytes	16-bit sum of values of all bytes in the payload section
End Sequence	b0b3	Permanent part of each frame specifying its end

**Fig. 1.** The basic elements of SiRF Binary protocol frame

The above system is used in all messages in SiRF binary protocol. Data is given exactly in the presented order. Only the payload varies in byte length – for each message type we are dealing with a different length, which forces the addition of payload length information. Messages of the same type usually have a fixed length, but it is not the rule. The example of a hexadecimal string representing the entire SiRF binary message frame was presented in the Fig. 2:



**Fig. 2.** The sample SiRF Binary message in hexadecimal format

The information contained in the payload can be of various types. Most often we are dealing with numerical data: natural numbers, integers, single and double precision floating point numbers. There are also so-called bitmapped fields – usually individual bits are used here to determine the Boolean values. Arranging the data in payload section is performed in series: specific quantity of bytes is dedicated for a piece of information, for example 4 bytes. The first information is followed by the next one which also has assigned a specific number of bytes.

Messages in SiRF protocol can be divided into input messages and output messages. Input messages allow the user to control the device functions – it is possible to enable or disable individual output messages, to configure GNSS receiver parameters as well as to test the device by means of special modes of operation. The maximum buffer size for input messages is 912 bytes. This means that the size of one input message cannot exceed this value. Output messages allow to present the data and results of measurements and calculations to the user.

Diagnostic information which can be used in studies are contained in the various messages of SiRF protocol. Pseudorange were posted in message 28, DGPS corrections were

contained in message 27, receiver clock errors (strictly it is a value of receiver clock correction), were provided in message 7, satellite clock errors (strictly it is a value of satellite clock correction), and errors due to the ionospheric delay as well as positions and velocity of the satellites were included in message 30. Detailed description of the SiRF output messages can be found in the *SiRF Binary Protocol Reference Manual*. In the Tables 1 to 7 is a summary of some of more important parameters given by the SiRF protocol, together with the description.

**Table 1.** Message 2 parameters

Parameter	Description
<b>XYZ Coordinates</b>	XYZ User coordinates in Cartesian system
<b>XYZ Velocity</b>	User Velocity components relative to the axis X, Y and Z
<b>Bitmapped Data</b>	Bit tags indicating the number of parameters such as the position calculation mode, information about receiving DGPS corrections, the status of motion sensors, etc.
<b>HDOP</b>	Value of HDOP coefficient
<b>GPS Week</b>	Current GPS week number
<b>GPS TOW</b>	Number of seconds elapsed in the current GPS week
<b>SVs in Fix</b>	Number of satellites used in position solution
<b>SV PRN</b>	PRN numbers of satellites used in position solution

**Table 2.** Message 4 parameters

Parameter	Description
<b>GPS Week</b>	Current GPS week number
<b>GPS TOW</b>	Number of seconds elapsed in the current GPS week
<b>Channels</b>	Number of channels
<b>SV PRN</b>	PRN numbers of satellites used in position solution
<b>SV Azimuth</b>	Azimuth values of satellites used in position solution
<b>SV Elevation</b>	Elevation values of satellites used in position solution
<b>Bitmapped Data</b>	Information about the current state of measurement to satellites
<b>SNR</b>	Signal to noise ratios in 100-millisecond intervals

**Table 3.** Message 7 parameters

Parameter	Description
<b>Extended GPS Week</b>	Extended number of current GPS week
<b>GPS TOW</b>	Number of seconds elapsed in the current GPS week
<b>SVs</b>	Number of satellites used in position solution
<b>Clock Drift</b>	GNSS receiver clock drift
<b>Clock Bias</b>	GNSS receiver clock error - the difference in nanoseconds between GPS time and receiver time
<b>Estimated GPS Time</b>	Calculated GPS measurement time

**Table 4.** Message 27 parameters

Parameter	Description
<b>DGPS Source</b>	DGPS correction source: SBAS, serial port, internal beacon, etc.
<b>Correction Age</b>	Age of DGPS corrections
<b>SV PRN</b>	Satellite PRN number
<b>Correction</b>	DGPS correction

**Table 5.** Message 28 parameters

<b>Parameter</b>	<b>Description</b>
<b>Channel</b>	Number of channel used for tracking
<b>SV PRN</b>	Satellite PRN number
<b>GPS Software Time</b>	GPS time without the clock corrections
<b>Pseudorange</b>	Pseudorange without ionospheric, tropospheric and clock corrections
<b>Carrier Freq</b>	Frequency of the carrier
<b>Carrier Phase</b>	Phase of the carrier
<b>Time in Track</b>	Time of satellite tracking
<b>SNR</b>	Signal to noise ratios in 100-millisecond intervals
<b>Delta Range Interval</b>	Delta range interval
<b>Mean Delta Range Time</b>	Mean delta range time
<b>Extrapolation Time</b>	Extrapolation time
<b>Phase Error Count</b>	Count of occurrences of phase error
<b>Low Power Count</b>	Count of occurrences of low power error

**Table 6.** Message 30 parameters

<b>Parameter</b>	<b>Description</b>
<b>SV PRN</b>	Satellite PRN number
<b>GPS Time</b>	GPS time without the clock corrections
<b>SV XYZ Coordinates</b>	XYZ satellite position in the Cartesian system
<b>SV XYZ Velocity</b>	Satellite velocity components relative to the axis X, Y and Z
<b>SV Clock Bias</b>	Satellite clock bias
<b>SV Clock Drift</b>	Satellite clock drift
<b>Bitmapped Data</b>	Information about the age and source of ephemeris data
<b>Ionospheric Delay</b>	Ionospheric delay

Table 7. Message 41 parameters

Parameter	Description
<b>Bitmapped Data</b>	Information about the correctness of the position solution
<b>Extended GPS Week</b>	Extended number of current GPS week
<b>GPS TOW</b>	Number of seconds elapsed in the current GPS week
<b>UTC Time</b>	The year, month, day, hour, minute and second of the UTC time
<b>SV in Fix</b>	PRN numbers of satellites used in position solution
<b>Longitude and Latitude</b>	Longitude and Latitude
<b>Ellipsoidal Height</b>	Height measured from the ellipsoid
<b>Normal Height</b>	Height measured from the mean sea level
<b>Map Datum</b>	Ellipsoid used
<b>Speed Over Ground</b>	Speed over ground
<b>Course Over Ground</b>	Course Over Ground
<b>Climb Rate</b>	Climb Rate
<b>Estimated Horizontal Position Error</b>	Estimated Horizontal Position Error
<b>Estimated Vertical Position Error</b>	Estimated Vertical Position Error
<b>SVs in Fix</b>	Number of satellites used in position solution
<b>HDOP</b>	Value of HDOP coefficient

### 3. SIRF DECODING ALGORITHM

Using the rules described in the previous section one can manually decode every SiRF message, for example by using the hexadecimal editor. To automate the process of decoding of the format the SiRF Dekoder application was created. To write the program the Python programming language in IDLE development environment was used. The graphical interface was created using Glade editor and the GTK+ library. Application features include loading a file, validating the binary data contained therein, decoding the data to a text format and displaying the data in switchable tables and export of decoded data to CSV format (*Comma Separated Values*). Sample view of the main window of the SiRF Decoder program is shown in the Fig. 3:

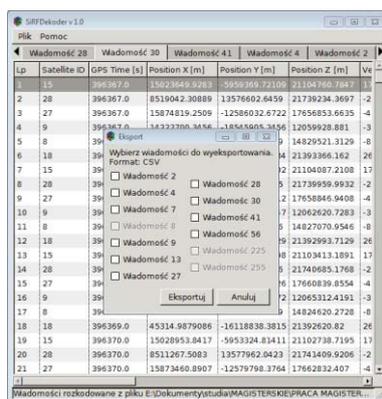


Fig. 3. The SiRF Decoder Application

In SiRF Decoder program 4 basic functions responsible for different actions were used:

- initiating function,
- checking function,
- decoding function,
- exporting function.

The whole decoding algorithm consists of two functions: checking and decoding function. This is the minimum necessary for performing error-free decoding process. The rest of the functions are an addition which provide more effortless workflow.

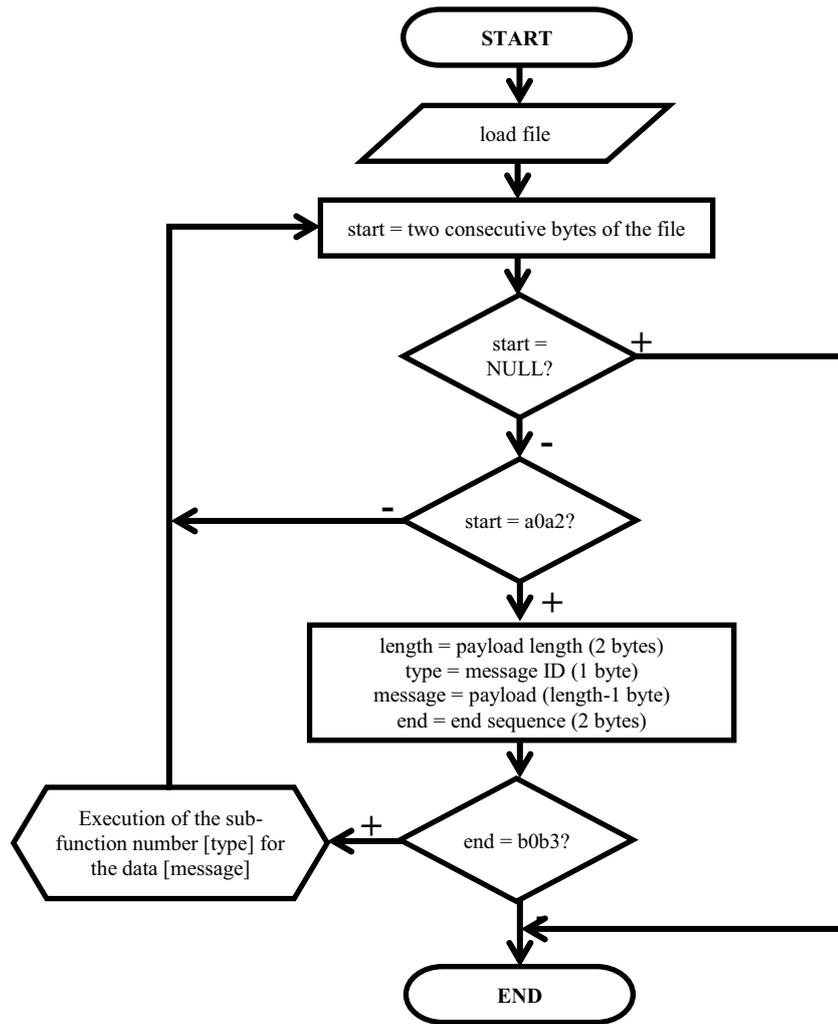
The decoding function performs basing on the principles described in section 2. It is possible to identify the beginning of the message frame by finding the start sequence of “a0a2” hexadecimal notation. Knowing the location of the start of a frame one can read the information about the type and length of the payload contained in the frame. With this information it is possible to load the payload data bytes to a variable and perform the decoding operation on it using appropriate sub-functions which act in accordance with the principles described for each message type in the *SiRF Binary Protocol Reference Manual*. In the Fig. 4 shown below the simplified block scheme of SiRF decoding algorithm was described. Start and end sequences were presented using the hexadecimal notation.

The functioning of the above algorithm is relatively simple. After loading into „start” variable the file two consecutive bytes from it are loaded. Then the “NULL” (no value) condition is checked. Meeting this condition would mean that the decoding process came to the end of file and the algorithm will be terminated. Another condition checks whether the loaded bytes represent the beginning of message frame. If not – next two bytes are read. If so – the process of reading the information about the message begins. Validation of the end sequence serves as control because there is a possibility that the last message in the file was not fully saved. Attempt of loading it could lead to an error. If the value of the end sequence agrees with the expected value „b0b3”, the sub-function appropriate for given message type is run.

It is also possible that the first message in the file is not saved correctly – it does not begin with the start sequence. The checking function, which counts the number of bytes to the first properly recorded message and then passes it to the decoding function, is detecting this situation. If in the first 200 bytes the correct SiRF message will not be found, the file is considered invalid and will not undergo the decoding process.

The structure of the algorithm used in the program allows its efficient update: when there is a need of adding a sub-function for decoding new message type one can simply write it and include it in algorithm loops. There is no need for further intervention in the structure of the code. The algorithm is also versatile - it can be used in any programming language, making it much easier to implement.

The algorithm is able to perform the correct decoding of SiRF binary format from SiRFstarIII-enabled devices, which is the most popular of all GNSS modules developed by SiRF Technology Inc. Over the years the binary format has not changed much, which leads to the conclusion, that the messages produced by the older generations of modules will also be correctly decoded, but this has not been confirmed by tests. The latest generation of chips, the SiRFstarIV, which was developed under the CSR brand, also has not been tested due to lack of available devices.



**Fig. 4.** Simplified block scheme of SiRF decoding algorithm

#### 4. ANALYSIS OF SIRF BINARY DATA

Availability of information such measured by the GNSS receiver as raw (unadjusted) values of pseudorange, values of clock errors and positions of satellites at the moment of observation is undoubted advantage of using the SiRF binary protocol.

In order to verify the given information the test measurement on point of known coordinates was carried out. Using the known coordinates of the measured point and the calculated satellite coordinates adjusted for speed, the geometrical distances between the satellite and the receiver were calculated using the formula:

$$r = \sqrt{(X^s - X_r)^2 + (Y^s - Y_r)^2 + (Z^s - Z_r)^2}$$

where  $r$  – the geometric distance between the satellite and the receiver,  $X^s, Y^s, Z^s$  – position of the satellite in the ECEF coordinate system,  $X_r, Y_r, Z_r$  – position of the receiver in the ECEF coordinate system. Then, using the remaining parameters given in the SiRF binary protocol, the corrected values of pseudoranges were calculated using the formula: (Misra, Enge 2006):

$$\rho = r + c(\delta_u - \delta^s) + I_\rho + T_\rho + \varepsilon_\rho$$

where  $\rho$  – measured unadjusted pseudorange,  $r$  – the geometrical distance between the satellite and the receiver,  $c$  – speed of light,  $\delta_u$  – receiver clock error,  $\delta^s$  – satellite clock error,  $I_\rho$  – ionospheric delay,  $T_\rho$  – tropospheric delay,  $\varepsilon_\rho$  – other non-modeled effects (measurement noises). All of the above parameters are contained in the SiRF binary format, except for the tropospheric delay, which is not given directly.

In the Table 8 is given from the obtained measurements information needed to calculate the corrected pseudorange. Measurement to the PRN 26 satellite was carried out 22. of June 2011 using the Glofish X500 palmtop with SiRFstarIII chip built-in.

**Table 8.** Sample data provided in SiRF protocol with measured pseudorange values and parameters needed for pseudorange correction

Time [hh:mm:ss]	$\rho$ [m]	$\delta_u$ [ns]	$\delta^s$ [s]	$I_\rho$ [m]
10:40:50	35945825,5954	53495781	-0,0000681318	3,334993
10:40:51	35964290,5914	53557200	-0,0000681319	3,334993
10:40:52	35982755,7197	53618625	-0,0000681319	3,334993
10:40:53	36001220,9618	53680052	-0,0000681319	3,334993
10:40:54	36019686,3901	53741479	-0,0000681319	3,334993
10:40:55	36038151,8751	53802906	-0,0000681319	3,334993
10:40:56	36056617,4658	53864333	-0,0000681319	3,334993
10:40:57	36075083,1893	53925760	-0,0000681320	3,334993
10:40:58	36093549,0821	53987188	-0,0000681320	3,334993
10:40:59	36112015,0791	54048615	-0,0000681320	3,334993

In the above summary the error due to tropospheric delay is missing – it is not given directly in the SiRF protocol. However, there is a possibility of modelling the value of error using the Hopfield or Saastamoinen formulas with the information about satellite elevation obtained from message 4.

Due to the considerable value of the receiver clock correction, the measured pseudorange error reaches very high value. With the presence of the corrective parameters it is possible to adjust the pseudorange value. The calculated adjusted pseudoranges were compared with the computed geometrical distances. The following table presents the summary with the calculated differences.

Occurred differences between geometrical distances and pseudoranges can be explained by the influence of troposphere, which was not included in pseudorange correction and by other impossible to model effects, such as multipath. The satellite coordinates' errors which are obtained from the ephemeris and can be up to 5 – 10 meters in value, also should have been taken into account (Hofmann-Wellenhof 1997). Given the fact that not all errors were included in the calculations, the obtained result can be considered as satisfactory and confirming the thesis that the diagnostic parameters present in the SiRF binary protocol are relatively reliable and capable of being used in specialized scientific research.

**Table 9.** Comparison of adjusted pseudoranges and computed geometrical distances

<b>Time [hh:mm:ss]</b>	<b><math>\rho_{corr}</math> [m]</b>	<b><math>r</math> [m]</b>	<b><math>r - \rho_{corr}</math> [m]</b>
<b>10:40:50</b>	19887765,1692	19887759,9677	<b>-5,2015</b>
<b>10:40:51</b>	19887817,2073	19887809,4656	<b>-7,7417</b>
<b>10:40:52</b>	19887867,5790	19887859,0912	<b>-8,4878</b>
<b>10:40:53</b>	19887917,4650	19887908,8444	<b>-8,6206</b>
<b>10:40:54</b>	19887967,5371	19887958,7252	<b>-8,8119</b>
<b>10:40:55</b>	19888017,6659	19888008,7337	<b>-8,9322</b>
<b>10:40:56</b>	19888067,9004	19888058,8698	<b>-9,0306</b>
<b>10:40:57</b>	19888118,2677	19888109,1337	<b>-9,1340</b>
<b>10:40:58</b>	19888168,5046	19888159,5251	<b>-8,9795</b>
<b>10:40:59</b>	19888219,1454	19888210,0442	<b>-9,1012</b>

## 5. CONCLUSIONS

The possibilities of using the decoded SiRF parameters are significant. It is possible for example to program the position calculation algorithm, gross error elimination system etc. Simple decoding application created by authors and the measurement results show, that the SiRF binary protocol may be useful and at the same time easy-to-use tool for conducting research, offering a rich set of data for analysis.

Data in the SiRF protocol, however, is burdened with some errors. During measurement a temporary loss of satellite signal was observed. In many cases, for a moment before a loss of signal, the pseudorange measurement error increased to 600 – 900 meters. There was also noticed the phenomenon of automatic receiver clock error correction after exceeding a specified value (0,15 s). Moments in which the clock error values changed were not the same as the moments of changes in the measured pseudorange. Pseudorange value was correctly determined a second after the moment of the GNSS receiver clock tuning, which caused one second (single for the moment) gross errors. Another phenomenon is the time shift - in message 41 the GPS and UTC time are present simultaneously. Analogous situation of correction the UTC time relative to the GPS time was noted.

Despite the found problems the SiRF protocol has great potential. Besides the obvious advantages of the presence of given measured pseudorange and correction parameters' values, an interesting function is the receiver operating parameters such as signal to noise ratios and HDOP values passed to the user. There is also a set of values defining, for example, the current ephemeris data age, satellite tracking status, error counts associated with time or a weak signal etc. This gives great opportunities in creating own applications. Caution in conducting the analysis of data provided by SiRF devices and including substantial corrections guarantees the reliability and appropriate accuracy of the obtained results.

**REFERENCES**

- Gurtner W. (2007) *RINEX. The Receiver Independent Exchange Format. Version 3.00*.
- Hofmann-Wellenhof B. (1997) *Global Positioning System. Theory and Practice*. Springer-Verlag Wien, NewYork.
- Klaus B. (2001) *The NMEA 0183 Protocol*.
- Oszczak B., Serżysko K., Tanajewski D. (2011) – „*Analiza danych protokołu SiRF*”. Logistyka 6/2011
- Oszczak B., Serżysko K., Tanajewski D. (2011) - „*Aplikacja SiRFLogger*”. Logistyka 6/2011
- Mihai (2004) *NMEA-0183 Protocol Description*.
- Misra P. and Enge P. (2006) *Global Positioning System. Signals, Measurements, and Performance*. Ganga-Jamuna Press, Massachusetts.
- Norton P. and others (2006) *Python. Od podstaw*. Helion, Gliwice.
- Specht C. (2007) *System GPS*. Bernardinum, Pelplin.
- Standard (2011) *Differential GNSS Services - Version 3*.

*Received: 2011-09-07,*

*Reviewed: 2012-02-08, by M. Grzegorzewski, and 2012-02-21,*

*Accepted: 2012-02-23.*