Time Server Based on NMEA and SNTP Protocols

Z. Łukasik, W. Nowakowski & T. Ciszewski
Kazimierz Pulaski University of Technology and Humanities in Radom, Radom, Poland

ABSTRACT: Modern technical ship systems very often use information and communication technologies. One of the basic requirements of these systems is ensuring time synchronization that aim is to coordinate clocks of all devices working in a computer network. This requirement results from, among others, the necessity to log events with a unified timestamp. To achieve this goal, most frequently expensive time servers are used. This article presents an easy method of time synchronization, which can serve as a reserve or alternative solution to the methods currently used. That is why a proprietary software has been developed, allowing to expand the functionality of a PC so that it can operate as a time server. A time reference in proposed solution would be provided by a GPS receiver. The developed application allows, with the use of the NMEA protocol, to synchronize the current time with the GPS receiver and then, with the help of the SNTP protocol, to distribute it to all network devices operating on the ship.

1 INTRODUCTION

A necessity to keep time precisely has been accompanying the humanity since ages, but precise measurement of time became available to people since 1946, when the atomic clock was invented. This invention made it possible to define the international standard time called International Atomic Time (TAI fr. Temps Atomique International). The atomic clocks allowed to introduce a common for the whole world time reference - UTC (Universal Time Coordinated). Precise and unified time keeping has become especially important since the development of computer systems and networks (Łukasik & Nowakowski 2017).

Technical systems of a ship i.e.: route planning and monitoring system, anti-collision and detecting system, radio communication system, voyage data recording system, automation and alarm system, steering control system, etc. commonly use information and communications technology (ICT). Time synchronization in all these subsystems is very important, which results mostly from the need to provide a unified time, among others, when exchanging data or during their archiving. One of the most popular time references are GNSS (Global Navigation Satellite System), which are commonly used to determine geographical position of an object (Bin et al. 2015, Kolar et al. 2012, Moussa et al. 2010). Because precise object location requires accurate time measurement, precise atomic clocks based on caesium and rubidium standards are located on the satellites (Jiang & Arias 2013). Data exchange with GNSS receivers is usually carried out using the NMEA protocol (National Marine Electronics Association) and allows to obtain, among others, such data as location, speed and time (Shoab et al. 2013, Ciszewski & Chrzan 2011). Time synchronization in the network is performed using the NTP (Network Time Protocol)

or SNTP (Simple Network Time Protocol) (Jie et al. 2017, Rybaczyk 2005).

2 NMEA PROTOCOL

The National Marine Electronics Association (NMEA) has developed a specification defining the interface between marine electronic equipment. Communication with GNSS receivers is defined within this specification. Two versions of the NMEA protocol are currently in use, i.e.: NMEA 0183 and NMEA 2000 (IEC 61162-3). NMEA 2000 can be considered as the successor to NMEA 0183 although these are not compatible protocols. Among others the differences concern such elements as:

- NMEA 2000 uses binary messages, unlike the ASCII serial communication used in NMEA 0183,
- NMEA 2000 allows multi-talker and multi-listener configuration, whereas NMEA 0183 allows only single-talker and multi-listener configuration,
- NMEA 2000 uses faster data transfer (250 kbps), whereas NMEA 0183 allows only 38400 bps.

Regardless the above limitations, the NMEA 0183 protocol is still very popular (Coleman 2008). Its current version is 4.11 and is dated on November 2018. It takes into consideration a dynamic development of the GNSS systems, including: GPS, GLONASS, GALILEO, BDS, QZSS and NAVIC (Petrov et al. 2016). The proposed by the authors time synchronization method is based on the NMEA 0183 protocol which, in the article, will be called NMEA.

The NMEA protocol uses a plain text encoded in ASCII. All NMEA sentences (messages) start with the character "$" or "!", then there is a GP prefix (indicating a GPS receiver) along with the message ID, then data fields ended with a pair of control characters <CR><LF> occur. Optionally, characters <CR><LF> could be prefixed by a checksum, which allows to check for communication errors or invalidate the received data. Sentence data fields are separated by commas and the checksum is prefixed by an asterisk ("*"). The number of characters in a single NMEA message, including beginning and end delimiters, should not exceed 82. In the NMEA standard there is a rule that if a GPS receiver data are unavailable the corresponding field remains blank, however the delimiting commas are left to specify the field boundaries. NMEA supports many possible message IDs, where only a few of them are used for time synchronization (Novick et al. 2018). A time server application recognizes and interprets the following IDs:

GGA - Global Positioning System Fixed Data

Table 1 contains the values for the following example:

*$GPGGA,135415.000,5115.6372,N,02107.9350,E,1,8,0.99 ,203.6,M,40.6,M,,*5D*

Table 1. GGA Data Format

Name	Example	Unit	Description
Message ID	$GPGGA		GGA protocol header
UTC Time	135415.000		hhmmss.sss
hhmmss.sss	5115.6372		ddmm.mmmm
N/S Indicator	N		N=north or S=south
Longitude	02107.9350		dddmm.mmmm
E/W Indicator	E		E=east or W=west
Position Fix Indicator	1		0 - Fix not available or invalid
			1 - GPS SPS Mode, fix valid
			2 - Differential GPS, SPS Mode, fix valid
			3-5 - Not supported
			6 - Dead Reckoning Mode, fix valid
Satellites Used	8		Range 0 to 12
HDOP	0.99		Horizontal Dilution of Precision
MSL Altitude	203.6	meters	
Units	M	meters	
Geoid Separation	40.6	meters	Geoid-to-ellipsoid Separation. Ellipsoid altitude = MSL Altitude + Geoid Separation.
Units	M	meters	
Age of Diff. Corr.		sec	Null fields when DGPS is not used
Diff. Ref. Station ID			
Checksum	*5D		
<CR> <LF>			End of message termination

An example of an Object Pascal function for decoding a GGA message is shown below:

```
function DecodeGGA(GPS: PNmeaGPS; A: TStringList):
Boolean;
begin
  Result := False;
  GPS^.fError := False;
  try
    GPS^.fValid := (StrToInt(A[6]) <> 0);
    if not GPS^.fValid then
      exit;
    GPS^.fSatCount := StrToInt(A[7]);
    if GPS^.fSatCount > GPS^.fMaxSatCount then
      GPS^.fMaxSatCount := GPS^.fSatCount;
  except
    GPS^.fError := True;
    Result := False;
  end;
end;
```

RMC - Recommended Minimum Specific GNSS Data

Table 2 contains the values for the following example:

*$GPRMC,135415.000,A,5115.6372,N,02107.9350,E,0.65, 192.12,231218,,,A*63*

Table 2. RMC Data Format

Name	Example	Unit	Description
Message ID header	$GPRMC		RMC protocol
UTC Time	135415.000		hhmmss.sss
Status	A		A=data valid or V=data not valid
Latitude	5115.6372		ddmm.mmmm
N/S Indicator	N		N=north or S=south
Longitude	02107.9350		dddmm.mmmm
E/W Indicator	E		E=east or W=west
Speed Over Ground	0.65	knots	
Course Over Ground	192.12	degrees	True
Date	231218		ddmmyy
Magnetic Variation		degrees	E=east or W=west
East/West Indicator			E=east
Mode	A		A=Autonomous, D=DGPS, E=DR
Checksum	*63		
<CR> <LF>			End of message termination

An example of an Object Pascal function for decoding a RMC sentence is shown below:

```
function DecodeRMC(GPS: PNmeaGPS; A: TStringList):
Boolean;
  begin
    Result := True;
    try
      GPS^.fValid := (A[2] = 'A');
      if not GPS^.fValid then
        exit;
      GPS^.fUTCTime := DateTimeNMEA(A[9], A[1]);
      if Length(A[3]) = 11 then
      begin
        GPS^.fLatitude := Copy(A[3],1,2) + '° '
        + Copy(A[3],3,7) + ' ' + A[4];
      end
      else
        GPS^.fLatitude := A[3] + ' ' + A[4];
      if Length(A[5]) = 12 then
      begin
        GPS^.fLongitude := Copy(A[5],1,3) + '° '
        + Copy(A[5],4,7) + ' ' + A[6];
      end
      else
        GPS^.fLongitude := A[5] + ' ' + A[6];
    except
      GPS^.fError := True;
      Result := False;
    end;
  end;
   Result := False;
  end;
end;
```

3 NTP AND SNTP PROTOCOLS

When defining the NTP protocol it was assumed that it would be designed for precise time synchronization for a big amount of electronic devices working in a network. Additionally, it was assumed that clock frequency will be adjusted to reduce the time offset gradually, without time shifts or the necessity to change the clock, and the whole procedure will not overload the network and will be fully automated. The NTP protocol also offers mechanisms protecting from an external interference of computer clock's time change. All these have contributed to a big popularity of the NTP protocol among time synchronization techniques. The protocol is not only used in typical computer and network systems, but also in many research areas and communication devices on boards of ships, planes and even spaceships.

The concept of time synchronization using NTP protocol is based on the tree structure that includes a certain number of devices. Some of them are reference time sources, others are responsible for time distribution, while remaining are NTP clients (which synchronizing their own clocks). Particular levels of the NTP hierarchy are called Stratum. They indicate on a distance between a given device and the reference clock. In Stratum 0 there are time references, i.e.: caesium atomic clocks, clocks with built-in rubidium generators or a GNSS systems. These reference clock sources are very accurate and reliable. Devices included in the next layer, called Stratum 1, take time directly from reference clocks included in Stratum 0 and are used for time synchronization of elements in the level below. Similarly, elements in the Stratum 2 play a role of reference clock for computers and local area net-works in next layer, Stratum 3. Therefore Stratum N components constitute the reference clock for Stratum N+1 components. The number of layers is limited to 16 (Stratum 0-15), and the number of devices in each layer is unlimited (Rybaczyk 2005).

Apart from the hierarchical structure, the NTP protocol has other crucial attributes. Timestamps are sent and received using the User Datagram Protocol (UDP) on port number 123 in packets of size 72 bytes. The protocol is capable to adjust clock's time for each client, on the basis of calculating time differences in relation to the reference clock. NTP is designed to produce 3 results: clock offset, roundtrip delay and dispersion. Clock offset represents a difference of time value indicated by a local clock in relation to the reference clock. Roundtrip delay provides the capability to launch a message to arrive at the reference clock at a specified time. Dispersion, on the other hand, represents a maximum error of the local clock relative to the reference clock (Novick et al. 2018). The NTP protocol can operate in a few modes, among others: *symmetric active, symmetric passive, client, server, broadcast*. A detailed description of the NTP protocol is included, among others, in book (Mills 2016) and specification (Mills et al. 2010).

A simplified version of the NTP protocol is the SNTP (Simple Network Time Protocol) protocol. A detailed specification of the SNTP protocol is described in the widely available RFC 4330 document (Mills 2006). The principle of the SNTP protocol operation is based on procedures found in the NTP protocol (Jv & Gao 2008). SNTP operate in client-server model and uses the connectionless user datagram protocol (UDP) to distribute reference clock obtained from time servers to the SNTP clients. The final product of the SNTP protocol, similarly as in the case of NTP, are three values: clock offset, roundtrip delay and dispersion. The architecture of SNTP is the same as in NTP, however, due to a simplified operation mode, there are differences between these two protocols. An example is an SNTP packet header, which does not have additional extension fields used for message authorization. Devices that distribute the

reference time using the SNTP protocol should be located in the Stratum 1 of the subnet. Servers using SNTP technology can support a large number of clients, but each client can only connect to one server. It is recommended to place the clients in the higher strata so that their location is at the end position in the hierarchy of a particular subnet If one of the devices is synchronized with the use of the SNTP protocol all the others, operating in the same subnet, should also be synchronized with this protocol.

The SNTP client can operate in three modes: *unicast*, *broadcast* or *manycast*. In the *unicast* mode it sends requests to a specific server for synchronization. In the *broadcast* configuration, the client acquires its time synchronization from data broadcast by any SNTP server to the network broadcast address. In the last, *manycast*, mode, the client sends requests to a designated broadcast address and waits for a response from any server with an active *manycast* mode. Having received the first response, the client connects only to the server that has sent the message. Messages from other servers are then ignored, which means that next operations are handled in the *unicast* mode. The client requests in the *unicast* and *manycast* mode are sent in time gaps depending on the client's clock frequency configuration and required precision.

4   TIME SERVER

Time server is an application that allows time synchronization of devices working in a computer network, in accordance to the reference clock retrieved from the GNSS receiver. In the main window the application shows the current, synchronized time on two panels: one displays the current time in the *hh-mm-ss* format, the other shows the current date in the *yyyy-mm-dd* format (Fig. 1).
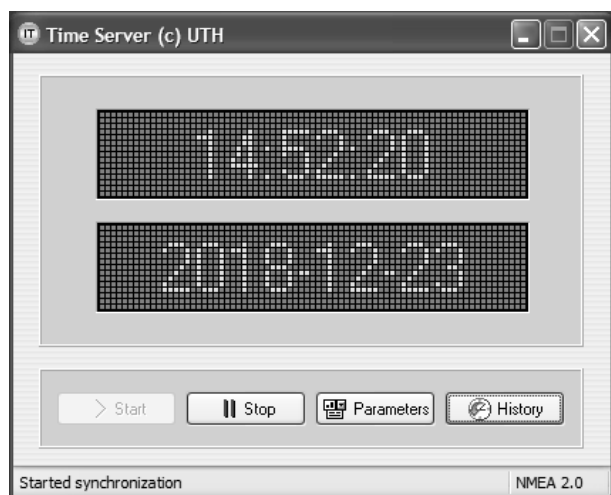

Figure 1. "Time Server" application's main window.

During the first start of the application there is a need of initial configuration. Among the others the following parameters should be configured (Fig. 2):
– maximum time difference between GPS and PC clock (0-1000ms),
– correction of GPS time (-2000ms / +2000ms),

– serial port settings, which allow proper communication between the GPS receiver and the computer (port, baud-rate, flow-control).

These parameters are set in the "General" tab in "Time Server" application. Additionally, if we require the time server functionality, the "Start the SNTP server automatically" option located in the "Server" tab (Fig. 3) should be enabled.

If the necessary configuration has been made, it is possible to select the "Start" button in the main window. (Fig. 1), which activates the reading of NMEA messages from the GPS receiver (Fig. 4).
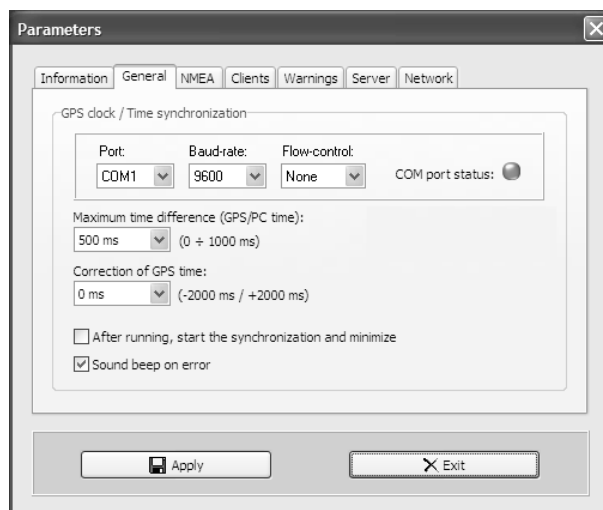

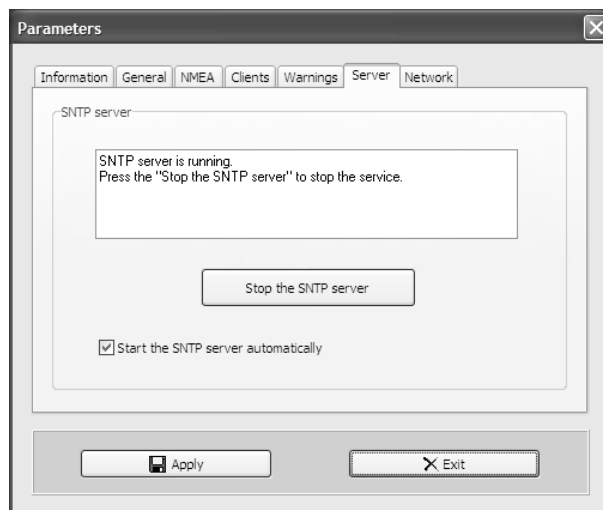Figure 2. An example setting of main parameters in "Time Server".

Figure 3. An example configuration of the SNTP server.

The "Time Server" application recognizes and interprets messages tagged with GGA and RMC identifiers, including:

– number of visible satellites,
– UTC time,
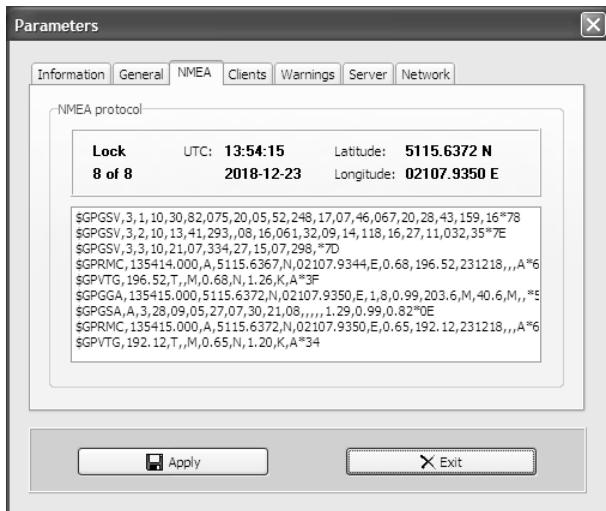– geographic coordinates (latitude and longitude).

Figure 4. Reading of NMEA messages from the GPS receiver.

Additionally, the application creates logs concerning particular actions, such as: starting and stopping synchronization and error occurrences. All messages are presented in a separate window that can be activated with the "History" button in the main window of the application. "Time Server" also has advanced functionality in the field of self-diagnosis. For example, it can check the activity of time clients which, according to the unicast mode, should send requests to servers at specified time intervals (Fig. 5).
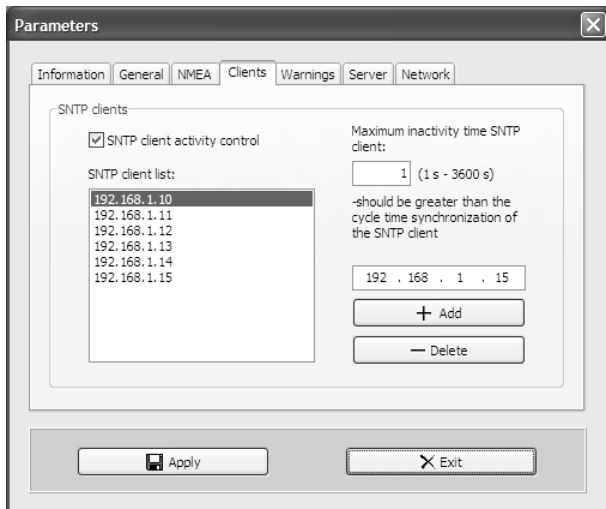

Figure 5. SNTP clients activity control.

"Time Server" also checks communication with the GPS receiver and controls correctness of the antenna's settings. All errors in the server operation are indicated to the operator in the form of displayed messages and SNMP traps sent to the network manager (Fig. 6). This functionality is available after correct configuration of OID MIB identifiers and after setting the IP address of the SNMP manager.

Figure 6. SNMP traps configuration.

5 CONCLUSIONS

Time synchronization in all electronic subsystems of the ship is very important, which results, among others, from the need to register events with a unified time stamp. The paper presents a proposal of a time synchronization method in accordance to the reference clock, which can be a time server equipped with a GPS receiver. Such a solution can be used as an alternative for methods based on expensive hardware servers. The proposed in this paper proprietary software "Time Server" allows retrieving time from a GPS receiver and then synchronize it to other network devices of the ship that are SNTP clients. "Time Server" application, in order to ensure high reliability, has advanced functions of self-diagnosis. The monitoring involves the status of the GPS receiver and correctness of NMEA communication, as well as the activity of selected SNTP clients. In addition, it is the possibility to cooperate with the SNMP network manager in monitoring these parameters.

REFERENCES

Bin, L., Guo, S. & Ma, S. 2015. Design and implementation of time synchronization system based on GPS signal, Advances in Intelligent Systems Research, Volume 124, pp. 2186-2191

Ciszewski, T. & Chrzan, M. 2011. Wykorzystanie technologii satelitarnej do precyzyjnego wyznaczania położenia wad w szynach kolei dużych prędkości, Logistyka nr 6/2011, str. 457-464

Coleman, R. 2008. A self-healing plug-in parser for NMEA streams, Proceedings of the 5th International Conference on Information Technology: New Generations, pp. 1023-1027

Jiang, Z. & Arias, E.F. 2013. Use of the Global Navigation Satellite Systems for the Construction of the International Time Reference UTC, Lecture Notes in Electrical Engineering, Volume 245, pp. 457-466

Jie, X., Liang, X., Lian, D., et al. 2017. Research on network timing system based on NTP, Proceedings of 2017 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI), Volume 1, pp. 356-360

Jv, H. & Gao, X. 2008. Application of Time Synchronization System Based on SNTP in Digital Substation, China International Conference on Electricity Distribution, Volume 1 and 2, pp. 531-534

Kolar, V., Thao, P. T. & Hrbac, R. 2012. Precise Time Synchronization with GPS, International Scientific Conference on Electric Power Engineering, pp. 329-331

Łukasik, Z. & Nowakowski, W. 2017. Synchronizacja czasu w systemach sterowania ruchem kolejowym, TTS Technika Transportu Szynowego, nr 12/2017, str. 254-257

Mills, D. L. 2016. Computer Network Time Synchronization: The Network Time Protocol on Earth and in Space, CRC Press

Mills, D. L. 2006. Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI, RFC 4330

Mills, D. L., Delaware D. & Martin J. 2010. Network Time Protocol Version 4: Protocol and Algorithms Specification, RFC 5905

Moussa, A., Ali, A.S. & El-Sheimy, N. 2010. The Effect of Time Synchronization on Real Time Implementation of Integrated GPS/INS Systems, Proceedings of the 2010 International Technical Meeting of the Institute of Navigation - ITM 2010, pp. 43-48

Novick, A.N., Lombardi, M.A. & Franzen, K. 2018. Improving packet synchronization in an NTP server, Proceedings of the 49th Annual Precise Time and Time Interval Systems and Applications Meeting, pp. 256-260

Petrov, D., Melnik, S. & Hamalainen, T. 2016. Distributed GNSS-Based Time Synchronization and Applications, International Conference on Ultra Modern Telecommunications and Control Systems and Workshops, pp. 130-134

Rybaczyk, P. 2005. Expert Network Time Protocol: An Experience in Time with NTP, Apress

Shoab, M., Jain, K., Anulhaq, M., et al. 2013. Development and Implementation of NMEA Interpreter for Real Time GPS Data Logging, Proceedings of the 2013 3rd IEEE International Advance Computing Conference (IACC), pp. 143-14