Research on GPS Timing Remote Synchronization Algorithm in High Altitude Meteorological Data Acquisition System

Min Qiu¹, Yang qiu², Yurong Yang¹, Yu Bai¹

¹Department of Physics and Chemistry, Heihe University, Heihe, 164300, China
²Communications research Center, Harbin Institute of Technology, Harbin 150001, China
qiuy@hit.edu.cn

Abstract. The various high altitude meteorological parameter influence each other, and test data of different locations need to be synchronized. In this paper, we propose a GPS timing remote synchronization algorithm for high altitude meteorological data acquisition system, which can make the meteorological data at different locations meet the requirements of synchronization. GPS module uses NEO-M8N chip of UBLOX to achieve GPS timing remote synchronization, which include hardware and software designs for the system. We use the designed system to measure high altitude meteorological data in two different locations, analyze experimental results and show the relationship between meteorological parameters. In the processing for experimental results, we employ Fletcher-Reeves algorithm based on artificial neural network to predict meteorological data, and analyze training errors of the algorithm. The proposed system can realize reception and processing of remote data synchronously, and predict the meteorological data for weather forecast.

1. Introduction

As a member of the World Meteorological Organization, our country has more than 4,000 stations of various types for weather forecast in various departments, whose meteorological department accounts for 65%. High altitude meteorological detections are explored every day [1][2]. Because of the vast land area and the need for meteorological experiments in different places, collecting meteorological data from different locations and sending them back to receivers is a very important issue. The uniform timing of meteorological data helps researchers analyze reasonably and obtain more accurate weather forecasts [3][4][5]. High altitude meteorological data acquisition at different positions to achieve synchronization is often encountered in meteorological works. Due to the acquisition equipment at different places, the performance of data acquisition systems largely depend on the synchronization performance. When the two places are close, they can use the network to achieve synchronization. However, when the distance between the two places is far, the network may be affected by surrounding environment and loss of transmission path, which makes the system complicated, higher cost, poor reliability, and difficult to guarantee synchronization [6][7][8][9]. In order to solve the above problems, we present a research on GPS timing remote synchronization algorithm in high altitude meteorological data acquisition system in this paper. The algorithm is mainly divided into three basic modules: GPS timing synchronization module, data acquisition module and FPGA control module. The basic principle of this algorithm is using GPS second pulse to trigger the acquisition devices in both places synchronously [6]. The rising edge of GPS second pulse exactly corresponds to a certain
UTC time (the error is several tens of ns). Using GPS timing synchronization can ensure high precision, all-weather and low-cost. Therefore, we focus on the performance of GPS timing synchronization module in meteorological data acquisition system.

Artificial neural network (ANN) is a kind of mathematical model algorithm, which uses distributed parallel information processing mode. Its characteristic is to imitate the behavior of animal neural network, whose basis is physiological research on brain [10][11]. We use Fletcher-Reeves algorithm in BP neural network to predict the meteorological data based on the previous studies [12].

The remaining of this article is as follows, Section II gives the system structure design, including the hardware and software designs. We focus on GPS timing remote synchronization algorithm. Section III is systematic experimental results and analysis. We present actual measurements values of meteorological data in the algorithm, and analyze the performance of the algorithm. In Section IV, we use Fletcher-Reeves algorithm to forecast meteorological data of the system, give the simulation results and analyze the performance, and verify the feasibility of the algorithm in meteorological data measurement and prediction. Finally, we conclude the work in our paper.

2. System Model
The high altitude meteorological radiosonde system in this paper mainly measures altitude, longitude, latitude, humidity, temperature, and atmospheric pressure. Among them, the humidity, temperature and atmospheric pressure are measured by the designed circuit, and the location (including altitude, longitude and latitude) of radiosondes are measured by the GPS module. The proposed GPS timing remote synchronization algorithm can achieve synchronization acquisition of high altitude meteorological data. The overall system structure is shown in Fig. 1. Then, we present the system model, including system hardware and software designs.

Figure 1 High altitude meteorological data acquisition system based on GPS timing remote synchronization algorithm

2.1 Hardware Design of System
The system hardware design includes brief introduction of GPS module, output characteristics of GPS second pulse, design of GPS command parsing and timing module.

2.1.1 Introduction of GPS Module
We choose NEO-M8N of UBLOX as GPS module, which is capable of receiving all global navigation systems (GPS, GLonass, Beidou, Galilei). The chip can provide high positioning accuracy, and support low power consumption. It can support RS232 serial communication and be compatible to TTL level. Its default communication rate is 9600 bps (adjustable), and the pin definitions as shown in
Table 1. The chip can produce high precision second pulse signal, whose pulse period is 1 s and the default value of pulse width is 100 ms (can set). The module uses the high precision second pulse and UTC time to synchronize calibration, and can also parse GPS information, report current location and so on.

Table 1 Function instructions of GPS module chip pin

| No | Name   | Function Description                                      |
|----|--------|------------------------------------------------------------|
| 1  | NC     | Reserved                                                   |
| 2  | D_SEL  | Interface select                                           |
| 3  | TIMEPULSE | Output of pulses per second (PPS)                         |
| 4  | EXTINT | External interrupt pin                                     |
| 5  | USB_DM | Positive end of USB differentia data                       |
| 6  | USB_DP | Negative end of USB differentia data                       |
| 7  | VDD_USB | USB supply                                                  |
| 8  | RESET_N | Reset chip pins                                            |
| 9  | VCC_RF | Voltage output of RF section                               |
| 10 | GND    | Ground                                                     |
| 11 | RF_IN  | Input signal of GNSS                                       |
| 12 | GND    | Ground                                                     |
| 13 | GND    | Ground                                                     |
| 14 | ANT_ON | Control of external low noise amplifier                     |
| 15 | NC     | Reserved                                                   |
| 16 | NC     | Reserved                                                   |
| 17 | NC     | Reserved                                                   |
| 18 | SDA/CS | DDC Data if D_SEL = 1 (or open); SPI Chip Select if D_SEL = 0 |
| 19 | SCL/SCLK | DDC Clock if D_SEL = 1 (or open); SPI Clock if D_SEL = 0     |
| 20 | TXD/SI | Serial Port if D_SEL = 1 (or open); SPI MISO if D_SEL = 0   |
| 21 | RXD/SO | Serial Port if D_SEL = 1 (or open); SPI MOSI if D_SEL = 0   |
| 22 | V_BCKP | Backup voltage supply                                      |
| 23 | VCC    | Supply voltage                                             |
| 24 | GND    | Ground                                                     |

2.1.2 Output Characteristics of GPS Second Pulse

The pulse per second (PPS) is a level signal, which usually output square wave. The high level indicates the PPS output (the default pulse width is 100 ms), and the low level means there is no signal output. The rising edge of high level is the exact moment of PPS output. The graphics is shown in Fig. 2.

![Figure 2 Output characteristic of pulses per second (PPS)](image)

When receivers obtain effective navigation solutions, the PPS rising edge differs from UTC moment within 50 ns. The output from UTC time has a delay (but generally less than 0.15 s) compared with the rising edge of PPS in RS-232 data transmission. That is the receivers provide users with PPS first, and then provide effective navigation solutions, which should be noted. The relationship between PPS and UTC is shown in Fig. 3.
2.1.3 GPS Command Parsing

NEO-M8N supports three protocols: NMEA protocol, UBX protocol, and RTCM protocol. The module uses NMEA protocol to parse, the specific format is shown as Fig. 4.

The module uses GNRMC information to extract time and location of UTC. The specific information is: $GNRMC, hhmmss. ss, A/V , ddmm. mmmmm, N/S, dddmm. mmmmm, E/W, spd, cog, ddmmyy, mv, mvEW, posMode, navStatus, cs, <CR><LF>. When the GNRMC information is received, useful information is extracted according to the format: GPS lock, time and location information of UTC.

2.1.4 Design for Timing Module

The overall module requires meteorological data acquisition once in 30 ms, which stored to the SD card. A day (24 h) is 2880000×30 ms, so we use the number of 30 ms to do timestamp synchronization for the collecting information. Thus, we packet data as seal frame processing, and the figure shown in Fig. 5.

In order to ensure the accuracy of synchronization, the module will calibrate the count every 3 s. If there is a deviation, the module will fine-tune the count to ensure the accuracy of data synchronization. The flow chart for specific implementation is shown as Fig. 6. First of all, GPS command is parsed and UTC time is extracted according to GPS information. The time format is: hh, mm, ss. Judging the module whether over 3 s, when it over 3 s, we calculate how many 30 ms contained in this period according UTC time. The system calibrates time once every 3 s when collects the valid PPS signal. If the system counts one hundred 30 ms during this period, we need to compensate a 30 ms to achieve synchronization with UTC time, and the count is output after the calibration is completed.
2.2 Software Design for System

In this Section, we mainly design driving module for UART receiving data, GPS protocol parsing module, and timestamp calibration module.

2.2.1 Design of Driving Module for UART Receiving Data

This module mainly process UART data returned by GPS module. In order to improve communication quality and reduce bit error rate, a tenfold baud rate is used to sample each bit in this module. After sampling, we compare data of the fourth, the fifth and the sixth, and take the most bit as output data. By doing so, we can achieve a simple error correction and reduce transmission error greatly.

2.2.2 Design of GPS Protocol Parsing Module

As mentioned earlier, the module uses GNRMC information to extract time and location of UTC, so the program must detect "$ GNRMC" first. After detecting the correct head, the subsequent parsing is followed. The end character of each message is ",". The "data is valid", "UTC time" and "location information" are is reported to the top-level module after parsing to do a sealed frame processing.

2.2.3 Design of Timestamp Calibration Module

The system achieves synchronization and calibration processing according to the state machine in Fig. 7. The system extracts the second information of UTC first, and generates an accurate 3 s effective signal through combing 3 s multiples and effective signal of PPS. The calibrated timestamp calculated as in (1). The normal timestamp of calculation is counted by the system clock, and counts up to 30 ms as a timestamp. If the counting error exceeds 1ms in the calibration status, the counter is restarted to count and jumps to counting status after supplying the calibration timestamp.

\[
T_t = 120000H + 2000M + \frac{100}{3}S \tag{1}
\]

Where, \(T_t\) is calibrated timestamp, \(H\) is hour, \(M\) is minute, \(S\) is second.

Figure 6 Flow chart of system implementation
2.2.4 Design of PC program
The PC program uses Labview of NI, which use graphical language. The Labview uses data stream mode, and the programming designers only need to create a block diagram to connect to achieve the function. The program do not have to write codes basically, whose development cycle is short, and simple to achieve and easy to maintain. In summary, the system uses Labview as a software development platform.

The PC program is mainly composed of the following parts: data upload (used to report serial data of the module), data parsing and processing (parse serial data), information display, exit and help form.

3. Experiment and Result Discussion
In this section, we use two sets of high altitude meteorological radiosondes based on the proposed GPS time remote synchronization algorithm to measure meteorological data. We give the experimental data and result analysis, and predict meteorological data using the previous neural network algorithm.

3.1 Experiment
High altitude detection mainly measure wind direction, wind speed, pressure, temperature, humidity and other meteorological parameters about 30000 meters above sea level. In the experiment, radiosondes are brought into air with hydrogen balloons, and measure air pressure, temperature, and humidity in the process of soaring. The detected GPS data are analyzed and processed by the system terminal, and are returned to monitoring center. The experimental data of A and B are shown in Table 2. Ground observers calculate the received data, and acquire wind direction, wind speed, pressure, temperature, humidity and other meteorological parameters at high altitude.

| Time   | Location | Latitude (°) | Longitude (°) | Altitude (m) | Temperature (°C) | Humidity (%) | Pressure (hPa) |
|--------|----------|--------------|---------------|--------------|------------------|--------------|---------------|
| 10:09:18 | A        | 45.43        | 126.44        | 632          | 9.3              | 50.9         | 913.0         |
|        | B        | 50.23        | 127.52        | 1083         | 4.2              | 54.7         | 864.8         |
| 10:14:17 | A        | 45.43        | 126.45        | 2171         | -4.3             | 55.7         | 757.2         |
|        | B        | 50.23        | 127.52        | 2826         | -8.3             | 52.5         | 698.4         |
| 10:19:17 | A        | 45.42        | 126.46        | 3819         | -14.1            | 44.9         | 616.5         |
|        | B        | 50.23        | 127.53        | 4528         | -18.6            | 38.9         | 563.0         |
| 10:24:17 | A        | 45.40        | 126.47        | 5561         | -26.2            | 30.4         | 492.1         |
|        | B        | 50.22        | 127.53        | 6287         | -32.1            | 25.3         | 446.8         |
| 10:29:17 | A        | 45.38        | 126.48        | 7232         | -40.3            | 20.2         | 393.0         |
|        | B        | 50.22        | 127.53        | 7751         | -44.8            | 18.1         | 365.8         |
| 10:34:17 | A        | 45.36        | 126.50        | 8885         | -53.5            | 15.1         | 311.6         |
|        | B        | 50.21        | 127.54        | 9431         | -57.4            | 14.2         | 288.0         |
| 10:39:17 | A        | 45.33        | 126.51        | 10583        | -62.9            | 13.1         | 242.9         |
|        | B        | 50.21        | 127.55        | 11892        | -65.0            | 11.9         | 199.1         |
3.2 Analysis and Discussion of Experimental Results

We compare the temperature, humidity and pressure data in the form of a three-dimensional map using two sets meteorological data of different locations from radiosondes, as shown in Figs. 8 (a), (b), and (c), respectively. As shown from the graphs, meteorological data of two radiosondes at different locations are different from each other. However, the received data can be transmitted back to the ground station at the same time based on the proposed GPS timing algorithm. Thus, the systems can avoid delay phenomenon from different locations, and ensure the accuracy of weather forecasting.
Figure 8 The three-dimensional curve for temperature, humidity and pressure data of radiosondes change with time and height.

We only give 10 sets of data in Fig. 8 as data acquisition density radiosondes are large relatively. We fit the received data and get the expressions for temperature, humidity and pressure changes with height as (2), (3) and (4), respectively.

\[ t = -0.001692a^2 + 0.06674a^4 - 0.8964a^3 + 4.983a^2 - 17.95a + 18.88 \]  \hspace{1cm} (2)

\[ h = 0.001313a^5 - 0.06446a^4 + 1.151a^3 - 8.754a^2 + 21.04a + 40.83 \]  \hspace{1cm} (3)

\[ p = 0.0032a^4 - 0.1844a^3 + 6.008a^2 - 116.8a + 984.5 \]  \hspace{1cm} (4)

Where \( a \) is altitude, \( t \) is temperature, \( h \) is humidity, and \( p \) is pressure.

Temperature collection effectively avoid high temperature phenomenon caused by thermal radiation, and humidity collection avoid sudden changes in humidity caused by condensation phenomenon. The response time of temperature and humidity is short, which can reduce data loss during transmission. The density of pressure data is also sufficient for later analysis of meteorological data.

4. Prediction of Meteorological Data based on Neural Network Algorithm

In this section, we use neural network algorithms to predict meteorological data measured by the system, including the employed algorithm and simulation results discussed.

4.1 Prediction Algorithm Based on Neural Network

We selected Fletcher-Reeves algorithm to predict the meteorological data, and make the high altitude experimental data measured by radiosondes as the analysis object based on the previous studies \(^{[12]}\). According to the trend of time, we predict the next set of data on the basis of exact data measured in the past to overcome the data loss phenomenon of a radiosonde and ensure the integrity of meteorological data.

4.2 Simulation Results and Discussion

Based on the previous analysis results, we select Fletcher-Reeves algorithm to train meteorological data. There are six input nodes (latitude, longitude, altitude, humidity, temperature and pressure), six hidden nodes and one output with a target error of \( 7.5 \times 10^{-10} \) in the training algorithm. Therefore, the system can achieve the effect of predicting loss of data \(^{[12]}\). The simulation results of the predicted meteorological data of locations A and B are shown in Figs.9 and 10.
(a) Simulation results of meteorological data at location A, the epochs of training is 43, and the best training performance is $5.5723 \times 10^{-10}$.

(b) Simulation results of meteorological data at location A, the epochs of training is 87, and the best training performance is $6.8784 \times 10^{-10}$. 
(c) Simulation results of meteorological data at location A, the epochs of training is 174, and the best training performance is $6.5505 \times 10^{-10}$.

Figure 9  Predict training results of meteorological data at location A.

(a) Simulation results of meteorological data at location B, the epochs of training is 56, and the best training performance is $5.4212 \times 10^{-10}$.
(b) Simulation results of meteorological data at location B, the epochs of training is 90, and the best training performance is $7.3033 \times 10^{-10}$

(c) Simulation results of meteorological data at location B, the epochs of training is 107, and the best training performance is $6.4032 \times 10^{-10}$

Figure 10 Predict training results of meteorological data at location B

From the simulation results of the meteorological data at locations A and B given in Figs. 9 and 10, we can see that when the epochs of trainings is small, the target error can be achieved, but the result is not optimal. With the increase of training epochs, not only the target error can be achieved, but also the target error and the optimal result can be overlapped. When the target error coincides with the optimal result, as shown in Figs. (b) and (c) of Figs. 9 and 10, the training error decreases as the epochs of training increases. We can conclude from the results of Figs. 9 and 10 that the proposed Fletcher-Reeves algorithm can well achieve the goal of meteorological data prediction and provide reliable support for weather forecasting.
5. Conclusion
High altitude meteorological data is complicated and changeable, and each parameter has certain influence directly. In this paper, we propose a GPS timing remote synchronization algorithm in high altitude meteorological data acquisition system to achieve high-altitude meteorological data synchronization processing in different locations. We use NEO-M8N chip of UBLOX in GPS module to realize GPS timing remote synchronization, including hardware and software designs for the system. The designed system test high altitude meteorological data in different locations, and synchronization error of received data are several tens of ns. Thus, this can meet delay requirement set by the system and achieve the synchronization requirements of different systems in different places. By using MATLAB software, the measured experimental results are processed and the relationships between temperature, humidity and pressure change with height are obtained. Finally, Fletcher-Reeves algorithm based on neural network is used to forecast the meteorological data and analyze the training error. Therefore, the system is proved to be able to receive and process remote data synchronously, predict the meteorological data, and provide support for meteorological prediction research.

Acknowledgements
This work is supported by the Natural Science Foundation of Heilongjiang Province, China: Research on High Altitude Meteorological Data Processing Method based on Neural Network ((F2016032)).

References
[1] Yang Qiu, Yunbo Shi, Yi Luo, Debin Xiu, Qi Sun, Chunzhi Ge and Huijie Nan. Research of High Altitude Meteorological Data Processing Method[C]. International Conference on Measurement, Information and Control (MIC 2012), 2012: 46-49.
[2] Wencheng Zhang, Xiaoxing Zheng, Xiaoguang Sun, Qiang Niu and Chengjia Bao. Short-term Photovoltaic Output Forecasting Based on Correlation of Meteorological Data [C]. IEEE Conference on Energy Internet and Energy System Integration (EI2), Beijing, China, 2017: 1-5.
[3] Elin Högström and Annett Bartsch. Impact of Backscatter Variations Over Water Bodies on Coarse-Scale Radar Retrieved Soil Moisture and the Potential of Correcting With Meteorological Data [J]. IEEE Transactions on Geoscience and Remote Sensing, January, 2017, 55(1): 3-13.
[4] Shigeomi Hara, Makoto Kasu, and Noriaki Matsui. Estimation Method of Solar Cell Temperature Using Meteorological Data in Mega Solar Power Plant [J]. IEEE Journal of Photovoltaics, September, 2016, 6(5): 1255-1260.
[5] Shobha N and Dr. Asha T. Monitoring Weather based Meteorological Data: Clustering approach for Analysis [C]. International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), Bangalore, India, 2017: 75-81.
[6] Wenqiu Wang. GPS-Based Time & Phase Synchronization Processing for Distributed SAR [J]. IEEE Transactions on Aerospace and Electronic Systems, July, 2009, 45(3): 1040-1051.
[7] H. C. Berm and R. J. Wilkes. GPS Time Synchronization System for K2K [J]. IEEE Transactions on Nuclear Science, April, 2000, 17(2):340-343.
[8] Y. Chen, D. Reale, J. Dickens, S. Holt, J. Mankowski and M. Kristiansen. Phased Array Pulsed Ring-down Source Synchronization with a GPS Based Timing System [J]. IEEE Transactions on Dielectrics and Electrical Insulation, August, 2011, 18(4): 1071-1078.
[9] Ali Khalajmehrabadi, Nikolaos Gatsis , David Akopian, and Ahmad F. Taha. Real-Time Rejection and Mitigation of Time Synchronization Attacks on the Global Positioning System [J]. IEEE Transactions on Industrial Electronics (ACCEPTED), DOI 10.1109/TIE.2017.2787581.
[10] J. Ida Jensona and E. Prayinli. Solar Radiation Forecasting Using Artificial Neural Network [C]. International Conference on Innovations in Power and Advanced Computing Technologies, Vellore, India, 2017: 1-7.
[11] Krishan L. Khatri and Lakshman S. Tamil. Early Detection of Peak Demand Days of Chronic
Respiratory Diseases Emergency Department Visits Using Artificial Neural Networks [J]. IEEE Journal of Biomedical and Health Informatics, January, 2018, 22(1): 285-290.

[12] Min Qiu, Yan Nan Mu, Xiu Ping Zhang and Dong Yang, Research on Data Processing Method of High Altitude Meteorological Parameters Based on Neural Network [J]. The Open Automation and Control Systems Journal, 2015, (7): 1597-1604.