

Marcin BAJER

ABB CORPORATE RESEARCH,
Starowińska 13a, 31-038 Kraków, Poland

Synchronization of current and voltage measurements in a modular motor diagnostic system

M.Sc. Marcin BAJER

Graduated from University of Science and Technology in Kraków, Poland, in field of Computer Control Systems. Since 2008 he has been working at ABB Corporate Research in Kraków, where he is responsible for programming and testing embedded devices. His research interests also cover industrial communication protocols, Linux kernel and mobile technology.



e-mail: marcin.bajer@pl.abb.com

Abstract

The purpose of this paper is to describe a problem of precise time synchronization between elements in a distributed automation system. At the beginning, to present current state of art, Network Time Protocol is described. Further, the main focus is placed on describing this problem in context of systems in which interconnection is done with RS-485 serial communication. The issue is presented based on the solution used for active power calculations in a modular motor control system. In this case, to determine the power factor, precise calculation of the delay between voltage and current zero crossings measured by discrete modules is required. The developed solution of this problem can be extended to most of distributed systems interconnected with a serial interface. Further in the article the method to validate algorithm accuracy for different voltage and current plots was presented.

Keywords: distributed diagnostic, RS-485, serial communication, time synchronization, active power, power factor.

Synchronizacja czasowa pomiarów prądu i napięcia w modułowym systemie diagnostyki silnika elektrycznego

Streszczenie

W artykule opisano zagadnienie precyzyjnej synchronizacji czasowej elementów rozproszonego systemu automatyki przemysłowej. W pierwszej części zostały zaprezentowane dostępne rozwiązania umożliwiające monitorowania zależności czasowych pomiędzy zdarzeniami w rozległej sieci komunikacyjnej. Jako przykład opisano działanie *Network Time Protocol* (NTP) służącego do synchronizacji czasowej urządzeń w sieci Internet. W dalszej części publikacji główny nacisk został położony na opisanie systemów w których komunikacja odbywa się przy użyciu magistrali szeregowy opartej o interfejs RS-485. Problem przedstawiono na podstawie synchronizacji czasowej pomiarów napięcia i prądu dla celów wyznaczenia mocy czynnej w modułowym systemie sterownika silników elektrycznych. Wynalezione rozwiązanie tego zagadnienia zostało zgłoszone jako patent i może być rozszerzona na większość systemów rozporozonych w których wymiana danych odbywa się przez interfejs szeregowy. W kolejnej części przedstawiono rozwiązanie pozwalające na walidację dokładności algorytmu dla różnych przebiegów prądu oraz napięcia. Aby nie wpływać na poszczególne elementy systemu testy oparto na obserwacji danych wymienianych pomiędzy elementami systemu.

Słowa kluczowe: rozproszony system diagnostyki, RS-485, komunikacja szeregowy, synchronizacja, moc czynna, współczynnik mocy.

1. Time synchronization in wide spread communication networks

Increasing modularity is a global trend among manufacturers of industrial automation. The use of distributed architectures in industrial applications results with greater flexibility and allows integrators to tailor the system to exact customer's needs. In many

cases this involves using communication networks to spread sensors, actuators and control logic over a large area.

Synchronization in distributed systems is much more difficult in comparison to uniprocessor or multiprocessor devices. Network synchronization deals with the problem of distributing time and frequency among widespread remote locations, often interconnected by non-deterministic, bandwidth limited bus. In such systems each module has its own internal clock. Even if all clocks are once synced, eventually, after time they will lose synchronization due to the clock drift. Depending on the bus and application several time synchronization algorithms were designed.

Before going into details, it is recommended to first clarify meaning of two synchronization quality indicators: accuracy and precision. A precise algorithm will generate all results close to each other, but a constant error may be introduced. In case of an accurate algorithm there is no constant error but results are less repeatable.

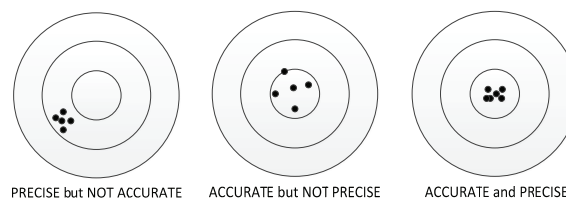


Fig. 1. Time synchronization quality indicators [1]
Rys. 1. Wyznaczniki jakości synchronizacji czasowej [1]

For further discussions about time synchronization it is worth also dividing industrial automation buses into two main types basing on the media access algorithm.

The traditional fieldbus are based on Master-Slave/Token passing technology which does not involve any packet switching. Nodes are interconnected point-to-point or with serial connection (Profibus, Modbus RTU). RS-485 is commonly used as a physical layer. In this case the time to transport data between nodes is deterministic.

Recently, due to popularity of Ethernet, fieldbuses which use multiple medium access and packet switching technology are growing in popularity (i.e. Profinet, EthernetIP, EtherCAT). In those networks data to be transferred is grouped into packets. Switching networks can share common paths for transferring data between network segments. Moreover, sometimes multiple paths can be used for communication between two network nodes.

Although, Ethernet based industrial communication protocols were modified to increase its real-time characteristics, data transfer delays in those networks might vary. Time synchronization in such networks is more difficult.

2. Network Time Protocol

Network Time Protocol is the most popular protocol for time synchronization in packet-switched, variable latency networks. Although the details of the algorithm are complicated, its general idea is pretty simple. A client requests the current time from a server, and uses the response to set its own clock.

Based on time stamps of sending and receiving messages in Slave and Master it is possible to calculate the transmission delay (1) and offset between the client and server clocks (2) [2].

$$\text{delay} = (T_4 - T_1) - (T_3 - T_2) \quad (1)$$

$$\text{offset} = \frac{(T_2 - T_1) + (T_3 - T_4)}{2} \quad (2)$$

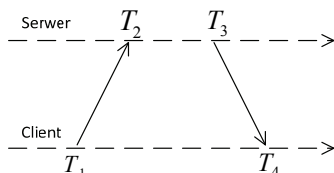


Fig. 2. Working principle of Network Time Protocol
Rys. 2. Zasada działania Network Time Protocol

Multiple levels were introduced in the global structure of NTP servers. Atomic clock sources are on the top, providing the most accurate clock source. Higher level servers are time servers for lower level computers, but it is not necessarily true that the stratum levels are indicators of quality or reliability [3]. The main purpose of leveling is to prevent cyclic dependency.

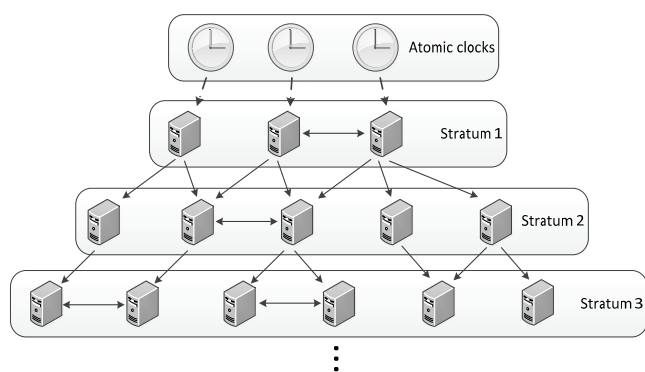


Fig. 3. Topology of NTP servers
Rys. 3. Topologia serwerów NTP

3. The purpose of distributed motor diagnostics

Industry consumes 42% of all electricity produced worldwide. Two third of this is used by electric motors, which gives 28% global energy consumption [4]. Large industrial facilities often contain thousands of electrical motors. Ensuring their correct work is crucial since breakdown can generate significant financial losses or even impact on safety of people on the site. Providing a flexible, cost efficient solution for motor control and diagnostic is a demanding task.

In many simple systems to provide required diagnostics functionalities it is sufficient to monitor feed motor currents. Based on this, problems such as phase lost, wrong phase sequence, overcurrent, undercurrent, earth fault and many others can be detected. With a thermal model of the motor embedded into a controller it is also possible to protect the motor from overheating.

The main problem with such approach is that the motor starting is needed to notice the problem. In many cases it is required to forbid the motor starting to protect the whole installation against the impact of faulty power supply. In order to achieve this, voltage monitoring is necessary.

4. Background of the current and voltage measurement synchronization problem

There are many reasons why it is expected to provide a system in which separate modules for voltage and current measurement are used. The main one is cost efficiency – customers can decide for which elements of the system supplementary protection is needed. In addition, bus interconnection between elements of the control system enables more flexibility of placing modules in the production cell.

Those factors lead to decision that the motor control system described in this paper is modular. The main unit measures a 3-phase current and provides an interface to access data by the field devices (PLCs, HMI). The system can be extended by Voltage Measurement Unit and Digital IO Unit.

In most cases measurements are independent – to provide the required information about the voltage quality (under/overvoltage, voltage sag and dip, high level of THD), the voltage measurement unit does not require any information about the current. The same statement applies to current monitoring.

The typical situation where it is necessary to combine voltage and current measurements is active power calculation.

$$P = U_{RMS} I_{RMS} \cos(\varphi) \quad (3)$$

To compute power factor, the exact time stamps of current and voltage zero crossings are needed.

5. Required accuracy of synchronization

To validate the feasibility of the idea of calculating active power by separate devices, there is a need to estimate the required precision of time synchronization.

The error of the active power measurement may be represented by the absolute error – the actual amount of error described in measured value units or by the relative error - ratio comparing the error to the size of measurements.

$$E_{absolute} = |x_{measured} - x_{real}| \quad (4)$$

$$E_{relative} = \frac{E_{absolute}}{|x_{real}|} \quad x_{real} \neq 0 \quad (5)$$

The highest absolute error of power factor measurements occurs for $\varphi = \frac{\pi}{2}$ since at this point $\frac{\partial \cos(\varphi)}{\partial \varphi} = \max$. To calculate the needed absolute accuracy of time synchronization, the following equation can be used:

$$\Delta t_{\phi} = \frac{1}{2\pi f} \arcsin(\Delta \cos(\phi)) \quad (6)$$

Assuming, if we would like to achieve the accuracy of $\cos(\varphi) \pm 0.01$, the synchronization error must be below $\Delta t_{\phi} = 31,8$ for the network frequency 50Hz.

The relative error approaches infinity for $\cos(\varphi) = 0$. To achieve meaningful results, we can assume that in most cases the motor works with the power factor between 0.4 and 0.8.

Based on the plots above, one can state that to achieve the absolute error of the power factor below ± 0.1 and percent in the standard operating range below 5% it is needed to calculate the time difference between voltage and current zero crossings with accuracy less than $32\mu s$. When assuming standard serial bus baud rate of 115200, this means synchronization needs to be done with the accuracy of $\sim 3,5$ bit.

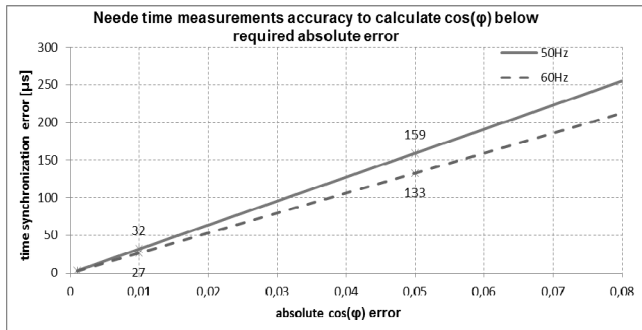


Fig. 4. Relation between allowed absolute error of power factor and accuracy of time synchronization needed
 Rys. 4. Zależności między dozwolonym błędem bezwzględnym współczynnika mocy a wymaganą dokładnością synchronizacji czasowej

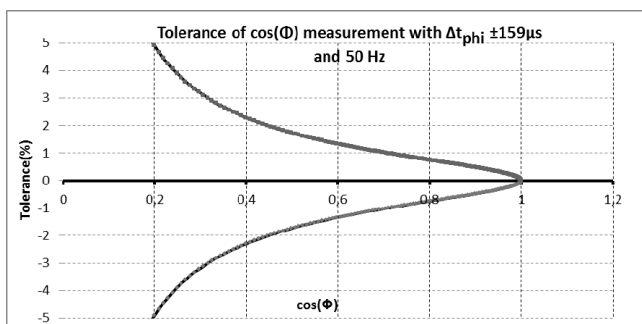


Fig. 5. Plot of the relative error of power factor calculation for the constant error in φ calculation
 Rys. 5. Zależność względnego błędu współczynnika mocy dla stałego błędu wyliczenia φ

6. Synchronization algorithm

The system architecture is presented in Fig. 6. The current measurement unit is a Master, there can be many extension modules which are Slaves on a bus.

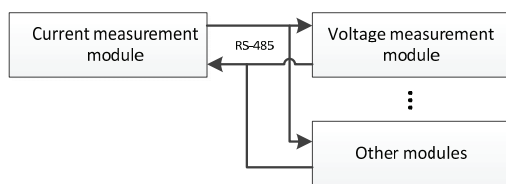


Fig. 6. Architecture of developed motor diagnostic system
 Rys. 6. Architektura rozwijanego systemu diagnostyki systemu

Since both devices use standard inexpensive quartz crystal to keep the common time base, it is needed to perform synchronization often to avoid a clock drift.

Another limitation is the assumption that a regular RS-485 interface should be used. This imposes the limitation of both physical interface and its hardware implementations in a microprocessor (IRQ handling).

Of course, using Ethernet would solve the problem since the accuracy of NTP is far better than the required one in this case. Although, Ethernet has many additional advantages, it is a complex protocol and its implementation is much more expensive – additional cost cannot be justified in this case.

All of those requirements lead to the solution described further in this paper.

The current state of art is that in most cases to calculate the time difference between events captured by distributed measurement units a time stamp to each of the events is added. The alternative solution would be to add a separate bus wire, to be used only for synchronization purpose.

Since both devices operate asynchronously to each other, with a different clock signal frequency, a special telegram sequence was designed for data exchange and synchronization. The complete message exchange sequence is described in Fig. 7.

$$T_{Diff_Ev1_Ev2} = [T_{First_Byte} - ((T_{Send_Delay} - T_{Ev1}) + T_{Transmission} + T_{Interrupt})] - T_{Ev2} \quad (7)$$

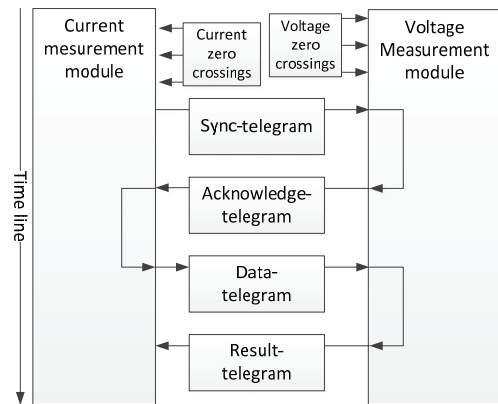


Fig. 7. The power factor calculation algorithm
 Rys. 7. Schemat algorytmu obliczania współczynnika mocy

The details of the synchronization algorithm are presented in Fig. 8. Each telegram contains CRC. To reduce the needed computation power and speedup frame composition, “Sync-Telegram” and “Acknowledge Telegrams” have the fixed content. Their only purpose is to indicate a common reference event in time base on which synchronization is done.

Both devices save the time stamp of the event base on their internal clock (T_{Send_Delay} , T_{First_Byte}). As it was described before in this paper, the transport delay in case of RS485 is deterministic. It consists of the time of sending one byte over the bus ($T_{Transmission}$) and the time to handle serial port interrupt in the receiver ($T_{Interrupt}$). With acceptable error it can be assumed that a sum of those is constant. It can be calculated using the serial port baud rate or determined by a measurement and is therefore known in advance. In this way the required corrections can be introduced to T_{First_Byte} on the second device.

The "Data Telegram" contains the event time (T_{Ev1}) and the time of sending the synchronization telegram (T_{Send_Delay}). Based on those information the second device is able to calculate the time difference ($T_{diff_Ev1_Ev2}$) between events 1 (T_{Ev1}) and 2 (T_{Ev2}) (7).

The serial data transmission is independent of asynchronous Events 1 and 2. The time between the synchronization telegram and the data telegram can be extended to fit application needs (CPU/communication bus load). Also, the time between the first event and the synchronization telegram can be longer, but it needs to be remembered that the clock drift will impact the results. In both cases the limitation is the size of a register used for saving time stamps.

With the presented approach the frequency of system timers running on Device 1 and 2 can be different. Recalculation to the common time base can be made during calculation of difference between events ($T_{diff_Ev1_Ev2}$).

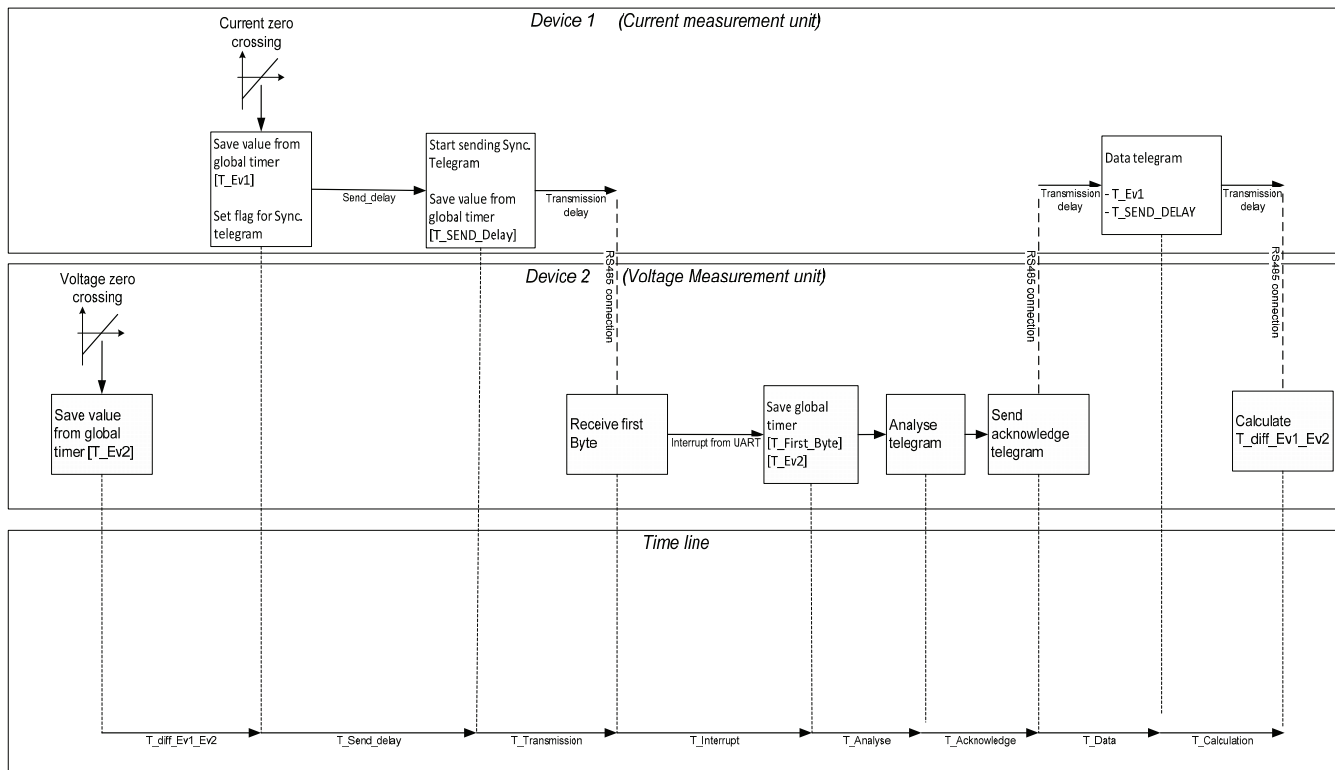


Fig. 8. Details of the synchronization algorithm
 Rys. 8. Szczegółowy opis algorytmu synchronizacji

Another advantage of such approach is the fact that the synchronization takes place after each event $[T_{Ev1}]$ detection. The algorithm is resistant to the clock drift since only the time between the first event and the synchronization event is important for calculations.

7. Tests of the solution

To ensure the correct behavior of the algorithm, two kinds of tests were performed. First, to ensure the correct solution work, the system described in Fig. 9 was constructed. The idea was to test only the feasibility of the proposed solution. To avoid hardware differences (A/D converters, UART), the same hardware was used.

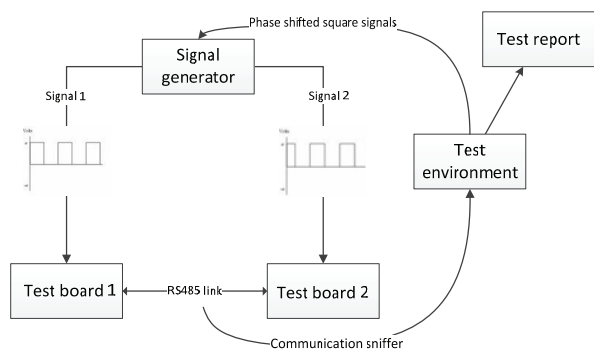


Fig. 9. Idea of algorithm feasibility tests
 Rys. 9. Idea testów weryfikacyjnych poprawności działania algorytmu

All tests were performed with help of the dedicated test environment which could control the signal generator to generate square wave signals with different phase shift. On the test board mock software was implemented. In addition, the test environment was sniffing RS-485 to capture the calculated value of the phase

shift between signals. As the result of the test the report was generated.

The second test was performed on the end product before its release. Instead of a square wave, the current and voltage plots were generated. A different level of THD was introduced in addition, to test its impact on the power calculation accuracy. The tests showed there were no additional corrections needed to the algorithm.

8. Summary

In this paper the solution for synchronizing asynchronous events detected by distributed modules is presented. The tests show that this solution works and provides significant amount of flexibility in implementation. The synchronization error is negligible in comparison to other delays (A/D conversion delay and filter, measurement offset etc).

9. References

- [1] Dr. Sarath Kumar, M Vinod Kumar: Synchronization In Packet Switched Networks. A Survey of Existing Techniques, Tejas Networks 2013.
- [2] Mills, D., Network Time Protocol Version 4: Protocol and Algorithms, BCP 78, RFC 5905, June 2010.
- [3] Eckert, Jason W.: Linux+ Guide to Linux Certification, Cengage Learning 2011.
- [4] ABB Infographic - Powering the world economy - Energy efficiency
- [5] Baetz Ingo, Bajer Marcin, Oestreich Rolf, Mueller Peter, Stegmaier Peter: Method for determining the time difference between two events recorded with asynchronously running timers in different devices. European Patent EP2595332.