Simulation model of DC servo motor control

Article · September 2010

DOI: 10.1109/EPEPEMC.2010.5606599

M. Kostadinovic
University of East Sarajevo

Mile Stojcev
University of Niš

Simulation model of DC servo motor control

Miroslav Kostadinović*, Mile Stojčev**, Zlatko Bundalo***, Dušanka Bundalo****

* University of East Sarajevo, Faculty of Traffic and Transportation, Doboj, Bosnia and Herzegovina, e-mail: *kostadinovicm@gmail.com*

** University of Niš, Faculty of Electronic Engineering, Niš, Serbia, e-mail: *mile.stojcev@elfak.ni.ac.rs*

*** University of Banja Luka, Faculty of Electrical Engineering, Banja Luka, Bosnia and Herzegovina, e-mail: *zbundalo@etfbl.net*

**** UniCredit Bank, Banja Luka, Bosnia and Herzegovina, e-mail: *zbundalo@etfbl.net*

***Abstract*—In the paper are explained in details all phases of adjusting TrueTime simulation surrounding necessary for analysis of behavior of WirelessHART protocol implemented on the example of control system with three nodes for control of DC servo motor. At the end of paper are given obtained results of simulation for control and planing of execution tasks in WirelessHART network.**

***Keywords*—DC machine, Industrial communications, Modelling, Sensor, Simulation.**

I. Introduction

WirelessHART protocol has relatively low speed of data transfer in the comparison with the IEEE 802.11g standard for computer wireless networks. This protocol works on the frequency of 2.4GHz in the ISM radio range using the Time Division Multiple Access (TDMA) for access to communication medium [1]. The complete time of communications executes inside of predetermined time slot of 10 ms. Series of the time slots form superframe for the data transfer and WirelessHART makes possible the hopping between communication channels in order to avoid interference and reduce multi-path fading effect.

One or more appliances or devices which send data and one or more appliances or devices which receive data can be determined that communicate mutually in one beforehand envisaged time slot. Time slot may be reserved for communication of only one appliance or can support dividing of data transfer across one time slot [2].

WirelessHART is communication protocol that is organized like the ISO/OSI 7 communication model. Data transfer from all communication appliances in WirelessHart network passes across the gateway (Fig. 1). It can be seen that gateway must direct packages towards beforehand quoted given destination. Gateway uses standard HART commands for communication with network devices and host applications. The network manager creates start superframe and configures Gateway [3].

II. Simulator Description

Here is described the usage of original simulator (dissembler) TrueTime based on the MATLAB/Simulink, which simulates regulating and control mechanism in execution of tasks in real-time systems, networks (wired or wireless) and dynamic plants.

Originally have been implemented two types of communication protocols, 802.11b/g (WLAN) and 802.15.4 (ZigBee), and the usage possibility of WirelessHart has been added later. Aim of WirelessHart development is to establish standard for wireless communication for usage in process automatics. WirelessHart makes possible cheap and relatively slow, in the comparison with 802.11b/g, wireless connection with HART-enabled devices. The WirelessHART network protocol is time divided into slots where every slot lasts 10ms. Slots can be dedicated to one node and use TDMA technique for access to the medium or can be divided between a few nodes and use CSMA/CA technique for access to the medium.

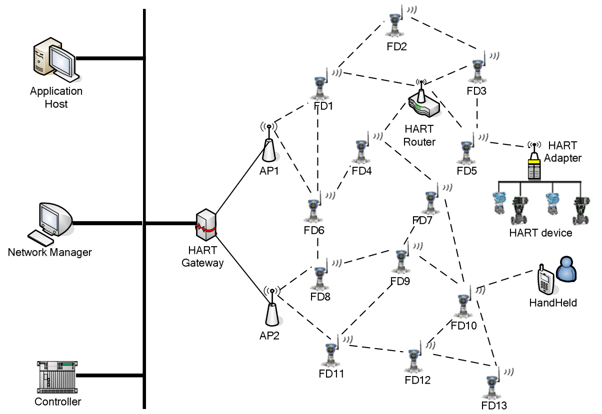

Fig. 1. Structure of WirelessHART network

TrueTime consists of the library of blocks as shown in the Fig. 2. with following functions:

1) TrueTime Kernel: Executes user defined tasks and interrupts which for example present input/output tasks, control algorithms and network interfaces.

2) TrueTime Network: This block is used to simulate access to the medium and transfer of packages according to the chosen network model.

3) TrueTime Wireless Network: Function of this block is similar as of the TrueTime Network block, but instead of wire network it uses wireless network.

4) TrueTime Battery: This block is used to simulate the battery-powered supply.

5) ttGetMsg: It is used for reception of messages from the network.

6) ttSendMsg: It is used for sending messages on the network.

TrueTime is not compose only from library blocks (Fig. 2) but also of collections of C++ functions with the suitable MATLAB MEX interfaces. Some of the functions make possible by simulation creation of tasks, manually interrupting, surveillance, timers, etc. The other functions are real-time which are called by the code during execution of task and enable AD/DA conversion, sending and receiving of messages, etc.

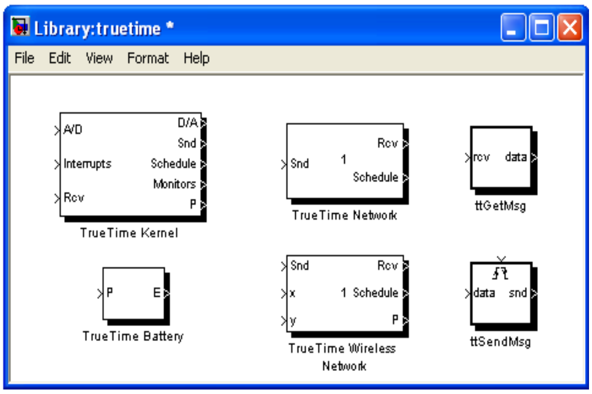

Fig. 2. The TrueTime 2.0 block library

TrueTime has been developed in Simulink, which takes care of the managing system in the meaning of performances, stability and endurance, and primarily has been intended for usage together with the MATLAB/Simulink [4].

III. Adjustment of Kernel and Wireless Network Block

Configuration of the TrueTime Kernel has been presented in the Fig. 3., where the input ports „Interrupts“ and „Rcv“ are connected with grounding which is in „Sources“ menu of Simulink library. Output ports „Snd, „Monitors” and „P“ are connected with terminators which are in „Sinks“ menu of Simulink library. Output port „Schedule“ has been connected with the oscilloscope which has been marked with „Schedule“ and is in „Sinks“ menu of Simulink library.

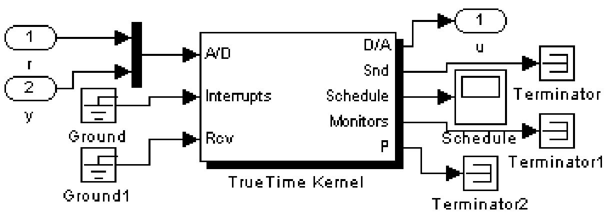

Fig. 3. Configuration of TrueTime Kernel ports

A/D input port has been connected with two input ports marked with r and y, via the multiplexer which is in „Signal Routing“ menu of Simulink library, and D/A output port of kernel has been connected on output port and marked with „u“.

Wireless Network block makes possible simulation of communication between two nodes, and in the original TrueTime simulator have been supported two communication protocols: IEEE 802.11b/g (WLAN) and IEEE 802.15.4 (ZigBee), while the WirelessHART protocol has been subsequently added [4].

IV. Control Performances of WirelessHART

Already two decades HART communication presents the standard for the simplicity, safety and reliability in the process industry. WirelessHART has been designed primarily to cover very wide range of needs in the process industry from the simple supervision to the control in closed loops [5]. Testing and experiments in the field with wireless appliances and devices have shown that these appliances provide correctness of communications, stability, reliability and that can satisfy all needs of supervision and control in industrial processes.

Control applications require periodical samples and on this occasion appear disturbances and delays which especially appear in WirelessHART technology. Actually, control performances with WirelessHART can be compared with the wired system that uses conventional field bus highway. Farther have been presented the some of the factors which appear in uses of WirelessHART.

Sampling speed for WirelessHART is determined from the condition that needs to be fulfilled on the basis of the requirement of concrete control loop while it is necessary at the same time to minimize influence of energy consumption of field appliances which can be supplied using batteries. Usual rule based on experience from practical systems of control is that the feedback control information is taken 4 to 10 times faster than is the speed of process response, where the time of response of process is equal to sum of time constant and dead time of process.

Measuring systems are often unsynchronized with the control system and measured values are usually sampled 2 to 10 times faster than is the answer of process on the change of parameter. In wireless systems it is desirable to reduce the sampling frequency and speed of communication with the measuring device in order to extend the life cycle of battery.

Communication using WirelessHART protocol is realized without endangering the reliability of control [6]. Fig. 4. shows two methods that are used by the sampling:

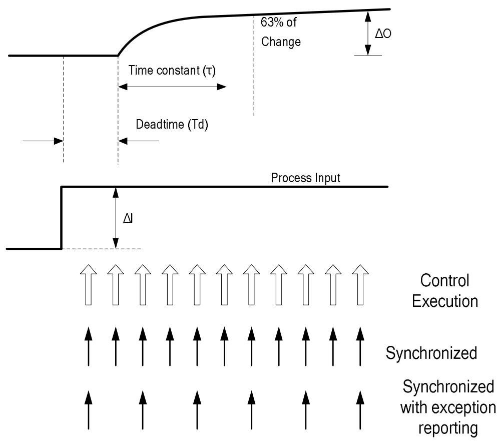

Fig. 4. Synchronized sampling of time slots

1) Synchronized. Samples are taken only when it is necessary control above the process.

2) Synchronized with exception reporting. Samples are taken for beforehand envisaged intervals, for example 4 to 10 times faster than is the response time of process, but transfer is performed only if it is changed measured value or if it is run out the time which was beforehand defined as the time between sending of two messages.

More frequent sending of measured values is possible as well as by the appliances and devices in the wired network. In the case of appliances which use batteries or the appliances where important is to preserve energy, WirelessHART offers users possibility to use the optimal balance at searching of compromise between speed of communication and lifetime of battery.

V. SAFETY OF WIRELESSHART NETWORKS

As in other wireless networks, key management is very difficult and important aspect of network security. WirelessHART networks use network manager as a central authority to distribute keys and for other network management functions. Automatic key management is very important in any network security system.

In the past, the lack of automatic control keys was a major lack of cancellation of some network security solutions. One good example is WEP (Wired Equivalent Privacy) algorithm for secure communication via the IEEE 802.11 wireless networks. While the introduction of WEP in 1997. the idea was to ensure the security of wireless communication, the same as what is in wired networks, but due to manual distribution of keys and the weak algorithm, today is enough a few minutes to break WEP security.

Modern cryptology algorithms are based on secret infornation, so-called keys and only using them can be carryed out the functions of encryption of original information and the inverse decryption function Encryption algorithms must be such that security and confidentiality of encrypted information not depend on the encryption algorithm, but only of secret keys used. Secrecy of keys is a basic assumption of the application of cryptology algorithms in the realization of security services. In symmetric encryption algorithms it is necessary that one key is secret and known only to the partners in communication.

Problem of need for a large number of main keys for all comunnicating entities can be solved by introducing one unique center for distribution of the keys KDC (Key Distribution Center). Using this concept all entities in a safe manner are taking their main key from the KDC, which provides them communication with KDC. Entities who wish to communicate do not possess mutual main keys and at no time know them, but communicate only by using the session key generated and assigned by the KDC. Disadvantage of this protocol is the existence of a single center which contains all the keys, to whom all a priori absolutely believe and whose safety at no time should be questioned. Violation of safety of such center compromises communication of all entities.

In order to satisfy the security requirements of wireless media standard for WirelessHART network determines needs of security managers, who then acts as centralised KDC which is easy and safe way to manage keys. Under the management of keys is implied:

1) Generation of keys,

2) Storage of keys,

3) Distribution of keys,

4) Restoring of keys,

5) Withdrawal of keys,

6) Check of keys.

Network manager is obtaining keys from the security manager by sending request for keys, which is then distributed to the appropriate device to ensure secure wireless communication.

VI. TRUETIME SIMULATION MODEL OF NETWORKED CONTROL SYSTEM

To analyze WirelessHART protocol in the industrial plant it will be shown simulation of networked control system whose scheme has been presented in Fig. 5. Simulations have been made in the Simulink surrounding using the modified TrueTime dissembler.

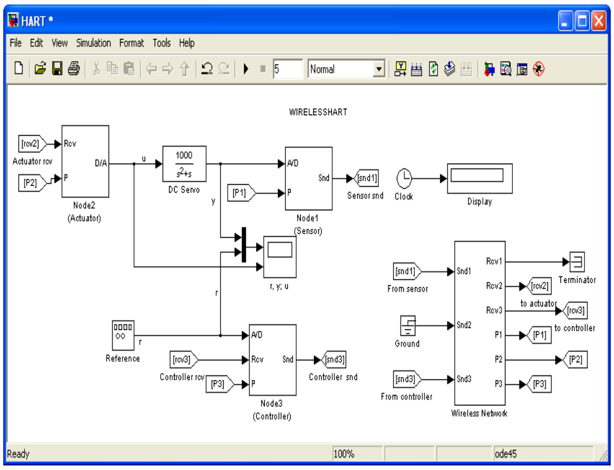

Fig 5. Model of control system

Model of simulation may be described in the following way: Sensor periodically converts analog signal from the process into digital value and sends it to controller. Controller, after taking message from the sensor, calculates the output according to appropriate control algorithms and sends the control signal to the actuator using WirelessHART network (Fig. 6). Actuator converts the control signal into analog and sends it in the process.

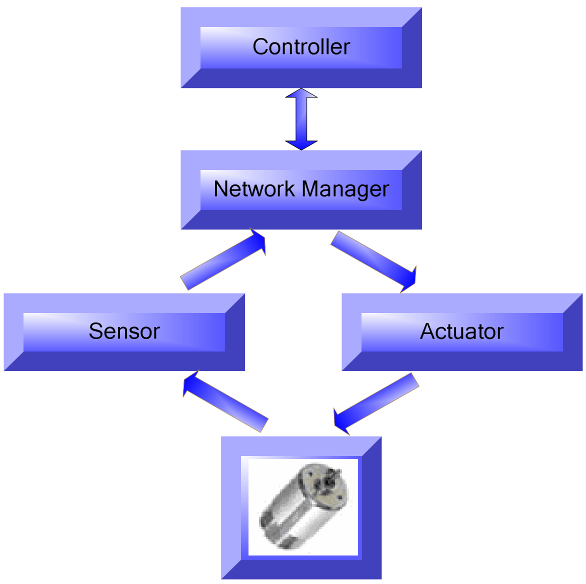

Fig. 6. Scheme of communication between devices

VII. Adjustment of Simulation Environment

In the TrueTime simulation model which has been presented in the Fig. 5. has been designed the networked control system where have been connected three network devices using communication via WirelessHART network [7]. Double-click on Wireless Network block (Fig. 7.) opens the dialog window in which we put the number of devices/knots (three) and select the WirelessHART protocol.

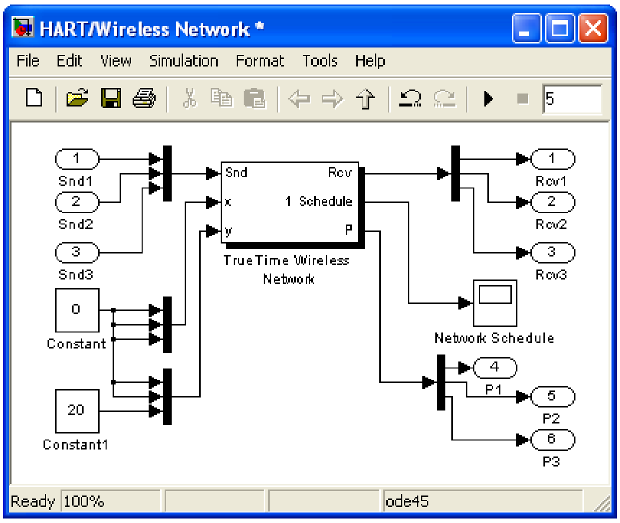

Fig. 7. TrueTime Wireless Network block

A. *Initialization of sensor*

Network devices/nodes have been simulated in the subsystem with the TrueTime kernel block. Details of subsystem for sensor/node1 have been given in the Fig.8. Sensor/node1 uses one A/D converter in the input part and one network output (Snd) at the output.

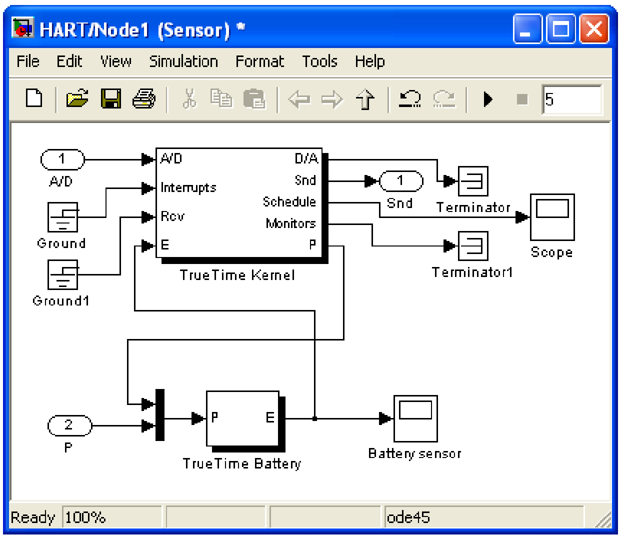

Fig. 8. TrueTime kernel block of sensor

B. *Initialization of actuator*

Details of subsystem for actuator/node2 have been given in Fig. 9. Actuator/node2 uses one network input (Rcv) in the input part and one D/A converter on the output.

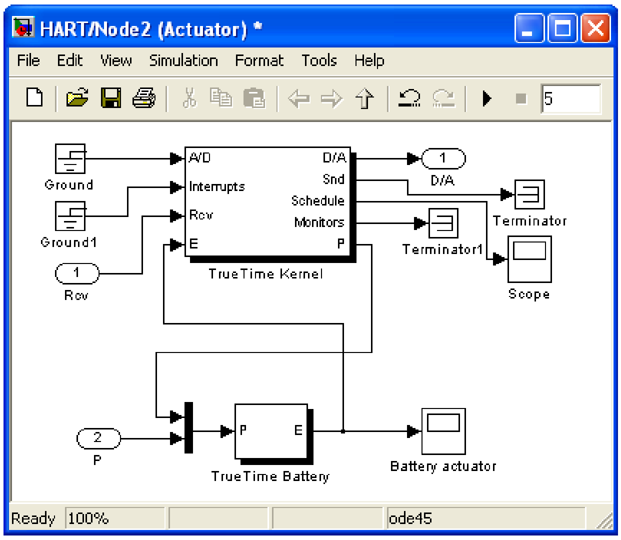

Fig. 9. TrueTime kernel block for actuator

C. *Initialization of controller*

Controller receives messages from the sensor and sends data back on the network i.e. to the actuator. Details of subsystem for the controller/node have been given in Fig. 10., where is used one A/D converter and one network input (Rcv) on the input part and one network output (Snd) on the output part.

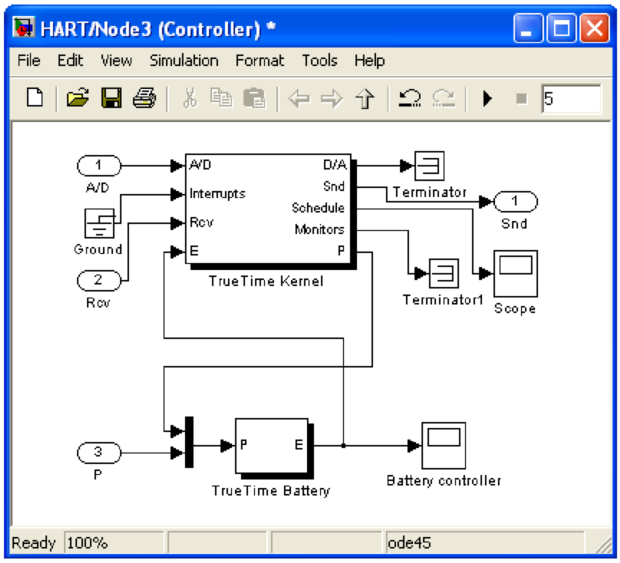

Fig. 10. TrueTime kernel block of controller

After adjusting parameters of kernel block it is necessary to create m-file ctrlcode (Fig. 11.) which calculates the control output signal and uses equations for PD controllers.

When are connected all previously given Simulink blocks it is obtained the model of control system (like in Fig. 5.) whose the wireless communication is realized by the WirelessHART protocol [7].

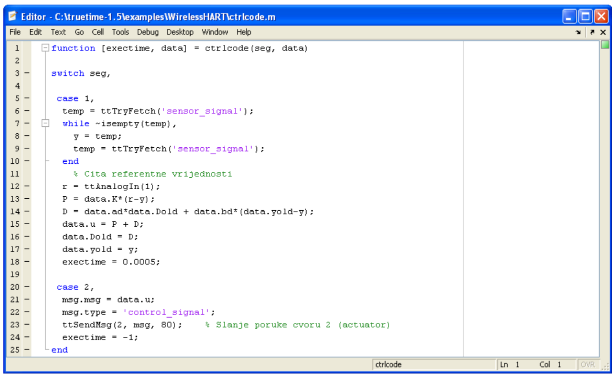

Fig. 11. Ctrlcode function

VIII. SIMULATION RESULTS

If the simulator does not report errors on the occasion of starting simulation, by double-click in the model (Fig. 5) on the oscilloscope opens the window like in Fig. 12. The new opened window makes possible to follow results of simulation of data transfer using WirelessHART protocol implemented on the model of control system for DC servo motor control. Results of simulation of described model have been presented in Fig.12.

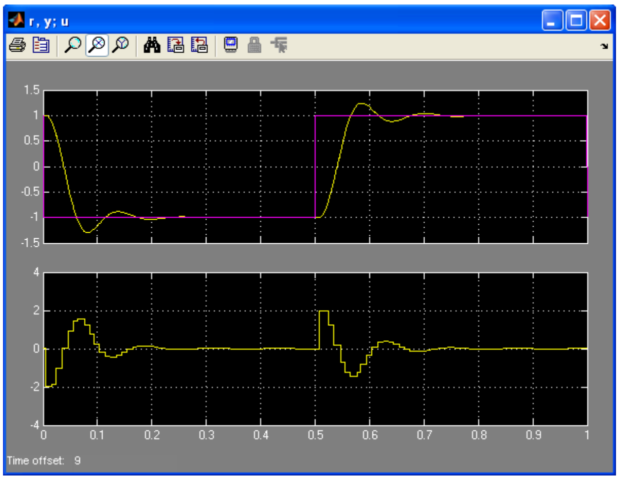

Fig. 12. Simulation results of simple TrueTime control

IX. CONCLUSION

In the paper has been described the new way of implementation of WirelessHART with the modified TrueTime simulator based on the MATLAB/Simulink which can simulate the control device in the execution of tasks in real-time systems, networks and dynamic plants.

WirelessHART protocol makes possible secure, high reliable communication with small delay without influence on the throughput range and performances of process. All this possibilities are integrated in WirelessHART standard in order to enable: simplicity, reliability and safety.

In details have been explained all phases of adjusting TrueTime simulation surrounding necessary for the analysis of behavior of WirelessHART protocol implemented on the example of control system with three nodes for control of DC servo motor. Also, in the form of graphs have been illustrated results of simulations for control and scheduling of executions of tasks in WirelessHART network.

REFERENCES

[1] HART Communication Foundation, *Control with WirelessHART*, HCF LIT-127, Revision 1.0, June30, 2008.

[2] HART Communication Foundation, *System Redundancy with WirelessHART*, HCF LIT-128, Revision 1.0, June30, 2008.

[3] HART Communication Foundation, *Peer-to-Peer Communication with Wireless HART*, HCF LIT-129, Revision 1.0, September 5, 2008.

[4] A. Cervin, D. Henriksson, M. Ohlin, *TrueTime 2.0 beta-Reference Manual*, Department of Automatic Control, Lund University, Lund, Sweden, January 2009.

[5] M. Kostadinović, M. Stojčev, Z. Bundalo, D. Bundalo, "Management of WirelessHART network", *Symposium INFOTEH2009.*, Jahorina, Bosnia and Herzegovina, Vol. 8, Ref. A-16, pp. 71-75, March 2009.

[6] M. Kostadinović, M. Stojčev, Z. Bundalo, D. Bundalo, "Design, Implementation and Simulation of WirelessHart Network", *9th International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Services TELSIKS 2009.*, Niš, Serbia, October 2009, pp. 556-559.

[7] M. Kostadinović, M. Stojčev, Z. Bundalo, D. Bundalo, "Application of modified TrueTime simulator in control of WirelessHart networks", *Proceedings of Intrenational Symposium Ee 2009.*, Novi Sad, Serbia, October, 2009, Paper No. T4-2.5, pp. 1-5.