Acta Universitatis Sapientiae
Electrical and Mechanical Engineering, 9 (2017) 43-55

DOI: 10.1515/auseme-2017-0009

Multilevel Distributed Embedded System for Control of the DC Magnetron Sputtering Process

Albert-Zsombor FEKETE, András KELEMEN,
László JAKAB-FARKAS

Department of Electrical Engineering,
Faculty of Technical and Human Sciences,
Sapientia University, Tg. Mureş,
e-mail: zsombor.fekete@tetronic.ro, kandras@ms.sapientia.ro, jflaci@ms.sapientia.ro

Manuscript received September 15, 2017; revised October 19, 2017.

Abstract: The paper presents in detail a unique multilevel control architecture designed for the comprehensive management of the DC magnetron sputtering process and of all subsystems of the sputtering equipment. The ultimate goal is to increase the repeatability, stability and the controllability of the complex process. The presented topics include embedded and distributed electronics, data acquisition and supervisory control, networking, data management, redundant local and remote data-archiving. There are presented platform independent algorithms managing the data exchange between computational devices, and conclusions are drawn regarding the efficiency of the various algorithms used.

Keywords: sputtering process, multilevel distributed system, embedded electronics, networking, data management

1. Introduction

The DC magnetron sputtering has been used for several decades for producing a wide variety of thin film coatings, serving purely decorative or more functional purposes, providing versatile enhanced mechanical properties, such as increased wear resistance.

The literature in the field is rich and there exist many valuable contributions to the development of different versions of the magnetron sputtering equipment and of the thin film deposition process [8-10].

The process itself consists of the deposition on a surface, named substrate, of different compounds formed in a chamber with controlled atmosphere. This atmosphere consists of an inert gas (usually Ar) used for bombing a surface named target, particles sputtered from the target, and different reactive gases expected to participate in the composition of the thin film deposited on the substrate. Special power supplies are required to form the plasma containing the bombing ions and particles at different energy levels. High vacuum has to be created in the sputtering chamber and the admission and evacuation of different gases has to be controlled in order to ensure the proper composition of the atmosphere. These control tasks are carried out by customized embedded systems, such as the dynamic pressure, the substrate temperature and the various mass flow controllers. Special measurement devices like vacuum gauges, flow meters, mass spectrometer, thin film growth rate monitor, different temperature sensors, sputtering voltage and current measuring units are used to make possible output feedback and state estimation.

The experimental equipment contains two interconnected vacuum chambers. One of them is the location of the sputtering process, while a significantly smaller secondary chamber provides the ultra-low pressure operating conditions for the mass spectrometer, which is used to determine the partial pressures of the gases and the composition of the gas mixture formed. Both chambers are equipped with rotary and turbomolecular vacuum pumps, water cooling systems, safety elements, as well as several control and monitoring units.

Ensuring the repeatability of the process assumes the presence of a controlled environment, achieved by monitoring and controlling as many process parameters as possible with the help of proper data acquisition and control electronics developed for these purposes.

2. The multilevel distributed control system

Even from the first step of the automation process, it became quite obvious that a multilevel control system [7] was needed for the adequate and comprehensive management of the sputtering equipment, due to the complexity of the process. The aim was to create a modular, easily expandable, well-structured, low cost system, which offers redundant data-archiving and remote access to the various subsystems.

When creating a multilevel control system, beyond the basic requirements for data transfer rates, computational capacity, power consumption, compatibility between different units, there are several other criteria that need to be taken into consideration such as implementation and maintenance costs, delivery time, platform flexibility and development time.

The initial approach was to adapt an industrial network such as ProfiBus or ProfiNet. The first test bench included a Unigate Deutschmann ProfiBus PBDPX - V3704 type slave unit acting as a protocol converter and a Vipa CPU-115 type of programmable logic controller (PLC) acting as master unit and as a central data processing unit. In the second setup, a Unigate Deutschmann

ProfiNet PN - V3804 module and a Siemens S7-300 type of PLC was used. The presented Unigate modules feature galvanic isolation, high reliability, small size, increased electromagnetic noise rejection, reduced development time, simple hardware demands, with only a few external components needed.

Thorough tests were carried out in order to determine the highest data transfer rate of the various units in one specific circumstance: constant length user packets of 30B and full message processing in order to perform protocol conversion. Even though the Unigate embedded units feature a high speed fieldbus interface, the global transfer rate was limited by the slow onboard data processing to 1.5kB/s in case of the PBDPX - V3704 module and to 4kB/s in case of the PN - V3804 module. Another significant disadvantage is the relatively high installation cost (PLCs: 800-900€/unit, Unigate modules: 130€/unit, accessories: 15€/unit). The purchasing time is also high, varying from two to three weeks depending on the supplier`s stock. Based on these results, the proposed architecture has proven to be unuseful in our application field.

In order to fulfill the imposed requirements, a four-level, low cost microcontroller based architecture was proposed which can be seen in *Fig. 1*.

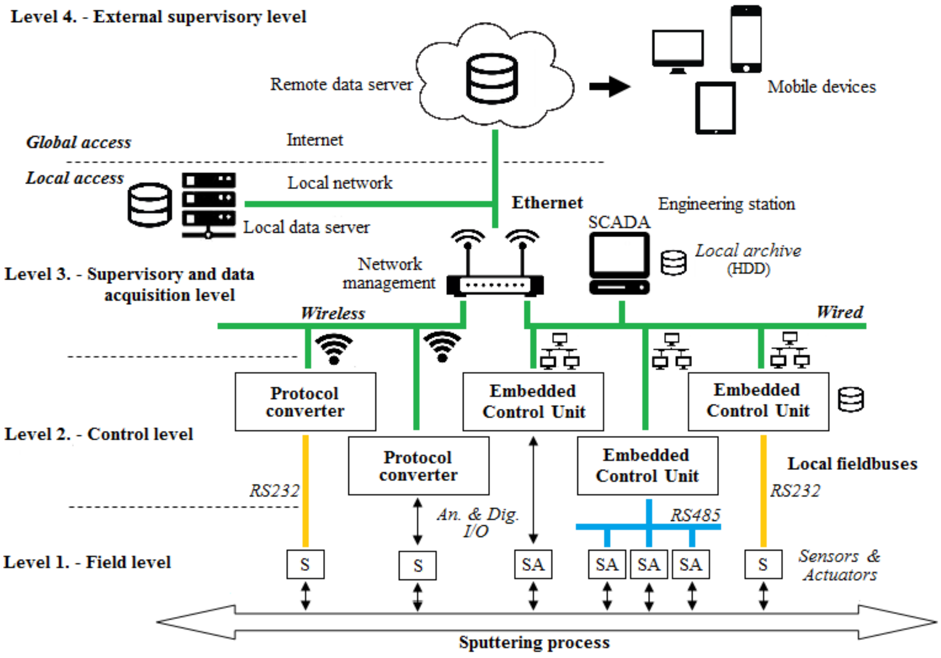

Figure 1: The multilevel distributed system developed for magnetron sputtering

On a small scale, the basic architecture and operation principle of the proposed multilevel control system was previously developed for the mass spectrometer unit [5,6]. Based on the results obtained [5,6] and after the revision of the entire communication stack of the mass spectrometer unit by increasing the modularity of the various algorithms, the small scale version was adapted to the entire DC magnetron sputtering process.

In the presented architecture there can be distinguished four levels: a field level, a control level, a supervisory and data acquisition level and an external supervisory level [1]. All the components can be classified into one of the levels presented. The field level consists of the different vacuum equipments, sensors and actuators which ensure a link between the sputtering process itself and the different embedded control systems situated at the inner level. At this level, generally there are implemented processes like data acquisition, local data processing and fast control (e.g. dynamic pressure controller, substrate temperature controller). At level three, all data and information available can be accessed through different communication channels, enabling local monitoring, data-archiving and the control of slow process parameters through online TCP connection with a dedicated Matlab application capable of executing several complex control tasks. The external supervisory level provides the same features as level three, providing the possibility of access control.

As presented in *Fig. 1*, the supervisory and data acquisition level incorporates a local Ethernet network with star topology [1]. This means that the network is managed by a router [2] enabling wired and wireless connectivity, as well as providing the possibility of online extension, without system shutdown. Also, if one or more nodes (embedded devices) are disconnected from the router, the network remains operational. This gives high reliability to the system in hand [1].

In contrast to the star topology found at level three, at lower levels the majority of the available measuring equipments demanded low speed (9600 and 19200 Baud) local buses and point to point topologies [1]. Typical examples are the different pressure gauges (e.g. Pfeiffer Vacuum MPT100, CMR365 and PKR251) mounted on the vacuum chambers and the digital multimeters (e.g. Appa 103N for measuring the polarization voltage of the substrate and Appa 305 for measuring the temperature of the substrate with a thermocouple). These units or components provide only serial RS232 or RS485 communication interfaces, thus low level, low speed topologies are needed. Furthermore, the baud rates are well defined by the manufacturers and limited to 9600 bps or 19200 bps.

Another typical equipment where the baud rates and communication topologies are restricted by the manufacturers is the Quadrupole Mass Spectrometer (QMS). In order to operate correctly and safely the measuring unit of the QMS, a multi-microcontroller based system must carefully supervise the vacuum pumps, the safety valve, the pressure levels, the various cooling conditions (water flow and temperature, equipment temperature), the power consumptions and the different equipment states. Each safety function is solved with the help of individual embedded units and different sensors connected to a bus type RS485 network [5, 6], featuring a master-slave type of serial communication. In order to increase the safety factor of the installation, cross monitoring is introduced between the units presented above [5]. This means that in case of the malfunction of one or more subunits, the bus network stays functional and the remaining electronics can trigger the shutdown procedure due to the cross-monitoring.

The presented safety mechanism stays functional even if only one unit or subassembly remains operational. The conclusion is that the bus topology in this case provides the necessary conditions to implement the much needed cross-monitoring [1]. As a side benefit, the local network created is easily expandable and the packet collisions are eliminated due to the master-slave type of communication.

Usually a point to point connection [1] is used between a smart field device (e.g. MPT100 full range pressure gauge, APPA 304N digital multimeters, and temperature sensors) and a local data processing and control unit.

The quality of a network strongly depends on the speed, reliability and the security of the data-exchange [1,2]. The speeds of the different networks presented reach from 9600kb/s (field level) to 100Mb/s (supervisory and data acquisition level). The reliability is increased by eliminating ground loops [3] and introducing proper cable shielding [1-3], line isolation and overvoltage protections, as well as different EMI filters in order to assure noise immunity. All the networks use S/FTP type cables, which feature double shielding for noise cancellation. The surge protection mainly consists of transient voltage suppressor (TVS) diodes, which feature low capacitance. This means that high speed data lines are not influenced by the presence of these diodes and every node (every device or unit) can incorporate them [3]. The security of the data-exchange is achieved by restricting global access to different communication channels.

In comparison with the first approach of forming a control network, the presented multilevel architecture presents a lower installation cost (Ethernet network: existent, wired embedded microcontroller units: 30-50€/unit, wireless embedded microcontroller units: 12-20€/unit, accessories: 1-5€/unit). The prices for the embedded units, in contrast to the ProfiBus and ProfiNet modules, include housing, internal wiring, power supplies and local Human-Machine Interfaces.

3. Embedded systems

One of the requirements regarding the development of the embedded systems is to create independent modules with both wired and wireless Ethernet connectivity and with isolated ground planes and power supplies in order to eliminate interdependencies between sensitive measuring circuits such as low voltage (e.g. 800mV) high resolution (e.g. 24bit) analog to digital converters used for Pirani type pressure gauges.

For the various tasks, three types of microcontroller based embedded electronics have been designed and built. Regardless of the chosen architecture, the modules feature a central digital processing unit and all the task specific peripheral circuits (e.g. analog and digital signal conditioner circuits, power electronics) are connected in form of add-on or extension cards. This method provides modularity to the systems in discussion. Every embedded software mainframe uses the same core which means that the development time for each subsystem is optimized and reduced significantly. The main difference between the circuits is the computational capacity of the microcontroller used. For less complex tasks (e.g. protocol converters, sputtering process state monitors, water cooling and substrate heating controllers) there have been used two slightly different ESP8266 based circuits featuring wireless connectivity, onboard RS232, digital I/O pins and a high resolution LTC2410 sigma-delta type analog to digital converter with dedicated reference source. The more complex tasks (e.g. Quadrupole mass spectrometer controller) require a multi-core distributed embedded system [5], based on PIC32MX795F512L microcontroller for wired communication and dsPIC33EP512MC806for high speed signal processing and PID or Fuzzy control algorithms.

Regardless of the architecture used, the software mainframes are constructed using the cooperative multitasking principle [4], limiting the execution time of each section to a maximum of 30ms in order to avoid partial or total lockups inside the software`s vital parts, resulting in an unwanted restart of the microcontroller. Because there exists no higher level arbitration mechanism to monitor the execution times, all the tasks that have to be performed by the microcontroller are distributed in the development phase and are executed based on a predefined order or sequence [4]. The execution timeout was determined empirically. To further conserve valuable microcontroller resources, the data transfer between the internal memory and the communication peripheral is executed by the DMA controller resulting in a 30µs time saving for a 30B packet in every sampling period. The values presented are valid for a 70MIPS, 32bit platform using a sampling frequency of 10kHz. This method is quite useful when a 4kB packet is sent containing the measurement results of the mass spectrometer.

Every connection type featured (TCP, UDP) is implemented in the form of different network services, each having his dedicated socket and port assigned to it. The TCP connection [2] is used to exchange data with the centralized supervisory control application, the UDP connection [2] is used to stream data to a remote data-server for archiving. The different IoT protocols (e.g. MQTT) are utilized to establish unidirectional connection with two separate Cloud type of data-servers. Note that every connection forwarded outside of the local Ethernet network is restricted to a unidirectional connection, only to serve remote monitoring and supervisory functions such as the embedded Web servers.

In order to meet the redundant archiving requirement, data and event archiving is implemented on different levels: control level, local and external supervisory and data acquisition levels. This means that every type of data or event is stored at least on two different computational systems, reducing the risk of irreversible data loss. Note that by moving upwards on the multilevel architecture the storing capacity is increasing. Therefore, on the control level the embedded units only store fault states and important events with or without timestamp. On the other hand, the storage speed is the highest at the supervisory levels, peaking at 1kB/s. The data stored on remote data-servers can be accessed from within the local network or through the Internet with mobile devices as shown in *Fig. 1*.

4. Communication management

By introducing a network connection, it became necessary to develop a unified, well-structured and configurable high level communication protocol which can be used on all embedded systems and computers regardless of their architecture and communication module. The proposed structure is shown in *Fig. 2*.

The developed stack features three layers: user layer, packet management layer and physical layer [6]. When implementing the stack on different platforms, only the physical layer needs to be adapted. The inner layer is responsible for assembling and dissembling the packets, transferring data from and to the memory via DMA, checking for faults and for scheduling the outgoing messages selected by the user.

The top layer is probably the simplest, because the user only needs to set the priority bit of the desired message (‘0’ for excluding the message, ‘1’ for one time and ‘2’ for cyclic transmission). This automated packet forming and sending mechanism assumes the presence of predefined message templates. This communication management protocol enabled us to send measurements periodically and systems events (faults, warnings, states) only upon change.

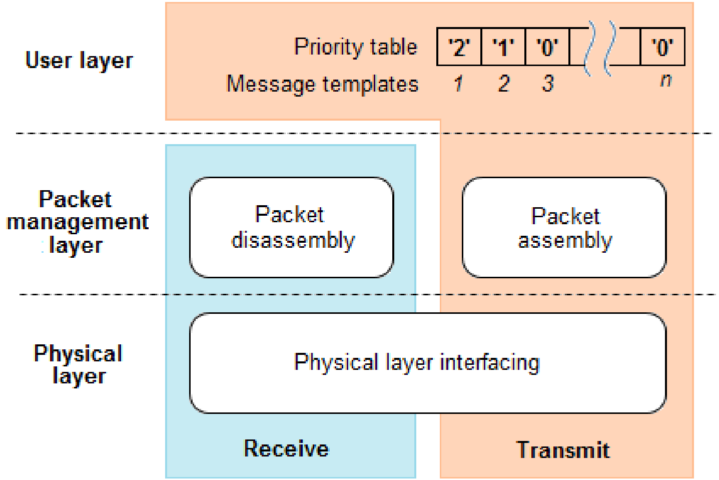

Figure 2: Platform independent communication algorithm structure

5. Supervisory control and data acquisition application

The main idea behind the multilevel distributed control system is to have access to all the data and system parameters at a superior level. This claim assumes the presence of a supervisory control and data acquisition application (SCADA). The mainframe of the application in discussion was developed in CVI Labwindows from National Instruments.

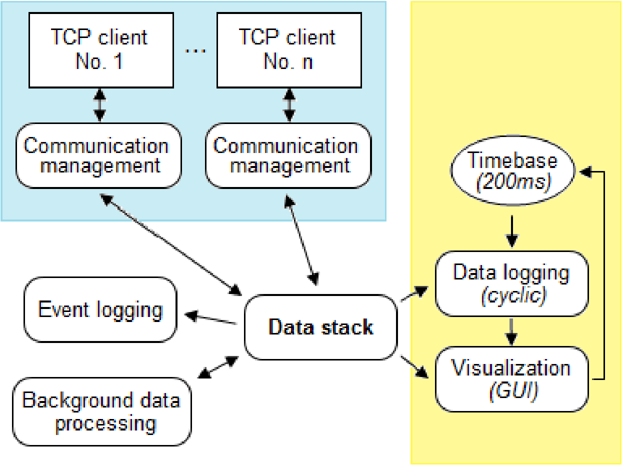

Figure 3: Simplified structure of the mainframe

One of the main criteria regarding the development of the software was to create an efficient application, by distribution of the tasks and by executing them with well-structured algorithms. The simplified structure of the mainframe is shown in *Fig. 3*. The software developed and executed on a process computer has three main functions or features which correspond to the basic components of the application: establishing TCP connection with all the embedded units, data processing and visualization, and data archiving.

Despite of the fact, that all the embedded electronics use the same message structure and communication protocol presented earlier, an iterative TCP/IP model was used. This way, one socket and one port is permanently allocated to every device and connection used. The reason for choosing this type of model is to separate every connection in order to increase program transparency, to implement independent packet processing algorithms and to facilitate the extension procedure of the mainframe with new connections and devices.

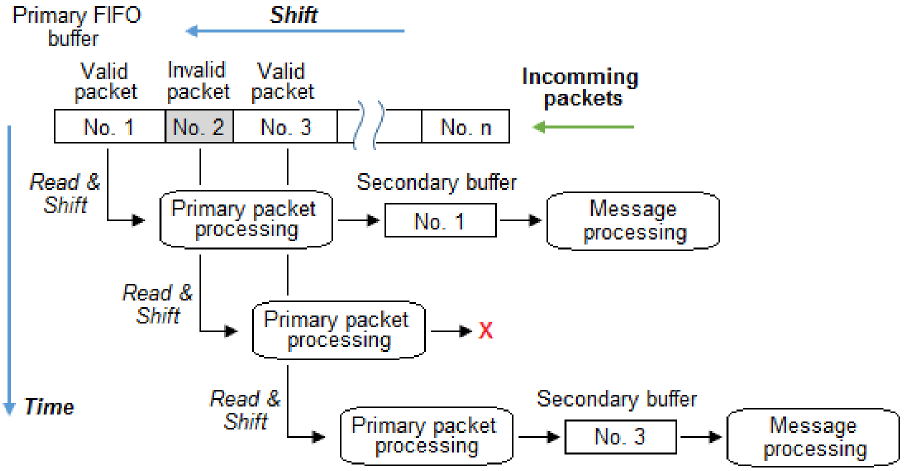

Figure 4: Supervisory control and data acquisition application – TCP buffer structure

It is important to emphasize that every TCP connection uses two separate buffers for storing incoming data messages: primary FIFO buffer, containing the incoming raw packets and a secondary buffer containing valid, demarcated messages which can be individually processed by the corresponding algorithm. The main FIFO type buffer can contain multiple consecutive messages, hence the need for the primary packet processing and the presence of the secondary software buffer or array as presented in *Fig. 4*. The message processing algorithm uses the top and inner layers of the generalized communication management algorithm (*Fig. 2*).

The incoming and derived data are divided into two groups: data that are refreshed periodically and data that are refreshed only upon change. This is available for the corresponding graphical elements of the graphical user interface (GUI) as well. Periodically received data are subcategorized into two classes: fast changing data (e.g. pressures, sputtering voltage and power) and slowly changing data (e.g. cooling and temperature parameters, thin film growth rate), which are determined in function of the variation speed of process parameters. The borderline between the two classes is set to 2s, and the adequate refresh rate is derived from the 200ms time-base. The background data processing is also simplified due to the fact that the majority of the received data is already pre-processed. This method of categorizing the data, the software structure and refreshing particular graphics only if there is a value change contributes to the efficiency of the application by optimizing the utilization of the computer's hardware resources. The average processor utilization on a single core 2.4GHz computer is less than 12%. By turning off the presented method and by keeping the same 200ms refresh and packet rate, the average utilization increases to 75%.

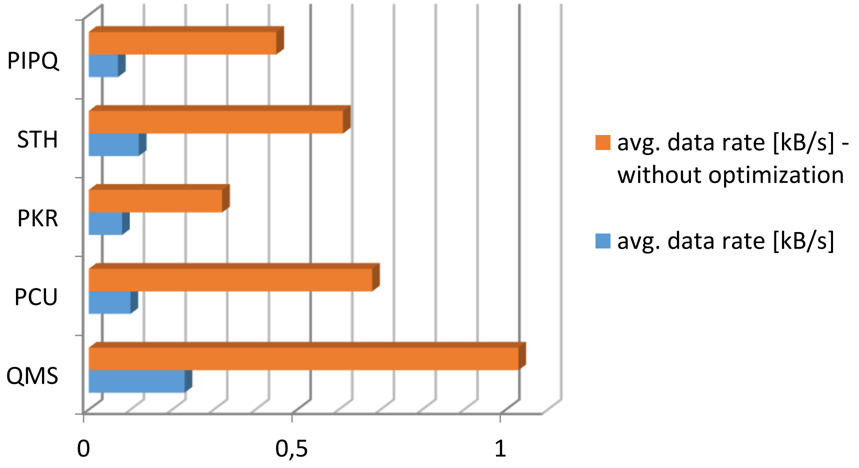

Figure 5: Average data rate with and without traffic optimization

The supervisory application features a built-in TCP traffic monitoring function, with the help of which the user can easily keep track of different network related data traffic. Due to the fact, that the outgoing TCP data rate does not have the same order of magnitude as the incoming data rate, in the following only the latter will be discussed. The positive effect of the optimization presented above is shown in *Fig. 5* through 5 different embedded units (PIPQ – Pirani pressure gauge, STH – Substrate thermometer and controller, PKR – Full range pressure gauge, PCU – dynamic pressure controller, QMS – Quadrupole mass spectrometer). The obtained traffic data presented in *Fig. 5* shows that without the different optimization algorithms, data management and data message types introduced above, the incoming average data rate would increase by approximately 5 times, resulting in unnecessary network and microcontroller CPU load.

The GUI is split into two well delimited zones as shown in *Fig. 6*. Zone 1 is basically a permanent frame which contains graphic elements displaying high priority information like measured and derived data, alarms, events, gauge states, system faults and important system messages. Events which may alter the course of the ongoing experiment and need the immediate attention of the user, such as communication errors, hardware malfunctions, cooling and power failures, process or system parameters out of the allowed operating range.

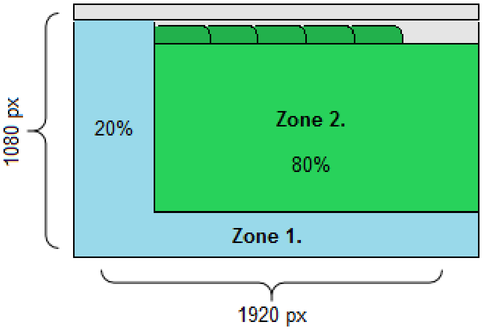

Figure 6: Subdivision of the graphical user interface created for the supervisory control and data acquisition application

All the received and derived data are further categorized in function of the application area (e.g. water-cooling, substrate temperature management) and subsystem (e.g. dynamic pressure controller, Quadrupole mass analyzer, networking, power grid and power distribution, high voltage DC power sources), and they are displayed in separate tabs, which make up Zone number 2. Each tab contains a graphic illustration or process diagram of the sub-process in hand; making it easier to identify the location of the different sensors and actuators, as well as the different systems and components that take part is the construction of the complex sputtering equipment. There are a total of 12 tabs, featuring a main tab containing a user-configurable chart or trend for the various system parameters, with the help of which the user can monitor the variation in time of the different measurements, helping in identifying patterns and dependencies between the process parameters.

The last, and possibly one of the most important features is the local data-archiving. The application creates automatically three types of data tables on the hard drive of the process computer: event log, scan log and process log. The first type contains time stamped system events, states, faults, errors, system shutdowns in chronological order as a result of background data processing. There are well over 100 different event types registered. The log itself has proven to be helpful in case of equipment troubleshooting. The second type is generated only when there is a valid scan result regarding the composition of the gas mixture inside the sputtering chamber. The third type of log contains measurements and derived values. Every 200ms a new entry is created with over 35 selected system and process parameters/states. The application is archiving locally at a rate of 1kB/s. The event and process logs are created 24 hours a day and contribute to the redundant data-archiving presented earlier. These structured logs are indispensable in the offline data processing, based on which mathematical models are created and fine-tuned with the goal of better understanding the complex phenomenon during a sputtering process.

6. Conclusion

The developed multilevel control system has proven to be a valuable asset in the comprehensive management of the complex reactive magnetron sputtering process and of all its subsystems, providing a stable environment for both online and offline data processing, as well as for the control and the monitoring of various interdependent process parameters and states.

Taking advantage of the fact, that the distributed system can be easily updated by connecting online new systems to it, the developed embedded control units, as well as the multilevel architecture ensures a modular and expandable platform. The ideas regarding the presented software, network topology and the implemented electronics can be used for the control of other complex processes.

The distributed system incorporates a total of 11 embedded control and data acquisition units developed over the last 6 years, monitoring 30 process parameters and measurements, over 80 electronic system states and parameters, approximately 200 systems event messages and controlling a total of 18 processes and system parameters combined. The constantly improving thin film structure and composition properties, as well as the increased process reproducibility justify the necessity and the usefulness of the system in discussion.

It is important to emphasize that the system developed can serve educational purposes as well, granting a fully functional environment, where multilevel control systems, embedded hardware, microcontroller programming, data acquisition, supervisory control and remote management can be tested under real circumstances.

Acknowledgements

The authors thank the members of the Thin Solid Films Research Group from the Sapientia Hungarian University of Transylvania, Domokos Biró, Sándor Papp and István Szöllösi for their indispensable contributions to the development of the research work presented.

References

[1] Westermo Handbook 5.0, “Industrial Data Communication – Theoretical and General Applications”, Westermo, Sweden, 2004.

[2] Westermo Handbook 5.0, “Industrial Data Communication – Industrial Ethernet”, Westermo, Sweden, 2004.

[3] Kugelstadt, T., “Protecting RS-485 Interfaces Against Lethal Electrical Transients”, in *Application report, SLLA292A-May 2009-Revised March 2011*.

[4] *** TCP/IP application note: Microchip TCP/IP Stack Help (version: 5.42.08 - 2013).

[5] Fekete, A. Zs., Jakab-Farkas, L., “Development of an Embedded System for Accessing Mass Spectrometry Measurements through Ethernet Network”, *in Proceedings of the XXIth International Scientific Conference of Young Engineers*, Cluj-Napoca, Romania, March 17-18, 2016, pp. 161-164.

[6] Fekete, A. Zs., “Automation of the reactive magnetron sputtering process”, *in Proceedings of the XVIIth International Conference of Technical Sciences*, Cluj-Napoca, Romania, November 26, 2016, pp. 79-84.

[7] Kopetz, H., “Real-Time Systems: Design Principles for Distributed Embedded Applications”, Second Edition, Springer Science + Business Media, 2011.

[8] Jonsson, L. B., Nyberg, T., and Berg, S., “Target compound layer formation during reactive sputtering”, *J. Vac. Sci. Technol. A 17(4)*, Jul/Aug 1999, pp. 1827-1831.

[9] Görgy, K., „Cercetări privind dezvoltarea unor electrotehnologii pentru depunerea straturilor metalice subţiri”, *Teză de doctorat, Universitatea Tehnică din Cluj-Napoca*, 2010.

[10] Berg, S., Katardijev, I. V., „Preferential sputtering effects in thin film processing”, *J. Vac. Sci. Technol. A 17(4).*, Jul/Aug 1999, pp. 1916-1925.