MONITORING SYSTEM FOR GRINDING MACHINE OF TURBINE-ENGINE BLADES

Wojciech Batko, Witold Cioch, Ernest Jamro

University of Science and Technology (AGH-UST)
al. Mickiewicza 30, 30-059 Kraków, Poland
tel.: +48 12 6173622, fax.: +48 12 6332314
e-mail: cioch@agh.edu.pl

Abstract

Monitoring operation of grinders of aviation-turbine blades is especially vital for the demanded product quality and economic expenses associated with defected products and production stoppages. In technological process of grinding, not always even wear of tools occurs and machine tools work definitely in non-stationary conditions. Self-induced vibrations are often observed. Therefore monitoring system of module structure was designed, dedicated to non-stationary signal processing. The system is composed of modules for recording and preliminary hardware signal processing, database servers and user's terminals.

Keywords*: monitoring, diagnostics, grinding machine,*

1 Introduction

Despite varied production technologies, machining is still commonly applied nowadays. It is the result of the fact that it provides a very high accuracy, high efficiency and could be easily automated. Perfect examples of devices carrying out machining process are grinding machines for aviation turbine blades.

Problems regarding grinders operations can have various causes. They can be associated with machine-tool defect, wear process of a tool or self-induced vibrations [2]. Hence it is advisable to diagnose machine-tool condition not only before machining process but also during the actual process. The condition of machine tool is vital for the product quality and for continuity of production process, the stoppage of which would bring significant economic losses.

Monitoring systems enable to detect changes in condition or operation parameters of machine tools [1]. Based on data collected, it is also possible to project the technical condition, which is important for production-process planning. It is of great significance in aviation industry, where manufactured elements are of high quality and precision. Therefore their production is expensive and loss of the whole batch of product is not acceptable.

Machine tool condition and machining process characteristics are vitally affected by dynamic phenomena. Hence vibration is a basic quantity measured in a monitoring process. Vibration is also measurement quantity bringing the most valuable diagnostical information [5]. Monitoring tool condition and self-induced vibrations is still an unsolved problem.

The most significant features, determining functional properties of monitoring systems are according to [4]:

- purpose (machine tool, machining type),
- type of selected diagnostical signals and their measurement manner,
- signal transformation methods,
- method of determining boundary values,

- diagnostical inference methods,
- maintenance characteristics, interfaces.

Having those items in mind, an innovative monitoring system for grinder machine of aviation turbine blades has been designed.

2. Monitoring system

The monitoring system for grinding machine includes methods and algorithms for signals acquisition, pre-processing, transmission and data processing and storage. Consequently, the whole system consists of the following modules:

- Programmable Unit for Diagnostic (PUD)
- Dedicated Server for data collection and processing
- Dedicated server for data storage with limited access for users.
- External users terminals for diagnostic signals analysis.

The block diagram of whole system is presented in Fig. 1.

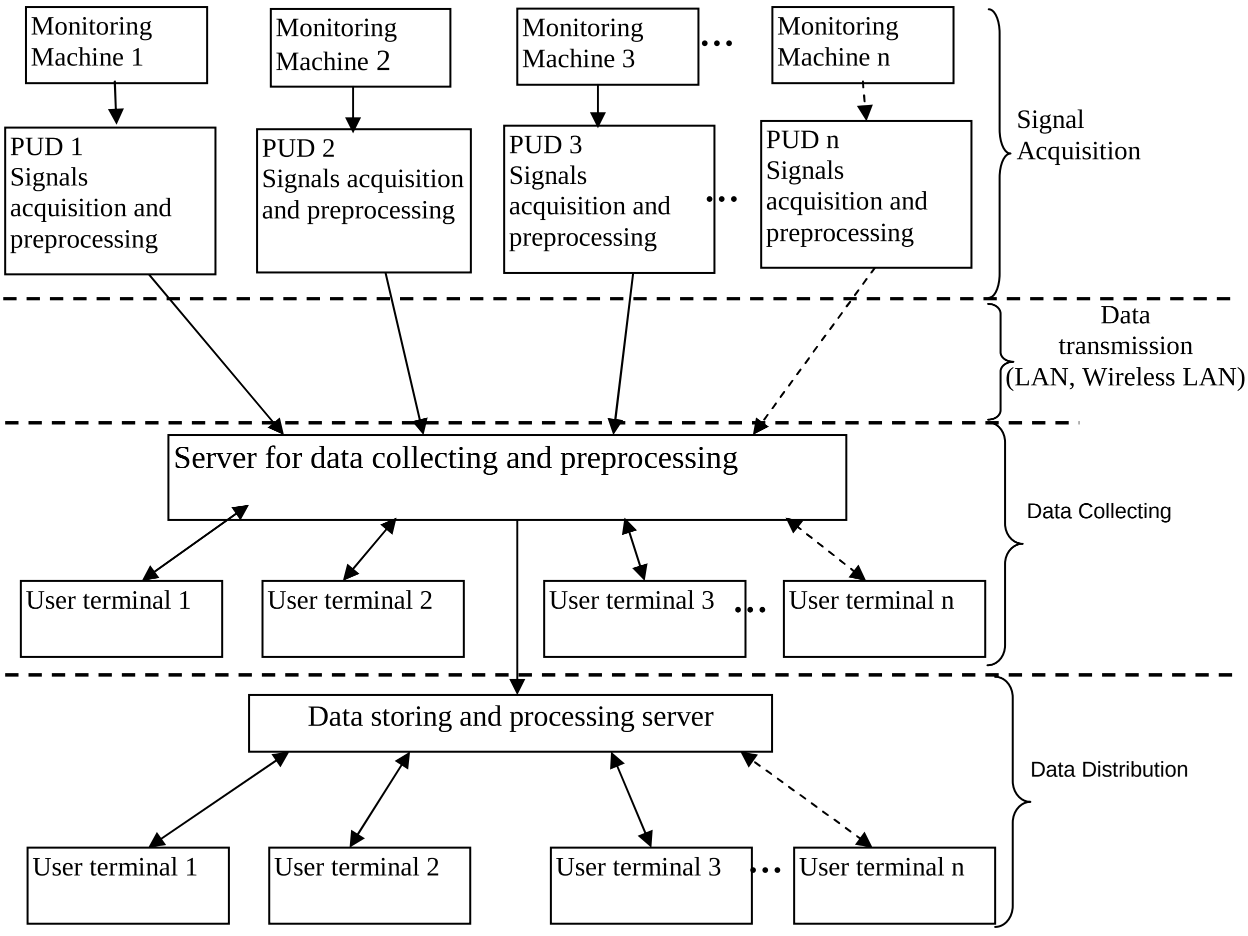

Fig. 1. The monitoring system block diagram

3. Signals acquisition and preprocessing

Electronic implementations of signals acquisition and preprocessing are often disregarded in academic considerations. Nevertheless they often determine the diagnostic procedure and final results. Therefore in this section the electronic solutions for high-speed diagnostic devices will be considered.

Usually three different electronic solutions are considered:

1) General-purpose processors or digital signal processors (DSP)
2) Dedicated VLSI chipset – Application Specific Integrated Circuit (ASIC)
3) Field Programmable Gate Arrays (FPGAs)

General-purpose processors (e.g. Pentium) or DSPs are very flexible, easy to be programmed and relatively cheap, nevertheless the signal sampling frequency and signal processing speed is relatively small. Consequently this solution is suitable for relatively slow diagnostic systems, for which signal-sampling frequency is roughly below 100kHz. The given threshold frequency differs for different algorithm complexity, number of channels, processor computing power and so on, therefore it should be regarded only as a rough number.

Dedicated (ASIC) solution is the best solution for mass-production devices for which the solved problem and algorithm is stable. This holds as device designing time and initial costs are high, but device per unit price is relatively low. Unfortunately, diagnostic devices are usually used only by specialists and therefore are produced in low-volume. Besides, the diagnostic algorithms are often adapted to different problems. This causes that this solution has rather only historical significance.

For diagnostic systems for which data sampling frequency is above roughly 100kHz the FPGAs (Field Programmable Gate Arrays) solution is the best one. FPGAs can be programmed by an end-user in similar way as microprocessors, nevertheless they employ different design procedures, which causes that they can be designed only by electronics engineers. For FPGAs, sampling frequency is up to roughly 500 MHz; similarly data processing speed is also very high.

The most important feature of FPGAs is programmability, i.e. logic or arithmetic functions performed by FPGAs can be programmed according to user requirements. Therefore FPGAs resemble microprocessors. The most important difference is that microprocessors fetch instructions from external memory, and FPGAs have built-in configuration memory. Consequently, microprocessors waste time for fetching and decoding instructions, which limits their speed and level of parallelism; nevertheless the executed program can be relatively easily changed. Conversely, FPGAs can execute user functions relatively quickly and in parallel, as they do not waste time for fetching and decoding instructions.

The main disadvantage of FPGAs is that they can execute a limited number of logic (instructions) at a time and changing logic functionality (reconfiguring FPGAs) is time consuming. For microprocessors, the size of a machine instruction is usually about 1-8 Bytes. Conversely, for FPGAs configuration size is e.g. roughly 50 kB (Xilinx XC3S50) ÷ 1.5MB (XC3S5000). Consequently, FPGAs reconfiguration requires relatively large amount of time (about 10-1000 ms) when FPGAs cannot execute any logic. Besides storing many different FPGA configurations is also difficult to be obtained because of large memory size. Summing up, FPGAs execute very quickly (in parallel) a limited number of instructions, but branching to another set of instructions (reconfiguring the FPGAs) is usually unacceptable. Consequently, it is recommended that FPGA perform only instructions that are executed in loop millions of times. Usually this is satisfied for data-driven algorithms, for which the same relatively simple algorithm is executed for a great number of times for different data.

Hardware software co-design

FPGAs and microprocessors complement each other, i.e. usually the most computationally intensive algorithms are data-driven algorithms, therefore they can be speed-up by FPGAs. On the others hand, program-driven algorithms, i.e. complex algorithms processed a limited number of times, cannot be easily implemented in FPGAs. Usually these algorithms are not computationally intensive and therefore are well suited for microprocessors. Partitioning an algorithm into software part (executed by microprocessors) and hardware part (executed by logic incorporated in FPGAs) is denoted as hardware-software co-design [49]. As hardware-software co-design is very efficient, FPGAs are often connected with microprocessors, i.e. two different chips (FPGA and microprocessors) are incorporated on the same PCB (Printed Circuit Board). Furthermore, FPGAs often incorporate microprocessors in the same chip, therefore combining hardware and software is more effortless.

Embedded Development Kit (EDK)

FPGA design cycle is relatively difficult and time consuming. Almost every engineer can program microprocessors employing C/C++ or other high-level programming languages. Unfortunately, FPGAs can be designed only by limited-number of electronic engineers. Besides design cycle is difficult, error-prune and time-consuming. And last but not least, finding errors (debugging) FPGA designs is a major drawback, which is often overlooked. Therefore a dedicated Advance Programmable System Interface (APSI) [26] was employed in the developed system.

In order to speed-up FPGA design cycle, modular design is often adopted. Modules, denoted as Intellectual Property (IP) cores, which functions are well defined and tested, are supplied by different vendors and connected with each other. Each module is in charge for different tasks, e.g. external SDRAM memory interface, Analog Digit Converter (ADC) interface. To speed-up modular design Xilinx Embedded Development Kit (EDK) was employed. This software packet allows connecting different modules graphically. Consequently adding a microprocessor and employing hardware/software co-design is significantly quicker and easier. An example of a Huffman compression system design in EDK is given in fig. 2.

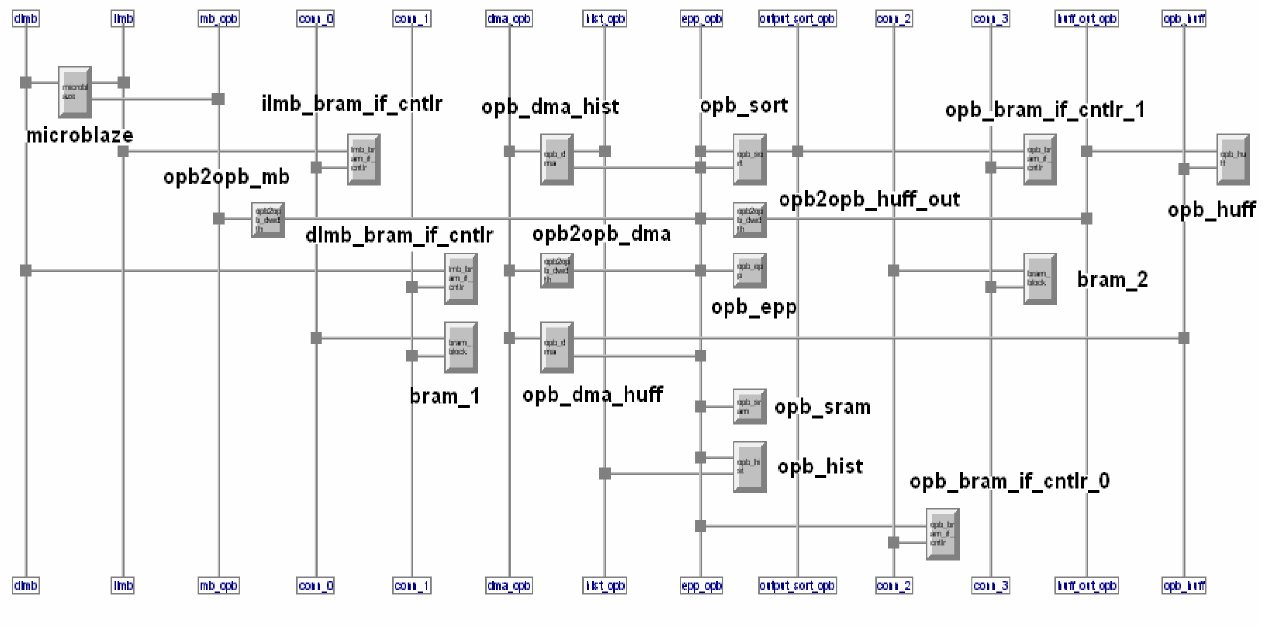

Fig. 2. An example of EDK design [27]

Programmable Diagnostical Unit (PDU)

Programmable Diagnostical Unit (PDU) is a device which core is FPGA device [6]. The block diagram of the PDU is shown in Fig. 3 and it incorporates the following parts:

- Four independent analog / digital modules on separate PCB.
- Two independent SDRAM memory banks, 64MB each, employed to store acquired data from analogue / digital modules and other temporal data.

- CPLD (Xilinx XC95144XL device) – module employed to configure FPGA and to control the PUD in power stand-by mode.
- Flash memory (4MB) to store FPGA configuration, MicroBlaze program and other non-volatile data.
- Hard Disk Drive (HDD) to store high volume data
- LCD display employed to visualise the state of the device and results for acquired data.
- Keyboard – allows user to control the PUD and to start / stop data acquisition.
- PC computer communication by Parallel or Serial Ports.

Besides, the PDU incorporates some optional devices: Compact Flash memory, VGA display, PC keyboard, and Ethernet and Radio Communication modules.

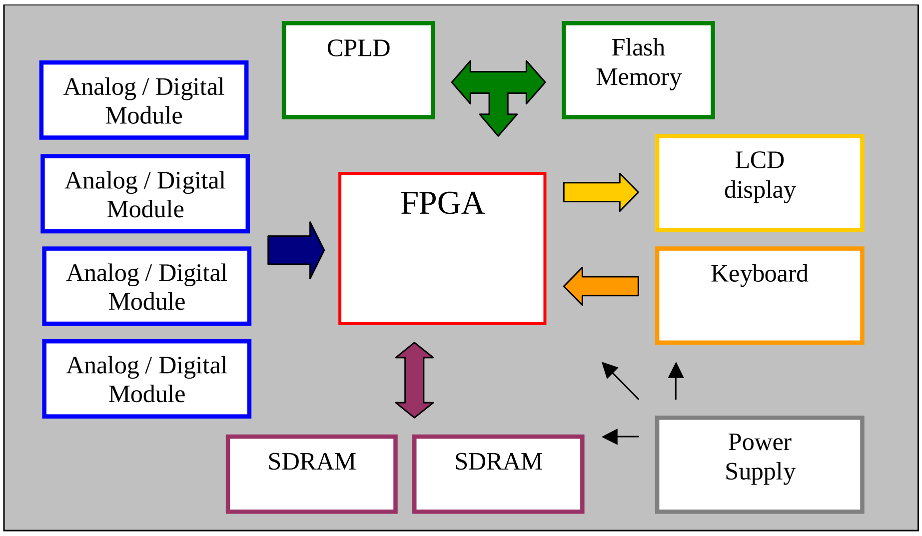

Fig. 3 Block diagram of the PDU

The PDU incorporates four independent analog / digital boards. These boards include Analog Digital Converters (ADC), input signal amplifiers with digitally controlled level of amplification and ICP sensor interface. Each analog board incorporates 2 channels 16-bit 500kS/s each or 4 channels 16-bit 250kS/s each, consequently up to 8 (16-channels) can be acquired by the PUD. It should be noted that changing analog / digital board requires FPGA configuration to be changed. Fortunately modular design in EDK significantly reduces designing time.

Data servers

To increase data security and to improve data visualization, data are stored on two independent servers. On the first server, data coming directly from the signal acquisition unit are collected. Then they may be preprocessed in order to decrease the data volume and to select only important diagnostic estimates. Then preprocessed data are stored in a local database and are transferred to the second server. The main task of the second server is data backup and data distribution as a database or by HTTP protocols on WWW. As a result end users can access data by users terminals, which need not advance diagnostic programs. This solution allows controlling data access to a limited number of users. Besides any data access or modification can be recorded in the system.

In the database apart from acquired signals some additional information, as acquisition time, sampling frequency and amplifier settings may be stored. The database allows also storing some additional information as an exact place of signals acquisition, grinding machine information and users comments. The database allows to group selected signals as a result of acquisition time, data processing type, etc. It allows also to process recorded signals by externals programs such as MATLAB.

5. Conclusions

Application of Programmable Diagnostical Unit into preliminary signal processing system enable to record data with high frequency and data hardware processing [7]. Using programmable FPGAs brought the possibility of non-stationary signal analysis in real time. The applied PDU module enabled, apart from measuring diagnostical signals, also to record parameter signals, especially rotation speeds of spindles.

Application of two servers for data collecting and for data distribution separately improved the safety of database.

It also gave the vast possibilities of processing (implementing Matlab suite) and surveying of collected signals by authorised users. Thanks to this solution, classifying grinding-machine condition and diagnosing operational process can be carried out by means of:

- numeral measures,
- functional measures,
- neural networks,
- parametrical models.

The modular construction of the monitoring system allows the whole system to be easily extended or modified to new grinding machines and acquisition points.

6. References

[1] Batko, W., Korbiel T., *System monitoringu diagnostycznego przekładni planetarnej pracującej w warunkach zmiennego obciążenia,* Diagnostyka nr 1, 2006.

[2] Pod red. Batko, W., Dąbrowski, Z., *Procesy wibroakustyczne w technice i środowisku*, Wydawnictwo Wydziału Inżynierii Mechanicznej i Robotyki, AGH, 2006.

[3] Bodnar, A., *Vibration Monitoring on a Milling Machine Using the Analysis of Stochastic Features of Generated Noise,* Ninth World Congress on the Theory of Machines and Mechanisms, August 29/September 2, 1995, Milano, Vol. 4, s. 2910-2914.

[4] Bodnar, A., *Diagnostyka i nadzorowanie drgań obrabiarek***.** Materiały Sekcji Podstaw Technologii Komitetu Budowy Maszyn PAN z 50 posiedzenia w dniu 24.06.93 w Szczecinie, W: Prace Inst. Technologii Mechanicznej Politechniki Szczecińskiej, 1993, s. 5-16.

[5] Cempel, Cz., *Diagnostyka wibroakustyczna maszyn*, Wydawnictwo Politechniki Poznańskiej, Poznań, 1985.

[6] Jamro, E., Adamczyk, A., Krzyworzeka, P., Cioch, W., *Programowalne urządzenie diagnostyczne stanów niestacjonarnych pracujące w czasie rzeczywistym*, XXXIII Ogólnopolskie Sympozjum Diagnostyka Maszyn, Węgierska Górka, 8.-11.03. 2006 r.

[7] Krzyworzeka, P., Adamczyk, J., Cioch, W., Jamro, E., *Monitoring of nonstationary states in rotating machinery,* Instytut Technologii i Eksploatacji, Radom 2006.

This work has been executed as part of research project at KBN no 6T0720005C/06545.