Military Technical College
Kobry El-Kobbah,
Cairo, Egypt

9th International Conference
on Electrical Engineering
ICEENG 2014

Multi-Agent Data Fusion System Based on Multiple–FPGA Parallel Computing Architecture

By

Assoc. Prof. Ahmed E. Abdalla* Dr. khaled M. Shafie * Eng.Mohamed F. Tawfic*

Abstract:

The multi-agent system concepts appeared recently and it is extremely distributed in all research areas to solve problems by many agents cooperation. In some applications, quick, accurate, and complete data is highly required for supporting decision making in order to reduce the decision cycle and to minimize the loss. Multi-agent data fusion has been used for such applications where the process of integration of multiple data and knowledge is turned into a consistent, accurate, and useful representation. The benefit of using multi-agent data fusion is that it can use for large structural, collect as much data as possible using different kinds of sensors with low cost. This paper presents a multi-agent cooperation system using a real time processor based on FPGA which is used as parallel processing to speed up the processing, measuring the time and the amount of data being processed. This makes real-time or near-real-time damage detection possible. The proposed multi-agent data fusion system is evaluated by a bridge aluminum structure monitoring experiment in which strain distribution are monitored by a set of sensors.

Keywords:

Multi-Agent System; Data Fusion; Field Programmable Gate Array; Structural Health Monitoring.

---

* Electrical Eng. Dept. Military Technical College.

1. Introduction:

Agents and Multi-agent system (MAS) over the past few years realized as essential technology for effectively handling the increasing availability of various, mixed and distributed online information sources, and also as a framework for building large, complex and reliable distributed information processing systems which take advantage of the efficiencies of organized behavior [1].This will bring extra conceptual power, new methods and techniques to our computer applications.
Nowadays, MAS technology is being used for a wide range of control applications including intelligent systems that analyze data to provide data management and online diagnostic information[2]. For example a modern distributed control systems perform various tasks including online condition monitoring, control, and protection [3, 4].
One of the important application go to the direction of the development of structure health monitoring system (SHM) where the studying of active monitoring methods for damage detection are applied. In some applications, SHM architecture are based on multi-agent wireless sensor network (MAS-WSN) technology [5]; While, others present a multi-agent cooperation system based on mobile agent technology to migrate among different sensor nodes and perform network self-organization data processing, information fusion and damage estimation [6].
While, others focused on the coordination in multi-agent system as NASA's goal of an aerospace vehicles which requires the development of vehicles that are capable of structural self-assessment and repair. These functions can be carried out by distributed sensors, intelligent processing and communication within the structure [7]. Intelligent software agents for distributed machinery health monitoring and diagnostics have been developed to continually monitor machinery, to identify future failures and to precisely predict its remaining useful life time [8].

This paper presents a general framework for the design of a multi-agent data fusion system based on multiple–FPGA parallel computing architecture Where, a parallel processing used to speed up the processing the time.

The outline of this paper is organized as follows, Section 2, gives an introduction to Multi-Agent Data Fusion System (MADFS), usage and its benefits. Section 3, discusses the development of multiple FPGA parallel processing. Section 4, presents the architecture of MADFS based on multiple FPGA parallel processing. Section 5, presents case study to discuss how to apply the proposed architecture on SHM system.
Section 6, introduces the analysis and performance evaluation of the results. At last, Section 7, gives a discussion and conclusion of this work.

2. Multi-Agent Data Fusion System:

An agent is anything that can make out its environment through sensors and act upon that environment through actuators .However, agents are often stand-alone systems. In many situations they coexist and interact with other agents in several different ways. Such a system that consists of a group of agents that can potentially interact with each other is called a multi-agent system (MAS).

The concept of Data fusion, is the process of integration of multiple data and knowledge representing the same or different real-world object to get more accurate and reliable information, can be applied on MAS to have what is called multi-agent data fusion system (MADFS).

MADFS is important in some applications including SHM, environment monitoring, automatic target detection and tracking, battlefield investigation, remote sensing and global awareness. They are usually time-critical, cover a large area, and require reliable delivery of accurate information for their achievement.

The Proposed general architecture of MADFS can be established in four basic levels as shown in Figure (1). First level, the sensors are embedded in or bonded to the sub-system to sense there parameters (data monitoring layer). Second level, appropriate signal or information processing methods are adopted to fuse and analyze data and extract the features sensitive to the sub-system failure from the sensed data (data interpretation layer). Third level, the local status of the sub-system can be deduced using corresponding failure evaluation methods. Fourth level, inform the user about the status and supporting decisions. These levels can be considered as four layers forming the MADFS. Where, they are data monitoring layer, data interpretation layer, information analysis and decision support layer and user interface layer.

3. Development of Multiple FPGA Parallel Processing:

Field programmable gate arrays (FPGAs) are emerging in many areas of high performance computing, either as tailor made signal processor, embedded algorithm implementation, software accelerator or application specific architecture. As shown in Figure (2), [9] the FPGA chip are comprised of different number of logic elements, interconnects, IOBs (Input/output blocks). All these components are user-configurable. It can help in implementing complex digital circuits.
FPGAs are so flexible and reconfigurable that they are capable of massively parallel operations, explicitly tailored to the problem at hand. There are a lot of paradigms to put FPGAs at work in a high performance computing environment, they provide the possibility of changing the hardware design easily more than software implementations on general purpose processors.

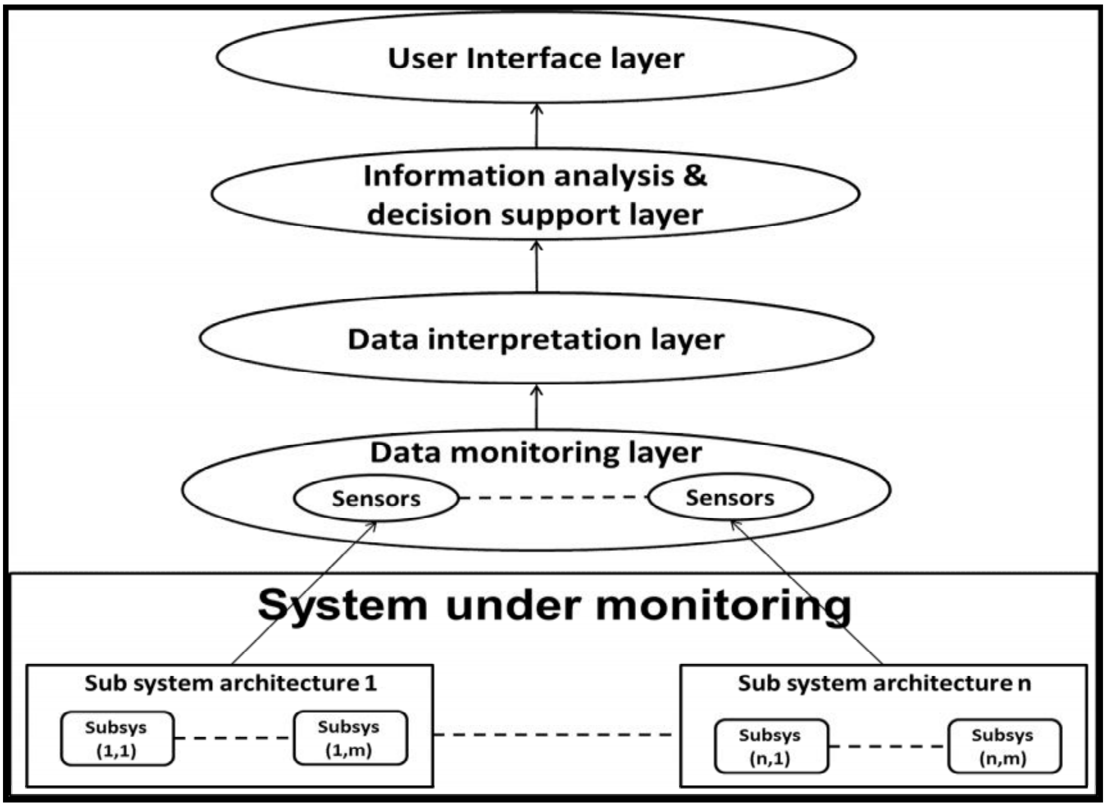

Figure (1): *Proposed general Architecture of MADFS*

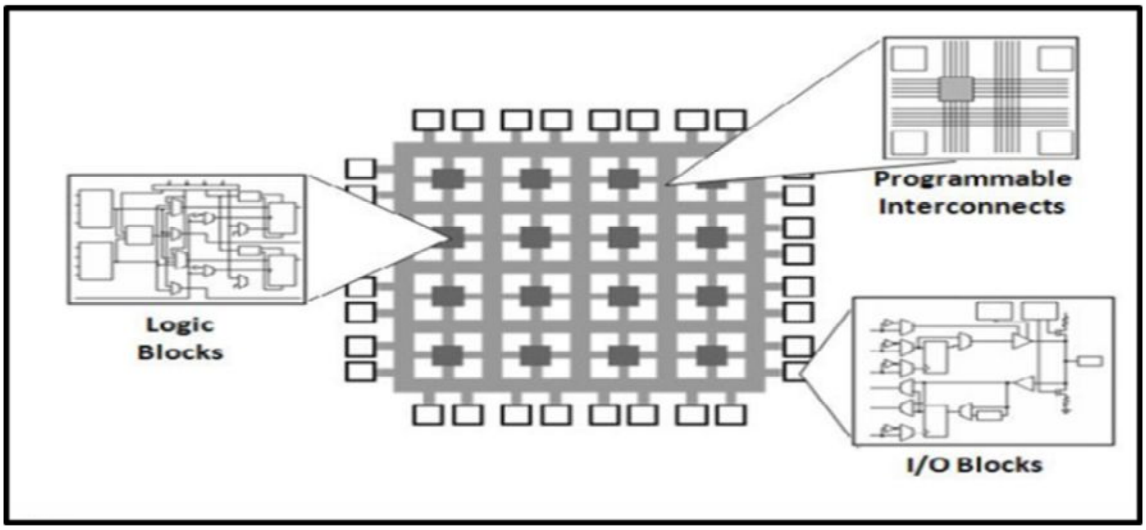

Figure (2): *FPGA internal structure composed of configurable logic and I/O blocks tied together with programmable interconnects [10].*

High performance reconfigurable computing (HPRC) are parallel computing system that allows multiple FPGAs embedded into it. HPRC has the ability to exploit inherent parallelism and match computation with application data. Reconfigurable logic consists of programmable computational matrix with a programmable interconnected network which is used within that computational matrix. There are the basic differences between reconfigurable logic and traditional processing which are [11] :

- Spatial computation: the data is processed by spatially distributing the computations rather than temporally sequencing.
- Configurable data path: the functionality of the computational units and the interconnection network can be adapted at run-time.
- Distributed control: the computational units' process data based on local configuration rather than an instruction broadcast to all the functional units.
- Distributed resources: the required resources for computation such as computational units and memory.

4. Architecture of MADFS Based on Multiple FPGA Parallel Processing:

The whole Proposed architecture of MADFS based on multiple FPGA parallel processing is shown in Figure (3). The structure, is divided into some substructures.

In the architecture, the facilitator, Where agents may register their services with it or query to find out which agent can provide the service they need, provides services to every agent and it is unique in every subsystem..

Each FPGA is responsible for a subsystem, processes the raw different kinds of data, extracts the feature related to the damage, and evaluates the damage information such as damage kind, damage area and damage degree. Then, every FPGA inform the manager, which in turn combine the evaluated data from all FPGAs to provide the user with the analysis and decisions.

To apply MADFS based on multiple FPGA parallel processing technology, each subsystem represents an agent. According to the subsystem’s functions, Agent are defined as follows:

1. Sensing agents: responsible for monitoring structure parameters, such as stress, stain, displacement, acoustic, pressure, and temperature.

2. Signal processing agents: responsible for processing data monitored by sensing agents and extracting signatures to estimate the structure state.

3. Estimation agents: responsible for estimating the structure state or life time using signatures extracted by signal processing agents.

4. Fusion agents: data from different sensor agents are processed by different signal processing agents and estimated by different estimation agents. Data from the same sensor agents can also be processed using different signal processing agents and estimated by different estimation agents. Thus fusion agents are required to fuse all the results from different methods to obtain the most reliable and precise conclusion. There are 2 types of fusion agents:

   1- Local fusion: Found in the subsystem, it fuses locally the data results from DEA.

   2- Central fusion agent: Fuses the monitoring results from every local fusion agent and gives a global estimation result of the sub structure to the manager.

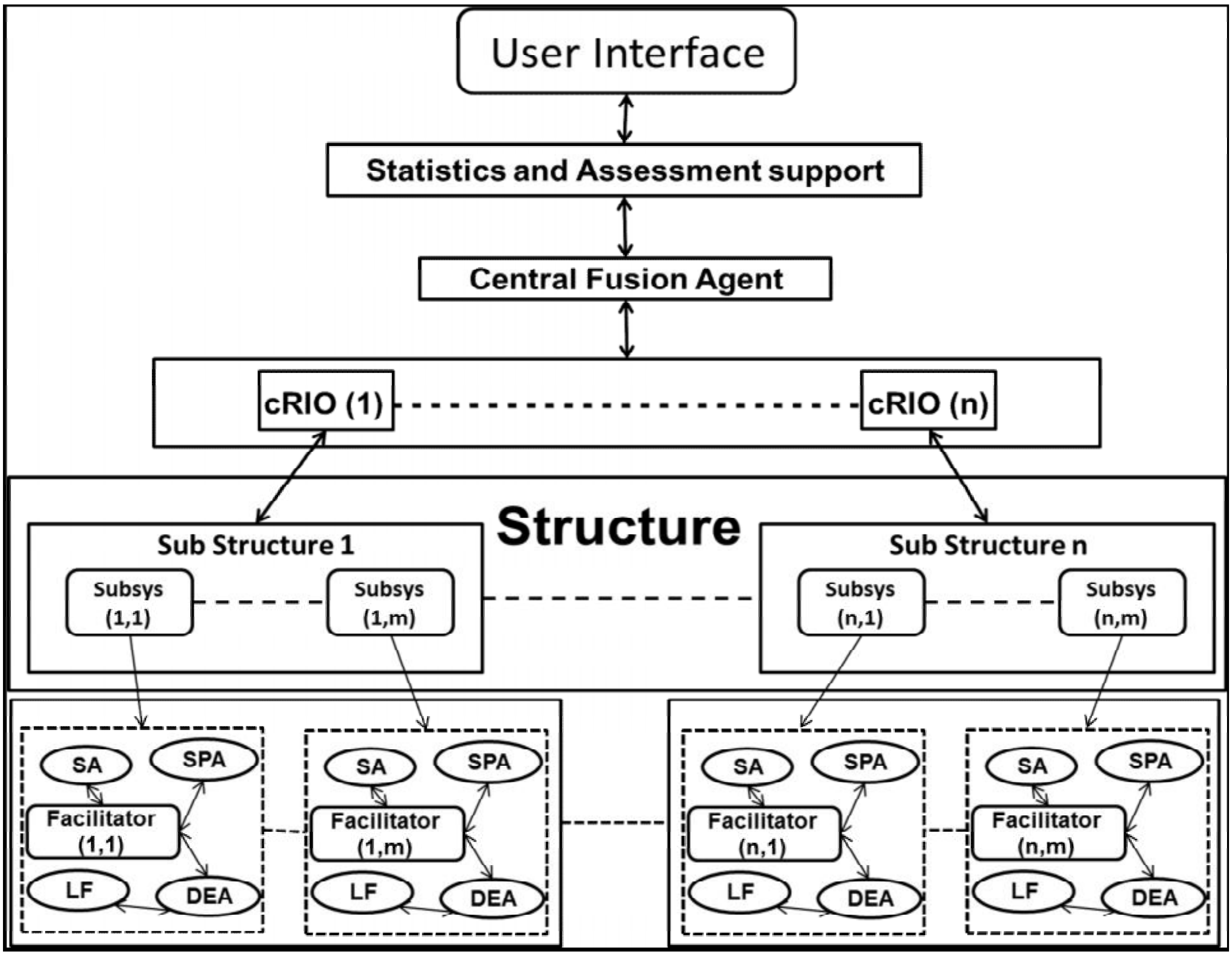

Figure (3): *Proposed Architecture of MADFS based on multiple FPGA parallel processing*

5. Case Study:

In order to verify the effectiveness of MADFS, in this paper, an aluminum plate structure is simulated to study as experimental object. The sensor array is adopted to monitor typical structure damages such as strain distribution change.

5.1. System Setup:

In Figure (4), it gives the plate structure and the sensor distribution diagram, and the photo of the structure. The experimental plate simulates an aerospace hard aluminum and the sensors arrangement. The subsystem is an 80 cm X 80 cm aluminum plate with 2 mm thick, around the structure, there are 4 legs. The subsystem divided averagely into 9 sub-areas, labeled from 1 to 9. The dimension of every sub-area is 23.3 cm X 23.3 cm.

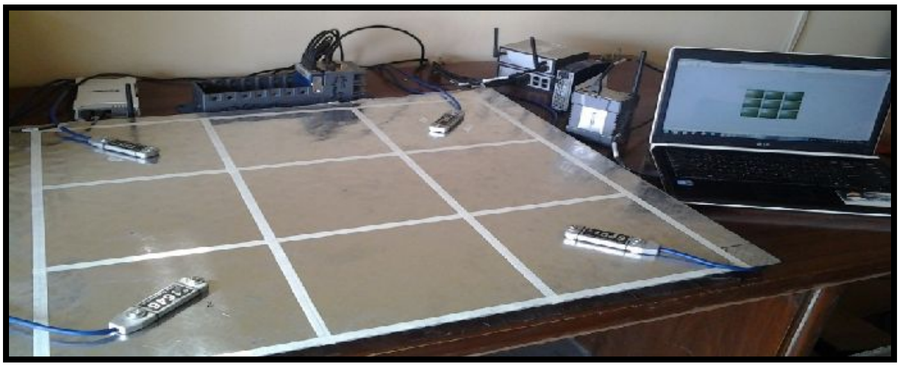

Figure (4) : *System Setup*

A strain gauge sensor, see Figure (5), is applied to each sub-areas (1-3-7-9) to monitor the concentrated load applied to the subsystem.

Figure (5) : *Strain Gauge Sensor*

Figure (6-a), The NI cRIO-9014 embedded real-time controller features an industrial 400 MHz Free scale real-time processor for deterministic, reliable real-time applications. It contains 128 MB of DRAM memory and 2 GB of nonvolatile storage. The cRIO embedded controller is designed for extreme ruggedness, reliability, and low power consumption with dual 9 to 35 VDC supply inputs that deliver power to the cRIO chassis/modules and a -40 to 70 °C operating temperature range, so it can function for long periods of time in remote applications using a battery or solar power. Also, there is a fault-tolerant file system embedded in the cRIO-9014 that provides increased reliability for data-logging applications. The cRIO-9014 runs the NI LabVIEW Real-Time Module on the VxWorks real-time operating system (RTOS) for extreme reliability and determinism.

Figure (6-b), The NI cRIO-9104 is an eight-slot, 3M gate reconfigurable embedded chassis that offers ultimate processing power and the ability to design custom hardware using NI LabVIEW software. The cRIO-9104 provides low-level hardware access to any cRIO I/O module that can create unprecedented timing, triggering, and synchronization schemes. For best performance, National Instruments recommends the cRIO-9014 when developing applications for the cRIO-9103/9104 3M gate reconfigurable chassis [12, 13].

(a) : *cRIO-9014*

(b) : *cRIO 9104 chassis* ***(c) :*** *modules*

Figure (6) : *NI cRIO*

Figure (6-c), I/O modules contain isolation, conversion circuitry, signal conditioning, and built-in connectivity for direct connection to industrial sensors/actuators. By offering a variety of wiring options and integrating the connector junction box into the modules, there more than 70 NI C Series I/O modules for CompactRIO to connect to almost any sensor or actuator. Module types include thermocouple inputs, simultaneous sampling, analog I/O, digital I/O and strain measurements.

The measurements taken from changes of strain distribution of the structure are used to illustrate the successful execution of concentrated loading localizations. Four strain gauges are incorporated and adapted to monitor the strain distribution changes caused by applying a concentrated load of weight 10 kg at different positions on the structure. There are total of nine areas at which the load may be applied. The strain gauges are incorporated in area 1, area 3, area 7 and area 9, respectively. The load position monitoring is based on the strain distribution change monitored by sensor agent. The four outputs of the strain gauge nodes represent the strain distribution of the structure. When the concentrated load position changes, the strain distribution changes correspondingly, and the output of the strain gauge nodes changes too. Since the Euclidean distance measure is simple and easy to implement, it is chosen to run for pattern recognition to decide different load positions [14, 15] .This used to process the structural responding information and decide the structural health status. The differences between the real-time loading output values and the no-loading values of the strain gauges create the input vector: $x(\theta_1, \theta_2, \theta_3, \theta_4)$, and the trained differences between nine areas' load output and the no-loading output compose the standard difference $y(\propto_{i1}, \propto_{i2}, \propto_{i3}, \propto_{i4}), i \in [0,9]$ The distance between two patterns is calculated using the Euclidean distance equation (1).

$$d(x,y) = \left[\sum_{j=1}^{n} |x_j - y_j|^2\right]^{\frac{1}{2}} \quad (1)$$

Where n is the number of SA, x_j refers to strain monitored by each SA and y_j refers to the reference strain stored in the data base related to different load position. Figure (7) shows the pattern recognition process.

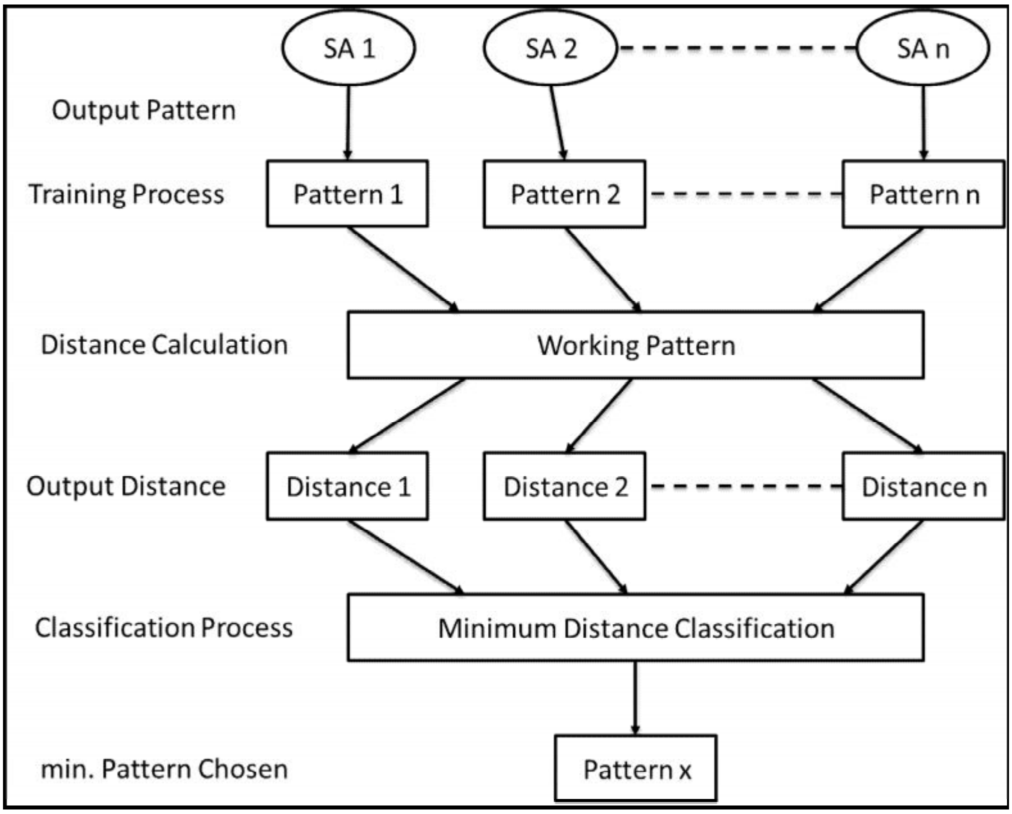

***Figure (7):** Pattern recognition process*

6. Analysis & Performance Evaluation Results:

In the proposed experiment, nine possible load positions areas are numbered from (1-9) as shown in Figure (8). For the strain load analysis, the strain changes of the four strain

sensors are combined to be a feature signature of the system under test at different load position.
A test load of weight 10 kg is applied on area (1), the collected signal waveforms is shown in figure (9).The load was applied for a 10 sec. The Euclidean distance is utilized to classify the calculated features of the strain distribution change. The smaller the minimal Euclidean distance is, the more accurate the diagnostic result of the subsystem with the minimal distance is.

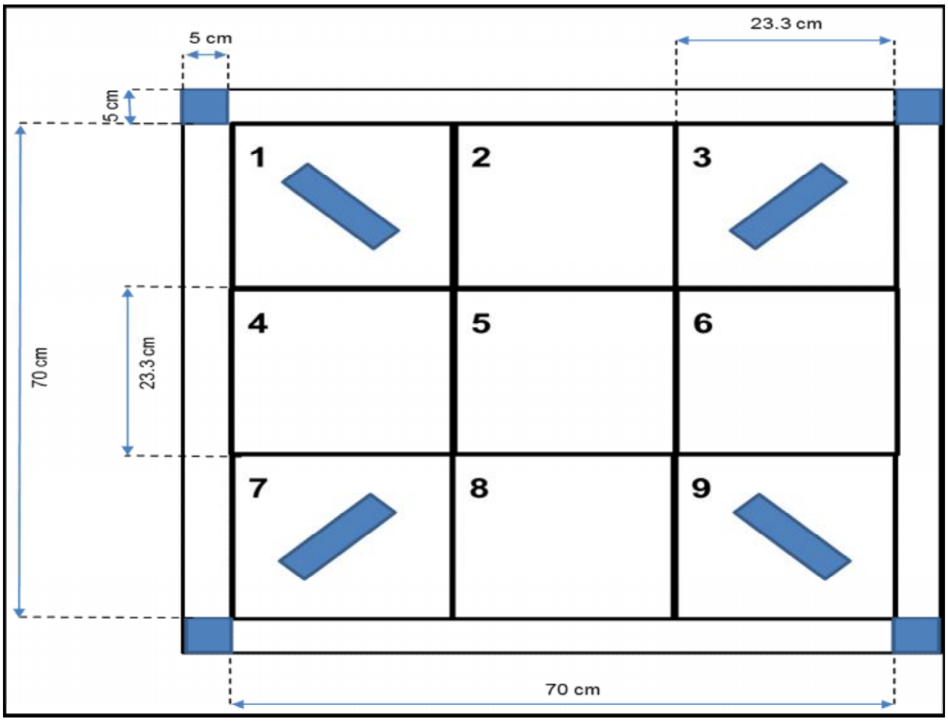

Figure (8): *shows The sensors distribution on the plate*

The proposed system can successfully detect the position of the applied load. The advantage of the proposed system is that by adopting multiple–FPGA used for parallel computing, the signal processing, damage estimation and local fusion all are done in parallel in the real time cRIO, thus reduces the data that needs to be transferred to the computer system. Since each subsystem has its own SPA, DEA and LF all this are done in parallel, thus the system speed is improved.

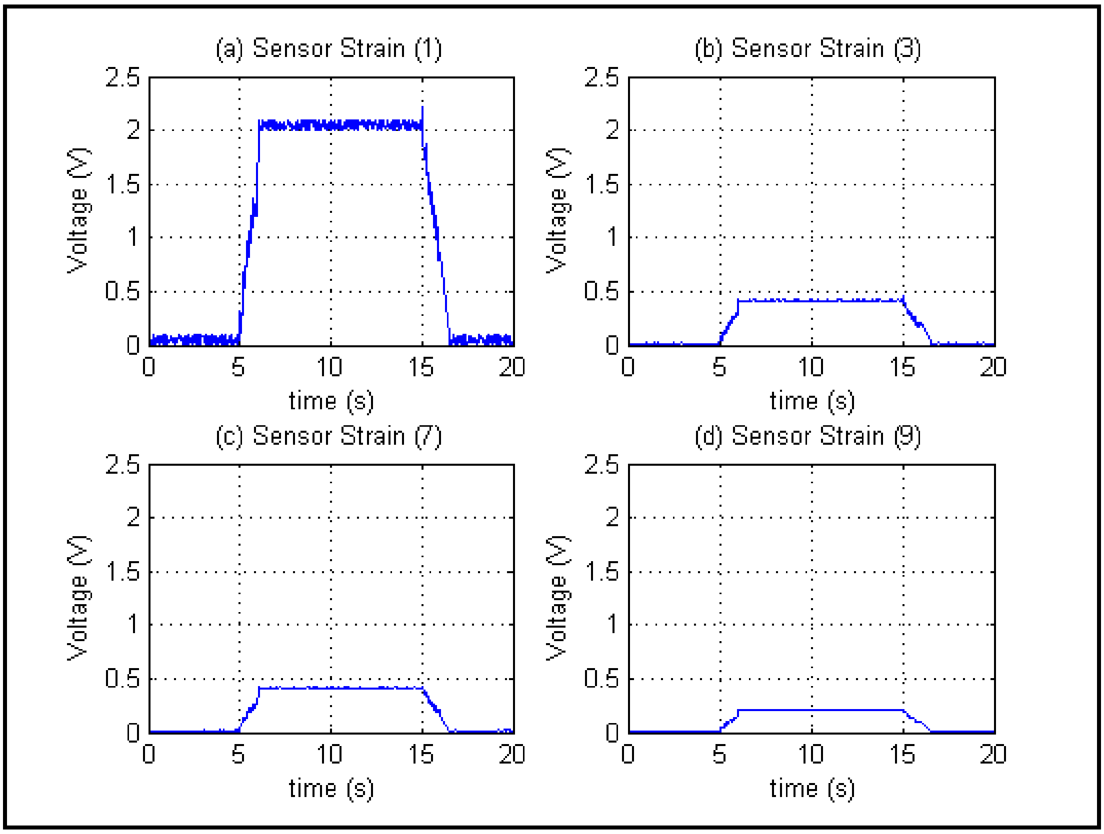

Figure (9): *Sensor strain signal of the plate*

7. Conclusions:

In this paper multi-agent data fusion system based on FPGA is developed.
Multiple FPGA are used as time processors in parallel computing architecture.
Labview program is used to fit the parallel architectures of FPGAs (cRIO) where traditional processors can only process a single instruction at a time and therefore need complex operating systems to schedule parallel tasks.
Another benefit of using the proposed system is one can add more functionality to the system architecture without seriously impacting the performance of the applied applications, this makes using FPGA better than using Microprocessors.
So the proposed multi-agent data fusion system is evaluated by a bridge aluminum structure monitoring experiment in which strain distribution are monitored by a set of sensors. Experimental results show that the system is able to detect early-stage structural damage, prevent disastrous failures, and improve structural management. It is significantly improving the monitoring performance for large-scale structures.

8. References:

[1] Jennings, Nicholas R., "A roadmap of agent research and development," *Autonoumous Agents and Multi-agent Systems*, 1998.

[2] Euan M. Davidson, Stephen D. J. McArthur,,James R. McDonald,,Tom Cumming,,Ian Watt, "Applying multi-agent system technology in practice: automated management and analysis of SCADA and digital fault recorder data," *IEEE Transactions on Power Systems*, 2006.

[3] D. P. Buse , Q. H. Wu, "Mobile agents for remote control of distributed systems," *IEEE Transactions on Industrial Electronics*, 2004.

[4] J. Galdun , L. Tak´aˇc,, J. Ligus,, J. M. Thirie,, J. Sarnovsk´y, "Distributed control systems reliability: consideration of multi-agent behavior," *6th International Symposium on Applied Machine Intelligence and Informatics* 2008.

[5] S. Yuan, Lai, Xiaosong, Zhao, Xia, Xu, Xin, & Zhang, Liang "Distributed structural health monitoring system based on smart wireless sensor and multi-agent technology," *Smart Materials and Structures*, 2006.

[6] S. Yuan, Dong Liang, Lei Qiu, & Menglong Liu "Mobile Multi-Agent Evaluation Method for Wireless Sensor Networks-Based Large-Scale Structural Health Monitoring," *International Journal of Distributed Sensor Networks*, 2012.

[7] Abbott, David et al. , "Development and Evaluation of sensor concepts for ageless aerospace vehicles," *NASA* 2002.

[8] Logan and P. Kevin, "Prognostic software agents for machinery health monitoring," *IEEE*, 2003.

[9] D. Buell, El-Ghazawi, T., Gaj,K.,& Kindratenko, "High-Performance reconfigurable computing," *IEEE Computer Society*, 2007.

[10] JONATHEN ROSE, ABBAS EL GAMAL, ALBERTO SANGIOVANNI-VINCENTELLI, "Architecture of Field-Programmable gate array," *IEEE*, 1993.

[11] K. Bondalapti, Prasanna,V., "Reconfigurable computing systems," *IEEE*, 2002.

[12] http://sine.ni.com/nips/cds/view/p/lang/en/nid/14159.

[13] http://sine.ni.com/nips/cds/view/p/lang/en/nid/203500.

[14] S. Yuan and T. Baoqi, "Research on damage self-diagnostic strength smart structure," *2nd ECCOMAS Conf. on NumericalMethod Engineering (Paris)* 1996.

[15] S. Yuan , L. Wang G. Peng "Neural network method based on a new damage signature for structural health monitoring," *Thin Walled Struct.*, 2005.