

# IMPLEMENTATION ISSUES OF A HIGH-SPEED DISTRIBUTED MULTI-CHANNEL ADDA SYSTEM

<sup>1</sup>Johan M. Wesselink, <sup>2</sup>Arthur Berkhoff, <sup>3</sup>Geert Jan Laanstra, <sup>4</sup>Henny Kuipers

<sup>1</sup>j.m.wesselink@utwente.nl

<sup>2</sup>arthur.berkhoff@tno.nl; <sup>2</sup>A.P.Berkhoff@utwente.nl

<sup>3</sup>G.J.Laanstra@utwente.nl

<sup>4</sup>H.Kuipers@utwente.nl

<sup>1 2 3 4</sup> University of Twente, Faculty of Electrotechnical Mathematics and Computer Science,  
P.O. Box 217 7500 AE, Enschede, The Netherlands  
<sup>2</sup> TNO Science and Industry, Acoustics Department,  
P.O. Box 155 2600 AD, Delft, The Netherlands

## ABSTRACT

A multi-channel ADDA controller is used in many active noise cancellation and active vibration control problems. Such a controller is able to yield good performance, however it also requires a lot of hardware on a centralized place and a lot of sensitive wiring. A practical work around for this problem would be to use a local single channel controller. However such a controller would reduce the overall system performance and may introduce instability. In this paper a system will be presented that acts as a hybrid form and combines the performance of a local feedback loop with a large multi-channel controller. To reduce the wiring and the influence of disturbances on this wiring a local analog to digital and digital to analog converter will be used. These systems will be interconnected using a high-speed serial communication system. To reduce the sample rate for the overall system, a local decimation and interpolation filter will be implemented. Further performance improvements will be realized by means of a simple local feedback system. The implementation issues concerning such a system are the subject of this paper.

## 1. INTRODUCTION

A normal MIMO control system will consist of a centralized controller, a control algorithm and all electronics needed to convert the analog signals. A major drawback of such a system is that it requires a lot of wiring. Another problem is that the signals measured are small and therefore sensitive for electromagnetic interference. A better approach would be to amplify and convert these signals into digital signals locally. The digital signals can then be sent through a high-speed serial interconnection network. In this project a high-end single board PC based computer will be used. The board contains an Intel PentiumM processor running at 2.0 GHz and uses the PCI104 form factor [1]. The aim of this system is to test and evaluate dif-

ferent algorithms and select the most appropriate one. To test the system a simple FXLMS algorithm will be used. The platform will be running realtime(RT) Linux, making software development and testing easier. The PCI104 standard only offers a PCI interface that uses a different form factor (see [2]) or normal PC peripherals. This makes it necessary to develop a dedicated interface card for the PCI104 platform. This interface card will offer a high-speed serial interconnection with the local controller(s) and is implemented in an FPGA. This high-speed serial interconnect can contain up to 16 boards resulting in 16 local controllers and offers a flexible architecture that scales up to 256 AD and DA channels.

A local controller that controls an AD-DA channel or even several channels will be constructed using an FPGA. This local controller will feature a high-speed serial receiver and transmitter, a decimation and interpolator filter and a local feedback controller. The local sample rate is higher than that of the centralized controller. To reduce the sample rate a decimation and interpolation filter is used. The centralized controller is still running at a higher sample rate, it is now however possible to balance the analog input/output filters and digital decimation/interpolation filter, to reduce the overall system delay. Using a controller with a higher sample rate means a faster response and higher bandwidth, this fact can be used in the local controller to implement a fast but simple local controller. The centralized controller uses a lower sample rate for two reasons. Firstly it reduces the computational complexity of the controller and secondly it reduces the communication bandwidth.

The paper will elaborate on the theory of the used control algorithm, the overall system, the communication protocol, the implementation on an FPGA and finally the demonstrator.

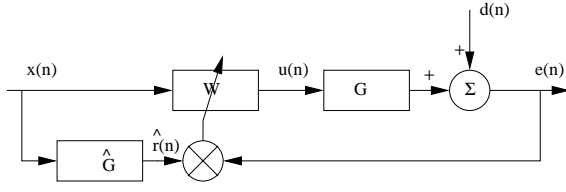


Figure 1: *Feed-forward adaptive controller*

## 2. THEORY

The demonstrator build will be a smart-panel. The purpose of such a panel is to actively reduce the sound that is transmitted through the panel. The realization will be based on an adaptive filter with internal model control in a feedback configuration. The adaptive filter will adopt to changes in the primary signal and path. Resulting in a good performance robustness. The block diagram of a feed-forward adaptive controller can be found Figure 1. The first prototype demonstrator will use the FXLMS algorithm in a multiple-input and multiple-output configuration (MIMO). It is however the intention of this research to test and evaluate different algorithms on the demonstrator. The error signal can be written as:

$$\mathbf{e}(n) = \mathbf{d}(n) + \mathbf{w}^T \mathbf{R}(n) \quad (1)$$

In this formula  $\mathbf{R}(n)$  is the matrix of filtered reference signal and can thus be written as:

$$\mathbf{R}(n) = \begin{bmatrix} \mathbf{r}_1^T(n) & \cdots & \mathbf{r}_1^T(n-I+1) \\ \mathbf{r}_2^T(n) & \cdots & \mathbf{r}_2^T(n-I+1) \\ \cdots & \cdots & \cdots \\ \mathbf{r}_L^T(n) & \cdots & \mathbf{r}_L^T(n-I+1) \end{bmatrix} \quad (2)$$

And the weight vector  $\mathbf{w}$  can be written as:

$$\mathbf{w}(n) = [ \mathbf{w}_0^T \quad \mathbf{w}_0^T \quad \cdots \quad \mathbf{w}_I^T - 1 ] \quad (3)$$

It can be seen from these equations that it is necessary to generate LKM reference signals. In this case  $L$  is the number of error sensors,  $K$  is the number of reference signals and  $M$  is the number of actuators.

The philosophy of the LMS algorithm is that it will adopt in the direction of the instantaneously negative gradient of the mean square error with respect to the filter coefficients. The quadratic error can thus be written as:

$$\mathbf{e}^T(n)\mathbf{e}(n) = \mathbf{w}^T(n)\mathbf{R}^T(n)\mathbf{R}(n)\mathbf{w}(n) + 2\mathbf{w}^T(n)\mathbf{R}^T(n)\mathbf{d}(n) + \mathbf{d}(n)^T\mathbf{d}(n) \quad (4)$$

Taking the derivative with respect to the coefficients result in:

$$\begin{aligned} \frac{\partial(e^T(n)\mathbf{e}(n))}{\partial \mathbf{w}} &= 2\mathbf{R}^T(n)\mathbf{R}(n)\mathbf{w}(n) + \mathbf{R}^T(n)\mathbf{d}(n) \\ &= 2\mathbf{R}^T(n)\mathbf{e}(n) \end{aligned} \quad (5)$$

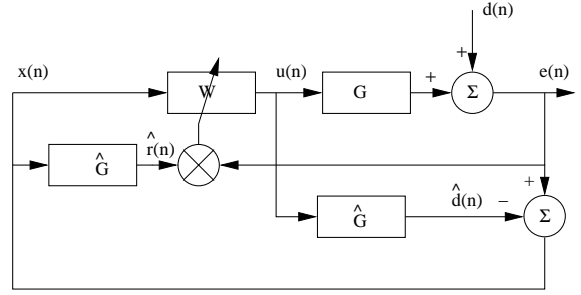


Figure 2: *Feedback controller with IMC*

The LMS algorithm can therefore be written in the following form:

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \alpha \mathbf{R}^T(n)\mathbf{e}(n) \quad (6)$$

The filtered reference signal is only available as an estimate resulting in:

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \alpha \hat{\mathbf{R}}^T(n)\mathbf{e}(n) \quad (7)$$

A more profound elaboration on the theory can be found in the textbook [3].

The theory derived so far is only suitable for a feed-forward controller. The a feedback controller with an internal model controller, makes it possible to use the theory from the feed-forward controller, in a feedback scheme. The block diagram of an IMC controller can be found in Figure 2 It can be shown that a feedback IMC controller is equivalent to a feed-forward controller with  $\mathbf{x}(n) = \mathbf{d}(n)$ . This is only the case when the estimated plant is equal to the real plant  $\hat{G} = G$ .

As shown in the derivation of the formulas it is necessary to generate an estimate of the reference signals  $\mathbf{R}(n)$ . For this it is necessary to generate LKM estimated reference signals. The process of generating these references signals will therefore require the most computations. Resulting in a system that spends most of its time in the update process of the filter coefficients.

The performance of a feedback controller largely depend on the group delay of such a system. In the textbook [3] a rough estimate for the bandwidth limitation due to group delay can be found:

$$Bandwith(Hz) < \frac{1}{6 \times Delay(sec)} \quad (8)$$

This delay will also degrade the performance of the overall system. It can be shown that most of the delay is within the reconstruction and anti-aliasing filters. A rule of thumb is, the lower the cut-off frequency of the filter, the larger the delay. So it would be a good idea to use filters that have a high cut-off frequency. However this will result in aliasing problems. It is also possible

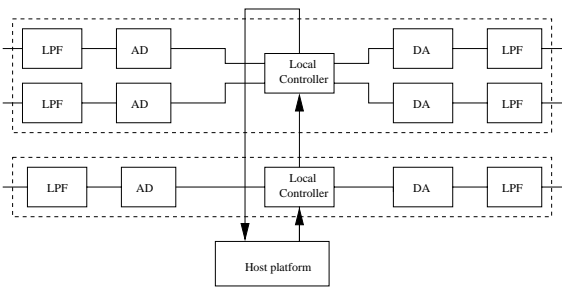


Figure 3: *ADDA distributed system*

to increase the sample frequency. This however will increase the computational complexity of the filtering reference signals. Each input signal is filtered with a plant response, however if the sample rate is doubled this would mean that the filters also need to be twice as long to gain the same dynamic behavior of the plant. The amount of computations needed would therefore raise quadratically. To work around this a multi-rate signal processing approach has been proposed in [4]. In this project a similar method will be used to reduce the overall group delay.

### 3. SYSTEM OVERVIEW

A general system overview can be found in Figure 3. It is clear that the converters and the local electronics are located near to the place where the actual physical quantity needs to be measured. Basically such a system consists of three parts, an AD and DA unit with filters, a local controller and a host platform.

The complexity of the FXLMS algorithm makes it necessary to select a platform that offers sufficient computational power. The PCI104 PentiumM platform delivers on this constraint. Also the fact that it runs RT-linux, makes it easier to implement and evaluate different algorithms. The PCI104 platform will need a simple PCI target interface that generates an interrupt whenever data is ready.

The PCI-interface card will be connected to the local controller by means of a simple four wire protocol. The protocol uses a low swing differential voltage system LVDS (see [5]). The LVDS signalling standard reduces the impact of electromagnetic compatibility issues. The local controllers and the PCI-controller will be connected in series (see Figure 3). The system will send 8 bits in every frame clock cycle. The frame-clock will be derived from the 33 MHz PCI clock. The data rate is therefore  $8 \text{ bits} * 33 \text{ MHz} = 266 \text{ MBit/sec}$ . The host controller needs a separate receiver and transmitter clock, which makes it necessary to select an FPGA family/device with at least 2 PLLs. The Altera Cyclone FPGA has two PLLs and supports LVDS pins it also has a reasonable price and is

therefor a suitable candidate.

The local controller translates the serial data and writes it into the local registers. The local controller generates a read action when an AD action has been completed and sends the value to the host-controller. The local-controller will be implemented in an FPGA. The use of an FPGA makes it possible to implement an interpolator and decimation filter and a local-feedback algorithm on the local-controller.

#### 3.1. PCI interface

The host controller will need a PCI to high-speed serial interface. A good reference for the PCI standard can be found in [6]. This PCI interface will have one single memory mapped IO range in which all IO registers are located. The PCI card will generate an interrupt in case of a timer event, or if an AD or DA conversion is ready. The interface board will also provide a master/slave like timer. This architecture makes it possible to start an AD and DA conversion at different times and can be used to improve system delay and jitter.

#### 3.2. Physical implementation of the protocol

The protocol is a simple point to point protocol. The clock is transmitted in the same direction as the data. The frame-clock will be multiplied by 4 to generate the data-clock. The data is transmitted on every rising and falling edge of the data-clock. This results in a data rate of 266 Mbits if the frame-clock is 33 MHz ( $33\text{MHz} * 2 \text{ bits} * 4$ ). This however will require a transmitter and receiver PLL and serializer/de-serializer. Every frame-clock period will therefore contain 8 bits. The FPGA used does not contain a high-speed serializer or de-serializer. Altera describes a method to implement it using normal logical elements [7].

### 4. PROTOCOL LAYER

The communication protocol between the PCI-interface card and the local controller is a simple byte oriented protocol. Every byte has an MSB bit, that if set to one, indicates the start of a packet. Every packet consists of 40 bits resulting in 5 bytes. Every none starting byte in the packet has this bit set to zero. A major advantage is that it is self synchronizing and that it can recover from communication problems. When the start bit is detected half way down a packet it will be ignored. The protocol does not support any further error handling, which is also not very useful, due to the realtime nature of the system.

#### 4.1. Packet layout

The layout of the packet is simple. Three bits in a 40 bits packet are reserved for commands : read , write and synchronization commands. Up to 16 controllers can be accessed resulting in a 4 bits controller ID. Each controller can have up to  $2^{12}$  register with a width of 16 bits. The remaining bits, each MSB bit of a byte, are used for synchronization.

#### 4.2. Host commands

The host can generate two types of commands: write and synchronization commands. The write command fills one of the internal registers of the local-controller with the 16 bits data word. The synchronization event does not deliver any data and the address field contains a value that indicates at what time the conversion should take place relative to the moment of receiving the synchronization packet. This relative time will be loaded into a timer that starts the AD or DA conversion. The first time when the sync package passes through a local-controller the host will set this relative delay-time to an arbitrary value. This value will be decremented with the number of clock cycles it takes for the packet to pass through a local-controller. This results in a new delay value for the next board, the value contained within the address is a 2's complement number. The first time a synchronization packet travels through all local-controllers, it is possible for the host-controller, to determine the time for a packet to travel around the ring. The next time the delay value will be adjusted and all controllers will fire at the same time.

#### 4.3. Local responses and commands

The local controller can only respond to a write or sync packet. It will then generate a read type packet. This packet will contain a 16 bits data word and the destination address of the register within the local-controller and the controller ID. To prevent collisions the local-controller will queue incoming data. Only if the packet is meant for the local-controller will it be consumed otherwise it will be passed on to the next local-controller. An exception to this rule is the synchronization packet which is always past to the next controller. This protocol can also be used to detect the presence of a smart-panel in the ring.

### 5. PROOF OF CONCEPT

At this moment a lot of work still needs to be done. The first prototype printed circuit boards are almost finished and the author is working on the hardware for the FPGAs. A demonstrator will be used to show the feasibility of such the implementation. The high-speed serial interface has

been tested and is working. The goal is to construct a demonstrator that can be used to evaluate different algorithms in a fast and easy way.

#### 5.1. The smart-panel

The smart-panel will be used to reduce the sound transmitted from the vibrating area. This will be realized by means of an adaptive multi-channel controller that uses off-line plant estimation. Local feedback loops are used to improve the performance. The overall system delay is improved by balancing between the analog and digital decimation and interpolation filters (see [4]).

### 6. CONCLUSION

In this paper a distributed approach for ADDA conversion has been proposed. The goal is to define a standard approach to be able to connect several boards in cascade. The use of high sample rates will reduce the size of the necessary analog filters and will improve overall system performance. A method to make all boards start synchronously has been proposed.

### 7. ACKNOWLEDGMENTS

This research has been performed within the scope of the EU 6th framework project Intelligent Materials for Active Noise Reduction (InMAR).

### 8. REFERENCES

- [1] Lippert Automationstechnik GmbH, *Cool Roadrunner 4 PCI-104 CPU-Board*, September 2004.
- [2] *PCI-104 Specification*, PC/104 Embedded Consortium, November 2004.
- [3] Stephen Elliot, *Signal Processing for Active Control*, Academic Press, 2001.
- [4] Mingsian R. Bai, Yuanpei Lin, and Jienwen Lai, "Reduction of electronic delay in active noise control systems a multi rate signal processing approach," *Journal of the Acoustical Society of America*, vol. **111**, pp. 916–924, February 2002.
- [5] National Semiconductor, *LVDS Owner's Manual Low Voltage Differential Signaling*, 3 edition.
- [6] Don Anderson Tom Shanley, *PCI System Architecture*, MindShare, Inc, 2000.
- [7] *Cyclone Device Handbook, Volume 1*, chapter 9, Altera, 2000.