Adaptive Noise Canceller Demo based on the LS Lattice Filter

Z. Pohl, J. Kadlec, M. Tichy
zdenek.pohl@utia.cas.cz

Contents

Revision history

Rev.	Date	Author	Description
0	12.11.2007	M.T.	Document creation
1			
2			

Licensing information

1. Description

This adaptive noise canceller (ANC) demo presents the use of our least-square lattice (LSL) IP core. The LS lattice adaptive filter employed within the core is implemented as four-stage pipelined unit. The core operates on 16-bit two's complement integer (fixed-point) input/output data. However, the LS lattice algorithm and all other calculations are implemented in the 19-bit logarithmic number system (LNS) arithmetic. Conversions from integer to LNS representation and vice versa are integrated within the LS lattice core.

The ANC LSL demo contained in this package has been prepared for testing on two different prototyping boards:

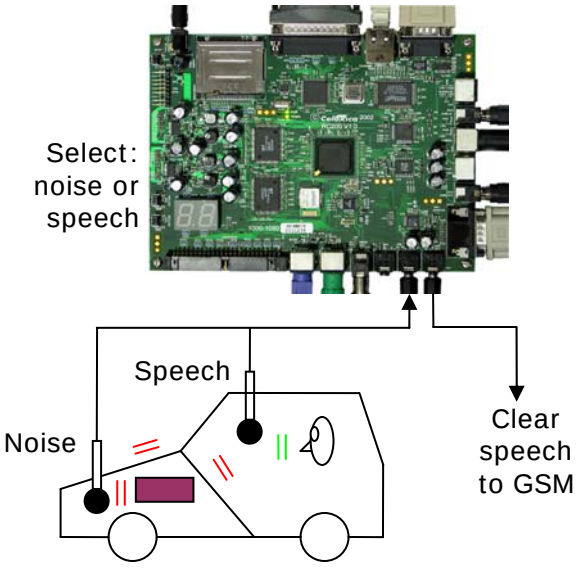
1. The Celoxica RC200(E) board equipped with the Xilinx Virtex-II XC2V1000-4 FPGA.
2. The XESS XSV-800 board equipped with the Xilinx Virtex XCV800-4 FPGA.

2. RC200(E) Board Set-Up and Parameters

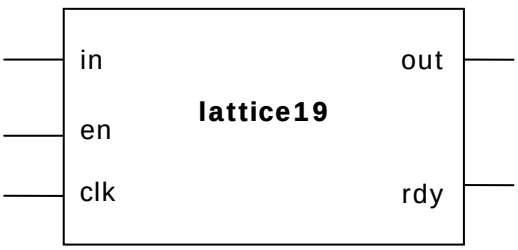
The following diagram shows typical noise canceller application in a car hands-free system, where the car noise on the other side of the line is disturbing but it can be effectively suppressed by an adaptive filter. In such application, an additional microphone (reference sensor) is required to acquire the car noise. The task of adaptive filter is to estimate the transfer function of the noise coming inside of the passenger cabin. When the outside noise is known and the transfer function estimated, the disturbance can be easily suppressed by subtracting the transformed car noise from the signal acquired by the primary sensor.

On the right-hand side, general features of the LS lattice core and of the demo are presented. The design implementing the filter of order 141 used in the demo application is clocked at 50 MHz, operating on an audio signal sampled at rate of 22050 Hz. This represents the overall performance around 90 MFLOPS.

RC200 Demo Board



LS Lattice IP Core



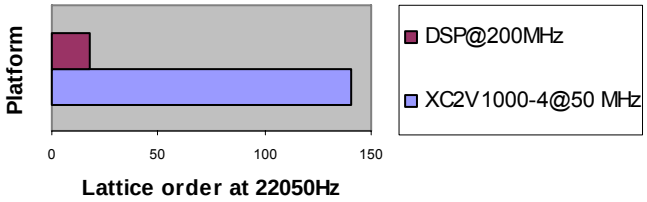
- Recursive LS lattice adaptive filter
- 19-bit LNS implementation
- 16-bit integer input & output
- Adjustable order and forgetting
- Demo implementation:
  - Design clock: 50MHz
  - Sampling rate: 22050 Hz
  - Filter order: 141
  - Performance: 90 MFLOPS

Available **lattice19** IP core versions and their parameters:

Inner pipeline	Max. lattice order	Max. sampling rate**	LUTs	MFLOPS	BRAMs
1 stage	63	17150	1571	30	13
2 stages	126	17150	2964	59	13
3 stages*	189	17150	4279	89	18
4 stages	252	17150	5682	119	18

* lattice19 IP core used in RC200 demo application
** design clock at 50MHz, maximal lattice order

Virtex II XC2V1000-4 lattice solution compared to a DSP microprocessor



General Information

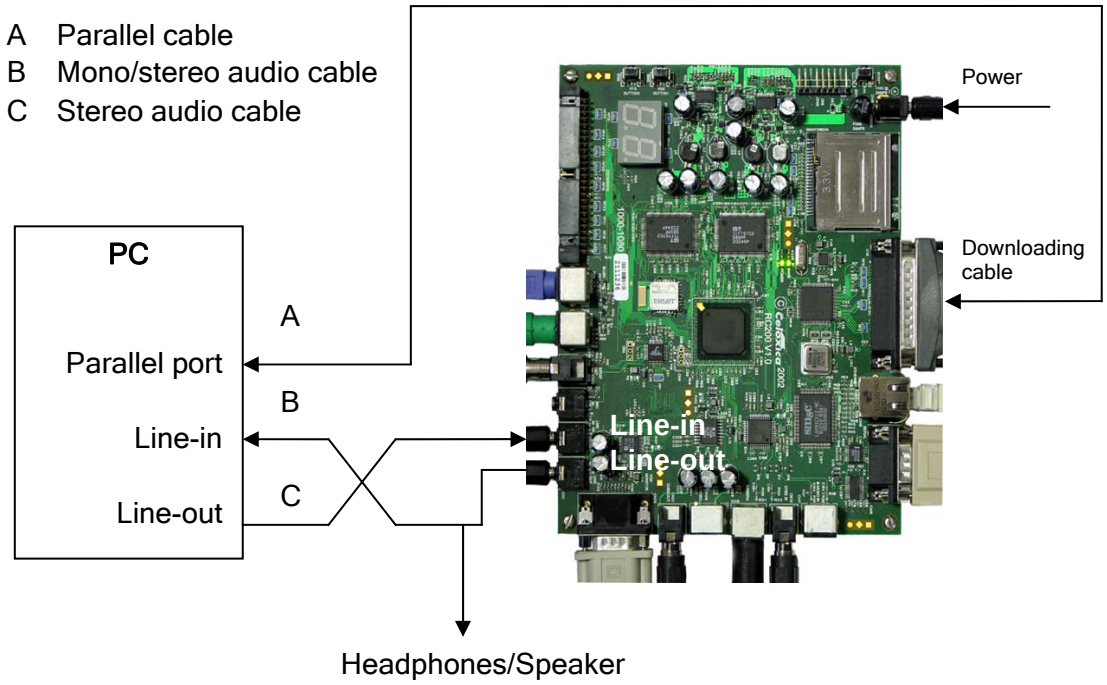
Board **RC200** available from Celoxica
http://www.celoxica.com

ANC LSL Demo available from:
http://sp.utia.cz

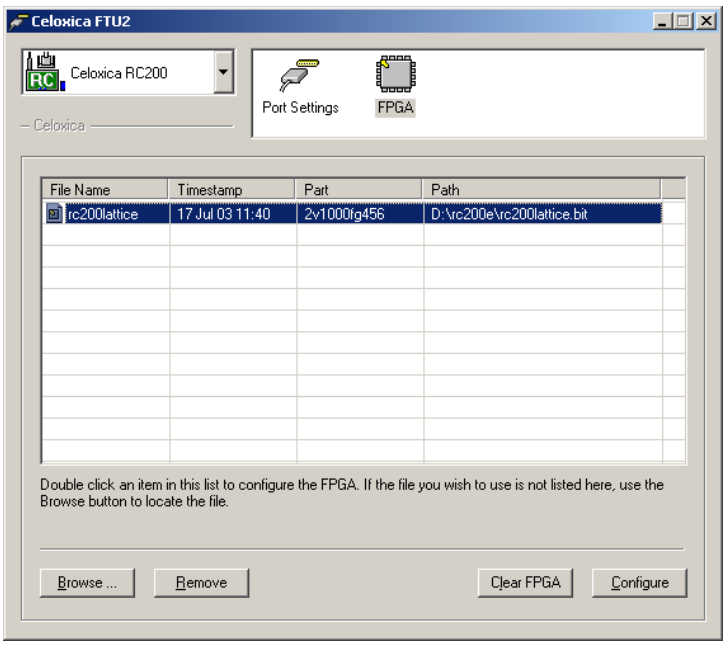
2.1 How to Run the Demo on the RC200(E) Board

To run the ANC LSL demo on the RC200(E) prototyping board, the Celoxica PDK must be properly installed. Use the following steps to run the demo:

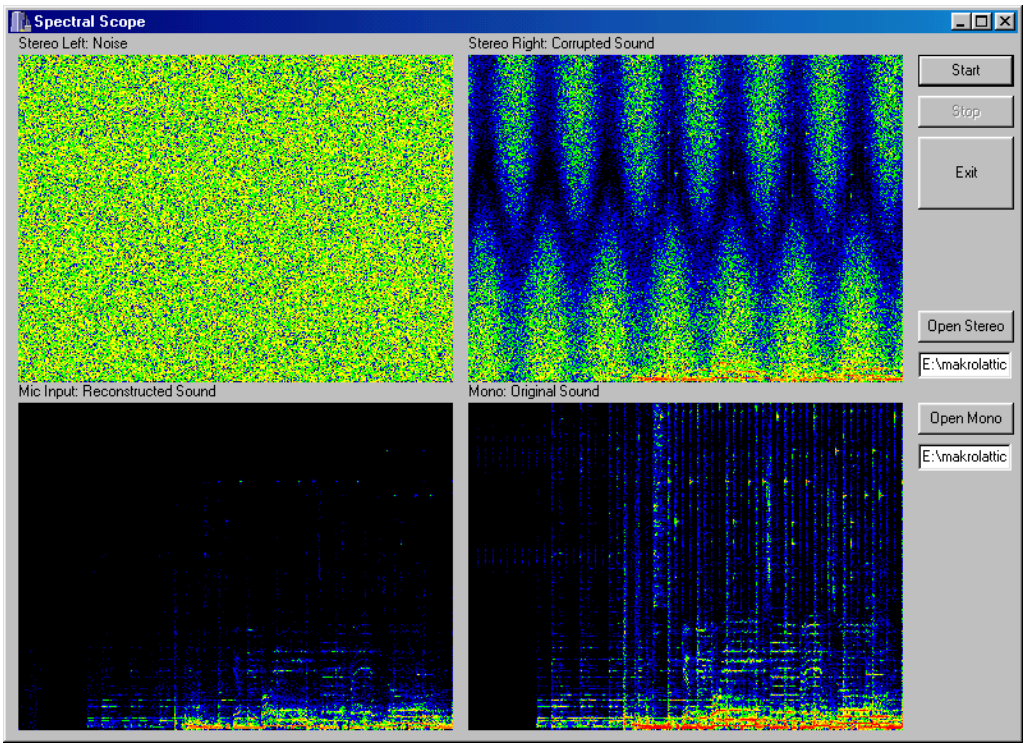
1. Connect the RC200(E) board has to the PC as depicted in the following figure.



2. After the RC200(E) has been powered on, use the FTU2 application (part of the Celoxica PDK) to upload the bitstream `rc200e/rc200lattice.bit` to the FPGA. See the following figure.



3. Finally, use the FFTScope application (`tools/FFTScope.exe`) to open audio files and to display the results. Open the stereo audio file `example/sendToCard.wav` and start audio playback. The application also requires the original (uncorrupted) audio signal as its input – use the mono audio file `example/original.wav`.

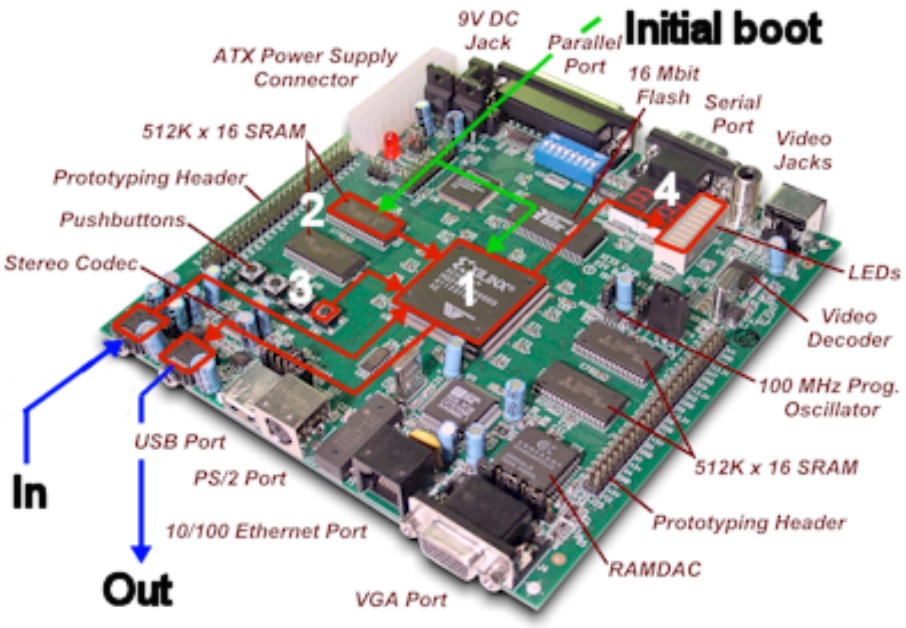


The figure above presents FFT scopes of the resulting signals in the demo. As can be seen of the figure, during playback, the FFT scopes of the noise (left-upper scope), of the speech corrupted by the transformed noise (right-upper scope) and of the original signal (left-lower scope) are displayed. If an audio cable connects the card's line-out to the PC's line-in, the FFT scope of the resulting reconstructed signal (right-lower scope) is displayed too.

3. XSV-800 Board Set-Up and Parameters

The ANC LSL demo has also been prepared for the XESS XSV-800 prototyping board. The same scheme of noise suppression as in the case of RC200(E) card is used. The following diagram shows schematically where to connect input and output signals.

On the right-hand side, general features of the LS lattice core in the XCV800 chip are presented. The LS lattice design occupies less than three quarters of the chip area, whereas it fully utilizes available block memory resources. The design implementing the filter of order up to 252 used in the demo application is clocked at 33.3 MHz.



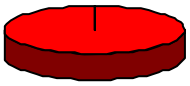
XCV800-4 Design Summary
Slice utilization  72 %
Block RAMs  100 %
Clock rate: 33.3 MHz

Diagram legend:

1) FPGA with uploaded LS lattice design
2) Start-up lattice configuration in SRAM
3) Filter `on/off` switch
4) Info LEDs

Depending on the actual lattice order, the filter can operate on audio signals sampled at rate of up to 31 kHz. The overall performance represented by the LS lattice core, depending on its configuration, is more than 75 MFLOPS. (See the table on the next page.)

Available **lattice19** IP core configurations and their parameters:

Oscillator division factor	Design clock [MHz]	Audio codec division factor	Audio sampling rate [Hz]	Max. pipe degree	Lattice degree	Mflops	Config. file**
3	33.33	2	31680	21	84	74.5	deg_084.XES
3	33.33	4	15840	43	172	76.3	deg_172.XES
3	33.33	8	7920	63*	252*	55.9	deg_252.XES

* Max. pipe degree is limited by the Block RAM size

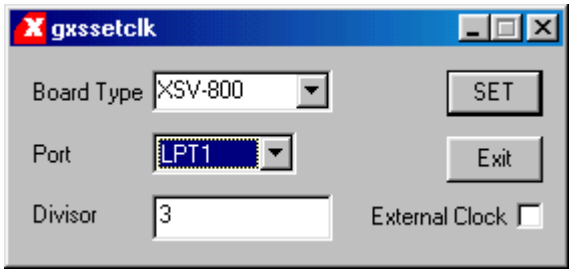
** Lattice design reads configuration from this file loaded into external RAM

3.1 How to Run the Demo on the XSV-800 Board

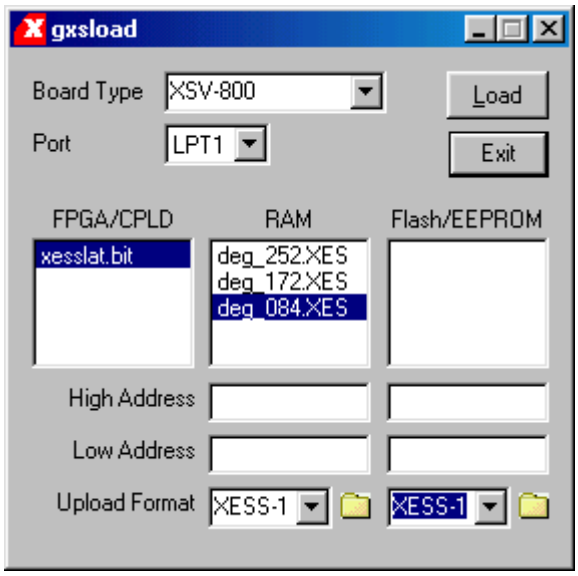
To run the ANC LSL demo on the XSV-800 card, board support package and tools provided by XESS (http://www.xess.com) must be properly installed. Use the following steps to run the demo:

1. Connect the XSV-800 board to PC using the parallel port and provide the card with the power supply. Audio cables should connect the PC's line-out with the XSV-800 audio input and the card audio output with the PC's line-in (and with loudspeakers or earphones).

2. Follow instructions in the XSV-800 board manual to set the appropriate oscillator division factor "3" of the programmable oscillator. The `gxssetclk` tool can be used as depicted in the following figure.



3. Load the appropriate configuration file (`*.HEX`) to the XSV-800 SRAM and then configure FPGA with the bitstream `xesslat.bit` using the `gxsload` tool as depicted here:



4. Then, open audio file `example/sendToCard.wav` and start audio playback from the PC. The filtered output comes to the PC's line-in and to loudspeakers (if connected to the card audio output). The filter `on/off` switch (3) can be used to send the original corrupted signal to the card audio output.

5. Once the LS lattice design has been loaded to the FPGA, the LED indicators show various states:
   - LED1: Input/output interconnection indicator; when ON, input signal goes directly to the loudspeakers (or to the PC's line-in).
   - LED2: Data are stored in SRAM; ON until maximum capacity has been reached.
   - LED3: When ON, initial delay is activated.
   - LED4: Synchronization LED; when ON, algorithm is waiting for synchronization from audio codec; higher light intensity indicates lower efficiency of the algorithm.

Note: Alternatively, the FFTScope application (`tools/FFTScope.exe`) can be used for sound playback, and also for displaying the results. Open the audio file `example/sendToCard.wav` and start audio playback. The application also requires the original (uncorrupted) audio signal as its input – use the file `example/original.wav`. During playback, the FFT scopes of the noise, of the speech corrupted by the transformed noise and of the original signal are displayed. If an audio cable connects the card's line-out to the PC's line-in, the FFT scope of the resulting reconstructed signal is displayed too. Example FFT scopes of signals used in this demo can be found in the last figure in Section 2.1.

4. Package contents

The following directory tree represents the contents of the CD-ROM (or ZIP package):

```
CD-ROM
      |-- autorun.inf
      |-- index.html
      |-- license.txt            Licensing information
      |-- _img                   Aux. files for html
      |   `-- logo_sp.gif
      |-- doc                    This documentation
      |   `-- anclsl_demo.pdf
      |-- example                Example audio files
      |   |-- original.wav
      |   `-- sendToCard.wav
      |-- rc200e                 Configuration bitstream for the RC200(E) board
      |   `-- rc200lattice.bit
      |-- tools
      |   |-- FFTScope.exe       FFTScope application
      |   `-- xstools4_0_2.exe   XESS tools
      `-- xsv800                 Configuration files for the XSV-800 board
          |-- deg_084.XES
          |-- deg_172.XES
          |-- deg_252.XES
          `-- xesslat.bit
```

Contact information

For more information or any comments on the package or this text, please contact:

Zdenek Pohl <zdenek.pohl@utia.cas.cz> or Milan Tichy <tichy@utia.cas.cz>
Department of Signal Processing
Institute of Information Theory and Automation
Academy of Sciences of the Czech Republic
Pod Vodarenskou vezi 4
182 08 Prague 8
Czech Republic
http://sp.utia.cz