D2.7 - Integration of the final version of the algorithms dedicated to intelligibility enhancement in confined public spaces – General description and exploitation

Project acronym:	I'CityForAll
Project name:	Age Sensitive ICT Systems for Intelligible City For All
Strategic objective:	Socio-acusis ICT solutions for a better social well-being of Ederly People
Project Duration:	July, 1st 2012 – Dec, 31th 2015 (42 months)
Co-ordinator:	CEA: Commissariat à l'Energie Atomique et aux Energies Alternatives
Partners:	UPD: Université Paris Descartes ENEA: Agenzia Nazionale per le Nuove tecnologie, l'EnergiA e lo sviluppo economico sostenibile TUM: Technische Universität München CRF: Centro Ricerche FIAT CENTICH: Centre d'Expertise National des Technologies de l'Information et de la Communication pour l'autonomie Active Audio EPFL : Ecole Polytechnique de Lausanne – Lab. D'Electromagnétisme et d'Acoustique

D2.7

Version:	3.00
Delivery Date:	2015-08-28
Due date:	2015-04-15
Task:	2.4
Leader:	Active Audio
Dissemination status:	PU

This project is co-funded by the Ambient Assisted Living (AAL) Joint program, by the German BMBF, by the Agence Nationale de la Recherche – ANR, by Caisse Nationale de la Solidarité pour l'Autonomie – CNSA, by the Ministero dellIstruzione dell'Università e della Ricerca – MIUR, and by Federal Office for Professional Education and Technology OPET

D2.7	Executive Summary

Foreword

This document presents a general description of the implementation in the NUT processor of the algorithms used to process vocal announces. The application is named IVA (Intelligible Vocal Announce).
It also presents the PC application, named IVAcontrol, which is used to control the IVA operation. Finally, the step-by-step procedure that should be followed for tuning a PA system using IVA is presented.

The algorithms used in IVA comprise:
- the speech Conformer
- the presbyacusis compensation
- the AGCpresby

The reader might refer to D2.4 for a description of the principles of these algorithms, and to dliverable D2.6 for a detailed description of their implementation in NUT.

Dissemination Level of this deliverable (*Source: I'CityForAll Technical Annex p20 & 22***)**	
PU	Public
Nature of this deliverable (*Source: I'CityForAll Technical Annex p20 & 22***)**	
R	Report **and prototype** *Even a demonstrator or a prototype shall be accompanied with a report, or which basic structure is explained on page 3.*

Due date of deliverable	15 April 2015
Actual submission date	28th august 2015
Evidence of delivery	Report

Date	Version	Reviewer	Recommendations
	3.0		

Authorisation			
No.	**Action**	**Company/Name**	**Date**
1	Prepared	ACTIVE AUDIO/XM	28th aug. 2015
2	Approved		
3	Released		

Disclaimer: The information in this document is subject to change without notice. Company or product names mentioned in this document may be trademarks or registered trademarks of their respective companies.

Table of contents

Introduction

The present document describes the implemetation of the algorithms used in the NUT processor to process the vocal announces.
The application is named IVA (Intelligible Vocal Announce).

The algorithms used in IVA have already been described in D2.4. They comprise:

- the Speech Conformer, which "normalizes" the spectrum of the voices ;
- the presbyacusis compensation, which compensates for the mean hearing loss of presbyacusis subjects
- the AGCpresby, which adjusts the level of diffusion.

The originality of the IVA application consists in :

a- combining the speech Conformer with a presbyacusis compensation, so that the compensation applies on a signal of known spectrum corresponding to standard CEI268-16 ;
b- diffusing the vocal announces at a sound level set by a UDR based AGC, taking into account the recruitment phenomenon typical from presbyacusic persons.

The present document describes :

- the architecture of the DSP code
- how to interface the IVA code (running in NUT) using the PC application IVAcontrol, which has been designed by Active Audio within the I'City project.

A detailed description of the implementation of the algorithms can be found in deliverable D2.6.

The speech Conformer was developed by Active Audio. The presbyacusis compensation and the AGCpresby algorithms have been developed by the partners of the I'CityForAll project.

The NUT processor (running IVA) should be inserted in the PA system between the audio matrix and the loudspeaker power amplifiers.

The IVA code and the IVAcontrol interface have been developed for testing purposes within the IcityForAll project. They are not intended for commercial use in the current version presented in this report.

In the following, we give a short description of the NUT processor and the SigmaStudio programming environment. Then a general description of the IVA DSP code is presented in chapter 1, followed by a description of the control of IVA using the IVAcontrol application. Finally, a typical tuning process of a PA system using IVA is reviewed in chapter 3.

Description of the NUT processor and the SigmaStudio programming environment

NUT is a digital audio processor having 8 symmetrical analog inputs and 8 symmetrical analog outputs. It can be remote controlled with a PC or many other devices, via ethernet, USB, or RS232. Figure 1 shows the front and rear panels of the NUT processor.
More information on NUT can be found on www.activeaudio.fr.

Figure 1 : Front and rear panels of the NUT processor.

NUT is based on the ADAU1442 DSP processor, from Analog Device. The sampling frequency is 48kHz. In the ADAU1442, the numbers are coded on 28 bits, either using a fractionnal format (noted 5.23) with numbers ranging from -16.0 to +16.0 by 2^{-23} increments, or using a integer format (noted 28.0) with positive integer numbers ranging from 0 to 2^{28}-1.

It is important to note that a great part of the difficulty in programming our algorithms come from this technology. Indeed, our algorithms often needs to perform slow operations, like estimating RMS levels with a very long time constant (several seconds or even minutes). This implies multiplications involving extremely small constants, yeilding problems of underflow which would not arise with floating point processors.

For more information about the ADAU range of DSPs, please refer to www.analog.com.

The ADAU1442 runs the same code at every sample processing : no multirate operation, no background task. The ADAU DSPs are programmed using the SigmaStudio language (http://wiki.analog.com/resources/toolssoftware/sigmastudio). It is a graphial language which has both full audio algorithms (such as compressors, filters...), and low level functions (such as additions, multiplications...). Code appears in boxes, connected together via their inputs and outputs. A box may contain sub-boxes, which can be seen as the equivalent of the subroutines in text-programming.

In chapter 1 below, the codes will be presented with screenshots showing block diagrams that represent the code.

We have also developed a Windows application for interacting with the I'City code on NUT. This application, called IVAcontrol, consists in reading and writing data in the memory of NUT. The IVAcontrol is presented in chapter 2 of this document. Chapter 3 presents the tuning methodology.

1. Description of the IVA code

1.1 Main program

Figure 1.1 below shows the main program.

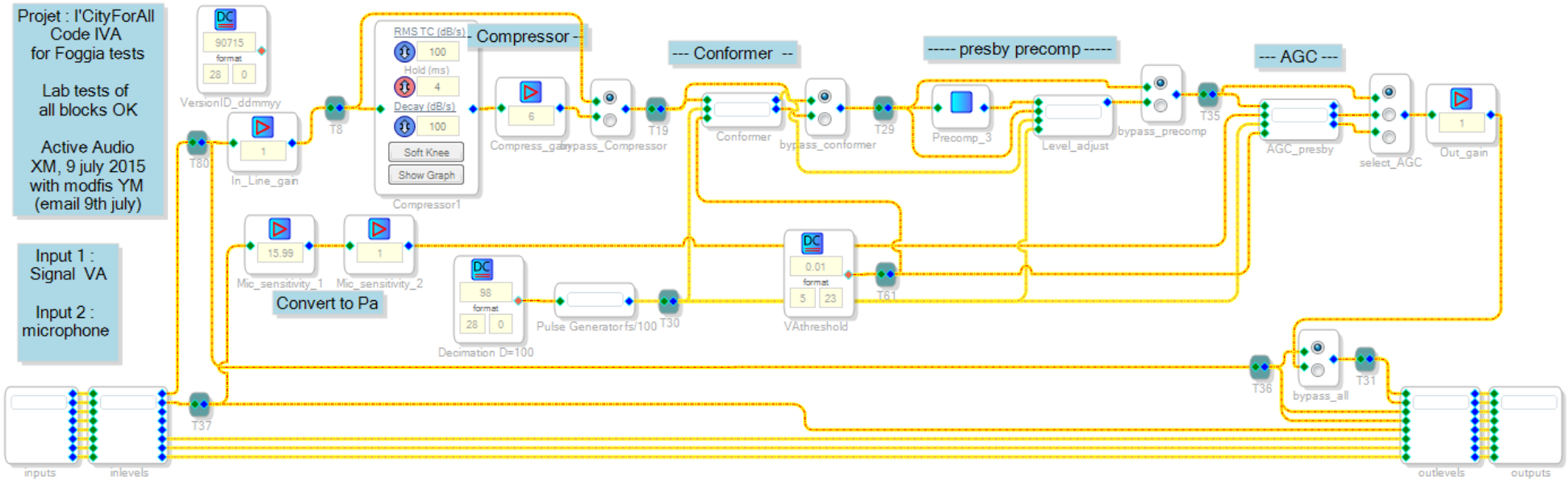

Figure 1.1 : Main program of IVA code.

The block named 'inputs' assigns the physical analog (XLR) inputs to the software input signals.
Block 'outputs' assigns the physical analog (XLR) outputs to the software output signals.
Block 'inlevels' monitors the level on the 8 inputs. These levels are used to drive the leds of the front panel.
Block 'outlevels' does the same for output signals.

The announce (VA) should be connected to input 1.
The microphone output should be connected to input 2 The two gains entitled 'Mic_sensitivity_1' and 'Mic_sensitivity_2' on input 2 converts the signal delivered by the microphone into Pascals : signal = 1 when pressure = 1 Pa.

The signal applied on input 1 is copied on outputs 3 and 4. The treated VA is sent to outputs 1 and 2. Chanels 5-8 are bypassed : input copied to output.

Gain 'in_line_gain' is used to adjust the amplitude of the signal to the adequate level for the various algorithms.
Gain 'Out_Gain' is used to adjust the level of the output signal to the sensitivity of whatever device is connected downstream NUT.

The presbyacusis compensation is downstream the conformer, so that it applies to a voice having a « normal » spectrum.
Five switches allows to switch on/off all the steps of the processing :

- switch 'bypass_compressor' allows bypassing the compressor
- switch 'bypass_conformer' allows bypassing the conformer
- switch 'bypass_precomp' allows bypassing the presbyacusis compensation
- switch 'select_AGC' allows selecting the AGC algorithm (type SNR or UDR), or bypassing the AGC.
- switch 'bypass_all' allows switching OFF all algorithms at once.

The pulse generator delivers a pulse once every 100 sampling periods. This clock is used in the Conformer and the AGC to freeze data during 100 sampling periods, for numerical precision reasons (see below).
The processed signal is sent to outputs 1 and 2 via an output gain which allows to adjust the output level to the device downstream.
The unprocessed signal is sent to output 1.

1.2 Compressor

Compression is a very standard audio function. It should be seen as a preliminary step to asjust the level of speech in order to ensure proper operation of the Conformer and the AGCpresby.

SigmaStudio has several pre-programmed compressors.
We use the one called « RMS (no gain) », which has the following settings :

- RMS TC is the time constant for attack (in dB/sec)
- Decay is the time constant for release (in dB/sec)
- Hold controls the time (in ms) the compressor maintains its current output gain setting before it starts decreasing as the input level decrease.

- The « show graph » button allows viewing and modifying the in/out characteristic.
- The « Soft knee » button has a minor effect, consisting in softening the angle of the articulation (at -20dB on the graph shown).

As can be seen, we have adjusted the parameters so that the compressor reacts rapidly both on attack and release, and limits the output to approximatively -20dB.

As a consequence, the "compress_gain" which is applied to the output of the compressor (see fig 1.1) is set to 6 (i.e. 16dB), so that the signal downstream is of correct amplitude.

These settings ensure that the amplitude of the signal downstream is relatively constant, which ensures optimal performance of subsequent algorithms.

1.3 Speech Conformer

The Speech Conformer algorithm developed by Active Audio is a kind of adaptive equalizer. Within a few seconds, it analyses the spectrum of the input signal in real time, and applies an equalization determined so that the spectrum of the output signal matches a specified target spectrum chosen by the user. In the IVA code, the target spectrum has been chosen as the spectrum specified in standard CEI268-16.

The effect of this algorithm is to correct the timbre of voices only when necessary. Typically, a person having a clear voice will not be corrected, whareas a person having a veiled voice will be corrected.

As a result, at the output of the Conformer, the spetrum of the signal is known, and almost independent of the speaker.

A detailed description of the implementation of the Conofrmer in NUT can be found in deliverable D2.6.

.1.4. Presbyacusis compensation

The presbyacusis compensation consists in a cascade of biquad cells, which closely approximates the compensation curve corresponding to the mean hearing loss curve of presbyacusis persons at the age of 45.

1.5. AGC_presby

Block AGC_presby computes a gain which is applied to the VA signal, ensuring optimal level of diffusion of the announce.

Two algorithms are implemented, named AGC_SNR and AGC_UDR. The desired algorithm can be switched on the main program, as seen above.

2. Description of the IVAcontrol application

The IVAcontrol application has been developed for debug and test purposes. It can only be used by users having a deep knowledge of its functioning. In particular, it is not intended for commercial use.

2.1 Loading the IVAcontrol application

To run the IVAcontrol application, the user must first start the NUT application from a PC, connect the PC to the NUT processor via USB or ethernet, and then click on the connected device as shown on figure 2.1.

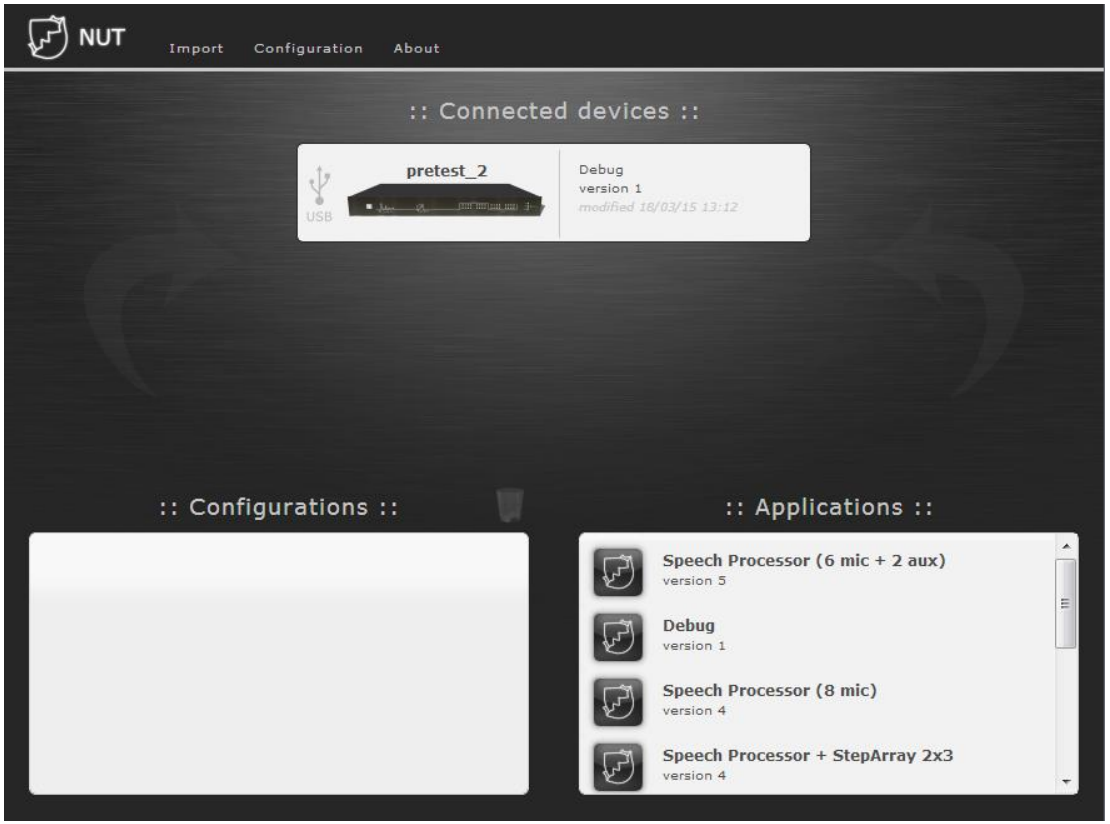

Figure 2.1 : To launch IVAcontrol, click on the connected device.

2.2 The IVAcontrol interface

When the NUT PC application starts a connection with a NUT processor running the IVA code (see above), the user interface presented in figure 2.2 appears. Table 2.1 lists the variables with their format and their meanings.

- In column 'NAME' are the names of the variables, in alphabetical order.
- The values of these variables is display in column 'VALUE'.
- For each variable, if the checkbox in the third column is checked, then the value is constantly refreshed.
- For each variable, clicking on the 'X' in the last column will cause the variable to disappear from the table.

For introducing a new variable, the user must type the variable name in the field marked with a red arrow. Variable names can be seen on the figures of chapter 1 above. All names in lower caps. For example, variable GainMin in block AGCforward is specified by typing "agcforward_gainmin".

Note : bypass_xxx value must be set to 2 for bypassing ; or set to 1 for not bypassing (i.e. algorithm ON).

Variable	Type	Write	Meaning
Agc_presby_readVAon	Float	No	1 if VA is detected, else 0
Agc_snr_read_eva	Float	No	Read Eva in AGC_SNR
Agc_snr_read_gainsnr	Float	No	Read the AGC_SNR gain
Agc_udr_read_eva	Float	No	Read Eva in AGC_UDR
Agc_udr_read_gainsnr	Float	No	Read the AGC_UDR gain
bypass_ compressor	binary	yes	Bypass switch of compressor
bypass_ conformer	binary	yes	Bypass switch of conformer
bypass_precomp	binary	yes	Bypass switch of presby compensation
Compress_gain	Float	No	Read the compression gain
conformer_sig_present	binary	No	1 if signal is detected in Conformer ; else 0
Estim_..._read_alphau	Float	No	Read alphau in estim_noiselevel
Estim_..._read_cmpval	Float	No	Read cmpval in estim_noiselevel
Estim_._read_ noisespl2	Float	No	Read noisespl2 in estim_noiselevel
in_line_gain	Float	yes	Input gain of VA signal
inlevels_level1	Float	No	Level of the input VA signal
inlevels_level2	Float	No	Level of the input microphone signal
Level_adjust_readgain	Float	No	Read gain of the Level_adjust algorithm
Level_adjust_readsplva	Float	No	Read SPLVA of the Level_adjust algorithm
Liss_mic_incr	Float	No	Read incr in Liss_mic algorithm
liss_mic_read_noisespl1	Float	No	Read noiseSPL1 in Liss_mic algorithm
Mic_sensitivity_2	Float	Yes	Normalisation gain 2 for mic signal

Figure 2.2 : Variables used by default in the IVAcontrol interface.

Table 2.1 : Variables of the IVAcontrol

3. Tuning IVA

Table 3.1 lists all the steps that must be completed in order to tune the IVA.

The reverberation time RT and the definition Dtmix of the hall must be measured first, as they are required to set the AGCpresby parameteres.

Step	Variable	Set as	Block
1	Mic_sensitivity	Apply a steady noise of known level Pmic on the microphone. Set mic_sensitivity to Pmic/sqrt(L_1), where L_1 is the value of agcforward_read_eb shown in IVAcontrol. (note : Eb is computed in Pa^2, see)	Main
2	In_line_gain	Set in_line_gain so that the level at the input of the conformer (ReadAref) is approximately 0.1 when the compressor is OFF. Lin value.	Main
3	sig_present	Make sure that when a VA is applied on input 1, flag sig_present of the conformer is set.	sig_presence
4	ReadVAon	Make sure that when a VA is applied on input 1, flag VAon of the conformer is set.	VApresence
5	Out_gain	Set the value of out_gain so that, with a VA of nominal amplitude, the output does not clip, and the amplitude of the output signal is correct for the devices downstream. Lin value.	Main
6	Eta ; C ; D	In block AGC_UDR : * Set eta = (squared mean pressure level on the audience) / (Eva) * Set constant C to UDRref, and constant D to coef*(Dtmix-(1-Dtmix)*UDRref).	AGC_UDR
7	Eta ; C ; D	In block AGC_SNR : * Set eta = (squared mean pressure level on the audience) / (Eva) * Set constant C and D so that C/D = desired SNR, i.e. (Energy of signal) / (Energy of background noise).	AGC_SNR
8	GainMin GainMax	Set GainMin so that, with a VA of nominal level Eva and AGC gain = GainMin (e.g. mic disconnected), the SPL in the listening area is correct for quiet periods (typcally at night). Lin value. Set GainMax so that, with a VA of nominal level Eva and AGC gain = GainMax (high noise level on mic), the SPL in the listening area corresponds to the max SPL allowed (in terms of the recruitment constraint, and the power capbility of the amplifiers and speakers). Lin value.	AGC_UDR and AGC_SNR
9	Bypass_xxx	Set the bypass switches	Main

Table 3.1 : steps to be completed to tune the IVA.

Conclusion

In this document, we have presented the implementation in the NUT processor of the algorithms that were used for processing the voice announcements (VA) messages within the I'CityForAll project. These algorithms comprise essentially the original combination of the speech Conformer with a presbyacusis compensation, and two AGC algorithms : a traditional SNR based AGC, and a the UDR based AGC developed within the I'CityForAll project.

They were described in deliverable D2.4. The NUT application is called IVA.

The IVA application was programmed using the SigmaStudio graphical programming environment used in NUT. Great care was taken in order to avoid numerical problems (overflow, underflow, computation noise...) due to the coding format of numbers in the DSP that is used in NUT.

A PC application named IVAcontrol has been developed in order to control the IVA code (running on NUT) and set the paramters.

IVA and IVAcontrol were tested succesfully during the listening tests carried out in Nantes and Foggia.

If Active Audio decides to implement these algorithms (together with other algorithms) in a commercial NUT application, then a dedicated graphical user interface would have to be developed.

Keywords

Public Address
Audio signal processing
DSP
Vocal
Presbyacusis

Acronyms used

DSP	Digital Signal Processor
PA	Public address
VA	Vocal announces
IVA	Name of the NUT application which runs the I'City algorithms
IVAcontrol	Name of the PC application which was developed for controling the IVA code
UDR	Useful to Detrimental Ratio
AGC	Automatic Gain Control