Acoustical Echo Canceller with Dynamic Echo Suppressor

I. Abreviations

Tableau 1 : Abreviations

AEC	Acoustical Echo Canceller
DSP	Digital Signal Processeur
PSD	Power Spectral Density (of 2nd order)
NLMS	Normalized Least Mean Square
SC	System Controller
EC	Echo Cancellation - Used to name the EC DSP package
FFT	Fast Fourier Transform
DES	Dynamic Echo Suppression - Uses spectral substraction
ERLE	Echo Return Loss Enhancement
ERL	Echo Return Loss

II. Preliminary info

The algorithms included in the AEC are suited ONLY if the coupling between speakerphone / microphone is LINEAR.

The delay estimation and compensation stuff is a priori suited ONLY with NON TOO COLORED signals. This implies that it is a priori NOT SUITED with DTMF and other very short band signals. A countermeasure will be taken in the future to cope with these kind of signals. But hopefully the delay stuff does not take into account the delay obtained for impulse responses that have many pics of rougly the same amplitude as it is the case when the apdative filter is excited by too colored signals.

III. Introduction

The aim of this document is to present how to use the AEC integrated in the MT firmware. It presents the signification of the coefficients transmitted to the AEC and also where and when to do the initialization and how to handle particuliar situations such as handovers, switch between one audio mode to another, e.g. from handset mode to handsfree carkit mode.

IV. Different modes overview

For the moment, only the two following operating modes will be described :

- *Handset echo cancellation mode* used to cancel the echo due to the coupling between earpiece and microphone through the handset.
- *Handsfree car kit mode* used in car kit mode.

V. Description of the configuration for the different modes

V.1 General description of the parameters

The DSP has two functions that can be invoked by the SC in order to configure the EC functionnality :

- **EC_init**
- **EC_init_param**

EC_init actually includes **EC_init_param**. It means that **EC_init** does the same thing as **EC_init_param**, which job is only to set some configuration values in EC's static ram, plus some initialization of the EC's static ram buffers. Both **EC_init** and **EC_init_param**

have the same software interface.

V.1.1 EC_init_param / EC_init

This function needs the following parameters in the following order :

- [0] EC_mode
- [1] smooth_factor
- [2] shift
- [3] dt_level2
- [4] nlms_factor
- [5] hf_threshold
- [6] alpha_rev
- [7] erle
- [8] beta_rev
- [9] 32768-beta_rev
- [10] gamma_nsp
- [11] gamman_sp
- [12] spdet
- [13] ycomp
- [14] history_buffer_size
- [15] ptr_ls

V.1.1.1 EC_mode

This short word is actually a bit field as follows :

b15 .. b14	**b13 .. b8**	**b7 .. b5**	**b4**	**b3**	**b2**	**b1**	**b0**
X	delay_max (6 bits)	X	2nd flag for delay stuff.	activation flag for delay stuff .	algo switch		EC_update flag

Here after is the description of the bits depicted below :

- If b0 = 1 then EC_update is executed.
- if b2b1 = 0b00 then the AEC is deactivated.
- if b2b1 = 0b01 then only the AEC runs in NLMS mode only.
- if b2b1 = 0b10 then the AEC runs in NLMS+DES (normal mode).
- if b2b1 = 0b11 then the AEC runs in other mode that may not have been defined at this date.
- if b3 = 1 then the AEC will try to estimate the delay between microphone and speakerphone and apply the correction when the estimation is complete. See also b4.
- if b4 = 1 and b3 = 1 then the AEC will try to always do an estimation and a correction otherwise if b4 = 0 and b3 = 1, the correction is applyed once and after that correction the feature of delay compensation is inhibed.
- b13-b8 (6 bits) : delay_max used to bound the correction of the delay. As the EC_fifo is limited in size, the correction can not be larger than a certain limit. The maximum limit bound is EC_fifo_size - audio_fifo_size - audio_frame_length. For example, on MT Av1 EC_fifo_size is 447, then the limit will be 447 - 254 - 160 = 33.

V.1.1.2 smooth_factor

This short word is related to the low pass filter used to smooth the energies in the NLMS part of the AEC's algorithm. We can link the value of the short word to the time constant τ of a first order low pass filter :

$$\tau = \frac{-1}{F_s \cdot \ln(1 - 2^{smoothfactor})}$$

Tests showed that we can take a value of smooth factor = -6 that is related to τ = 8 milliseconds.

Note : This parameter is negative.

V.1.1.3 shift

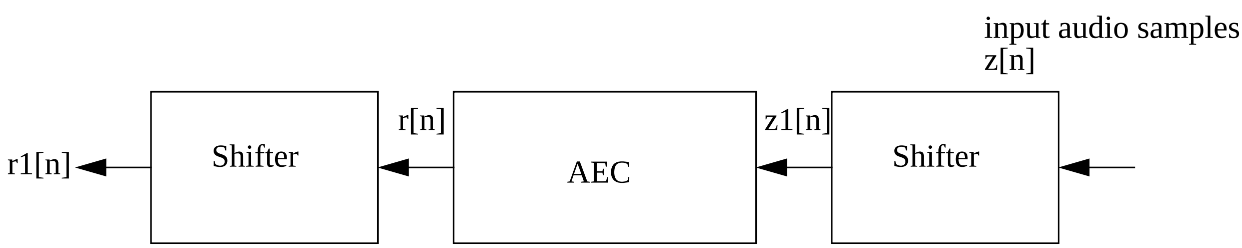
It is used to apply a scaling operation over the samples from the microphone and after treatment over the ouput samples.

This short word actually contains two informations, which are two signed 8 bits words, one from bit 0 to bit 7 (low part) and the other from bit 8 to bit 15.

Tableau 2 : shift

bit 15 - bit 8	bit 7 - bit 0
-32 <= shift_out <= 32	-32 <= shift_in <= 32

The input samples are shifted shift_in times to the left and the output samples are shifted shift_out time to the left. if the shift value is negative, then the shift is done to the right indeed !



r1[n] = r[n] << shift_out

z1[n] = z[n] << shift_in

V.1.1.4 dt_level2

This short word is related to the NLMS part double talk threshold. The actual constant is 2^dt_level2.

The adaptation step must go to full speed if there is no near-end signal. In case there is near-end signal, the adaptation step must be lowered.

For this purpose, the dt_level2 constant is used in the NLMS part to lower the adaptation step in case near-end signal is present.

In the software, the adaption step is lowered if (Px is the power of the far-end signal while Pr is the power of the residual signal which is the near-end signal minus the estimated echo)

$$\frac{P_x}{P_r} < 2^{dtlevel2}$$

This constant should be choosed in the way that the echo is not detected as near-end speech. Thus one must know or <u>estimate the attenuation between the loudspeaker and the microphone</u> in order to set this constant.

Suppose that the attenuation between the far-end signal and the near-end echo signal is 6 dB. And suppose that 2^dt_level2 refers to 10 dB (log10(2^dt_level2) = 1) then at initialisation, since the NLMS does not give enough ERLE, the attenuation between Px and Pr is less than 10 dB (10*log10(Pr/Px) < 10 dB) <u>so the adaptation step is lowered</u>. <u>But as soon as, the adaptive filter gives 4 dB of ERLE</u>, this means the total attenuation between Px and Pr is about 10 dB, then the adaptation goes to full speed.

So in order to optimize the NLMS, one have to know :

- **The total attenuation between the far-end signal and the echo. This can be measured.**
- **The ERLE we can expect from the adaptive filter. This can also be measured or estimated. For long impulse responses, the ERLE of the adaptive filter may be very low, but for impulse responses about the same size of the adaptive filter's length, the ERLE can be great. This has to be measured.**

In a future version, a new coding of this parameter will permit to set it in a more accurate way.

<u>Note : this parameter is negative.</u>

V.1.1.5 nlms_factor

nlms_factor is the step used to update the coefficients of the adaptive filter. The actual step is 2^nlms_factor/power of far-end signal. There are constraints related to this factor :

Stability of the adaptive filter : **nlms_step_factor < -6**

But because of the kind of algorithm used to identify the echo path, the mean power of the residual signal is proportional to 2^nlms_factor, to the power of the noise introduced in the microphone and to the noise due to the fact that the order of the adaptive filter is insufficient. Thus we have to choose the best step that lowers the mean power of the residual echo while keeping good tracking capabilities of the impulse response. Try **nlms_factor = -7.**

For those who are unfamiliar with NLMS algorithm, the step is of the form mu/N/P where N is the order of the filter and P the power of signal that is filtered. The filter is stable in practice for mu < 0.66 i.e. for mu/N < 0.66/64 = 0.01 # 2^-6.

Note : this parameter is negative.

V.1.1.6 hf_threshold

hf_threshold is used to decide wether or not we have to update the FIR filter of the NLMS part. hf_threshold should be choosed so that minimum misdecisions are made. Try 65535. As this parameter can have huge values, the EC should be modified in order to code this threshold in a logarithm form, for example as a power of 2. The parameter will be then the exponent.

Note : this parameter is unsigned short positive value.

V.1.1.7 alpha_rev

alpha_rev depends on the T60 reverberation time (T60 is in milliseconds) :

$$alpharev = 32768 \cdot \exp\left(-\frac{69}{T_{60}}\right)$$

For a car environnment we can take T_{60} = 60ms leading to **alpha_rev** = 10376

Note : this parameter is signed short positive value and lies between 0 and 32767.

V.1.1.8 erle

erle is Q_{15} factor used to limit the amount of the cancellation. The cancellation is limited to erle, that is to say that if the cancellation cannot be greater than 20*log10(1/erle). For an unlimited cancellation, use erle = 0, that is the default value at this date.

V.1.1.9 beta_rev

This short word in Q_{15} format (That is to say that the real value is the signed short integer value divided by the scaling factor 32768) is related to the low pass filter used to smooth the gains that are applyed over the FFT of signal (z-y). Signal z is the microphone signal while signal y is the estimated echo from the NLMS part. See alpha_rev.

Using too low values lead to musical tones but good capabilities for tracking the variations of the impulse response. Using too high capabilities reduce the tracking capabilities but the cancellation can be bad at transition between presence of echo and non presence of echo (the phenomenon often occurs at the beginning of a sentence).

We decided to emphasize on good cancellation and tracking capabilities.

Note : this parameter is positive value and lies between 0 and 32767.

V.1.1.10 gamma_nsp

The short word in Q_{14} format gamma_nsp is the over substraction factor used for the case of non near-end speech. In this case we can do over substraction, as the near-end signal contains only the echo of the far-end speech.

So we can use value greater than 16384.

Note : this parameter is positive value and lies between 0 and 32767.

V.1.1.11 gamma_sp

The short word in Q_{14} format gamma_sp is the over substraction factor used for the case of presence of near-end speech. In this case we should lower the over substraction, as the near-end signal contains the echo plus possibly the far-end speech.

Note : this parameter is positive value and lies between 0 and 32767.

V.1.1.12 spdet

This short word in Q_{14} format is used in the near-end speech detection of the DES part.

Near-end speech is detected if P_s > spdet * P_y where P_s is the power of the near-end signal and P_y is the power of the estimated echo from the NLMS part. This has to be tuned by trial and error. See (cf. § VI. page 12)

Note : this parameter is positive value and lies between 0 and 32767.

V.1.1.13 ycomp

ycomp is a short word in Q_{15} format. At this date it is not documented and is set to zero.

Note : this parameter is positive value and lies between 0 and 32767.

V.1.1.14 history_buffer_size

history_buffer_size is the size of the EC_fifo we want to use. The EC_fifo has a nominal size fixed in the firmware. But to allow to use the end of the fifo for additionial functions such as delay compensation, the argument passed to the AEC's initialization functions can be lower than this nominal value. For the Av1 MT firmware, the nominal size if 449. We use 447 because we introduce the delay compensation.

V.1.1.15 ptr_ls

This is value of the pointer used to read the samples from the EC_fifo. At initialisation the pointer used to write samples to the EC_fifo is set at the beggining of the fifo. Because there is a delay between the x samples and the z samples, ptr_ls is set to the beggining of the fifo minus the delay, this modulo the size of the fifo.

There is an incompressible delay due to the MT architecture that is equal to audio_fifo_size samples. Thus ptr_ls should have the value -audio_fifo_size + EC_fifo_size = -254 + 447 = 193. This is the default value for handset operation. For car kit the delay is greater so ptr_ls < 193.

For handset operation, a delay of 10 samples is taken into account => ptr_ls = 183.

V.2 Configuration for handset echo cancellation

V.2.1 Default configuration

Tableau 3 : default configuration for handset mode

Name	Default value, starting point	Comment
EC_mode	0x211D	one delay compensation after initialization. Max delay = 33.
smooth_factor	-6 (0xFFFA)	
shift	0x0000	
dt_level2	-7 (0xFFF9)	to be tuned
nlms_factor	-7 (0xFFF9)	
hf_threshold	4096 (0x1000)	
alpha_rev	104 (0x0068)	to be tuned
erle	0	
beta_rev	24576 (0x6000)	to be tuned
32768-beta_rev	32768-24576	to be tuned
gamma_nsp	24576 (0x6000)	to be tuned
gamman_sp	18000 (0x4650)	to be tuned
spdet	28672 (0x7000)	to be tuned
ycomp	0	
history_buffer_size	447 = 0x01BF	
ptr_ls	447-254 - 10 = 183 (0x00B7)	delay = 10

We expect that the coupling between speakerphone and microphone is linear.

NOTE : The delay compensation feature must be dowloaded in addition to the initial MT AV1 DSP firmware if one expect to use it.

V.2.2 Where and when to do initialization.

The initialization should be done a priori every time the audios are opened as the echo patch may has changed. There is a particuliar situation where the audios are not stopped and reopened but where the AEC may need to be restarted.

When an handover occurs, the audios interrupts are not freezed. This means that the last samples from the audio fifo are still sent to the earpiece while the incoming audio samples from the microphone are stored in the audio fifo. Thus the echo patch is not changed. The handover is schematically handled as follows :

- stop tch.
- start tch with audio synchro ordered.

This clearly means that the acoustical path is not changed but we still have to reset the EC_fifo and freeze the update of the adaptive filter until the audio synchronization is completed. The key is to stop the update of the NLMS part until audio synchronization is finished. This applies also to the carkit operation. In the current implementation of the software, some sound artefacts can be heard at the far-end side.

V.3 Configuration for handsfree car kit operation

In order to use the AEC, one must verify that the coupling between speakerphone /

microphone is linear.

Name	Default value	Comment
EC_mode	0x211D	Delay compensation is always done. Max delay = 33.
smooth_factor	-5 (0xFFFB)	
shift	0x0000	
dt_level2	-2 (0xFFFE)	to be tuned
nlms_factor	-6 (0xFFFA)	
hf_threshold	(0x4000)	
alpha_rev	(0x3600)	related to T60 = 80 ms. to be tuned
erle	0	
beta_rev	(0x5000)	to be tuned
32768-beta_rev		to be tuned
gamma_nsp	(0x7FFF)	to be tuned
gamman_sp	(0x4800)	to be tuned
spdet	0x6000	to be tuned
ycomp	0	
history_buffer_size	447 (0x01BF)	
ptr_ls	447-254-25 = 173 (0x00A8)	Initial delay = 15 + 10

VI. Tuning the DES

For each kind of application, either handset echo cancellation or echo cancellation in a car, some work has to be done to best tune the parameters. In the following of the text I will give directives to help in tuning the AEC.

As the AEC has several parameters and the performance depends on them, it is difficult to, a priori, tune all the parameters at the same time. Hopefully, some parameters can be tuned independently.

We can split the parameters into families :

- gamma_nsp, gamma_sp, spdet that belongs to double talk handling.
- alpha_rev, ycomp that belongs to T60 reverberation time.
- beta_rev has to be tuned by trial and error.

VI.1 alpha_rev ycomp

alpha_rev depends on the reverberation time of the impulse response. So it can be fixed if we know the environnment of the application. For a carkit, the reverberation is around 50 ms - 60 ms. For the handset echo it is more problematic as the impulse response decays rapidly.

As the adaptive filter estimates the echo that is due to the first samples of the impulse response, alpha_rev is involved in estimating the power of the echo that is due to the tail of the impulse response.

Sometimes the impulse response does not decay as it is expected. It decays more rapidly and in order to avoid distortion of the near speech, we have to compensate for that. Thus ycomp should be greater than 0. The tuning of this parameter should be done by trial and error.

VI.2 gamma_nsp gamma_sp spdet

These parameters impact the way the AEC works during double talk and single talk. In particuliar, spdet is an important threshold on which the AEC relies, to decide wether there is double talk or not. Depending on this decision, the AEC chooses one of the two oversubstraction parameters :

- gamma_nsp if there is no double talk
- gamma_sp is there is double talk.

We will consider in the following that the double talk problem can be reduced to the detection of the near-end speech. In the case there is only near-end speech, the AEC detects double talk. Does this mean that it make a misdecision. In practice this is no important as that if there is no far-end signal, there is no echo further, and choosing any oversubstraction leads roughly to the same results.

The procedure to tune these parameters is as follows :

- First, let’s tune spdet. Set :
  - gamma_sp to the lowest value (i.e. 0) => No suppression if selected by the algorithm
  - gamma_nsp to the highest value (i.e. 32767) => Maximum suppression if selected.
- Then you will be able to hear echoes only when gamma_sp is selected, i.e. during doubletalk. You have to set then spdet with trial and error. If you can hear echoes during far-end talk only, then it means that spdet is too low. If near-end speech is suppressed too much then it means that spdet is too high.
- Once spdet has been set correctly, then you can set gamma_nsp is adjusted so that near-end speech is not distorted too much and the echoes during double talk are not annoying.
- Then you can go on tuning gamma_nsp. gamma_nsp has to be set so that no echoes can be heard during far-end only speech and the beginning and end-phases of the near-end speech is not distorted too much.

VI.3 beta_rev

beta_rev is a problematic parameter. If it is too low, then the filter shape of the DES varies rapidly and some musical tones can ne heard. The near-end speech can also be distorted. If it is too high then it will make the DES filter shape vary slowly and thus track the echo path variations slowly.

This parameter has to be tuned by trial and error.