Acoustic EC Software Library

Profiling has been done in order to evaluate the resource consumption in terms of MIPS. RAM and FLASH figures may change depending on specific use case optimizations. The following figure shows the profiling results for acoustic echo cancellation using a STM32F446 MCU based on an ARM M4 core with floating point unit running at 168 MHz (210 DMIPS available) and the IAR embedded workbench tool chain, version 7.70. Optimization has been set on High, speed.

<table>
<thead>
<tr>
<th>Options</th>
<th>Tail Length</th>
<th>CPU (MIPS)</th>
<th>FLASH (Bytes)</th>
<th>RAM (Bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AEC</td>
<td>128</td>
<td>34.1</td>
<td>32460</td>
<td>24412</td>
</tr>
<tr>
<td>AEC + Denoiser</td>
<td>128</td>
<td>73.9</td>
<td>43236</td>
<td></td>
</tr>
<tr>
<td>AEC + Denoiser + AGC</td>
<td>128</td>
<td>75.2</td>
<td>43236</td>
<td></td>
</tr>
<tr>
<td>AEC</td>
<td>512</td>
<td>47.5</td>
<td>35164</td>
<td></td>
</tr>
<tr>
<td>AEC + Denoiser</td>
<td>512</td>
<td>87.4</td>
<td>32460</td>
<td>53988</td>
</tr>
<tr>
<td>AEC + Denoiser + AGC</td>
<td>512</td>
<td>88.8</td>
<td>53988</td>
<td></td>
</tr>
</tbody>
</table>

Memory footprint depends on the length of the filter tail. Data has been collected using a filter tail length equal to 128 and 512.
Here is a list of all modules:

<table>
<thead>
<tr>
<th>MIDDLEWARES</th>
<th>ACOUSTIC_EC</th>
<th>AcousticEC Exported Constants</th>
<th>AcousticEC Exported Types</th>
<th>AcousticBF Exported Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Acoustic_EC_errors</td>
<td>Echo cancellation errors</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Acoustic_EC_preprocessor</td>
<td>Echo cancellation preprocessor</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Acoustic EC Software Library

<table>
<thead>
<tr>
<th>Main Page</th>
<th>Related Pages</th>
<th>Modules</th>
<th>Data Structures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Files</td>
<td>Profiling</td>
<td></td>
<td>Modules</td>
</tr>
</tbody>
</table>

MIDDLEWARES
<table>
<thead>
<tr>
<th>Modules</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACOUSTIC_EC</td>
</tr>
</tbody>
</table>
# Acoustic EC Software Library

<table>
<thead>
<tr>
<th>Main Page</th>
<th>Related Pages</th>
<th>Modules</th>
<th>Data Structures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Files</td>
<td>Profiling</td>
<td>Modules</td>
<td></td>
</tr>
</tbody>
</table>

**ACOUSTIC_EC**  
MIDDLEWARES
## Modules

<table>
<thead>
<tr>
<th>Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AcousticEC Exported Constants</td>
<td></td>
</tr>
<tr>
<td>AcousticEC Exported Types</td>
<td></td>
</tr>
<tr>
<td>AcousticBF Exported Functions</td>
<td></td>
</tr>
</tbody>
</table>
# Acoustic EC Software Library

<table>
<thead>
<tr>
<th>Main Page</th>
<th>Related Pages</th>
<th>Modules</th>
<th>Data Structures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Files</td>
<td>Profiling</td>
<td>Modules</td>
<td></td>
</tr>
</tbody>
</table>

**AcousticEC Exported Constants**

MIDDLEWARES » ACOUSTIC_EC
<table>
<thead>
<tr>
<th><strong>Modules</strong></th>
</tr>
</thead>
</table>
| **Acoustic_EC_errors**  
Echo cancellation errors. |
| **Acoustic_EC_preprocessor**  
Echo cancellation preprocessor. |
Acoustic EC Software Library

Acoustic_EC_errors

Echo cancellation errors. More...
### Macros

```c
#define ACOUSTIC_EC_TAIL_LENGTH_ERROR ((uint32_t)0x00000001)
#define ACOUSTIC_EC_AEC_LEVEL_ERROR  ((uint32_t)0x00000002)
#define ACOUSTIC_EC_PTR_CHANNELS_ERROR  ((uint32_t)0x00000004)
#define ACOUSTIC_EC_PREPROCESS_ERROR  ((uint32_t)0x00000010)
#define ACOUSTIC_LOCK_ERROR  ((uint32_t)0x10000000)
```
Detailed Description

Echo cancellation errors.
Acoustic EC Software Library

**Acoustic_EC_preprocessor**
MIDDLEWARES » ACOUSTIC_EC »
AcousticEC Exported Constants

Echo cancellation preprocessor. More...
### Macros

```c
#define ACOUSTIC_EC_PREPROCESS_ENABLE ((uint32_t)0x00000001)
```

```c
#define ACOUSTIC_EC_PREPROCESS_DISABLE ((uint32_t)0x00000000)
```
Detailed Description

Echo cancellation preprocessor.

Generated by doxygen 1.8.11
# Acoustic EC Software Library

<table>
<thead>
<tr>
<th>Main Page</th>
<th>Related Pages</th>
<th>Modules</th>
<th>Data Structures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Files</td>
<td>Profiling</td>
<td></td>
<td>Data Structures</td>
</tr>
</tbody>
</table>

## AcousticEC Exported Types

MIDDLEWARES » ACOUSTIC_EC
### Data Structures

<table>
<thead>
<tr>
<th>struct</th>
<th><strong>AcousticEC_Handler_t</strong></th>
<th>Library handler. It keeps track of the static parameters and it handles the internal state of the algorithm. More...</th>
</tr>
</thead>
<tbody>
<tr>
<td>struct</td>
<td><strong>AcousticEC_Config_t</strong></td>
<td>Library dynamic configuration handler. It contains dynamic parameters. More...</td>
</tr>
</tbody>
</table>
Detailed Description

Generated by doxygen 1.8.11
Acoustic EC Software Library

AcousticEC_Handler_t
Struct Reference

MIDDLEWARES » ACOUSTIC_EC »
AcousticEC Exported Types

Library handler. It keeps track of the static parameters and it handles the internal state of the algorithm. More...

#include <acoustic_ec.h>
## Data Fields

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>uint16_t</td>
<td>tail_length</td>
</tr>
<tr>
<td>uint32_t</td>
<td>preprocess_init</td>
</tr>
<tr>
<td>uint16_t</td>
<td>ptr_primary_channels</td>
</tr>
<tr>
<td>uint16_t</td>
<td>ptr_reference_channels</td>
</tr>
<tr>
<td>uint16_t</td>
<td>ptr_output_channels</td>
</tr>
<tr>
<td>uint32_t</td>
<td>internal_memory_size</td>
</tr>
<tr>
<td>uint32_t*</td>
<td>pInternalMemory</td>
</tr>
</tbody>
</table>
Detailed Description

Library handler. It keeps track of the static parameters and it handles the internal state of the algorithm.
Field Documentation

`uint32_t internal_memory_size`

Keeps track of the amount of memory required for the current setup. It's filled by the `libSpeexAEC_getMemorySize()` function and must be used to allocate the right amount of RAM.

`uint32_t* plInternalMemory`

Pointer to the memory allocated by the user.

`uint32_t preprocess_init`

Specifies the option for the preprocessor uinitialization. This parameter can be a value of `Acoustic_EC_preprocessor`.

`uint16_t ptr_output_channels`

Specifies the number of channel in the Output Stream. Default Value is 1.

`uint16_t ptr_primary_channels`

Specifies the number of channel in the Primary Stream. Default Value is 1.

`uint16_t ptr_reference_channels`
Specifies the number of channel in the Reference Stream. Default Value is 1.

**uint16_t tail_length**

Specifies the length of the filter tail. Default Value is 512.

The documentation for this struct was generated from the following file:

- D:/Documents/Repositories/DMIC/Private/Fw/OSX_Libraries/EchoCancellation/trunk/export/
Acoustic EC Software Library

AcousticEC_Config_t
Struct Reference
MIDDLEWARES » ACOUSTIC_EC » AcousticEC Exported Types

Library dynamic configuration handler. It contains dynamic parameters. More...

#include <acoustic_ec.h>
<table>
<thead>
<tr>
<th>Data Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>uint32_t</strong> preprocess_state</td>
</tr>
<tr>
<td><strong>uint32_t</strong> AGC_value</td>
</tr>
<tr>
<td><strong>uint32_t</strong> residual_echo_remove</td>
</tr>
<tr>
<td><strong>int32_t</strong> noise_suppress_default</td>
</tr>
<tr>
<td><strong>int32_t</strong> echo_suppress_default</td>
</tr>
<tr>
<td><strong>int32_t</strong> echo_suppress_active</td>
</tr>
</tbody>
</table>
Detailed Description

Library dynamic configuration handler. It contains dynamic parameters.
## Field Documentation

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>uint32_t AGC_value</code></td>
<td>Specifies the threshold for the AGC, if activated</td>
</tr>
<tr>
<td><code>int32_t echo_suppress_active</code></td>
<td>Specifies the echo suppress active parameter of the preprocessor</td>
</tr>
<tr>
<td><code>int32_t echo_suppress_default</code></td>
<td>Specifies the echo suppress default parameter of the preprocessor</td>
</tr>
<tr>
<td><code>int32_t noise_suppress_default</code></td>
<td>Specifies the noise suppress default parameter of the preprocessor</td>
</tr>
<tr>
<td><code>uint32_t preprocess_state</code></td>
<td>Enable or disable pre-process function</td>
</tr>
<tr>
<td><code>uint32_t residual_echo_remove</code></td>
<td>Activate residual echo removal</td>
</tr>
</tbody>
</table>

The documentation for this struct was generated from the following file:
• D:/Documents/Repositories/DMIC/Private/Fw/OSX_Libraries/EchoC

Generated by doxygen 1.8.11
Acoustic EC Software Library

<table>
<thead>
<tr>
<th>Main Page</th>
<th>Related Pages</th>
<th>Modules</th>
<th>Data Structures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Files</td>
<td>Profiling</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

AcousticBF Exported Functions

MIDDLEWARES » ACOUSTIC_EC
<table>
<thead>
<tr>
<th>Function Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>uint32_t AcousticEC_getMemorySize (AcousticEC_Handler_t *pHandler)</code></td>
<td>Fills the &quot;internal_memory_size&quot; of the pHandler parameter passed as argument with a value representing the right amount of memory needed by the library, depending on the specific static parameters adopted. More...</td>
</tr>
<tr>
<td><code>uint32_t AcousticEC_Init (AcousticEC_Handler_t *pHandler)</code></td>
<td>Library initialization. More...</td>
</tr>
<tr>
<td><code>uint32_t AcousticEC_Data_Input (void *ptrPrimary, void *ptrReference, void *ptrBufferOut, AcousticEC_Handler_t *pHandler)</code></td>
<td>Library data input and output function. More...</td>
</tr>
<tr>
<td><code>uint32_t AcousticEC_Process (AcousticEC_Handler_t *pHandler)</code></td>
<td>Library run function, performs audio analysis when all required data has been collected. More...</td>
</tr>
<tr>
<td><code>uint32_t AcousticEC_setConfig (AcousticEC_Handler_t *pHandler, AcousticEC_Config_t *pConfig)</code></td>
<td>Library setup function, it sets the values for dynamic parameters. It can be called at runtime to change dynamic parameters. More...</td>
</tr>
<tr>
<td><code>uint32_t AcousticEC_getConfig (AcousticEC_Handler_t *pHandler, AcousticEC_Config_t *pConfig)</code></td>
<td>Fills the pConfig structure with the actual dynamic parameters as they are used inside the library. More...</td>
</tr>
<tr>
<td><code>uint32_t AcousticEC_GetLibVersion (char *version)</code></td>
<td>To be used to retrieve version information. More...</td>
</tr>
</tbody>
</table>
Detailed Description

Function Documentation

```c
uint32_t AcousticEC_Data_Input ( void * ptrPrimary, void * ptrReference, void * ptrBufferOut, AcousticEC_Handler_t * pHandler )
```

Library data input and output function.

Parameters

- **ptrPrimary**: pointer to an array that contains PCM samples (16 bit signed int) of the primary signal (1 ms = 16 samples at 16 KHZ).
- **ptrReference**: pointer to an array that contains PCM samples (16 bit signed int) of the reference (echo) signal (1 ms = 16 samples at 16 KHZ).
- **ptrBufferOut**: pointer to an array that will contain output PCM samples (1 ms for processed channel = 16 samples at 16 KHZ).
- **pHandler**: pointer to the handler of the current SpeexAEC instance running.

Return values

1 if data collection is finished and libSpeexAEC_Process must be called, 0 otherwise.

Note

Input/output function reads and write samples skipping the
required number of values depending on the ptr_Mx_channels configuration

```c
uint32_t AcousticEC_getConfig ( AcousticEC_Handler_t * pHandler,
                                AcousticEC_Config_t * pConfig
                            )
```

Fills the pConfig structure with the actual dynamic parameters as they are used inside the library.

**Parameters**
- **pHandler** AcousticEC_Handler_t filled with desired parameters.
- **pConfig** pointer to the dynamic parameters handler that will be filled with the current library configuration.

**Return values**
- 0 if everything is ok, 1 otherwise.

```c
uint32_t AcousticEC_GetLibVersion ( char * version )
```

To be used to retrieve version information.

**Parameters**
- **none**

**Return values**
- **Version** Number.

```c
uint32_t AcousticEC_getMemorySize ( AcousticEC_Handler_t * pHandler )
```

Fills the "internal_memory_size" of the pHandler parameter passed as argument with a value representing the right amount of memory.
needed by the library, depending on the specific static parameters adopted.

**Parameters**

- `pHandler AcousticEC_Handler_t` filled with desired parameters.

**Return values**

- 0 if everything is fine.

```c
uint32_t AcousticEC_Init ( AcousticEC_Handler_t * pHandler )
```

Library initialization.

**Parameters**

- `pHandler AcousticEC_Handler_t` filled with desired parameters.

**Return values**

- 0 if everything is fine. Different from 0 if erroneous parameters have been passed to the Init function and the default value has been used. The specific error can be recognized by checking the relative bit in the returned word.

```c
uint32_t AcousticEC_Process ( AcousticEC_Handler_t * pHandler )
```

Library run function, performs audio analysis when all required data has been collected.

**Parameters**

- `pHandler` pointer to the handler of the current SpeexAEC instance running

**Return values**

- 0 if everything is ok, 1 otherwise.
uint32_t AcousticEC_setConfig (AcousticEC_Handler_t * pHandler,
    AcousticEC_Config_t * pConfig)

Library setup function, it sets the values for dynamic parameters. It can be called at runtime to change dynamic parameters.

Parameters
  * pHandler AcousticEC_Handler_t filled with desired parameters.
  * pConfig pointer to the dynamic parameters handler that will be filled with the current library configuration.

Return values
  0 if everything is ok, 1 otherwise.
Data Structures

Here are the data structures with brief descriptions:

<table>
<thead>
<tr>
<th>Data Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AcousticEC_Config_t</td>
<td>Library dynamic configuration handler. It contains dynamic parameters</td>
</tr>
<tr>
<td>AcousticEC_Handler_t</td>
<td>Library handler. It keeps track of the static parameters and it handles the internal state of the algorithm</td>
</tr>
</tbody>
</table>
# Data Structure Index

<table>
<thead>
<tr>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>AcousticEC_Handler_t</td>
</tr>
<tr>
<td>AcousticEC_Config_t</td>
</tr>
<tr>
<td>A</td>
</tr>
</tbody>
</table>

Generated by [doxygen](https://www.stackoverflow.com) 1.8.11
Here is a list of all documented struct and union fields with links to the struct/union documentation for each field:

- AGC_value: `AcousticEC_Config_t`
- echo_suppress_active: `AcousticEC_Config_t`
- echo_suppress_default: `AcousticEC_Config_t`
- internal_memory_size: `AcousticEC_Handler_t`
- noise_suppress_default: `AcousticEC_Config_t`
- pInternalMemory: `AcousticEC_Handler_t`
- preprocess_init: `AcousticEC_Handler_t`
- preprocess_state: `AcousticEC_Config_t`
- ptr_output_channels: `AcousticEC_Handler_t`
- ptr_primary_channels: `AcousticEC_Handler_t`
- ptr_reference_channels: `AcousticEC_Handler_t`
- residual_echo_remove: `AcousticEC_Config_t`
- tail_length: `AcousticEC_Handler_t`
### Acoustic EC Software Library

- AGC_value: `AcousticEC_Config_t`
- echo_suppress_active: `AcousticEC_Config_t`
- echo_suppress_default: `AcousticEC_Config_t`
- internal_memory_size: `AcousticEC_Handler_t`
- noise_suppress_default: `AcousticEC_Config_t`
- ptrInternalMemory: `AcousticEC_Handler_t`
- preprocess_init: `AcousticEC_Handler_t`
- preprocess_state: `AcousticEC_Config_t`
- ptr_output_channels: `AcousticEC_Handler_t`
- ptr_primary_channels: `AcousticEC_Handler_t`
- ptr_reference_channels: `AcousticEC_Handler_t`
- residual_echo_remove: `AcousticEC_Config_t`
- tail_length: `AcousticEC_Handler_t`
## File List

Here is a list of all documented files with brief descriptions:

<table>
<thead>
<tr>
<th>[detail level 1 2 3 4]</th>
<th>▼ EchoCancellation</th>
<th>trunk</th>
<th>export</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>acousti<strong>c_ec.h</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Generated by [doxygen](http://www.doxygen.org) 1.8.11
# Acoustic EC Software Library

<table>
<thead>
<tr>
<th>Main Page</th>
<th>Related Pages</th>
<th>Modules</th>
<th>Data Structures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Files</td>
<td>Profiling</td>
<td></td>
<td></td>
</tr>
<tr>
<td>File List</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EchoCancellation</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

EchoCancellation Directory Reference
# Acoustic EC Software Library

<table>
<thead>
<tr>
<th>Main Page</th>
<th>Related Pages</th>
<th>Modules</th>
<th>Data Structures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Files</td>
<td>Profiling</td>
<td></td>
<td></td>
</tr>
<tr>
<td>File List</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EchoCancellation</td>
<td>trunk</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**trunk Directory Reference**
Acoustic EC Software Library

<table>
<thead>
<tr>
<th>Main Page</th>
<th>Related Pages</th>
<th>Modules</th>
<th>Data Structures</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Files</strong></td>
<td><strong>Profiling</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>File List</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EchoCancellation</td>
<td>trunk</td>
<td>export</td>
<td></td>
</tr>
</tbody>
</table>

**export Directory Reference**
# acoustic_ec.h

```c
/* Define to prevent recursive inclusion -------------------------------*/
#ifdef __ACOUSTIC_EC_H
#define __ACOUSTIC_EC_H

/* Includes ----------------------------------*/
#include "stdint.h"

/* Exported constants -------------------------------*/

/* Exported macro --------------------------------*/

/* Exported define --------------------------------*/

/* Exported functions -------------------------------*/

/* Exported types --------------------------------*/

#define ACOUSTIC_EC_TAIL_LENGTH_ERROR ((uint32_t)0x00000001)
#define ACOUSTIC_EC_AEC_LEVEL_ERROR ((uint32_t)0x00000002)
#define ACOUSTIC_EC_PTR_CHANNELS_ERROR
```
#define ACOUSTIC_EC_PREPROCESS_ERROR ((uint32_t)0x00000004)

#ifndef ACOUSTIC_LOCK_ERROR
#define ACOUSTIC_LOCK_ERROR
((uint32_t)0x10000000)
#endif

#define ACOUSTIC_EC_PREPROCESS_ENABLE ((uint32_t)0x00000001)

#define ACOUSTIC_EC_PREPROCESS_DISABLE ((uint32_t)0x00000000)

typedef struct {
    uint16_t tail_length;
    uint32_t preprocess_init;
    uint16_t ptr_primary_channels;
    uint16_t ptr_reference_channels;
    uint16_t ptr_output_channels;
    uint32_t internal_memory_size;
    uint32_t * pInternalMemory;
} AcousticEC_Handler_t;

typedef struct {
    uint32_t preprocess_state;
    uint32_t AGC_value;
    uint32_t residual_echo_remove;
    int32_t noise_suppress_default;
    int32_t echo_suppress_default;
    int32_t echo_suppress_active;
} AcousticEC_Config_t;
uint32_t AcousticEC_getMemorySize(AcousticEC_Handler_t * pHandler);

uint32_t AcousticEC_Init(AcousticEC_Handler_t * pHandler);

uint32_t AcousticEC_Data_Input(void *ptrPrimary, void *ptrReference, void *ptrBufferOut, AcousticEC_Handler_t * pHandler);

uint32_t AcousticEC_Process(AcousticEC_Handler_t * pHandler);

uint32_t AcousticEC_setConfig(AcousticEC_Handler_t * pHandler, AcousticEC_Config_t * pConfig);

uint32_t AcousticEC_getConfig(AcousticEC_Handler_t * pHandler, AcousticEC_Config_t * pConfig);

uint32_t AcousticEC_GetLibVersion(char *version);

#endif /*__ACOUSTIC_EC_H*/
Related Pages

Here is a list of all related documentation pages:

profiling