

Typedefs

typedef struct **AWEOSAudioRecordNotification** **AWEOSAudioRecordNotification_t**
Audio recording notification callback argument structure.

typedef void(* **recordNotificationCallbackFunction**)
(**AWEOSAudioRecordNotification_t** *)
Audio recording notification callback type.

typedef struct **AWEOSVersionInfo** **AWEOSVersionInfo_t**
Versioning structure returned by **aweOS_getVersion**.

typedef void **AWEOSInstance**
The AWE Core OS Instance instance type. [More...](#)

typedef struct **AWEOSConfigParameters** **AWEOSConfigParameters**
AWEOSConfigParameters. [More...](#)

Functions

INT32 **aweOS_getParamDefaults** (**AWEOSConfigParameters** *aweParams)
Populates an AWEOSConfigParameters structure with defaults. [More...](#)

INT32 **aweOS_init** (**AWEOSInstance** **pAWEOS, const **AWEOSConfigParameters** *aweParams, const void *pModuleDescriptorTable, UINT32 moduleDescriptorTableSize)
Initialize the AWEOSInstance with the specified configuration parameters. [More...](#)

INT32 **aweOS_tuningSocketOpen** (**AWEOSInstance** **pAWEOS, INT32 portNo, UINT32 numInstances)
Initialize and open an integrated TCP/IP tuning interface socket. [More...](#)

void **aweOS_tuningSocketClose** (**AWEOSInstance** *pAWEOS)
Close a running integrated TCP tuning socket. [More...](#)

INT32 **aweOS_tuningLoggingEnable** (**AWEOSInstance** *pAWEOS, char *path, char *baseName, UINT32 verbosity)
Enable logging of the tuning packets sent and received by the AWEOSInstance. [More...](#)

INT32 **aweOS_loadAWBFromArray** (**AWEOSInstance** *pAWEOS, const UINT32 *pArray, UINT32 arraySize, UINT32 *pErrorOffset)
Executes packet commands from an in-memory array. [More...](#)

INT32 **aweOS_loadAWBFile** (**AWEOSInstance** *pAWEOS, const char *binaryFile, UINT32 *pErrorOffset)
Executes packet commands from an AWB file on the filesystem. [More...](#)

INT32 **aweOS_layoutGetChannelCount** (const **AWEOSInstance** *pAWEOS, UINT32 *inCount, UINT32 *outCount)
Returns the number of input and output channels in the loaded layout. [More...](#)

INT32 **aweOS_layoutGetBlockSize** (const **AWEOSInstance** *pAWEOS, UINT32 *blockSize)
Returns the block size of the loaded layout. [More...](#)

INT32 **aweOS_layoutGetSampleRate** (const **AWEOSInstance** *pAWEOS, FLOAT32 *sampleRate)

Returns the sample rate of the loaded layout. [More...](#)

INT32 **aweOS_audioPumpAll** (**AWEOSInstance** *pAWEOS)
Pump one fundamental block size of audio through the loaded layout and all of its sublayouts.
[More...](#)

INT32 **aweOS_audioImportSamples** (**AWEOSInstance** *pAWEOS, void *inSamples, INT32 inStride, INT32 channel, **SampleType** inType)
Import samples from an audio buffer to a specific channel of the AWEOSInstance's input pin.
[More...](#)

INT32 **aweOS_audioExportSamples** (**AWEOSInstance** *pAWEOS, void *outSamples, INT32 outStride, INT32 channel, **SampleType** outType)
Export samples to a user buffer from a specific channel of the AWEOSInstance's output pin. [More...](#)

INT32 **aweOS_layoutIsValid** (const **AWEOSInstance** *pAWEOS)
Determines if a layout is loaded and valid. [More...](#)

INT32 **aweOS_audiolsStarted** (const **AWEOSInstance** *pAWEOS)
Check if this instance has received an Audio Start command. [More...](#)

const char * **aweOS_errorToString** (INT32 errorCode)
Convert an error code (INT32) to its corresponding error string. [More...](#)

INT32 **aweOS_tuningPacketProcess** (**AWEOSInstance** *pAWEOS)
Process the packet buffer in the AWEOSInstance. [More...](#)

INT32 **aweOS_ctrlSetValue** (const **AWEOSInstance** *pAWEOS, UINT32 handle, void *value, INT32 arrayOffset, UINT32 length)
Set a scalar or array value(s) of a module parameter by handle. [More...](#)

INT32 **aweOS_ctrlGetValue** (const **AWEOSInstance** *pAWEOS, UINT32 handle, void *value, INT32 arrayOffset, UINT32 length)
Get a scalar or array value(s) of a module parameter by handle. [More...](#)

INT32 **aweOS_ctrlSetStatus** (const **AWEOSInstance** *pAWEOS, UINT32 handle, UINT32 *status)
Set the status of a module. [More...](#)

INT32 **aweOS_ctrlGetStatus** (const **AWEOSInstance** *pAWEOS, UINT32 handle, UINT32 *status)
Get the status of a module. [More...](#)

INT32 **aweOS_ctrlSetValueMask** (const **AWEOSInstance** *pAWEOS, UINT32 handle, void *value, INT32 arrayOffset, UINT32 length, UINT32 mask)
Set a scalar or array value(s) of a module variable by handle with mask. [More...](#)

INT32 **aweOS_ctrlGetValueMask** (const **AWEOSInstance** *pAWEOS, UINT32 handle, void *value, INT32 arrayOffset, UINT32 length, UINT32 mask)
Get a scalar or array value(s) of a module variable by handle with mask. [More...](#)

INT32 **aweOS_ctrlGetModuleClass** (const **AWEOSInstance** *pAWEOS, UINT32 handle, UINT32 *pClassID)
Get a module's object class from its handle. [More...](#)

INT32 **aweOS_destroy** (**AWEOSInstance** **pAWEOS)

Destroys the AWEOSInstance and closes all associated threads. [More...](#)

INT32 **aweOS_setProfilingStatus** (AWEOSInstance *pAWEOS, UINT32 enable)
Enable or disable the profiling ability of the AWE Core OS Instance. [More...](#)

INT32 **aweOS_getAverageLayoutCycles** (AWEOSInstance *pAWEOS, UINT32 idx, UINT32 *averageCycles)
Get the average cycles of a running layout, in units of cycles at profileSpeed. [More...](#)

void **aweOS_getVersion** (AWEOSVersionInfo_t *versionInfo)
Get the version information of the AWE Core OS library. [More...](#)

INT32 **aweOS_audioRecordingEnable** (AWEOSInstance *pAWEOS, char *path, char *baseName, UINT32 bufferSize, SampleType sampleType)
Enables recording of all input and output audio of AWEOSInstance. [More...](#)

INT32 **aweOS_audioRecordingRegisterNotificationCallback** (AWEOSInstance *pAWEOS, recordNotificationCallbackFunction recordNotificationCallback, UINT32 recordNotificationMask)
Register a callback function for audio recording event notifications. [More...](#)

INT32 **aweOS_audioRecordingDisable** (AWEOSInstance *pAWEOS)
Disable recording of input and output audio of AWEOSInstance. [More...](#)

INT32 **aweOS_wavFileOpen** (const char *file, FLOAT32 *sampleRate, UINT32 *numChannels, UINT32 *sampleSize, UINT32 *numSamples, FILE **fp)
Open a .wav file and populate the user arguments with the header information in the file. [More...](#)

INT32 **aweOS_wavFileCreate** (const char *file, FLOAT32 sampleRate, UINT32 numChannels, UINT32 sampleSize, FILE **fp)
Create a .wav file and populate the header with the passed in arguments. [More...](#)

INT32 **aweOS_wavFileWrite** (FILE *fp, void *samples, UINT32 numSamples, UINT32 sampleSize)
Write audio data to .wav file created using **aweOS_wavFileCreate**. [More...](#)

INT32 **aweOS_wavFileRead** (FILE *fp, void *samples, UINT32 numSamples, UINT32 sampleSize)
Read audio data from .wav file opened using **aweOS_wavFileOpen**. [More...](#)

INT32 **aweOS_wavFileClose** (FILE *fp)
Close the .wav file opened using **aweOS_wavFileOpen** or **aweOS_wavFileCreate**. [More...](#)
