/\*\*

@copyright COPYRIGHT 2017 - PROPERTY OF TOBII AB

@copyright 2017 TOBII AB - KARLSROVAGEN 2D, DANDERYD 182 53, SWEDEN - All Rights Reserved.

@copyright NOTICE: All information contained herein is, and remains, the property of Tobii AB and its suppliers, if any. The intellectual and technical concepts contained herein are proprietary to Tobii AB and its suppliers and may be covered by U.S.and Foreign Patents, patent applications, and are protected by trade secret or copyright law. Dissemination of this information or reproduction of this material is strictly forbidden unless prior written permission is obtained from Tobii AB.

\*/

/\*\*

\* @file tobii\_research.h

\* @brief <b>Generic SDK functions.</b>

\*

\*/

#ifndef TOBII\_RESEARCH\_H\_

#define TOBII\_RESEARCH\_H\_

#include <stddef.h>

#include <stdint.h>

#if \_WIN32 || \_WIN64

#ifdef TOBII\_STATIC\_LIB

#define TOBII\_RESEARCH\_CALL

#define TOBII\_RESEARCH\_API

#else

#define TOBII\_RESEARCH\_CALL \_\_cdecl

#ifdef TOBII\_EXPORTING

#define TOBII\_RESEARCH\_API \_\_declspec(dllexport)

#else

#define TOBII\_RESEARCH\_API \_\_declspec(dllimport)

#endif /\* TOBII\_EXPORTING \*/

#endif /\* TOBII\_STATIC\_LIB \*/

#else

#define TOBII\_RESEARCH\_API

#define TOBII\_RESEARCH\_CALL

#endif /\* \_WIN32 \*/

#ifdef \_\_cplusplus

extern "C" {

#endif

/\*\*

Status codes returned by the SDK.

\*/

typedef enum {

/\*\*

No error.

\*/

TOBII\_RESEARCH\_STATUS\_OK,

/\*\*

Fatal error. This should normally not happen.

\*/

TOBII\_RESEARCH\_STATUS\_FATAL\_ERROR,

/\*\*

Failed to initialize the API. This is a fatal error.

\*/

TOBII\_RESEARCH\_STATUS\_INITIALIZE\_FAILED,

/\*\*

Failed to terminate the API.

\*/

TOBII\_RESEARCH\_STATUS\_TERMINATE\_FAILED,

/\*\*

Failed to create browser for finding local devices.

\*/

TOBII\_RESEARCH\_STATUS\_LOCALBROWSER\_CREATE\_FAILED,

/\*\*

Failed to poll local devices.

\*/

TOBII\_RESEARCH\_STATUS\_LOCALBROWSER\_POLL\_FAILED,

/\*\*

Failed to create zero configuration browser.

\*/

TOBII\_RESEARCH\_STATUS\_ZEROCONFBROWSER\_CREATE\_FAILED,

/\*\*

Failed to poll devices from zero configuration browser.

\*/

TOBII\_RESEARCH\_STATUS\_ZEROCONFBROWSER\_POLL\_FAILED,

/\*\*

Failed to create browser that looks for devices in file.

\*/

TOBII\_RESEARCH\_STATUS\_FILEBROWSER\_CREATE\_FAILED,

/\*\*

Failed to poll devices from file browser.

\*/

TOBII\_RESEARCH\_STATUS\_FILEBROWSER\_POLL\_FAILED,

/\*\*

An invalid parameter was given to the method.

\*/

TOBII\_RESEARCH\_STATUS\_INVALID\_PARAMETER,

/\*\*

The operation was invalid.

\*/

TOBII\_RESEARCH\_STATUS\_INVALID\_OPERATION,

/\*\*

Internal core error code. Should never be returned by the SDK.

\*/

TOBII\_RESEARCH\_STATUS\_UNINITIALIZED,

/\*\*

A parameter is out of bounds.

\*/

TOBII\_RESEARCH\_STATUS\_OUT\_OF\_BOUNDS,

/\*\*

The display area is not valid. Please configure the eye tracker.

\*/

TOBII\_RESEARCH\_STATUS\_DISPLAY\_AREA\_NOT\_VALID,

/\*\*

The buffer is too small.

\*/

TOBII\_RESEARCH\_STATUS\_BUFFER\_TOO\_SMALL,

/\*\*

tobii\_research\_initialize has not been called.

\*/

TOBII\_RESEARCH\_STATUS\_NOT\_INITIALIZED,

/\*\*

tobii\_research\_initialize has already been called.

\*/

TOBII\_RESEARCH\_STATUS\_ALREADY\_INITIALIZED,

/\*\*

The license saved on the device failed to apply when connecting. It has probably expired.

\*/

TOBII\_RESEARCH\_STATUS\_SAVED\_LICENSE\_FAILED\_TO\_APPLY,

/\*\*

Internal stream engine error.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_INTERNAL = 200,

/\*\*

The operation requires a higher license type.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_INSUFFICIENT\_LICENSE,

/\*\*

The operations isn't supported in the current context.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_NOT\_SUPPORTED,

/\*\*

The device is unavailable.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_NOT\_AVAILABLE,

/\*\*

Connection to the device failed.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_CONNECTION\_FAILED,

/\*\*

The operation timed out.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_TIMED\_OUT,

/\*\*

Failed to allocate memory.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_ALLOCATION\_FAILED,

/\*\*

The API is already initialized.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_ALREADY\_INITIALIZED,

/\*\*

The API isn't initialized.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_NOT\_INITIALIZED,

/\*\*

An invalid parameter was given to the method.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_INVALID\_PARAMETER,

/\*\*

Calibration has already started.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_CALIBRATION\_ALREADY\_STARTED,

/\*\*

Calibration isn't started.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_CALIBRATION\_NOT\_STARTED,

/\*\*

Already subscribed.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_ALREADY\_SUBSCRIBED,

/\*\*

Not subscribed.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_NOT\_SUBSCRIBED,

/\*\*

The buffer is too small.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_BUFFER\_TOO\_SMALL,

/\*\*

Operation failed.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_OPERATION\_FAILED,

/\*\*

No response from firmware.

\*/

TOBII\_RESEARCH\_STATUS\_SE\_FIRMWARE\_NO\_RESPONSE,

/\*\*

Internal error.

\*/

TOBII\_RESEARCH\_STATUS\_FWUPGRADE\_INTERNAL = 400,

/\*\*

Firmware upgrade is not supported.

\*/

TOBII\_RESEARCH\_STATUS\_FWUPGRADE\_NOT\_SUPPORTED,

/\*\*

Unknown firmware version.

\*/

TOBII\_RESEARCH\_STATUS\_FWUPGRADE\_UNKNOWN\_FIRMWARE\_VERSION,

/\*\*

Connection failed.

\*/

TOBII\_RESEARCH\_STATUS\_FWUPGRADE\_CONNECTION\_FAILED,

/\*\*

Invalid parameter.

\*/

TOBII\_RESEARCH\_STATUS\_FWUPGRADE\_INVALID\_PARAMETER,

/\*\*

Device mismatch. The firmware package is not meant for the device.

\*/

TOBII\_RESEARCH\_STATUS\_FWUPGRADE\_PACKAGE\_DEVICE\_MISMATCH,

/\*\*

Parse response.

\*/

TOBII\_RESEARCH\_STATUS\_FWUPGRADE\_PARSE\_RESPONSE,

/\*\*

The firmware upgrade operation failed.

\*/

TOBII\_RESEARCH\_STATUS\_FWUPGRADE\_OPERATION\_FAILED,

/\*\*

Memory allocation failed during firmware upgrade.

\*/

TOBII\_RESEARCH\_STATUS\_FWUPGRADE\_ALLOCATION\_FAILED,

/\*\*

The firmware failed to respond during firmware upgrade.

\*/

TOBII\_RESEARCH\_STATUS\_FWUPGRADE\_FIRMWARE\_NO\_RESPONSE,

/\*\*

The firmware downgrade operation is not supported.

\*/

TOBII\_RESEARCH\_STATUS\_FWUPGRADE\_DOWNGRADE\_NOT\_SUPPORTED,

/\*\*

Unknown error. This is a fatal error.

\*/

TOBII\_RESEARCH\_STATUS\_UNKNOWN = 1000

} TobiiResearchStatus;

/\*\*

Opaque representation of an eye tracker struct.

\*/

typedef struct TobiiResearchEyeTracker TobiiResearchEyeTracker;

/\*\*

Contains all eye trackers connected to the computer or the network.

\*/

typedef struct {

/\*\*

An array of pointers to eye trackers.

\*/

TobiiResearchEyeTracker\*\* eyetrackers;

/\*\*

Number of eye tracker pointers in the array.

\*/

size\_t count;

} TobiiResearchEyeTrackers;

/\*\*

Source of log message.

\*/

typedef enum {

/\*\*

The log message is from stream engine.

\*/

TOBII\_RESEARCH\_LOG\_SOURCE\_STREAM\_ENGINE,

/\*\*

The log message is from the SDK.

\*/

TOBII\_RESEARCH\_LOG\_SOURCE\_SDK,

/\*\*

The log message is from the firmware upgrade module.

\*/

TOBII\_RESEARCH\_LOG\_SOURCE\_FIRMWARE\_UPGRADE

} TobiiResearchLogSource;

/\*\*

Log level.

\*/

typedef enum {

/\*\*

Error message.

\*/

TOBII\_RESEARCH\_LOG\_LEVEL\_ERROR,

/\*\*

Warning message.

\*/

TOBII\_RESEARCH\_LOG\_LEVEL\_WARNING,

/\*\*

Information message.

\*/

TOBII\_RESEARCH\_LOG\_LEVEL\_INFORMATION,

/\*\*

Debug message.

\*/

TOBII\_RESEARCH\_LOG\_LEVEL\_DEBUG,

/\*\*

Trace message.

\*/

TOBII\_RESEARCH\_LOG\_LEVEL\_TRACE

} TobiiResearchLogLevel;

/\*\*

Represents a normalized x- and y-coordinate point in a two-dimensional plane.

\*/

typedef struct {

/\*\*

Position of the point in the X axis.

\*/

float x;

/\*\*

Position of the point in the Y axis.

\*/

float y;

} TobiiResearchNormalizedPoint2D;

/\*\*

Represents an x-, y- and z-coordinate point in a three-dimensional space.

\*/

typedef struct {

/\*\*

Position of the point in the X axis.

\*/

float x;

/\*\*

Position of the point in the Y axis.

\*/

float y;

/\*\*

Position of the point in the Z axis.

\*/

float z;

} TobiiResearchPoint3D;

/\*\*

Represents a normalized x-, y- and z-coordinate point in a three-dimensional space.

\*/

typedef TobiiResearchPoint3D TobiiResearchNormalizedPoint3D;

/\*\*

@brief Log callback.

Implement this and send as a parameter to @ref tobii\_research\_logging\_subscribe.

@param system\_time\_stamp: The time stamp according to the computer's internal clock.

@param source: Source of log message.

@param level: Log message level.

@param message: The log message.

\*/

typedef void(\*tobii\_research\_log\_callback)(int64\_t system\_time\_stamp,

TobiiResearchLogSource source,

TobiiResearchLogLevel level,

const char\* message);

/\*\*

@brief Subscribes to logging.

@param callback: Callback that will receive log messages.

@returns A @ref TobiiResearchStatus code.

\*/

TOBII\_RESEARCH\_API TobiiResearchStatus TOBII\_RESEARCH\_CALL tobii\_research\_logging\_subscribe(

tobii\_research\_log\_callback callback);

/\*\*

@brief Unsubscribes from logging.

@returns A @ref TobiiResearchStatus code.

\*/

TOBII\_RESEARCH\_API TobiiResearchStatus TOBII\_RESEARCH\_CALL tobii\_research\_logging\_unsubscribe();

/\*\*

@brief Finds eye trackers connected to the computer or the network.

The eye trackers can be used in any tobii\_research function that requires an eye tracker.

\snippet find\_all\_eyetrackers.c FindAllEyetrackers

@param eyetrackers: Pointers to found eye trackers will be stored in this struct.

@returns A @ref TobiiResearchStatus code.

\*/

TOBII\_RESEARCH\_API TobiiResearchStatus TOBII\_RESEARCH\_CALL tobii\_research\_find\_all\_eyetrackers(

TobiiResearchEyeTrackers\*\* eyetrackers);

/\*\*

@brief Free memory allocation for the result received via @ref tobii\_research\_find\_all\_eyetrackers.

@param eyetrackers: Eye trackers to free.

\*/

TOBII\_RESEARCH\_API void TOBII\_RESEARCH\_CALL tobii\_research\_free\_eyetrackers(

TobiiResearchEyeTrackers\* eyetrackers);

/\*\*

@brief Gets data for an eye tracker given an address.

\snippet create\_eyetracker.c Example

@param address: Address of eye tracker to get data for.

@param eyetracker: Eye tracker object returned.

@returns A @ref TobiiResearchStatus code.

\*/

TOBII\_RESEARCH\_API TobiiResearchStatus TOBII\_RESEARCH\_CALL tobii\_research\_get\_eyetracker(

const char\* address,

TobiiResearchEyeTracker\*\* eyetracker);

/\*\*

@brief Retrieves the time stamp from the system clock in microseconds.

\snippet get\_system\_time\_stamp.c Example

@param time\_stamp\_us: The time stamp of the system in microseconds.

@returns A @ref TobiiResearchStatus code.

\*/

TOBII\_RESEARCH\_API TobiiResearchStatus TOBII\_RESEARCH\_CALL tobii\_research\_get\_system\_time\_stamp(int64\_t\* time\_stamp\_us);

/\*\*

@brief Free memory allocation for a string allocated by the SDK.

@param str: String to free.

\*/

TOBII\_RESEARCH\_API void TOBII\_RESEARCH\_CALL tobii\_research\_free\_string(char\* str);

/\*\*

SDK Version.

\*/

typedef struct {

/\*\*

Major.

\*/

int major;

/\*\*

Minor.

\*/

int minor;

/\*\*

Revision.

\*/

int revision;

/\*\*

Build.

\*/

int build;

} TobiiResearchSDKVersion;

/\*\*

Gets the SDK version.

@param sdk\_version: Version of the SDK.

@returns A TobiiResearchStatus code.

\*/

TOBII\_RESEARCH\_API TobiiResearchStatus TOBII\_RESEARCH\_CALL tobii\_research\_get\_sdk\_version(

TobiiResearchSDKVersion\* sdk\_version);

#ifdef \_\_cplusplus

}

#endif \_\_cplusplus

#endif /\* TOBII\_RESEARCH\_H\_ \*/