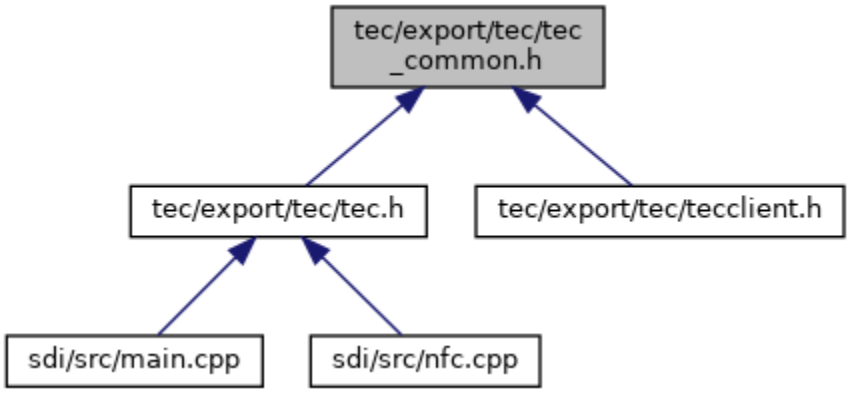
tec_common.h File Reference

This graph shows which files directly or indirectly include this file:



Go to the source code of this file.

Macros

#define	CTS_CHIP 1
#define	CTS_MSR 2
#define	CTS_CTLS 4
#define	CTS_SYNC 8
#define	CTS_DATA_TLV 0x80
#define	CTS_OK 0
#define	CTS_NO_CHIP 1
#define	CTS_IN_PROGRESS 2
#define	CTS_TIMEOUT 3
#define	CTS_PARAM 4
#define	CTS_NOT_STARTED 5
#define	CTS_STOPPED 6
#define	CTS_CTLS_INIT 7
#define	CTS_ERROR 8
#define	CTS_CTLS_NOT_ALLOWED 9
#define	CTS_CTLS_EMV_NO_CARD 10

#define	CTS_ERR_LOAD 11
#define	CTS_VAS_DECRYPTION_NOT_REQUIRED 12
#define	CTS_VAS_DATA_DECRYPTED_OK 13
#define	CTS_VAS_DATA_DECRYPTED_FAILED 14
#define	CTS_UX_MSRDATA_NOT_AVAILABLE_TIMEOUT 15
#define	CTS_NFC_INIT 16
#define	CTS_CHIP_WITH_WRONG_ATR 17
#define	CTS_API_NOT_ALLOWED 18
#define	CTS_VAS_DATA_ENCRYPTED_OK 19
#define	CTS_UX_NO_MSR_DATA_AFTER_CARD_REMOVL 20
#define	CTS_MSR_ONLY_REQUEST_BUT_NO_DEVICE 21
#define	CTS_PURE_CARD_DETECTION 0x01
#define	CTS_NO_POWERON 0x02
#define	CTS_POWERON_AFTER_CTLS_MSR_DEACTIVATION 0x04
#define	CTS_NFC_ENABLE 0x01
#define	CTS_VAS_ENABLE 0x02
#define	CTS_EMV_AFTER_NFC_ISO 0x04
#define	CTS_VAS_HANDLE_LED_BUZZ 0x08
#define	CTS_EMV_CTLS_TIMEOUT_AFTER_VAS 0x10
#define	CTS_VAS_DONT_DECRYPT 0x20
#define	CTS_MSR_AFTER_CTLS_FAIL 0x40
#define	CTS_DATA_TAG_NFC_RESULT 0xDFDB20
#define	CTS_DATA_TAG_CARD 0xFFDB20
#define	CTS_DATA_TAG_CARD_TYPE 0xDFDB21
#define	CTS_DATA_TAG_CARD_INFO 0xDFDB22
#define	CTS_DATA_TAG_VAS_RESULT 0xDFDB23

#define	CTS_DATA_TAG_VAS_DATA 0xDFDB24
#define	CTS_DATA_TAG_EMV_RESULT 0xDFDB25
#define	CTS_DATA_TAG_VAS_DECRYPT_DATA_RESULT 0xDFDB26
#define	CTS_DATA_TAG_CARD_TYPE_FULL 0xDFDB27
#define	CTS_DATA_TAG_CARDS_TOTAL_COUNT 0xDFDB28
#define	CTS_DATA_TAG_CARDS_A 0xDFDB29
#define	CTS_DATA_TAG_CARDS_B 0xDFDB2A
#define	CTS_DATA_TAG_CARDS_F 0xDFDB2B
#define	CTS_DATA_TAG_CUSTOM_POLL_RESULT 0xDFDB2C
#define	CTS_DATA_TAG_CARD_SAK 0xDFDB2D
#define	CTS_DATA_TAG_CARD_ATQ 0xDFDB2E
#define	CTS_DATA_TAG_CARD_RFU 0xDFDB2F
#define	CTS_OPTION_TAG_APP_ID 0xDFDB40
#define	CTS_OPTION_TAG_SYNC_CARD_TYPE 0xDFDB41
#define	CTS_OPTION_TAG_UX_MSR_TIMEOUT 0xDFDB42

Typedefs

typedef void(*	cts_TraceCallback) (const char *str, void *data)

Functions

void	cts_Version (char *version, unsigned char len)
void	cts_SetTraceCallback (cts_TraceCallback cbf, void *cb_data)

Detailed Description

Interface definitions for libtec, common part for tec.h and tecclient.h. This file defines the API for the technology selection library.

Author
Dirk Germann, GSS

Macro Definition Documentation

◆ CTS_DATA_TLV

```
#define CTS_DATA_TLV   0x80
```

if set in usedTechnology parameter of cts_WaitSelection(), dataBuffer is in TLV format (this is only the case if NFC or VAS are used)

Typedef Documentation

◆ cts_TraceCallback

```
typedef void(* cts_TraceCallback) (const char *str, void *data)
```

Type of function that is called for traces, see cts_SetTraceCallback();

Parameters

[in]	str	: Trace message.
[in]	data	: Data pointer provided by the application.

Function Documentation

◆ cts_SetTraceCallback()

void cts_SetTraceCallback	(	cts_TraceCallback	*cbf*,
		void *	*cb_data*
	)		

Set callback function for trace output.

Parameters

[in]	cbf	: Callback function for trace messages, may be NULL.

[in]	cb_data	: Data pointer that is passed on to the callback function cbf, may be NULL.

◆ cts_Version()

void cts_Version	(	char *	*version*,
		unsigned char	*len*
	)		

Get version of libtec.

Parameters

[out]	version	: Buffer to store null-terminated version string.
[in]	len	: Size of buffer version.