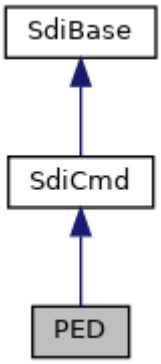
PED Class Reference

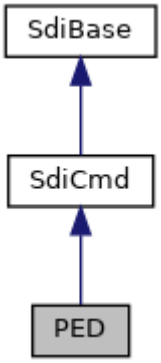
`#include <sdi_if.h>`

Inheritance diagram for PED:



[legend]

Collaboration diagram for PED:



[legend]

Public Types

enum	NavigatorMode { NAVIGATOR_MODE_OFF, NAVIGATOR_MODE_DOUBLE_TAB, NAVIGATOR_MODE_TACTILE_BUTTON }

Public Member Functions

void	setTimeout (unsigned seconds) Set PIN input dialog timeout (DFA005) for startPinEntry(). More...
bool	setDefaultTimeout (unsigned seconds) Configure PIN input dialog default timeout (22-02) More...
void	setTouchCoordinates (const unsigned char *array, unsigned size) configure touch coordinates (DFA024) for startPinInput() and startPinEntry(). More...
void	setNavigatorMode (enum NavigatorMode mode) Activate Navigator Mode (DFA025) for startPinInput() and startPinEntry(). More...
void	setPinDigitCountMinMax (unsigned char min, unsigned char max) Change PIN digit count limits (DFA02B and DFA02C) for sendPinInputParameters(), startPinInput() and startPinEntry(). More...
void	setLanguage (unsigned char lang) Set language (DFA008) for Pin input dialog of startPinInput(). More...
void	setAmount (const unsigned char *amount) Amount (9F02) for PIN input dialog of startPinInput() More...
void	setCurrency (const unsigned char *currency) Currency (5F2A) for PIN input dialog of startPinInput() More...
void	setAppLabel (const std::string &appLabel) Card application label (DFA037) for PIN input dialog of startPinInput() More...
int	startPinInput () Send command for PIN input (22-01). More...
int	startPinInput (bool enablePinBypass) Send command for PIN input (22-01). More...
bool	startPinEntry (unsigned pinBypassKey=0) Send command for asynchronous PIN input (22-03, polling mode). More...
bool	pollPinEntry (std::vector< unsigned char > &status) poll PIN input status after startPinEntry() (22-04) More...
bool	stopPinEntry () abort PIN entry (22-05) More...
bool	setPinInputClearKeyMode (bool clearAllDigits) Configure behavior of the backspace key for the PIN input dialog (22-06). More...
void	setClearKeyMode (bool clearAllDigits) Setter for clear key behavior (DFA02F) for sendPinInputParameters(), startPinInput() and startPinEntry(). More...
void	setPinBypassKeyAndMode (unsigned char value) Set PIN bypass key and mode bit mask (DFA104) for sendPinInputParameters() and startPinInput(). More...
void	setAutoConfirmation (bool enable)

	Set automatic confirmation when maximum digits are entered (DFA02E) for sendPinInputParameters(), startPinInput() and startPinEntry(). More...
void	setDialogOptions (uint32_t options)
	Set dialog option bit mask (DFA13D) for sendPinInputParameters() and startPinInput(). More...
void	setIntercharTimeout (uint32_t millis)
	Set inter-character timeout (DFA035) for sendPinInputParameters() and startPinInput(). More...
void	setHeaderLabel (const std::string &label)
	Set label for header area (DFA143) for sendPinInputParameters() and startPinInput(). More...
void	setEchoChar (uint32_t unicode)
	Set PIN echo character as unicode (DFA036) for sendPinInputParameters() and startPinInput(). More...
bool	sendPinInputParameters (bool epp=true)
	Send parameters that have been set before (22-06). More...
enum SDI_SW12	receiveGetPinResult ()
	Receive Pin Input result after startPinInput(). More...
bool	getPinBypassKey (unsigned char &value)
	Get PIN bypass key after startPinInput() and receiveGetPinResult(). More...

▸ Public Member Functions inherited from SdiCmd

	SdiCmd ()
virtual	~SdiCmd ()
enum SDI_SW12	sendReceive (unsigned char cla, unsigned char ins, unsigned char p1=0, unsigned char p2=0, unsigned maxResponseSize=2048)
	Generic SDI command exchange function combining sending of request, waiting and receiving the response. More...
int	send (unsigned char cla, unsigned char ins, unsigned char p1=0, unsigned char p2=0)
	Like sendReceive() but without waiting for response. More...
enum SDI_SW12	receive (unsigned maxResponseSize=2048)
	Receive SDI response after request has been send with send() and data availability has been signaled. More...
void	set (const char *path, int value, unsigned fixedLength=0)
	Set integer data object SDI command. More...
void	set (const char *path, uint32_t value, unsigned fixedLength=0)
	Set unsigned integer data object SDI command. More...
void	set (const char *path, const unsigned char *data, unsigned dataLen)
	Set byte array data object in SDI command. More...
void	set (const char *path, unsigned char byteValue)
	Set single byte value in SDI command. More...
void	set (const char *path, const std::vector< unsigned char > &data)
	Set byte array data object in SDI command. More...

void	set (const char *path, const std::string &value) Set character data object in SDI command. More...
void	setCommandDestination (bool epp, bool force=false) Set command destination for systems with EPP Unless force the command destination is set only if parameter epp is true. More...
virtual void	clear () Remove all SDI command command input data. More...
virtual void	clear (const char *path) Remove single input data stored in dataIn. More...
virtual void	clearResults () Remove all SDI response data. More...
void	importResults (const SdiCmd &intermediate) Transfer status word, client error and response data from an intermediate command while keeping command input data. More...
void	importResults (const unsigned char *sw12, const unsigned char *tlvData, unsigned tlvSize) Inject result data from a call back. More...
bool	get (const char *path, int &value)
bool	get (const char *path, uint32_t &value)
int	get (const char *path, unsigned char *buffer, unsigned bufferSize)
bool	get (const char *path, unsigned char &value)
bool	get (const char *path, std::vector< unsigned char > &buffer)
bool	get (const char *path, std::string &value)
std::string	getString (const char *path)
▸ Public Member Functions inherited from SdiBase	
	SdiBase ()
enum SDI_SW12	getSdiSw12 ()
int	getAdditionalResultValue () Access Additional Result Value if returned in SDI response. More...
SDICLIENT_ERROR	getClientError () Access client side error codes. More...
enum SDI_SW12	receiveSW12 () Receive SDI server response with no data. More...
void	clear () clear result data obtained from SDI communication More...
void	importResults (const SdiBase &intermediate) set result data obtained from intermediate SDI communication More...

Additional Inherited Members

Protected Member Functions inherited from SdiBase

void	setSdiSw12 (enum SDI_SW12 s)
void	setClientError (int libsdiprotocol_result)

Protected Attributes inherited from SdiCmd

void *	dataIn Command input buffer collecting TLV data items. More...
void *	dataOut Command response TLV data items. More...

Protected Attributes inherited from SdiBase

unsigned short	sw12
int	additionalResultValue
SDICLIENT_ERROR	clientErr

Detailed Description

Interface for PIN input dialog control *PED Interface*

Member Enumeration Documentation

? NavigatorMode

enum NavigatorMode

Navigator Mode is used for barrier-free PIN entry on touch-only devices

Enumerator	
NAVIGATOR_MODE_OFF	Navigator Mode disabled.
NAVIGATOR_MODE_DOUBLE_TAB	mode 1 with double tap
NAVIGATOR_MODE_TACTILE_BUTTON	mode 2 with tactile button

Member Function Documentation

? getPinBypassKey()

bool getPinBypassKey (unsigned char & *value*)

Get PIN bypass key after startPinInput() and receiveGetPinResult().

This is for the case that multiple bypass keys are possible

Parameters

[out] value key code (see 22-01 documentation, DFA038) return true if information had been received

? pollPinEntry()

bool pollPinEntry (std::vector< unsigned char > & *status*)

poll PIN input status after startPinEntry() (22-04)

Parameters

[out] status like described for tag DFA000

Returns

true for success

? receiveGetPinResult()

enum SDI_SW12 receiveGetPinResult ( )

Receive Pin Input result after startPinInput().

This is the alternative to receiveSW12() if data is expected, e.g. if two bypass keys are possible and getPinBypassKey() will be called later.

? sendPinInputParameters()

bool sendPinInputParameters (bool *epp* = `true`)

Send parameters that have been set before (22-06).

Note: This command should be used to configure standard mode behavior only and therefore the command destination defaults to EPP

Setters

setClearKeyMode() for DFA02F

setPinBypassKeyAndMode() for DFA104

setPinDigitCountMinMax() for DFA02B and DFA02C

setDialogOptions() for DFA13D

setHeaderLabel() for DFA143

setIntercharTimeout() for DFA035

setAutoConfirmation() for DFA02E

setEchoChar() for DFA036

Parameters

[in] epp command destination, default is true for EPP

? setAmount()

void setAmount (const unsigned char * *amount*)

Amount (9F02) for PIN input dialog of startPinInput()

this option is valid in standard mode only

Parameters

[in] amount n12 (6 bytes)

? setAppLabel()

void setAppLabel (const std::string & *appLabel*)

Card application label (DFA037) for PIN input dialog of startPinInput()

this option is valid in standard mode only

Parameters

[in] appLabel Card application as UTF-8 string

? setAutoConfirmation()

void setAutoConfirmation (bool *enable*)

Set automatic confirmation when maximum digits are entered (DFA02E) for sendPinInputParameters(), startPinInput() and startPinEntry().

Parameters

[in] enable enable auto confirmation

? setClearKeyMode()

void setClearKeyMode (bool *clearAllDigits*)

Setter for clear key behavior (DFA02F) for sendPinInputParameters(), startPinInput() and startPinEntry().

Parameters

[in] clearAllDigits true: Clear all input, false: Just clear one digit.

? setCurrency()

void setCurrency (const unsigned char * *currency*)

Currency (5F2A) for PIN input dialog of startPinInput()

this option is valid in standard mode only

Parameters

[in] currency n3 (2 bytes, padded left with '0')

? setDefaultTimeout()

bool setDefaultTimeout (unsigned *seconds*)

Configure PIN input dialog default timeout (22-02)

This timeout will be applied in case setTimeout() has not been invoked.

Parameters

[in] seconds PIN input dialog timeout

Returns

true in case of successful execution

? setDialogOptions()

void setDialogOptions (uint32_t *options*)

Set dialog option bit mask (DFA13D) for sendPinInputParameters() and startPinInput().

Parameters

[in] options Option bit mask

? setEchoChar()

void setEchoChar (uint32_t *unicode*)

Set PIN echo character as unicode (DFA036) for sendPinInputParameters() and startPinInput().

Parameters

[in] unicode unicode code

? setHeaderLabel()

void setHeaderLabel (const std::string & *label*)

Set label for header area (DFA143) for sendPinInputParameters() and startPinInput().

Setting empty value resets the default.

Parameters

[in] label Label

? setIntercharTimeout()

void setIntercharTimeout (uint32_t *millis*)

Set inter-character timeout (DFA035) for sendPinInputParameters() and startPinInput().

Parameters
[in] millis timout in milli seconds

? setLanguage()

void setLanguage (unsigned char *lang*)

Set language (DFA008) for Pin input dialog of startPinInput().

this option is valid in standard mode only

Parameters
[in] lang language code

? setNavigatorMode()

void setNavigatorMode (enum NavigatorMode *mode*)

Activate Navigator Mode (DFA025) for startPinInput() and startPinEntry().

? setPinBypassKeyAndMode()

void setPinBypassKeyAndMode (unsigned char *value*)

Set PIN bypass key and mode bit mask (DFA104) for sendPinInputParameters() and startPinInput().

Parameters
[in] value value for DFA104

? setPinDigitCountMinMax()

void setPinDigitCountMinMax (unsigned char *min*,
unsigned char *max*
)

Change PIN digit count limits (DFA02B and DFA02C) for sendPinInputParameters(), startPinInput() and startPinEntry().

The default values are 4 and 12. Calling this method is only required if other values shall be applied.

Parameters

[in] min minimal number of PIN digits [4..12]

[in] max maximal number of PIN digits [4..12]

? setPinInputClearKeyMode()

bool setPinInputClearKeyMode (bool *clearAllDigits*)

Configure behavior of the backspace key for the PIN input dialog (22-06).

Note: In contrast to other setters this function sends this single configuration immediately. This is somewhat obsolete as there are setClearKeyMode() and sendPinInputParameters() can do the same together with other parameters.

Parameters

[in] clearAllDigits true: Clear all input, false: Just clear one digit.

? setTimeout()

void setTimeout (unsigned *seconds*)

Set PIN input dialog timeout (DFA005) for startPinEntry().

This timeout will be used for PIN entries started with this object and replaces the SDI services default timeout configured via setDefaultTimeout.

Parameters

[in] seconds PIN input dialog timeout

? setTouchCoordinates()

void setTouchCoordinates (const unsigned char * *array*,
unsigned *size*
)

configure touch coordinates (DFA024) for startPinInput() and startPinEntry().

Each key consists of x-position b2 y-position b2 button width b2 button height b2 associated key ASCII code b1 ("0" .. "9" = '30' .. '31', cancel = '1B', correction = '08', enter = '0D')

Parameters

[in] array coordinates

[in] size size of coordinates array (n*9)

? startPinEntry()

bool startPinEntry (unsigned *pinBypassKey* = `0`)

Send command for asynchronous PIN input (22-03, polling mode).

Use pollPinEntry() and stopPinEntry() hereafter.

Setters

setPinDigitCountMinMax() for DFA02B and DFA02C

setAutoConfirmation() for DFA02E

setClearKeyMode() for DFA02F

setNavigatorMode() for DFA025

setTouchCoordinates() for DFA024

Parameters

[in] pinBypassKey

PIN bypass key (DFA02D),

0: deactivate (default),

ASCII code of bypass key,

256 for direct PIN bypass

Returns

true for success

? startPinInput() [1/2]

int startPinInput ( )

Send command for PIN input (22-01).

The response should be received with receiveGetPinResult() or receiveSW12(). SDI_SetSdiCallback() can be used to receive PIN input status messages, i.e. the digit count.

Setters

- setTimeout() for DFA005
- setPinBypassKeyAndMode() for DFA104
- setTouchCoordinates() for DFA024 (headless mode only)
- setNavigatorMode() for DFA025
- setPinDigitCountMinMax() for DFA02B and DFA02C
- setClearKeyMode() for DFA02F
- setIntercharTimeout() for DFA035
- setAutoConfirmation() for DFA02E
- setLanguage() for DFA008 (standard mode only)
- setAmount() for 9F02 (standard mode only)
- setCurrency() for 5F2A (standard mode only)
- setAppLabel() for DFA037 (stanard mode only)
- setDialogOptions() for DFA13D (standard mode only)
- setHeaderLabel() for DFA143 (standard mode only)
- setEchoChar() for DFA036 (standard mode only)

Returns

- return value from SDI_Send()

? startPinInput() [2/2]

int startPinInput (bool *enablePinBypass*)

Send command for PIN input (22-01).

This is for backward compatibility with previous libsdiclient versions having a default parameter. For description, see startPinInput() without parameters.

Parameters

[in] enablePinBypass if false, PIN bypass is disabled otherwise current configuration from setPinBypassKeyAndMode() is applied.

Returns

return value from SDI_Send()

? stopPinEntry()

bool stopPinEntry ( )

abort PIN entry (22-05)

Returns

true for success

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

- sdiclient/client/sdi_if.h