



USB To ST4 converter Software Development Kit

**Revision:1, 0
2017.4.7**

All material in this publication is subject to change without notice and is copyright
Zhen Wang Optical company.



Table of Contents

Change History	3
1 Introduction	3
2 Definition of enum-type and struct	3
2.1 #define USB2ST4_ID_MAX 128.....	3
2.2 typedef enum USB2ST4_DIRECTION{.....	3
2.3 typedef enum _USB2ST4_ERROR_CODE{.....	3
3 Function declaration	3
3.1 USB2ST4GetNum.....	3
3.2 USB2ST4GetID.....	4
3.3 USB2ST4Open.....	4
3.4 USB2ST4IsOpened.....	4
3.5 USB2ST4PulseGuide.....	4
3.6 USB2ST4Close.....	5
3.7 USB2ST4GetProductIDs.....	5
4 Suggested call sequence	5



Change History

Change date	revision	comment
2017.4.7	1.0	Create document

1 Introduction

This SDK is used to operate USB To ST4 converter, can be used by C, C++, C# and other develop tools, is suit for Windows, Linux, OSX operating system of x86 and x64.

Header file: USB2ST4_Conv.h

Under Windows the import library and dynamic library: USB2ST4_Conv.lib、USB2ST4_Conv.dll

Under Linux the dynamic library and static library: USB2ST4_Conv.so、USB2ST4_Conv.a

Under OSX the dynamic library and static library: USB2ST4_Conv.dylib、USB2ST4_Conv.a

Installation method:

Under Windows, extract the downloaded zip file to any directory, and add DLL's path to system environment variables, sometimes logout and re-login is required

2 Definition of enum-type and struct

2.1 #define USB2ST4_ID_MAX 128

```
2.2 typedef enum USB2ST4_DIRECTION{
    USB2ST4_NORTH=0,
    USB2ST4_SOUTH,
    USB2ST4_EAST,
    USB2ST4_WEST
}USB2ST4_DIRECTION;
```

```
2.3 typedef enum _USB2ST4_ERROR_CODE{
    USB2ST4_SUCCESS = 0,
    USB2ST4_ERROR_INVALID_INDEX,
    USB2ST4_ERROR_INVALID_ID,
    USB2ST4_ERROR_INVALID_VALUE,
    USB2ST4_ERROR_REMOVED, //failed to find the converter, maybe the converter has been
removed
    USB2ST4_ERROR_ERROR_STATE, //converter is in error state
    USB2ST4_ERROR_GENERAL_ERROR, //other error
    USB2ST4_ERROR_CLOSED,
    USB2ST4_ERROR_END = -1
}USB2ST4_ERROR_CODE;
    Returned error code
```

3 Function declaration

3.1 USB2ST4GetNum

Syntax: int USB2ST4GetNum()

Descriptions:

This should be the first API to be called, get number of connected USB2ST4 converter, call this API to



refresh device list if USB2ST4 is connected or disconnected

Return: number of connected USB2ST4 converter. 1 means 1 converter is connected.

3.2 USB2ST4GetID

Syntax: `USB2ST4_ERROR_CODE USB2ST4GetID(int index, int* ID)`

Descriptions:

Get ID of converter

Paras:

int index: the index of converter, from 0 to N - 1, N is returned by `USB2ST4GetNum()`

int* ID: pointer to ID. if the converter is not opened, the ID is negative, otherwise the ID is a unique integer between 0 to `USB2ST4_ID_MAX - 1`, after opened, all the operation is base on this ID, the ID will not change before the converter is closed.

Return:

`USB2ST4_ERROR_INVALID_INDEX`: index value is invalid

`USB2ST4_SUCCESS`: operation succeeds

3.3 USB2ST4Open

Syntax: `USB2ST4_ERROR_CODE USB2ST4Open(int ID)`

Descriptions:

Open converter

Paras:

int ID: the ID of converter

Return:

`USB2ST4_ERROR_INVALID_ID`: invalid ID value

`USB2ST4_ERROR_GENERAL_ERROR`: number of opened converter reaches the maximum value.

`USB2ST4_ERROR_REMOVED`: the converter is removed.

`USB2ST4_SUCCESS`: operation succeeds

3.4 USB2ST4IsOpened

Syntax: `USB2ST4_ERROR_CODE USB2ST4IsOpened(int ID)`

Descriptions:

determine if the converter is opened

Paras:

int ID: the ID of converter

Return:

`USB2ST4_ERROR_INVALID_ID`: invalid ID value

`USB2ST4_ERROR_CLOSED`: not opened

`USB2ST4_SUCCESS`: operation succeeds

`USB2ST4_ERROR_INVALID_VALUE`: Position value is invalid

`USB2ST4_ERROR_ERROR_STATE`: converter is in error state

`USB2ST4_ERROR_REMOVED`: converter is removed

3.5 USB2ST4PulseGuide

Syntax: `USB2ST4_ERROR_CODE USB2ST4PulseGuide(int ID, USB2ST4_DIRECTION Dirction, bool bSet)`

Descriptions:



pulse guide in the given direction

Paras:

int ID: the ID of converter

bool bSet: true is on, false is off

Return:

USB2ST4_ERROR_INVALID_ID: invalid ID value

USB2ST4_ERROR_CLOSED: not opened

USB2ST4_SUCCESS: operation succeeds

USB2ST4_ERROR_INVALID_VALUE: Position value is invalid

USB2ST4_ERROR_ERROR_STATE: converter is in error state

USB2ST4_ERROR_REMOVED: converter is removed

3.6 USB2ST4Close

Syntax: USB2ST4_ERROR_CODE USB2ST4Close(int ID)

Descriptions:

Close converter

Paras:

int ID: the ID of converter

Return:

USB2ST4_ERROR_INVALID_ID: invalid ID value

USB2ST4_SUCCESS: operation succeeds

3.7 USB2ST4GetProductIDs

Syntax: int USB2ST4GetProductIDs(int* pPIDs)

Descriptions:

get the product ID of each wheel, at first set pPIDs as 0 and get length and then malloc a buffer to load the PIDs

Paras:

int* pPIDs: pointer to array of PIDs

Return: length of the array.

4 Suggested call sequence

Get count of connected converters--> USB2ST4GetNum

Get converters' ID-> USB2ST4GetID

Open converter --> USB2ST4Open (Notes: this SDK can operate multiple converters, distinguish by each converter's ID)

Guide--> USB2ST4PulseGuide

Close converter-->USB2ST4Close