

# ***StoreX Function Library II***

## ***Middleware for StoreX Integration***

Version	1.0.0	Bemerkungen	
---------	-------	-------------	--

# Table of Content

<b>1</b>	<b>Introduction.....</b>	<b>4</b>
1.1	Definitions, Acronyms and Abbreviations .....	4
1.2	Specifications.....	5
<b>2</b>	<b>Command Set.....</b>	<b>6</b>
2.1	Initialisation Functions .....	7
2.1.1	int Connect(int STXPort) .....	7
2.1.2	int Initialize(int STXPort) .....	10
2.1.3	int Reset(int STXPort) .....	12
2.1.4	int SoftReset(int STXPort) .....	14
2.1.5	int Disconnect(int STXPort) .....	16
2.2	Status Functions .....	18
2.2.1	void GetErrorInfo(int ErrorCode, char* Reply) .....	18
2.2.2	int ReadInitFlag(int STXPort, bool* Value) .....	20
2.2.3	int ReadErrorFlag(int STXPort, bool* Value) .....	22
2.2.4	int ReadReadyFlag(int STXPort, bool* Value) .....	24
2.2.5	int ReadErrorCode(int STXPort, int* Value) .....	26
2.2.6	int ReadDoorFlag(int STXPort, bool* Value) .....	28
2.2.7	int ReadIsBusy(int STXPort, bool* Value) .....	30
2.2.8	int ReadIsInitialized(int STXPort, bool* Value) .....	32
2.3	Setting Functions .....	34
2.3.1	int SetUseSystemStatus(int STXPort, bool IsUse) .....	34
2.3.2	int SetCassetteConfig(int STXPort, int CassetteId, int MaxLevels, int ZPitch, int BCRZOffset) .....	36
2.3.3	int SetHandlerZPitch(int STXPort, int ZPitch) .....	38
2.3.4	int ReadHandlerZPitch(int STXPort, int* Value) .....	40
2.3.5	int SetBCRZOffset(int STXPort, int ZOffset) .....	42
2.3.6	int ReadBCRZOffset(int STXPort, int* Value) .....	44
2.3.7	int ReadSerialNumber(int STXPort, int* SNumber) .....	46
2.3.8	int ReadFirmwareVersion(int STXPort, int* FWVersion) .....	48
2.3.9	int SetPlateTrace(int STXPort, bool PlateTrace) .....	50
2.3.10	int ReadPlateTrace(int STXPort, bool* Value) .....	52
2.3.11	int SetLowLevelLogPath(char* LogPath) .....	54
2.3.12	int SetConfigFilePath (char* Path) .....	55
2.4	Environmental Functions .....	57
2.4.1	int ReadTemperature(int STXPort, float* Temperature) .....	57
2.4.2	int ReadHumidity(int STXPort, float* Humidity) .....	59
2.4.3	int ReadCO2(int STXPort, float* co2) .....	61
2.4.4	int ReadN2(int STXPort, float* n2) .....	63
2.4.5	int ReadO2(int STXPort, float* o2); .....	65
2.4.6	int SetTemperature(int STXPort, float Temperature) .....	67
2.4.7	int SetHumidity(int STXPort, float Humidity) .....	69
2.4.8	int SetCO2(int STXPort, float co2) .....	71
2.4.9	int SetN2(int STXPort, float n2) .....	73
2.4.10	int SetO2(int STXPort, float o2) .....	75
2.4.11	int ReadSetTemperature(int STXPort, float* Temperature) .....	77
2.4.12	int ReadSetHumidity(int STXPort, float* Humidity) .....	79
2.4.13	int ReadSetCO2(int STXPort, float* co2) .....	81
2.4.14	int ReadSetN2(int STXPort, float* n2) .....	83
2.4.15	int ReadSetO2(int STXPort, float* o2) .....	85
2.5	Plate Handling Functions .....	87
2.5.1	int LoadPlate(int STXPort, int Cassette, int Level) .....	87

2.5.2	int UnloadPlate(int STXPort, int Cassette, int Level) .....	89
2.5.3	int PickPlate(int STXPort, int Cassette, int Level) .....	91
2.5.4	int PlacePlate(int STXPort, int Cassette, int Level) .....	93
2.5.5	int GetPlate(int STXPort) .....	95
2.5.6	int SetPlate(int STXPort) .....	97
2.5.7	int ReadPlatePresence(int STXPort, int Cassette, int Level, bool* Presence) .....	99
2.5.8	int ReadShovelDetector(int STXPort, bool* Presence) .....	101
2.5.9	int ReadXferDetector(int STXPort, bool* Presence) .....	103
2.5.10	int ReadXferDetector2(int STXPort, bool* Presence) .....	105
2.6	<b>Barcode Reader Functions .....</b>	<b>108</b>
2.6.1	int BCConnect(int STXPort, int BCPort) .....	108
2.6.2	int BCDisconnect (int STXPort) .....	110
2.6.3	int ReadBarcode(int STXPort, int Cassette, int Level, char *Barcode) .....	112
2.6.4	int Inventory(int STXPort, char* FileName) .....	114
2.7	<b>Shaker Functions .....</b>	<b>117</b>
2.7.1	int SetShakerSpeed(int STXPort, int Value) .....	117
2.7.2	int ReadShakerSpeed(int STXPort, int* Value) .....	119
2.7.3	int RunShaker(int STXPort, bool Run) .....	121
2.8	<b>Transfer Station functions .....</b>	<b>124</b>
2.8.1	int SwapXferStation(int STXPort, bool Swap) .....	124
2.8.2	int AlternateXferStation(int STXPort, bool Swap) .....	126
2.9	<b>Gate functions .....</b>	<b>128</b>
2.9.1	int OpenTransferGate(int STXPort) .....	128
2.9.2	int CloseTransferGate(int STXPort) .....	130

# 1 Introduction

## 1.1 Definitions, Acronyms and Abbreviations

BCR	Barcode Reader
CU	Control Unit
GUI	Graphical User Interface
HW	Hardware
SW	Software
FW	Firmware
MPC	Micro Processor Controller
PLC	Programmable Logic Controller
DLL	Dynamic Link Library
PRD	Product Requirements Document
RoMA	Robotic Manipulator Arm
SCR	Software Change Request
SOP	Standard Operating Procedure
TC	Test Case
OS	Operation System
IDE	Integrated Development Environment

## 1.2 Specifications

The STXLib.dll is a communication library that provides an interaction between custom software and devices of the StoreX family using the serial communication interface RS-232.

The STXLib.dll library contains all necessary functions of Environment, Plate Operations, Error Diagnostic and Periphery control. The Command Set was developed to ease integration of Liconic products and to provide maximum functionality to the end-user.

The STXLib.dll library doesn't contain any COM-objects and doesn't require registration in the Operation System.

All functions return an integer result, where a value of 1 corresponds to success, and a negative value to an error.

It is possible to use the GetLastErrorInfo function to get a description of the error code.

Functions for reading device parameters or other values such as barcode or temperature return the result at the address of an argument of the appropriate type.

Each function has at least one int parameter that corresponds to the identifier of the serial port or serial number of device where the device is connected. Such way one entity of the library provides interaction with a large number of devices distinguished by the serial port identifier.

All functions exported from the library are using “\_cdecl” calling convention.

## 2 Command Set

- 4.1 Initialisation Functions
- 4.2 Setting Functions
- 4.3 Environmental Functions
- 4.4 Plate Handling Functions
- 4.5 Barcode Reader Functions

## 2.1 Initialisation Functions

### 2.1.1 int Connect(int STXPort)

Sets the name of serial port as id of the device, opens Serial Port and sends Communication Request command ("CR") to the device. If serial port is opened and communication with the device established then returns Serial Number of the device otherwise an error.

Parameter:

STXPort – Number of a Serial Port (e.g. "1")  
In Linux version of the library this parameter has char\* type and uses as value a Serial Port File name such as "/dev/ttyS0"

Return values:

Serial Number – Serial Port and communication with a device are successfully opened.

- 1 - Wrong device ID
- 12 - Error of read or write value to device
- 13 - Serial port already in use
- 14 - Serial port does not exist
- 15 - Error opening serial port
- 16 - No communication with the device
- 17 - Wrong Communication with the device

Examples:

#### 2.1.1.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* STXConnect)(int);  
STXConnect conn = (STXConnect)GetProcAddress(hDLL, "Connect");
```

'Execution:

```
int STXPort = 1;  
int result = conn(STXPort);
```



#### 2.1.1.2 C#

##### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "Connect")]  
public static extern int conn(int port);
```

##### //Execution:

```
int STXPort = 1;  
int result = conn(STXPort);
```

### 2.1.2 int Initialize(int STXPort)

Initializes the Device, initializes the handling, reads Device's system constants, writes Parameters to the Device.

Parameter:

STXPort – Number of a Serial Port or Serial Number of the Device

Return values:

- 1 - Device is successfully initialised
- 1 - Wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 4 - System is not initialized
- 6 - Device error flag is true
- 7 - Device initialization flag is false
- 8 - Door opened
- 9 - Error of waiting for completion
- 10 - Timeout of waiting for completion

Examples:

#### 2.1.2.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* Initialize)(int);  
Initialize STXInitialize = (Initialize)GetProcAddress(hDLL,  
"Initialize");
```

'Execution:

```
int STXPort = 1;  
int result = STXInitialize(STXPort);
```

#### 2.1.2.2 C#

##### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "Initialize")]  
public static extern int STXInitialize(int port);
```

##### //Execution:

```
int STXPort = 1;  
int result = STXInitialize(STXPort);
```

### 2.1.3 int Reset(int STXPort)

Resets the Device and puts it in the idle state.

The user should call the Reset function after any error of the machine. The user should call Initialize again to continue operations or press manually the "Reset" button of the machine.

Parameter:

STXPort – Number of a Serial Port or Serial Number of the Device

Return values:

- 1 - Device is reset
- 1 - Wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 12 - Error of read or write value to device

Examples:

#### 2.1.3.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));
typedef int(__cdecl* STXReset)(int port);
STXReset reset = (STXReset)GetProcAddress(hDLL, "Reset");
```

'Execution:

```
int STXPort = 1;
int result = reset(STXPort);
```

#### 2.1.3.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "Reset")]  
public static extern int reset(int);
```

//Execution:

```
int STXPort = 1;  
int result = reset(STXPort);
```

#### 2.1.4 int SoftReset(int STXPort)

Recover the Device after a soft error. In case if Device was recovered from the Error puts it in the Ready state.

The user should call the Soft Reset function after some errors such as PlateTrace Error etc. then Device will be ready to to continue operations.

Parameter:

STXPort – Number of a Serial Port or Serial Number of the Device

Return values:

- 1 - Device is reset
- 1 - Wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 12 - Error of read or write value to device

Examples:

##### 2.1.4.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));
typedef int(__cdecl* STXSoftReset)(int port);
STXSoftReset softReset = (STXSoftReset)GetProcAddress(hDLL,
"SoftReset");
```

'Execution:

```
int STXPort = 1;
int result = softReset(STXPort);
```

##### 2.1.4.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "SoftReset")]  
public static extern int softReset(int port);
```

//Execution:

```
int STXPort = 1;  
int result = softReset(STXPort);
```

### 2.1.5 int Disconnect(int STXPort)

Closes Serial Port and Communication with the Device through the active Serial Port.

Parameter:

STXPort – Number of a Serial Port or Serial Number of the Device

Return values:

- 1 – Communication is closed
- 1 – wrong device ID
- 2 – Unknown device ID
- 18 – Serial port not opened
- 19 – Error of closing serial port

Examples:

#### 2.1.5.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* STXDisconnect)(int port);  
STXDisconnect dconn = (STXDisconnect)GetProcAddress(hDLL,  
"Disconnect");
```

'Execution:

```
int STXPort = 1;  
int result = dconn(STXPort);
```

#### 2.1.5.2 C#

//Declaration:



```
[DllImport("STXLib.dll", EntryPoint = "Disconnect")]  
public static extern int dconn(int port);
```

//Execution:

```
int STXPort = 1;  
int result = dconn(STXPort);
```

## 2.2 Status Functions

### 2.2.1 void GetErrorInfo(int ErrorCode, char\* Reply)

Provides information about commands result error by Error Code.

Parameters:

ErrorCode - Error code as reply of a Command

Reply - Pointer to a value of Error Information.

Examples:

#### 2.2.1.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));
typedef void(__cdecl* GetErrorInfo)(int, bool*);
GetErrorInfo STXGetErrorInfo = (GetErrorInfo)GetProcAddress(hDLL,
"GetSTXErrorInfo");
```

'Execution:

```
int STXPort = 1;
int result = Initialize(STXPort);
if (result < 0)
{
    char reply[50];
    STXGetErrorInfo(result, &reply[0]);
}
```

#### 2.2.1.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "GetSTXErrorInfo")]  
public static extern void STXGetErrorInfo(int port, StringBuilder  
reply);
```

//Execution:

```
int STXPort = 1;  
int result = LoadPlate(STXPort, Cassette, Level);  
if (result < 0)  
{  
    StringBuilder reply = new StringBuilder();  
    STXGetErrorInfo(result, reply);  
}
```

## 2.2.2 int ReadInitFlag(int STXPort, bool\* Value)

Reads the Initialization Flag of the Device.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device  
Value - Pointer to Initialization Flag Value  
true: Device is initialized  
false: Device is not initialized.

Return values:

1 - value has been read  
-1 - Wrong device ID  
-2 - Unknown device ID  
-3 - Communication with the device not opened  
-12 - Error of read or write value to device

Examples:

### 2.2.2.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadInitFlag)(int, bool*);  
ReadInitFlag readInitFlag = (ReadInitFlag)GetProcAddress(hDLL,  
"ReadInitFlag");
```

'Execution:

```
int STXPort = 1;  
bool initFlag;  
int result = readInitFlag(STXPort, &initFlag);
```

#### 2.2.2.2 C#

##### //Declaration:

```
[DllImportSTXLib.dll", EntryPoint = "ReadInitFlag")]  
public static extern int reaInitFlag(int port, ref bool initFlag);
```

##### //Execution:

```
int STXPort = 1;  
bool initFlag;  
int result = readInitFlag(STXPort, ref initFlag);
```

### 2.2.3 int ReadErrorFlag(int STXPort, bool\* Value)

Reads the Error Flag of the Device.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device  
Value - Pointer to Error Flag Value  
true: Device has error status  
false: Device does not have error

Return values:

1 - value has been read  
-1 - Wrong device ID  
-2 - Unknown device ID  
-3 - Communication with the device not opened  
-12 - Error of read or write value to device

Examples:

#### 2.2.3.1 Microsoft C++

Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadErrorFlag)(int, bool*);  
ReadErrorFlag readErrorFlag = (ReadErrorFlag)GetProcAddress(hDLL,  
"ReadErrorFlag");
```

Execution:

```
int STXPort = 1;  
bool errorFlag;  
int result = readErrorFlag(STXPort, &errorFlag);
```

### 2.2.3.2 C#

#### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadErrorFlag")]  
public static extern int readErrorFlag(int port, ref bool errorFlag);
```

#### //Execution:

```
int STXPort = 1;  
bool errorFlag;  
int result = readErrorFlag(STXPort, ref errorFlag);
```

## 2.2.4 int ReadReadyFlag(int STXPort, bool\* Value)

Reads the Ready Flag of the Device.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device  
Value - Pointer to Ready Flag value  
true: Device is ready  
false: Device is not ready

Return values:

1 - value has been read  
-1 - Wrong device ID  
-2 - Unknown device ID  
-3 - Communication with the device not opened  
-12 - Error of read or write value to device

Examples:

### 2.2.4.1 Microsoft C++

Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadReadyFlag)(int, bool*);  
ReadReadyFlag readReadyFlag = (ReadReadyFlag)GetProcAddress(hDLL,  
"ReadReadyFlag");
```

Execution:

```
int STXPort = 1;  
bool readyFlag;  
int result = readReadyFlag(STXPort, &readyFlag);
```



#### 2.2.4.2 C#

##### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadReadyFlag")]  
public static extern int rrf(int port, ref bool readyFlag);
```

##### //Execution:

```
int STXPort = 1;  
bool readyFlag;  
int result = rrf(STXPort, ref readyFlag);
```

### 2.2.5 int ReadErrorCode(int STXPort, int\* Value)

Reads the Error Code of the Device.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device  
value - Ponter to Error Code value

Return values:

1 - value has been read  
-1 - wrong device ID  
-2 - Unknown device ID  
-3 - Communication with the device not opened  
-12 - Error of read or write value to device

Examples:

#### 2.2.5.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadErrorCode)(int, int*);  
ReadErrorCode readErrorCode = (ReadErrorCode)GetProcAddress(hDLL,  
"ReadErrorCode");
```

'Execution:

```
int STXPort = 1;  
int errorCode;  
int result = readErrorCode(STXPort, &errorCode);
```

#### 2.2.5.2 C#

//Declaration:

```
using System;
using System.Runtime.InteropServices;

[DllImport("STXLib.dll", EntryPoint = "ReadErrorCode")]
public static extern int readErrorCode(int port, ref int errorCode);
```

//Execution:

```
int STXPort = 1;
int errorCode = 1;
int result = readErrorCode(STXPort, ref errorCode);
```

## 2.2.6 int ReadDoorFlag(int STXPort, bool\* Value)

Reads whether door is opened.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device

Value - Pointer to Door Flag Value

true: Door is closed  
false: Door is opened

Return values:

1 - value has been read

-1 - wrong device ID

-2 - Unknown device ID

-3 - Communication with the device not opened

-12 - Error of read or write value to device

Examples:

### 2.2.6.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadDoorFlag)(int, bool*);  
ReadDoorFlag readDoorFlag = (ReadDoorFlag)GetProcAddress(hDLL,  
"ReadDoorFlag");
```

'Execution:

```
int STXPort = 1;  
bool doorFlag;  
int result = readDoorFlag(STXPort, &doorFlag);
```

### 2.2.6.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadDoorFlag")]  
public static extern int readDoorFlag(int port, ref bool doorFlag);
```

//Execution:

```
int STXPort = 1;  
bool doorFlag = false;  
int result = readDoorFlag(STXPort, ref doorFlag);
```

## 2.2.7 int ReadIsBusy(int STXPort, bool\* Value)

Checks whether previous long operation is running. It is possible to check whether command like LoadPlate, UnloadPlate, SetPlate, GetPlate, PickPlate, PlacePlate, ReadPlatePresence or Inventory are running by means of usage this method.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device

Value - Pointer to the value  
true: Driver is busy, command is running  
false: No commands are running

Return values:

- 1 - Value has been read
- 1 - Wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened

Examples:

### 2.2.7.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadIsBusy)(int, bool*);  
ReadIsBusy readIsBusy = (ReadIsBusy)GetProcAddress(hDLL,  
"ReadIsBusy");
```

'Execution:

```
int STXPort = 1;  
bool isBusy;  
int result = readIsBusy(STXPort, &isBusy);
```

### 2.2.7.2 C#

#### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadIsBusy")]  
public static extern int readIsBusy(int port, ref bool isBusy);
```

#### //Execution:

```
int STXPort = 1;  
bool isBusy = false;  
int result = readIsBusy(STXPort, ref isBusy);
```

## 2.2.8 int ReadIsInitialized(int STXPort, bool\* Value)

Checks whether Device was initialized by Initialize command.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device  
Value - Pointer to the Initialization Value  
true: Device is initialized  
false: Device is not initialized

Return values:

1 - value has been read  
-1 - wrong device ID  
-2 - Unknown device ID  
-3 - Communication with the device not opened

Examples:

### 2.2.8.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadIsInitialized)(int, bool*);  
ReadIsInitialized rInit = (ReadIsInitialized)GetProcAddress(hDLL,  
"ReadIsInitialized");
```

'Execution:

```
int STXPort = 1;  
bool isInitialized;  
int result = rInit(STXPort, &isInitialized);
```

### 2.2.8.2 C#

//Declaration:



```
[DllImport("STXLib.dll", EntryPoint = "ReadIsInitialized")]  
public static extern int rInit(int port, ref bool isInitialized);
```

//Execution:

```
int STXPort = 1;  
bool isInitialized = false;  
int result = rInit(STXPort, ref isInitialized);
```

## 2.3 Setting Functions

### 2.3.1 int SetUseSystemStatus(int STXPort, bool IsUse)

Sets to read Status Register values (DM204) of the Device.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device

IsUse - true: Using System Status turned on  
false: Using System Status turned off

Return values:

- 1 - value has been set
- 1 - wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened

Examples:

#### 2.3.1.1 Microsoft C++

##### //Declaration

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* SetUseSystemStatus)(int, bool);  
SetUseSystemStatus setUseSysStat =  
(SetUseSystemStatus)GetProcAddress(hDLL, "SetUseSystemStatus");
```

##### //Execution

```
int STXPort = 1;  
int result = setUseSysStat(STXPort, true);
```

#### 2.3.1.2 C#

##### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "SetUseSystemStatus")]  
public static extern int setUseSysStat(int port, bool isUse);
```

//Execution:

```
int STXPort = 1;  
int result = setUseSysStat(STXPort, true);
```

### 2.3.2 **int SetCassetteConfig(int STXPort, int Cassetteld, int MaxLevels, int ZPitch, int BCRZOffset)**

Sets cassette configuration for defined cassette. Defines pitch value, number of levels and Barcode Reader Z-Lift Read Position offset for one cassette. The parameters will be set automatically during plate operations with specified cassette.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device

Cassette - Identifier of a Cassette.

MaxLevels - Value Levels of maximum the Cassette.

ZPitch - value of Barcode Reader Z-Lift Position offset

Return values:

1 - values are set

-1 - Wrong device ID

-2 - Unknown device ID

-21 - Wrong parameter value

Examples:

#### 2.3.2.1 *Microsoft C++*

##### //Declaration

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));
typedef int(__cdecl* SetCassetteConfig)(int, int, int, int, int);
SetCassetteConfig setCassetteConfig =
(SetCassetteConfig)GetProcAddress(hDLL, "SetCassetteConfig");
```

##### //Execution

```
int STXPort = 1;
int cassette = 1;
int maxLevels = 21;
int zPitch = 300;
int bcrzOffset = 100;

int result = setCassetteConfig(STXPort, cassette, maxLevels, zPitch,
bcrzOffset);
```

#### 2.3.2.2 C#

##### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "SetCassetteConfig")]
public static extern int setCassetteConfig(int port, int cassetteId,
int maxLevels, int zPitch, int BCRZOffset);
```

##### //Execution:

```
int STXPort = 1;
int cassette = 1;
int maxLevels = 21;
int zPitch = 300;
int bcrzOffset = 100;

int result = setCassetteConfig(STXPort, cassette, maxLevels, zPitch,
bcrzOffset);
```

### 2.3.3 int SetHandlerZPitch(int STXPort, int ZPitch)

Sets Handler z-Pitch.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device  
ZPitch - value of Z-Pitch

Return values:

1 - value is set to device.  
-1 - wrong device ID  
-2 - Unknown device ID  
-3 - Communication with the device not opened  
-12 - Error of read or write value to device  
-21 - wrong parameter value

Examples:

#### 2.3.3.1 Microsoft C++

##### //Declaration

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* SetHandlerZPitch)(int, int);  
SetHandlerZPitch setHandlerZ = (SetHandlerZPitch)GetProcAddress(hDLL,  
"SetHandlerZPitch");
```

##### //Execution

```
int STXPort = 1;  
int zPitch = 300;  
int result = setHandlerZ(STXPort, zPitch);
```

#### 2.3.3.2 C#

##### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "SetHandlerZPitch")]  
public static extern int setHandlerZPitch (int port, int zPitch);
```

//Execution:

```
int STXPort = 1;  
int zPitch = 300;  
int result = setHandlerZPitch(STXPort, zPitch);
```

### 2.3.4 int ReadHandlerZPitch(int STXPort, int\* Value)

Reads value of handler Z pitch.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
value – Pointer to value of Z Pitch

Return values:

1 – value has been read  
-1 – wrong device ID  
-2 – Unknown device ID  
-3 – Communication with the device not opened  
-12 – Error of read or write value to device  
-21 – wrong parameter value

Examples:

#### 2.3.4.1 Microsoft C++

##### //Declaration

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadHandlerZPitch)(int, int*);  
ReadHandlerZPitch readHandlerZPitch =  
(ReadHandlerZPitch)GetProcAddress(hDLL, "ReadHandlerZPitch");
```

##### //Execution

```
int STXPort = 1;  
int zPitch;  
int result = readHandlerZPitch(STXPort, &zPitch);
```

#### 2.3.4.2 C#

##### //Declaration:



```
[DllImport("STXLib.dll", EntryPoint = "ReadHandlerZPitch")]  
public static extern int readHandlerZPitch (int port, ref int  
zPitch);
```

//Execution:

```
int STXPort = 1;  
int zPitch = -1;  
int result = readHandlerZPitch(STXPort, ref zPitch);
```

### 2.3.5 int SetBCRZOffset(int STXPort, int ZOffset)

Sets Barcode Reader Z-Lift Read Position offset.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device  
ZPitch - value of Barcode Reader Z-Lift Read Position offset

Return values:

1 - value is set to device.  
-1 - wrong device ID  
-2 - Unknown device ID  
-3 - Communication with the device not opened  
-12 - Error of read or write value to device  
-21 - wrong parameter value

Examples:

#### 2.3.5.1 Microsoft C++

//Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* SetBCRZOffset)(int, int);  
SetBCRZOffset setBCRZOffset = (SetBCRZOffset)GetProcAddress(hDLL,  
"SetBCRZOffset");
```

//Execution:

```
int STXPort = 1;  
int zoffset = 100;  
int result = setBCRZOffset(STXPort, zoffset);
```

#### 2.3.5.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "SetBCRZOffset")]  
public static extern int setBCRZOffset(int port, int zoffset);
```

//Execution:

```
int STXPort = 1;  
int zoffset = 100;  
int result = setBCRZOffset(STXPort, zoffset);
```

### 2.3.6 int ReadBCRZOffset(int STXPort, int\* Value)

Reads Barcode Reader Z-Lift Read Position offset.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
value – Pointer to value of Z-Lift offset position

Return values:

- 1 – value has been read
- 1 – wrong device ID
- 2 – Unknown device ID
- 3 – Communication with the device not opened
- 12 – Error of read or write value to device
- 21 – wrong parameter value

Examples:

#### 2.3.6.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadBCRZOffset)(int, int*);  
ReadBCRZOffset readBCRZOffset = (ReadBCRZOffset)GetProcAddress(hDLL,  
"ReadBCRZOffset");
```

'Execution:

```
int STXPort = 1;  
int bcrzOffset;  
int result = readBCRZOffset(STXPort, &bcrzOffset);
```

#### 2.3.6.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadBCRZOffset")]  
public static extern int readBCRZOffset(int port, ref int zoffset);
```

//Execution:

```
int STXPort = 1;  
int bcrzOffset = -1;  
int result = readBCRZOffset(STXPort, ref bcrzOffset);
```

### 2.3.7 int ReadSerialNumber(int STXPort, int\* SNumber)

Reads Serial Number of the device.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
SNumber – Pointer to value of Serial Number

Return values:

1 – value has been read  
-1 – wrong device ID  
-2 – Unknown device ID  
-3 – Communication with the device not opened  
-12 – Error of read or write value to device  
-21 – wrong parameter value

Examples:

#### 2.3.7.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadSerialNumber)(int, int*);  
ReadSerialNumber readSerialNumber =  
(ReadSerialNumber)GetProcAddress(hDLL, "ReadSerialNumber");
```

'Execution:

```
int STXPort = 1;  
int serialNum;  
int result = readSerialNumber(STXPort, &serialNum);
```

#### 2.3.7.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadSerialNumber")]  
public static extern int readSerialNumber(int port, ref int serial);
```

//Execution:

```
int STXPort = 1;  
int serialNum = -1;  
int result = readSerialNumber(STXPort, ref serialNum);
```

### 2.3.8 int ReadFirmwareVersion(int STXPort, int\* FWVersion)

Reads Firmware Version of the device.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
FWVersion – Pointer to value of Firmware Version

Return values:

- 1 – value has been read
- 1 – wrong device ID
- 2 – Unknown device ID
- 3 – Communication with the device not opened
- 12 – Error of read or write value to device
- 21 – wrong parameter value

Examples:

#### 2.3.8.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadFirmwareVersion)(int, int*);  
ReadFirmwareVersion readFirmwareVersion =  
(ReadFirmwareVersion)GetProcAddress(hDLL, "ReadFirmwareVersion");
```

'Execution:

```
int STXPort = 1;  
int firmwareVersion;  
int result = readFirmwareVersion(STXPort, &firmwareVersion);
```

#### 2.3.8.2 C#

//Declaration:



```
[DllImport("STXLib.dll", EntryPoint = "ReadFirmwareVersion")]  
public static extern int readFirmwareVersion(int port, ref int  
firmware);
```

//Execution:

```
int STXPort = 1;  
int firmwareVersion = -1;  
int result = readFirmwareVersion(STXPort, ref firmwareVersion);
```

### 2.3.9 int SetPlateTrace(int STXPort, bool PlateTrace)

Sets Plate Trace mode of the device.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device  
PlateTrace - true: Plate Trace turned on  
              false: Plate Trace turned off

Return values:

1 - value has been set  
-1 - Wrong device ID  
-2 - Unknown device ID  
-3 - Communication with the device not opened  
-12 - Error of read or write value to device

Examples:

#### 2.3.9.1 Microsoft C++

##### //Declaration

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* SetPlateTrace)(int, bool);  
SetPlateTrace setPlateTrace = (SetPlateTrace)GetProcAddress(hDLL,  
"SetPlateTrace");
```

##### //Execution

```
int STXPort = 1;  
int result = setPlateTrace(STXPort, true);
```

#### 2.3.9.2 C#

##### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "SetPlateTrace")]  
public static extern int setPlateTrace(int port, bool isPlateOn);
```

//Execution:

```
int STXPort = 1;  
int result = setPlateTrace(STXPort, true);
```

### 2.3.10 int ReadPlateTrace(int STXPort, bool\* Value)

Reads whether Plate Trace mode is set on the device.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device  
Value - Pointer to Plate Trace value  
          true: Plate Trace is turned on  
          false: Plate Trace is turned off

Return values:

1 - value has been read  
-1 - wrong device ID  
-2 - Unknown device ID  
-3 - Communication with the device not opened  
-12 - Error of read or write value to device

Examples:

#### 2.3.10.1 *Microsoft C++*

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadPlateTrace)(int, bool*);  
ReadPlateTrace readPlateTrace = (ReadPlateTrace)GetProcAddress(hDLL,  
"ReadPlateTrace");
```

'Execution:

```
int STXPort = 1;  
bool isPlateTracing;  
int result = readPlateTrace(STXPort, &isPlateTracing);
```

### 2.3.10.2 C#

#### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadPlateTrace")]  
public static extern int readPlateTrace(int port, ref bool  
isPlateTraceOn);
```

#### //Execution:

```
int STXPort = 1;  
bool isPlateTracing = false;  
int result = readPlateTrace(STXPort, ref isPlateTracing);
```

### 2.3.11 int SetLowLevelLogPath(char\* LogPath)

Sets Path to a Folder where Serial Communication Log Files will be saved.

Parameters:

LogPath – Path to a Folder

Return values:

- 1 – value has been read
- 21 – wrong parameter value

Examples:

#### 2.3.11.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));
typedef void(__cdecl* SetLowLevelPath)(char*);
SetLowLevelPath STXSetLowLevelPath =
(SetLowLevelPath)GetProcAddress(hDLL, "SetLowLevelLogPath");
```

'Execution:

```
char logPath[100] = "C:\\Temp\\Projects\\Main"
STXSetLowLevelPath(&logPath[0]);
```

#### 2.3.11.2 C#

//Declaration:

```
[DllImport("C:\\Users\\admin\\b\\STXLib.dll", EntryPoint =
"SetLowLevelLogPath")]
public static extern void STXSetLowLevelLogPath(string path);
```

//Execution:

```
string filePath = "C:\\Users\\admin\\Documents\\DLLLog";
STXSetLowLevelLogPath(path);
```

### 2.3.12 int SetConfigFilePath (char\* Path)

Sets path to Configuration Files of the Devices. Configuration file is a text file which contains values of Flags, Data Memories, Timers as low-level commands which will be sent to Device at the time of Initialization. The name of file consists of Serial Number of the Device with "cfg" extension e.g. "3656.cfg".

Exampe of device configuration file:

```
ST 1611
ST 1612
RS 1606
WR DM27 450
WR DM23 788
```

Parameters:

STXPort - Number of a Serial Port  
IsUse - true: Using System Status turned on  
false: Using System Status turned off

Return values:

1 - value has been set  
-21 - wrong parameter value  
-33 - can not access the path

Examples:

#### 2.3.12.1 Microsoft C++

##### //Declaration

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));
typedef int(__cdecl* SetConfigFilePath)(char*);
SetConfigFilePath STXSetConfigFilePath =
(SetConfigFilePath)GetProcAddress(hDLL, "SetConfigFilePath");
```

##### //Execution

```
string ConfigPath = "C:\\Data\\DevicesConfig\\";
int result = STXSetConfigFilePath(&ConfigPath[0]);
```

### 2.3.12.2 C#

#### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "SetConfigFilePath")]  
public static extern int STXSetConfigFilePath(string path);
```

#### //Execution:

```
string path = "C:\\Data\\DevicesConfig\\";  
result = STXSetConfigFilePath(path);
```



## 2.4 Environmental Functions

### 2.4.1 int ReadTemperature(int STXPort, float\* Temperature)

Returns actual value of temperature in °C.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
Temperature – Pointer to value of temperature

Return values:

1 – value has been read  
-1 – wrong device ID  
-2 – Unknown device ID  
-3 – Communication with the device not opened  
-12 – Error of read or write value to device  
-21 – wrong parameter value

Examples:

#### 2.4.1.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadTemperature)(int, float*);  
ReadTemperature readTemperature =  
(ReadTemperature)GetProcAddress(hDLL, "ReadTemperature");
```

'Execution:

```
int STXPort = 1;  
float temperature;  
int result = readTemperature(STXPort, &temperature);
```

#### 2.4.1.2 C#

##### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadTemperature")]  
public static extern int readTemperature(int port, ref float temp);
```

##### //Execution:

```
int STXPort = 1;  
float temperature;  
int result = readTemperature(STXPort, ref temperature);
```

## 2.4.2 int ReadHumidity(int STXPort, float\* Humidity)

Returns actual value of Relative Humidity in percent.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device

Humidity – Pointer to value of humidity

Return values:

1 – value has been read

-1 – wrong device ID

-2 – Unknown device ID

-3 – Communication with the device not opened

-12 – Error of read or write value to device

-21 – wrong parameter value

Examples:

### 2.4.2.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));
typedef int(__cdecl* ReadHumidity)(int, float*);
ReadHumidity readHumidity = (ReadHumidity)GetProcAddress(hDLL,
"ReadHumidity");
```

'Execution:

```
int STXPort = 1;
float humidity;
int result = readHumidity(STXPort, &humidity);
```

### 2.4.2.2 C#

//Declaration:

```
using System;
using System.Runtime.InteropServices;

[DllImport("STXLib.dll", EntryPoint = "ReadHumidity")]
public static extern int readHumidity(int port, ref float hum);
```

//Execution:

```
int STXPort = 1;
float humidity;
int result = readHumidity(STXPort, ref hum);
```

### 2.4.3 int ReadCO2(int STXPort, float\* co2)

Returns actual value of CO2 concentration in percent.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
co2 – Pointer to value of CO2

Return values:

1 – value has been read  
-1 – wrong device ID  
-2 – Unknown device ID  
-3 – Communication with the device not opened  
-12 – Error of read or write value to device  
-21 – wrong parameter value

Examples:

#### 2.4.3.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadCO2)(int, float*);  
ReadCO2 readCO2 = (ReadCO2)GetProcAddress(hDLL, "ReadCO2");
```

'Execution:

```
int STXPort = 1;  
float co2;  
int result = readCO2(STXPort, &co2);
```

#### 2.4.3.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadCO2")]  
public static extern int readCO2(int port, ref float co2);
```

//Execution:

```
int STXPort = 1;  
float co2;  
int result = readCO2(STXPort, ref co2);
```

#### 2.4.4 int ReadN2(int STXPort, float\* n2)

Returns actual value of N2 concentration in percent.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
n2 – Pointer to value of N2

Return values:

1 – value has been read  
-1 – wrong device ID  
-2 – Unknown device ID  
-3 – Communication with the device not opened  
-12 – Error of read or write value to device  
-21 – wrong parameter value

Examples:

##### 2.4.4.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadN2)(int, float*);  
ReadN2 readN2 = (ReadN2)GetProcAddress(hDLL, "ReadN2");
```

'Execution:

```
int STXPort = 1;  
float n2;  
int result = readN2(STXPort, &n2);
```

##### 2.4.4.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadN2")]  
public static extern int readN2(int port, ref float n2);
```

//Execution:

```
int STXPort = 1;  
float n2;  
int result = readN2(STXPort, ref n2);
```



#### 2.4.5 int ReadO2(int STXPort, float\* o2);

Returns actual value of O2 concentration in percent.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
o2 – Pointer to value of O2

Return values:

1 – value has been read  
-1 – wrong device ID  
-2 – unknown device ID  
-3 – communication with the device not opened  
-12 – error of read or write value to device  
-21 – wrong parameter value

Examples:

##### 2.4.5.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadO2)(int, float*);  
ReadO2 readO2 = (ReadO2)GetProcAddress(hDLL, "ReadO2");
```

'Execution:

```
int STXPort = 1;  
float o2;  
int result = readO2(STXPort, &o2);
```

##### 2.4.5.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "Read02")]  
public static extern int read02(int port, ref float o2);
```

//Execution:

```
int STXPort = 1;  
float o2;  
int result = read02(STXPort, ref o2);
```

## 2.4.6 int SetTemperature(int STXPort, float Temperature)

Sets the target temperature in °C.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
Temperature – value of temperature

Return values:

1 – value has been set  
-1 – wrong device ID  
-2 – Unknown device ID  
-3 – Communication with the device not opened  
-12 – Error of read or write value to device  
-21 – wrong parameter value

Examples:

### 2.4.6.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* SetTemperature)(int, float);  
SetTemperature setTemperature = (SetTemperature)GetProcAddress(hDLL,  
"SetTemperature");
```

'Execution:

```
int STXPort = 1;  
float temperature = 36;  
int result = setTemperature(STXPort, temperature);
```

#### 2.4.6.2 C#

##### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "SetTemperature")]  
public static extern int setTemperature(int port, float temp);
```

##### //Execution:

```
int STXPort = 1;  
float temperature = 36;  
int result = setTemperature(STXPort, temperature);
```

### 2.4.7 int SetHumidity(int STXPort, float Humidity)

Sets the target relative humidity in percent.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
Humidity – value of Humidity

Return values:

1 – value has been set  
-1 – wrong device ID  
-2 – Unknown device ID  
-3 – Communication with the device not opened  
-12 – Error of read or write value to device  
-21 – wrong parameter value

Examples:

#### 2.4.7.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* SetHumidity)(int, float);  
SetHumidity setHumidity = (SetHumidity)GetProcAddress(hDLL,  
"SetHumidity");
```

'Execution:

```
int STXPort = 1;  
float humidity = 90;  
int result = setHumidity(STXPort, humidity);
```

#### 2.4.7.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "SetHumidity")]  
public static extern int setHumidity(int port, float hum);
```

//Execution:

```
int STXPort = 1;  
float humidity = 90;  
int result = setHumidity(STXPort, humidity);
```

#### 2.4.8 int SetCO2(int STXPort, float co2)

Sets the target CO2 concentration in percent.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
co2 – value of CO2

Return values:

1 – value has been set  
-1 – wrong device ID  
-2 – unknown device ID  
-3 – communication with the device not opened  
-12 – error of read or write value to device  
-21 – wrong parameter value

Examples:

##### 2.4.8.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* SetCO2)(int, float);  
SetCO2 setCO2 = (SetCO2)GetProcAddress(hDLL, "SetCO2");
```

'Execution:

```
int STXPort = 1;  
float co2 = 0;  
int result = setCO2(STXPort, co2);
```

##### 2.4.8.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "SetCO2")]  
public static extern int setCO2(int port, float co2);
```

//Execution:

```
int STXPort = 1;  
float co2 = 0;  
int result = setCO2(STXPort, co2);
```



### 2.4.9 int SetN2(int STXPort, float n2)

Sets the target N2 concentration in percent.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
n2 – value of N2

Return values:

1 – value has been set  
-1 – wrong device ID  
-2 – Unknown device ID  
-3 – Communication with the device not opened  
-12 – Error of read or write value to device  
-21 – wrong parameter value

Examples:

#### 2.4.9.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* SetN2)(int, float);  
SetN2 setN2 = (SetN2)GetProcAddress(hDLL, "SetN2");
```

'Execution:

```
int STXPort = 1;  
float n2 = 10;  
int result = setN2(STXPort, n2);
```

#### 2.4.9.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "SetN2")]  
public static extern int setN2(int port, float n2);
```

//Execution:

```
int STXPort = 1;  
float n2 = 10;  
int result = setN2(STXPort, n2);
```

## 2.4.10 int SetO2(int STXPort, float o2)

Sets the target O2 concentration in percent.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
o2 – value of O2

Return values:

1 – value has been read  
-1 – wrong device ID  
-2 – unknown device ID  
-3 – communication with the device not opened  
-12 – error of read or write value to device  
-21 – wrong parameter value

Examples:

### 2.4.10.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* SetO2)(int, float);  
SetO2 setO2 = (SetO2)GetProcAddress(hDLL, "SetO2");
```

'Execution:

```
int STXPort = 1;  
float o2 = 60;  
int result = setO2(STXPort, o2);
```

### 2.4.10.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "SetO2")]  
public static extern int setO2(int port, float o2);
```

//Execution:

```
int STXPort = 1;  
float o2 = 0;  
int result = setO2(STXPort, o2);
```

### 2.4.11 int ReadSetTemperature(int STXPort, float\* Temperature)

Returns the target temperature in °C.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
Temperature – Pointer to value of Temperature

Return values:

1 – value has been read  
-1 – Wrong device ID  
-2 – Unknown device ID  
-3 – Communication with the device not opened  
-12 – Error of read or write value to device  
-21 – wrong parameter value

Examples:

#### 2.4.11.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadSetTemperature)(int, float*);  
ReadSetTemperature readSetTemperature =  
(ReadSetTemperature)GetProcAddress(hDLL, "ReadSetTemperature");
```

'Execution:

```
int STXPort = 1;  
float setTemperature;  
int result = readSetTemperature(STXPort, &setTemperature);
```

#### 2.4.11.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadSetTemperature")]  
public static extern int readSetTemperature(int port, ref float  
temp);
```

//Execution:

```
int STXPort = 1;  
float temperature;  
int result = readSetTemperature(STXPort, ref temperature);
```

## 2.4.12 int ReadSetHumidity(int STXPort, float\* Humidity)

### Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
Humidity – Pointer to value of Humidity

### Return values:

1 – value has been read  
-1 – wrong device ID  
-2 – Unknown device ID  
-3 – Communication with the device not opened  
-12 – Error of read or write value to device  
-21 – wrong parameter value

### Examples:

#### 2.4.12.1 *Microsoft C++*

##### Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadSetHumidity)(int, float*);  
ReadSetHumidity readSetHumidity =  
(ReadSetHumidity)GetProcAddress(hDLL, "ReadSetHumidity");
```

##### Execution:

```
int STXPort = 1;  
float setHumidity;  
int result = readSetHumidity(STXPort, &setHumidity);
```

#### 2.4.12.2 *C#*

##### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadSetHumidity")]  
public static extern int readSetHumidity(int port, ref float hum);
```

//Execution:

```
int STXPort = 1;  
float humidity;  
int result = readSetHumidity(STXPort, ref hum);
```



### 2.4.13 int ReadSetCO2(int STXPort, float\* co2)

Returns the target CO2 concentration in percent.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
co2 – Pointer to value of CO2

Return values:

1 – value has been read  
-1 – wrong device ID  
-2 – Unknown device ID  
-3 – Communication with the device not opened  
-12 – Error of read or write value to device  
-21 – wrong parameter value

Examples:

#### 2.4.13.1 *Microsoft C++*

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadSetCO2)(int, float*);  
ReadSetCO2 readSetCO2 = (ReadSetCO2)GetProcAddress(hDLL,  
"ReadSetCO2");
```

'Execution:

```
int STXPort = 1;  
float setCO2;  
int result = readSetCO2(STXPort, &setCO2);
```

#### 2.4.13.2 *C#*

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadSetCO2")]  
public static extern int readSetCO2(int port, ref float co2);
```

//Execution:

```
int STXPort = 1;  
float co2;  
int result = readSetCO2(STXPort, ref co2);
```

#### 2.4.14 int ReadSetN2(int STXPort, float\* n2)

Returns the target N2 concentration in percent.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
n2 – Pointer to value of CO2

Return values:

1 – value has been read  
-1 – wrong device ID  
-2 – Unknown device ID  
-3 – Communication with the device not opened  
-12 – Error of read or write value to device  
-21 – wrong parameter value

Examples:

##### 2.4.14.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadSetN2)(int, float*);  
ReadSetN2 readSetN2 = (ReadSetN2)GetProcAddress(hDLL, "ReadSetN2");
```

'Execution:

```
int STXPort = 1;  
float setN2;  
int result = readSetN2(STXPort, &setN2);
```

##### 2.4.14.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadSetN2")]  
public static extern int readSetN2(int port, ref float n2);
```

//Execution:

```
int STXPort = 1;  
float n2;  
int result = readSetN2(STXPort, ref n2);
```

#### 2.4.15 int ReadSetO2(int STXPort, float\* o2)

Returns the target O2 concentration in percent.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
o2 – Pointer to value of O2

Return values:

1 – value has been read  
-1 – wrong device ID  
-2 – Unknown device ID  
-3 – Communication with the device not opened  
-12 – Error of read or write value to device  
-21 – wrong parameter value

Examples:

##### 2.4.15.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadSetO2)(int, float*);  
ReadSetO2 readSetO2 = (ReadSetO2)GetProcAddress(hDLL, "ReadSetO2");
```

'Execution:

```
int STXPort = 1;  
float setO2;  
int result = readSetO2(STXPort, &setO2);
```

##### 2.4.15.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadSet02")]  
public static extern int readSet02(int port, ref float o2);
```

//Execution:

```
int STXPort = 1;  
float o2;  
int result = readSet02(STXPort, ref o2);
```

## 2.5 Plate Handling Functions

### 2.5.1 `int LoadPlate(int STXPort, int Cassette, int Level)`

Loads a plate from the Transfer Station to specified Cassette and Level.

Parameters:

`STXPort` - Number of a Serial Port or Serial Number of the Device

`Slot` - plate slot position

`Level` - plate level position

Return values:

- 1 - Command completed successfully
- 1 - Wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 4 - System is not initialized
- 5 - Previous command is not completed
- 6 - Device error flag is true
- 7 - Device initialization flag is false
- 8 - Door opened
- 9 - Error waiting for completion
- 10 - Timeout of waiting for completion
- 11 - Device ready flag is false
- 12 - Error of read or write value to device
- 21 - Wrong parameter value
- 23 - Cassette exceed maximum value
- 24 - Level exceed maximum value

Examples:

#### 2.5.1.1 *Microsoft C++*

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));
typedef int(__cdecl* LoadPlate)(int, int, int);
LoadPlate loadPlate = (LoadPlate)GetProcAddress(hDLL, "LoadPlate");
```

'Execution:

```
int STXPort = 1;
int cassette = 3;
int level = 21;
int result = loadPlate(STXPort, cassette, level);
```

### 2.5.1.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "LoadPlate")]
public static extern int loadPlate(int port, int cassette, int level);
```

//Execution:

```
int STXPort = 1;
int cassette = 3;
int level = 21;
int result = loadPlate(STXPort, cassette, level);
```



## 2.5.2 int UnloadPlate(int STXPort, int Cassette, int Level)

Unloads a plate from the specified Slot and Level to the Transfer Station.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device

Slot - plate slot position

Level - plate level position

Return values:

- 1 - Command completed successfully
- 1 - wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 4 - System is not initialized
- 5 - Previous command is not completed
- 6 - Device error flag is true
- 7 - Device initialization flag is false
- 8 - Door opened
- 9 - Error waiting for completion
- 10 - Timeout of waiting for completion
- 11 - Device ready flag is false
- 12 - Error of read or write value to device
- 21 - wrong parameter value
- 23 - Cassette exceed maximum value
- 24 - Level exceed maximum value

Examples:

### 2.5.2.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* unloadPlate)(int, int, int);  
UnloadPlate unloadPlate = (UnloadPlate)GetProcAddress(hDLL,  
"UnloadPlate");
```

'Execution:

```
int STXPort = 1;  
int cassette = 3;  
int level = 21;  
int result = unloadPlate(STXPort, cassette, level);
```

#### 2.5.2.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "UnloadPlate")]  
public static extern int unloadPlate(int port, int cassette, int  
level);
```

//Execution:

```
int STXPort = 1;  
int cassette = 3;  
int level = 21;  
int result = unloadPlate(STXPort, cassette, level);
```

### 2.5.3 int PickPlate(int STXPort, int Cassette, int Level)

Picks Plate by Shovel from the specified Cassette and Level for further placing.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device

Slot - plate slot position

Level - plate level position

Return values:

- 1 - Command completed successfully
- 1 - wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 4 - System is not initialized
- 5 - Previous command is not completed
- 6 - Device error flag is true
- 7 - Device initialization flag is false
- 8 - Door opened
- 9 - Error waiting for completion
- 10 - Timeout of waiting for completion
- 11 - Device ready flag is false
- 12 - Error of read or write value to device
- 21 - wrong parameter value
- 23 - Cassette exceed maximum value
- 24 - Level exceed maximum value

Examples:

#### 2.5.3.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* PickPlate)(int, int, int);  
PickPlate pickPlate = (PickPlate)GetProcAddress(hDLL, "PickPlate");
```

'Execution:

```
int STXPort = 1;  
int cassette = 3;  
int level = 21;  
int result = pickPlate(STXPort, cassette, level);
```

### 2.5.3.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "PickPlate")]  
public static extern int pickPlate(int port, int cassette, int level);
```

//Execution:

```
int STXPort = 1;  
int cassette = 3;  
int level = 21;  
int result = pickPlate(STXPort, cassette, level);
```

## 2.5.4 int PlacePlate(int STXPort, int Cassette, int Level)

Places Plate from the Shovel to the specified Slot and Level.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device

Slot - plate slot position

Level - plate level position

Return values:

- 1 - Command completed successfully
- 1 - wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 4 - System is not initialized
- 5 - Previous command is not completed
- 6 - Device error flag is true
- 7 - Device initialization flag is false
- 8 - Door opened
- 9 - Error waiting for completion
- 10 - Timeout of waiting for completion
- 11 - Device ready flag is false
- 12 - Error of read or write value to device
- 21 - wrong parameter value
- 23 - Cassette exceed maximum value
- 24 - Level exceed maximum value

Examples:

### 2.5.4.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* PlacePlate)(int, int, int);  
PlacePlate placePlate = (PlacePlate)GetProcAddress(hDLL,  
"PlacePlate");
```

'Execution:

```
int STXPort = 1;  
int cassette = 3;  
int level = 21;  
int result = placePlate(STXPort, cassette, level);
```

#### 2.5.4.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "PlacePlate")]  
public static extern int placePlate(int port, int cassette, int  
level);
```

//Execution:

```
int STXPort = 1;  
int cassette = 3;  
int level = 21;  
int result = placePlate(STXPort, cassette, level);
```

### 2.5.5 int GetPlate(int STXPort)

Gets a Plate from the Transfer Station to the Shovel.

Parameter:

STXPort - Number of a Serial Port or Serial Number of the Device

Return values:

- 1 - Command completed successfully
- 1 - wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 4 - System is not initialized
- 5 - Previous command is not completed
- 6 - Device error flag is true
- 7 - Device initialization flag is false
- 8 - Door opened
- 9 - Error waiting for completion
- 10 - Timeout of waiting for completion
- 11 - Device ready flag is false
- 12 - Error of read or write value to device
- 21 - wrong parameter value

Examples:

#### 2.5.5.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* GetPlate)(int);  
GetPlate getPlate = (GetPlate)GetProcAddress(hDLL, "GetPlate");
```

'Execution:

```
int STXPort = 1;  
int result = getPlate(STXPort);
```

#### 2.5.5.2 C#

##### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "GetPlate")]  
public static extern int getPlate(int port);
```

##### //Execution:

```
int STXPort = 1;  
int result = getPlate(STXPort);
```



## 2.5.6 int SetPlate(int STXPort)

Moves a Plate from the Shovel to the Transfer Station.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device

Return values:

- 1 - Command completed successfully
- 1 - Wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 4 - System is not initialized
- 5 - Previous command is not completed
- 6 - Device error flag is true
- 7 - Device initialization flag is false
- 8 - Door opened
- 9 - Error waiting for completion
- 10 - Timeout of waiting for completion
- 11 - Device ready flag is false
- 12 - Error of read or write value to device
- 21 - wrong parameter value

Examples:

### 2.5.6.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* SetPlate)(int);  
SetPlate setPlate = (SetPlate)GetProcAddress(hDLL, "SetPlate");
```

'Execution:

```
int STXPort = 1;  
int result = setPlate(STXPort);
```

#### 2.5.6.2 C#

##### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "SetPlate")]  
public static extern int setPlate(int port);
```

##### //Execution:

```
int STXPort = 1;  
int result = setPlate(STXPort);
```

## 2.5.7 int ReadPlatePresence(int STXPort, int Cassette, int Level, bool\* Presence)

Reads whether Plate is present at Cassette/Level Location.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device

Slot - plate slot position

Level - plate level position

Presence - Pointer to Plate Presence value  
true: Plate has been detected  
false: Location is empty

Return values:

- 1 - Command completed successfully
- 1 - Wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 4 - System is not initialized
- 5 - Previous command is not completed
- 6 - Device error flag is true
- 7 - Device initialization flag is false
- 8 - Door opened
- 9 - Error waiting for completion
- 10 - Timeout of waiting for completion
- 11 - Device ready flag is false
- 12 - Error of read or write value to device
- 21 - Wrong parameter value
- 23 - Cassette exceed maximum value
- 24 - Level exceed maximum value

Examples:

### 2.5.7.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadPlatePresence)(int, int, int, bool*);  
ReadPlatePresence readPlatePresence = (GetPlate)GetProcAddress(hDLL,  
"ReadPlatePresence");
```

'Execution:

```
int STXPort = 1;  
int cassette = 3;  
int level = 21;  
bool isPlatePresent;  
int result = readPlatePresence(STXPort, cassette, level,  
&isPlatePresent);
```

## 2.5.7.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadPlatePresence")]  
public static extern int readPlatePresence(int port, int cassette, int  
level, ref bool isPlatePresent);
```

//Execution:

```
int STXPort = 1;  
int cassette = 3;  
int level = 21;  
bool isPlatePresent;  
int result = readPlatePresence(STXPort, cassette, level, ref  
isPlatePresent);
```

### 2.5.8 int ReadShovelDetector(int STXPort, bool\* Presence)

Reads whether Plate is present on the Shovel.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device

Presence - Pointer to Plate Presence value  
true: Plate has been detected on the Shovel  
false: No Plate on the shovel

Return values:

- 1 - Command completed successfully
- 1 - Wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 12 - Error of read or write value to device

Examples:

#### 2.5.8.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));
typedef int(__cdecl* ReadShovelDetector)(int, bool*);
ReadShovelDetector readShovelDetector =
(ReadShovelDetector)GetProcAddress(hDLL, "ReadShovelDetector");
```

'Execution:

```
int STXPort = 1;
bool isPlateShovelPresent;
int result = readShovelDetector(STXPort, &isPlateShovelPresent);
```

#### 2.5.8.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadShovelDetector")]  
public static extern int readShovelDetector(int port, ref bool  
isPlatePresent);
```

//Execution:

```
int STXPort = 1;  
bool isPlateShovelPresent = false;  
int result = readShovelDetector(STXPort, ref isPlateShovelPresent);
```

### 2.5.9 int ReadXferDetector(int STXPort, bool\* Presence)

Reads whether Plate is present on the Transfer Station.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device

Presence - Pointer to Plate Presence value  
true: Plate has been detected on the Transfer Station  
false: No Plate on the Transfer Station

Return values:

- 1 - Command completed successfully
- 1 - Wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 12 - Error of read or write value to device

Examples:

#### 2.5.9.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));
typedef int(__cdecl* ReadXferDetector)(int, bool*);
ReadXferDetector readXferDetector =
(ReadXferDetector)GetProcAddress(hDLL, "ReadXferDetector");
```

'Execution:

```
int STXPort = 1;
bool isPlateXferPresent;
int result = readXferDetector(STXPort, &isPlateXferPresent);
```

#### 2.5.9.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadXferDetector")]  
public static extern int readXferDetector(int port, ref bool  
isPlatePresent);
```

//Execution:

```
int STXPort = 1;  
bool isPlateXferPresent;  
int result = readXferDetector(STXPort, ref isPlateXferPresent);
```



### 2.5.10 int ReadXferDetector2(int STXPort, bool\* Presence)

Reads whether Plate is present on the second Transfer Station.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device

Presence - Pointer to Plate Presence value  
true: Plate has been detected on the Transfer Station  
false: No Plate on the Transfer Station

Return values:

- 1 - Command completed successfully
- 1 - Wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 12 - Error of read or write value to device

Examples:

#### 2.5.10.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));
typedef int(__cdecl* ReadXferDetector2)(int, bool*);
ReadXferDetector2 readXferDetector2 =
(ReadXferDetector2)GetProcAddress(hDLL, "ReadXferDetector2");
```

'Execution:

```
int STXPort = 1;
bool isPlateXfer2Present;
int result = readXferDetector2(STXPort, &isPlateXfer2Present);
```

#### 2.5.10.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadXferDetector2")]  
public static extern int readXferDetector2(int port, ref bool  
isPlatePresent);
```

//Execution:

```
int STXPort = 1;  
bool isPlateXfer2Present;  
int result = readXferDetector2(STXPort, ref isPlateXferPresent);
```



## 2.6 Barcode Reader Functions

### 2.6.1 int BCConnect(int STXPort, int BCPort)

Opens Serial Communication with Barcode Reader.

The barcode reader must be initialized with this function before being used.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device

BCPort – Name of Serial Port of Barcode Reader(e.g. "2")  
In Linux version of the library this parameter has char\* type and uses as value a Serial Port File name such as "/dev/ttyS1"

Returns values:

- 1 – Serial Port of Barcode Reader is successfully opened.
- 1 – wrong device ID
- 2 – Unknown device ID
- 13 – Serial port already in use
- 14 – Serial port does not exist
- 15 – Error opening serial port
- 20 – wrong barcode reader port.

Examples:

#### 2.6.1.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T(".dll"));  
typedef int(__cdecl* BCConnect)(int, int);  
BCConnect bconn = (BCConnect)GetProcAddress(hDLL, "BCConnect");
```

'Execution:

```
int STXPort = 1;  
int BCPort = 2;  
int result = bconn(STXPort, BCPort);
```

#### 2.6.1.2 C#

##### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "BCConnect")]  
public static extern int bconn(int port, int bcport);
```

##### //Execution:

```
int STXPort = 1;  
int BCPort = 2;  
int result = bconn(STXPort, BCPort);
```

## 2.6.2 int BCDisconnect (int STXPort)

Closes Serial Communication with Barcode Reader.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device

Returns values:

- 1 - Serial Port of Barcode Reader is successfully opened.
- 1 - Wrong device ID
- 2 - Unknown device ID
- 18 - Serial port not opened
- 19 - Error of closing serial port

Examples:

### 2.6.2.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));
typedef int(__cdecl* BCDisconnect)(int);
BCDisconnect bdconn = (BCDisconnect)GetProcAddress(hDLL,
"BCDisconnect");
```

'Execution:

```
int STXPort = 1;
int result = bdconn(STXPort);
```

### 2.6.2.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "BCDisconnect")]  
public static extern int bdconn(int port);
```

//Execution:

```
int STXPort = 1;  
int result = bdconn(STXPort);
```

### 2.6.3 int ReadBarcode(int STXPort, int Cassette, int Level, char \*Barcode)

Reads the Barcode of a Plate at specified Cassette and Level location.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device

Cassette - Cassette number

Level - Level number

Barcode - Pointer to a value of barcode

Return values:

- 1 - Wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 4 - System is not initialised
- 5 - Previous command is not completed
- 6 - Device error flag is true
- 7 - Device initialization flag is false
- 8 - Door opened
- 9 - Error waiting for completion
- 10 - Timeout of waiting for completion
- 11 - Device ready flag is false
- 12 - Error of read or write value to device
- 21 - wrong parameter value
- 22 - Barcode reader port not opened
- 23 - Cassette exceed maximum value
- 24 - Level exceed maximum value
- 25 - Plate not detected

Examples:



### 2.6.3.1 Microsoft C++

#### 'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadBarcode)(int, int, int, char*);  
ReadBarcode readBarcode = (ReadBarcode)GetProcAddress(hDLL,  
"ReadBarcode");
```

#### 'Execution:

```
int STXPort = 1;  
int cassette = 3;  
int level = 21;  
char barcode[50];  
int result = readBarcode(STXPort, cassette, level, &barcode[0]);
```

### 2.6.3.2 C#

#### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadBarcode")]  
public static extern int readBarcode(int port, int cassette, int  
level, StringBuilder barcode);
```

#### //Execution:

```
int STXPort = 1;  
int cassette = 1;  
int level = 2;  
StringBuilder barcode = new StringBuilder();  
int result = readBarcode(STXPort, cassette, level, barcode);
```

## 2.6.4 int Inventory(int STXPort, char\* FileName)

Implements inventory of entire Device, the result saves into the file – FileName.

Parameters:

PortNumber - Number of a Serial Port or Serial Number of the Device

FileName - Path and name of file for saving results of inventory.

Return values:

- 1 - Wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 4 - System is not initialised
- 5 - Previous command is not completed
- 6 - Device error flag is true
- 7 - Device initialization flag is false
- 8 - Door opened
- 9 - Error waiting for completion
- 10 - Timeout of waiting for completion
- 11 - Device ready flag is false
- 12 - Error of read or write value to device
- 21 - Wrong parameter value
- 22 - Barcode reader port not opened
- 31 - Error of creating a file
- 32 - Error of saving result to the file

The result of the Inventory is a file which consists of four columns separated by coma. The first column is a number of Cassette, the second column is a number of Level, the third column is a value of Plate Present Detector (1 - plate is present; 0 - plate is not present), the fourth column is value of a Barcode (<null> - No Barcode or barcode was not detected).

For example:

```
1,1,1,<null>
1,2,0,<null>
1,3,1,0484856654
...
```

Examples:

#### 2.6.4.1 Microsoft C++

##### 'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));
typedef int(__cdecl* Inventory)(int, char*);
Inventory STXInventory = (Inventory)GetProcAddress(hDLL,
"Inventory");
```

##### 'Execution:

```
int STXPort = 1;
char filePath[100] = "C:\\Data\\Inventory.txt"
int result = STXInventory(STXPort, &filePath[0]);
```

#### 2.6.4.2 C#

##### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "Inventory")]
public static extern int STXInventory(int port, string path);
```

##### //Execution:

```
int STXPort = 1;
string filePath = " C:\\Data\\Inventory.txt "
int result = STXInventory(STXPort, filePath)
```



## 2.7 Shaker Functions

### 2.7.1 int SetShakerSpeed(int STXPort, int Value)

Sets Shaker Speed.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
value – value of Shaker Speed

Return values:

- 1 – value is set to device.
- 1 – Wrong device ID
- 2 – Unknown device ID
- 3 – Communication with the device not opened
- 12 – Error of read or write value to device
- 21 – wrong parameter value

Examples:

#### 2.7.1.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* setShakersSpeed)(int, int);  
setShakersSpeed setShakersSpeed = (setShakersSpeed)GetProcAddress(hDLL,  
"setShakersSpeed");
```

'Execution:

```
int STXPort = 1;  
int shakersSpeed = 2;  
int result = setShakersSpeed(STXPort, shakersSpeed);
```

### 2.7.1.2 C#

#### //Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "SetShakersSpeed")]  
public static extern int setShakersSpeed(int port, int speed);
```

#### //Execution:

```
int STXPort = 1;  
int shakersSpeed = 2;  
int result = setShakersSpeed(STXPort, shakersSpeed);
```

## 2.7.2 int ReadShakerSpeed(int STXPort, int\* Value)

Reads value of Shaker Speed.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device  
value – Pointer to value of Shaker Speed

Return values:

- 1 – value has been read
- 1 – wrong device ID
- 2 – Unknown device ID
- 3 – Communication with the device not opened
- 12 – Error of read or write value to device
- 21 – wrong parameter value

Examples:

### 2.7.2.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* ReadShakersSpeed)(int, int*);  
ReadShakersSpeed readShakersSpeed =  
(ReadShakersSpeed)GetProcAddress(hDLL, "ReadShakersSpeed");
```

'Execution:

```
int STXPort = 1;  
int shakerSpeed;  
int result = readShakersSpeed(STXPort, &shakerSpeed);
```

### 2.7.2.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "ReadShakersSpeed")]  
public static extern int readShakersSpeed(int port, ref int speed);
```

//Execution:

```
int STXPort = 1;  
int shakersSpeed = -1;  
int result = readShakersSpeed(STXPort, ref shakersSpeed);
```



### 2.7.3 int RunShaker(int STXPort, bool Run)

Switches Shaker on and off.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device

Run –     true: Shaker switched on  
          false: Shaker switched off

Return values:

- 1 – Shaker was turned on/off
- 1 – Wrong device ID
- 2 – Unknown device ID
- 3 – Communication with the device not opened
- 12 – Error of read or write value to device

Examples:

#### 2.7.3.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* RunShaker)(int, bool);  
RunShaker runShaker = (RunShaker)GetProcAddress(hDLL, "RunShaker");
```

'Execution:

```
int STXPort = 1;  
int result = runShaker(STXPort, true);
```

#### 2.7.3.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "RunShaker")]  
public static extern int runShaker(int port, bool isRunning);
```

//Execution:

```
int STXPort = 1;  
int result = runShaker(STXPort, true);
```



## 2.8 Transfer Station finctions

### 2.8.1 int SwapXferStation(int STXPort, bool Swap)

Rotates Swap Station.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device

Swap - true: Rotates the swap station on 180 degree  
false: Rotates the swap station back to home position

Return values:

- 1 - Position of Transferstation has been swaped
- 1 - wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 12 - Error of read or write value to device

Examples:

#### 2.8.1.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* SwapXferStation)(int, bool);  
SwapXferStation swapXferStation =  
(SwapXferStation)GetProcAddress(hDLL, "SwapXferStation");
```

'Execution:

```
int STXPort = 1;  
int result = swapXferStation(STXPort, true);
```

#### 2.8.1.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "SwapXferStation")]  
public static extern int swapXferStation(int port, bool isRotating);
```

//Execution:

```
int STXPort = 1;  
int result = swapXferStation(STXPort, true);
```

## 2.8.2 int AlternateXferStation(int STXPort, bool Swap)

Changes Input and Output position of Transfer Station.

Parameters:

STXPort - Number of a Serial Port or Serial Number of the Device

Swap - true: Swaps Input and Output position of Transfer Station  
false: Set Transfer Station to default position

Return values:

- 1 - Input/Output Position of Transferstation has been changed
- 1 - wrong device ID
- 2 - Unknown device ID
- 3 - Communication with the device not opened
- 12 - Error of read or write value to device

Examples:

### 2.8.2.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));  
typedef int(__cdecl* AlternateXferStation)(int, bool);  
AlternateXferStation alternateXferStation =  
(AlternateXferStation)GetProcAddress(hDLL, "AlternateXferStation");
```

'Execution:

```
int STXPort = 1;  
int result = alternateXferStation(STXPort, true);
```

### 2.8.2.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "AlternateXferStation")]  
public static extern int alternateXferStation(int port, bool  
isAlternating);
```

//Execution:

```
int STXPort = 1;  
int result = alternateXferStation(STXPort, true);
```

## 2.9 Gate functions

### 2.9.1 int OpenTransferGate(int STXPort)

Opens Transfer Station Gate.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device

Return values:

- 1 – Input/Output Position of Transferstation has been changed
- 1 – Wrong device ID
- 2 – Unknown device ID
- 3 – Communication with the device not opened
- 12 – Error of read or write value to device

Examples:

#### 2.9.1.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));
typedef int(__cdecl* OpenTransferGate)(int);
OpenTransferGate openTransferGate =
(OpenTransferGate)GetProcAddress(hDLL, "OpenTransferGate");
```

'Execution:

```
int STXPort = 1;
int result = openTransferGate(STXPort);
```

#### 2.9.1.2 C#

//Declaration:



```
[DllImport("STXLib.dll", EntryPoint = "OpenTransferGate")]  
public static extern int openTransferGate(int port);
```

//Execution:

```
int STXPort = 1;  
int result = openTransferGate(STXPort);
```

### 2.9.2 int CloseTransferGate(int STXPort)

Closes Transfer Station Gate.

Parameters:

STXPort – Number of a Serial Port or Serial Number of the Device

Return values:

- 1 – Input/Output Position of Transferstation has been changed
- 1 – wrong device ID
- 2 – Unknown device ID
- 3 – Communication with the device not opened
- 12 – Error of read or write value to device

Examples:

#### 2.9.2.1 Microsoft C++

'Declaration:

```
HINSTANCE hDLL = LoadLibrary(_T("STXLib.dll"));
typedef int(__cdecl* CloseTransferGate)(int);
CloseTransferGate CloseTransferGate =
(CloseTransferGate)GetProcAddress(hDLL, "CloseTransferGate");
```

'Execution:

```
int STXPort = 1;
int result = CloseTransferGate(STXPort);
```

#### 2.9.2.2 C#

//Declaration:

```
[DllImport("STXLib.dll", EntryPoint = "CloseTransferGate")]  
public static extern int closeTransferGate(int port);
```

//Execution:

```
int STXPort = 1;  
int result = closeTransferGate(STXPort);
```