

# Instruction Manual

## AlprSDK\_JNI Interface

Thank you for choosing our product. Please read the instructions carefully before operation. Follow these instructions to ensure that the product is functioning properly. The images shown in this manual are for illustrative purposes only.



For further details, please visit our Company's website  
[www.zkteco.in](http://www.zkteco.in).

Copyright © 2026 ZKTECO CO., LTD. All rights reserved.

Without the prior written consent of ZKTeco, no portion of this manual can be copied or forwarded in any way or form. All parts of this manual belong to ZKTeco and its subsidiaries (hereinafter the "Company" or "ZKTeco").

### ZKTeco Biometrics India Pvt Ltd.

**Address** B-1, Khata No. 31, Devasandra Industrial Estate,  
Whitefield Main Road, Beside Phoenix mall,  
Mahadevapura, Bengaluru-560048, Karnataka, India.

**Phone** 080 6828 1342

For business related queries, please write to us at: [sales@zkteco.in](mailto:sales@zkteco.in).

To know more about our global branches, visit [www.zkteco.com](http://www.zkteco.com).

### About the Manual

This manual introduces the operations of AlprSDK\_JNI product.

All figures displayed are for illustration purposes only. Figures in this manual may not be exactly consistent with the actual products.

## Document Conventions

Conventions used in this manual are listed below:

### GUI Conventions

| For Software     |                                                                                                                             |
|------------------|-----------------------------------------------------------------------------------------------------------------------------|
| Convention       | Description                                                                                                                 |
| <b>Bold font</b> | Used to identify software interface names e.g. <b>OK</b> , <b>Confirm</b> , <b>Cancel</b>                                   |
| <b>&gt;</b>      | Multi-level menus are separated by these brackets. For example,<br>File > Create > Folder.                                  |
| For Device       |                                                                                                                             |
| Convention       | Description                                                                                                                 |
| <b>&lt; &gt;</b> | Button or key names for devices. For example, press <OK>                                                                    |
| <b>[ ]</b>       | Window names, menu items, data table, and field names are inside square brackets. For example, pop up the [New User] window |
| <b>/</b>         | Multi-level menus are separated by forwarding slashes. For example, [File/Create/Folder].                                   |

### Symbols

| Convention | Description |
|------------|-------------|
|------------|-------------|

|                                                                                   |                                                                                        |
|-----------------------------------------------------------------------------------|----------------------------------------------------------------------------------------|
|  | This implies about the notice or pays attention to, in the manual                      |
|  | The general information which helps in performing the operations faster                |
|  | The information which is significant                                                   |
|  | Care taken to avoid danger or mistakes                                                 |
|  | The statement or event that warns of something or that serves as a cautionary example. |

# Table of Contents

|                                                   |           |
|---------------------------------------------------|-----------|
| <b>Table of Contents</b>                          | <b>5</b>  |
| <b>1 Brief introduction</b>                       | <b>14</b> |
| 1.1 About Our Product                             | 14        |
| 1.2 Features                                      | 14        |
| 1.3 Programming Model                             | 15        |
| 1.4 About the value of Handle                     | 16        |
| <b>2 AlprSDK Architecture</b>                     | <b>18</b> |
| 2.1 Supported Platform                            | 18        |
| <b>3 Brief introduction</b>                       | <b>19</b> |
| 3.1 Initialization of the SDK                     | 19        |
| 3.2 Search for Local Devices                      | 19        |
| 3.2.1 Search for Local Devices                    | 19        |
| 3.3 Initialization handle                         | 20        |
| 3.3.1 Initialization of the Window Handle         | 20        |
| 3.4 Device Connections                            | 20        |
| 3.4.1 Set Connection Timeout                      | 20        |
| 3.4.2 Connect to Device                           | 21        |
| 3.4.3 Send HeartBeat                              | 21        |
| 3.5 Callback Functions                            | 21        |
| 3.5.1 License Plate Recognition                   | 21        |
| 3.5.2 Capture Image                               | 22        |
| 3.5.3 Device Status Callback functions            | 22        |
| 3.5.4 Device Connection Status Callback functions | 23        |

|                                                  |           |
|--------------------------------------------------|-----------|
| 3.5.5 Wiegand Card Data Callback functions.....  | 23        |
| 3.5.6 Video Callback functions.....              | 23        |
| 3.6 Video connection.....                        | 24        |
| 3.7 Close connection.....                        | 24        |
| 3.8 Release handle.....                          | 24        |
| 3.9 Release SDK.....                             | 24        |
| 3.10 Other Interface Functions.....              | 25        |
| 3.10.1 Recognition Area.....                     | 25        |
| 3.10.2 Device Information.....                   | 25        |
| 3.10.3 Device Control Functions.....             | 26        |
| 3.10.4 Information Parameter Settings.....       | 26        |
| 3.10.5 License Plate Information Management..... | 30        |
| 3.10.6 Data Delivery Management.....             | 30        |
| 3.10.7 RS485 Communication Settings.....         | 31        |
| 3.10.8 Video Stream Settings.....                | 31        |
| 3.10.9 Auxiliary Out and P2P Settings.....       | 31        |
| <b>4 AlprSDK Functions.....</b>                  | <b>33</b> |
| 4.1 AlprSDK_Startup.....                         | 33        |
| 4.2 AlprSDK_InitHandle.....                      | 34        |
| 4.3 AlprSDK_InitVideoHandle.....                 | 35        |
| 4.4 AlprSDK_InitVideoIntHandle.....              | 36        |
| 4.5 AlprSDK_SearchAllCameras.....                | 37        |
| 4.6 DevFindCallback.....                         | 38        |
| 4.7 AlprSDK_SetConnectTimeout.....               | 40        |
| 4.8 AlprSDK_ConnectDev.....                      | 41        |
| 4.9 AlprSDK_SendHeartBeat.....                   | 42        |
| 4.10 AlprSDK_CreateRecogAllInfoTask.....         | 43        |
| 4.11 RecogAllInfoCallback.....                   | 44        |
| 4.12 IRecogAllInfoCallback.....                  | 45        |
| 4.13 AlprSDK_ClearRecogAllInfoTask.....          | 46        |
| 4.14 AlprSDK_CreateCaptureJpgCallback.....       | 47        |

|                                             |    |
|---------------------------------------------|----|
| 4.15 DeviceCaptureCallback.....             | 48 |
| 4.16 IDeviceCaptureCallback.....            | 49 |
| 4.17 AlprSDK_ClearCaptureJpgCallback.....   | 50 |
| 4.18 AlprSDK_CaptureJpg.....                | 50 |
| 4.19 AlprSDK_CreateDevStatusCallback.....   | 52 |
| 4.20 DevStatusCallback.....                 | 53 |
| 4.21 IDevStatusCallback.....                | 54 |
| 4.22 AlprSDK_ClearDevStatusCallback.....    | 55 |
| 4.23 AlprSDK_SetDevConnStatusCallBack.....  | 55 |
| 4.24 DevConnStatusCallback.....             | 57 |
| 4.25 IDevConnStatusCallback.....            | 58 |
| 4.26 AlprSDK_CreateWiegandDataCallback..... | 58 |
| 4.27 WiegandDataCallback.....               | 59 |
| 4.28 AlprSDK_ClearWiegandDataCallback.....  | 60 |
| 4.29 AlprSDK_CreateEZStreamDataCB.....      | 60 |
| 4.30 StreamDataCallBack.....                | 63 |
| 4.31 IStreamDataCallBack.....               | 64 |
| 4.32 AlprSDK_ClearEZStreamDataCB.....       | 64 |
| 4.33 AlprSDK_StartVideo.....                | 65 |
| 4.34 AlprSDK_StopVideo.....                 | 66 |
| 4.35 AlprSDK_DisConnectDev.....             | 67 |
| 4.36 AlprSDK_UnInitHandle.....              | 68 |
| 4.37 AlprSDK_Cleanup.....                   | 68 |
| 4.38 AlprSDK_SetRoiEx.....                  | 70 |
| 4.39 AlprSDK_GetRoiEx.....                  | 71 |
| 4.40 AlprSDK_SetVirtualCoil.....            | 72 |
| 4.41 AlprSDK_GetVirtualCoil.....            | 73 |
| 4.42 AlprSDK_GetDeviceFunList.....          | 75 |
| 4.43 AlprSDK_GetDevLicenseInfo.....         | 77 |
| 4.44 AlprSDK_GetDevVerInfo.....             | 77 |
| 4.45 AlprSDK_GetDevSN.....                  | 78 |
| 4.46 AlprSDK_SetDevSN.....                  | 79 |
| 4.47 AlprSDK_GetDeviceParam.....            | 80 |
| 4.48 AlprSDK_OpenGate.....                  | 81 |
| 4.49 AlprSDK_ControlZoom.....               | 82 |
| 4.50 AlprSDK_ControlFocus.....              | 85 |
| 4.51 AlprSDK_RebootDevice.....              | 87 |
| 4.52 AlprSDK_SetGateCfg.....                | 87 |
| 4.53 AlprSDK_GetGateCfg.....                | 88 |
| 4.54 AlprSDK_SetAlprCfg.....                | 89 |
| 4.55 AlprSDK_GetAlprCfg.....                | 90 |
| 4.56 AlprSDK_SetNAlprCfg.....               | 91 |
| 4.57 AlprSDK_GetNAlprCfg.....               | 93 |
| 4.58 AlprSDK_SetOfflineParam.....           | 94 |

|                                           |         |
|-------------------------------------------|---------|
| 4.59 AlprSDK_GetOfflineParam.....         | 95      |
| 4.60 AlprSDK_SetBaseParam.....            | 96      |
| 4.61 AlprSDK_GetBaseParam.....            | 106     |
| 4.62 AlprSDK_OperateSDCard.....           | 107     |
| 4.63 AlprSDK_ResetFactory.....            | 108     |
| 4.64 AlprSDK_SetDevInfo.....              | 110     |
| 4.65 AlprSDK_GetDevInfo.....              | 111     |
| 4.66 AlprSDK_StartMDThrDetect.....        | 112     |
| 4.67 AlprSDK_GetMDThreshold.....          | 112     |
| 4.68 AlprSDK_GetNetworkParam.....         | 113     |
| 4.69 AlprSDK_SetCountryCode.....          | 114     |
| 4.70 AlprSDK_GetImageList.....            | 125     |
| 4.71 AlprSDK_SetNetworkParam.....         | 126     |
| 4.72 AlprSDK_SetP2PServerAddress.....     | 129     |
| 4.73 AlprSDK_SetOSD.....                  | 130     |
| 4.74 AlprSDK_GetOSD.....                  | 132     |
| 4.75 AlprSDK_SetFtpClientCfg.....         | 133     |
| 4.76 AlprSDK_GetFtpClientCfg.....         | 134     |
| 4.77 AlprSDK_SetSysTime.....              | 136     |
| 4.78 AlprSDK_GetSysTime.....              | 138     |
| 4.79 AlprSDK_SetVideoParam.....           | 140     |
| 4.80 AlprSDK_GetVideoParam.....           | 141     |
| 4.81 AlprSDK_AddPlateListExt.....         | 142     |
| 4.82 AlprSDK_DelplateListExt.....         | 143     |
| 4.83 AlprSDK_ClearPlateListExt.....       | 145     |
| 4.84 AlprSDK_ExportPlateListExt.....      | 146     |
| 4.85 AlprSDK_GetDeviceDataCount.....      | 147     |
| 4.86 AlprSDK_GetDeviceData.....           | 148     |
| 4.87 AlprSDK_SetDeviceData.....           | 150     |
| 4.88 AlprSDK_ClearDeviceData.....         | 153     |
| 4.89 AlprSDK_CommTransparentTransfer..... | 154     |
| 4.90 AlprSDK_SetDColorDLineSCNParam.....  | 155     |
| 4.91 AlprSDK_GetVideoSize.....            | 157     |
| 4.92 AlprSDK_SetVideoConfig.....          | 158     |
| 4.93 AlprSDK_GetVideoConfig.....          | 159     |
| 4.94 AlprSDK_OpenAuxOut.....              | 160     |
| 4.95 AlprSDK_EnableP2PReconnect.....      | 161     |
| 4.96 AlprSDK_DisableP2PReconnect.....     | 162     |
| <br>5 Structure definition.....           | <br>164 |
| <br>5.1 PLATERESULT.....                  | <br>164 |
| 5.2 DEVINFO.....                          | 166     |
| 5.3 ALPR_VIDEO_CONFIG.....                | 167     |

|                               |            |
|-------------------------------|------------|
| 5.4 DEV_VER_INFO.....         | 168        |
| 5.5 ALPR_CONFIG.....          | 169        |
| 5.6 OSD_CTRL.....             | 172        |
| 5.7 BASE_PARAM.....           | 174        |
| 5.8 PLATE_INFO_EXT.....       | 176        |
| 5.9 GATE_CONFIG.....          | 177        |
| 5.10 DSCREENDLIN.....         | 178        |
| 5.11 WIEGANDDATA.....         | 180        |
| 5.12 ZOOMCTLPARA.....         | 180        |
| 5.13 FOCUSCTLPARA.....        | 181        |
| 5.14 JPG_BYTES.....           | 182        |
| 5.15 XRECT.....               | 183        |
| <b>5.16 DEVLICENSE.....</b>   | <b>184</b> |
| 5.17 SYS_TIME.....            | 185        |
| 5.18 OFFLINEPARAM.....        | 186        |
| 5.19 SDCARDCAPACITY.....      | 189        |
| <b>5.20 NETWORKPARAM.....</b> | <b>190</b> |
| <b>5.21 VIDEOPARAM.....</b>   | <b>190</b> |
| 5.22 DEVICEPARAM.....         | 191        |
| 5.23 WERAPIVER.....           | 192        |
| 5.24 RECOG_ALL_INFO.....      | 193        |
| 5.25 DEVINFO.....             | 194        |
| 5.26 DEVSTATUS.....           | 196        |
| 5.27 DEFBRIGHTNESS.....       | 197        |
| 5.28 CHARGELOG.....           | 197        |
| 5.29 OSDPARAM.....            | 199        |
| 5.30 FPCLIENTCFG.....         | 199        |
| 5.31 RS485CONFIG.....         | 201        |
| 5.32 RS485DATA.....           | 202        |
| 5.33 AUXCAMERASTATUS.....     | 203        |
| 5.34 DEVS.....                | 204        |
| 5.35 NALPRCFG.....            | 205        |
| 5.36 COUNTRY_CODE.....        | 206        |
| 5.37 XPOINT.....              | 216        |
| 5.38 LICENSE_PLATE.....       | 216        |
| 5.39 XDATE.....               | 218        |
| <b>6 Appendixes.....</b>      | <b>219</b> |
| Appendix 1 – Error Code.....  | 219        |

|                                                                 |     |
|-----------------------------------------------------------------|-----|
| Appendix 2– Voice board and display communication protocol..... | 221 |
| Communication Data Format Definition.....                       | 221 |
| Communication Procedure.....                                    | 222 |
| Communication Command.....                                      | 222 |
| Appendix 3 - Voice Segment Definition Description.....          | 225 |
| Appendix 4 - RS485 Communication Protocol.....                  | 228 |
| Communication Data Format Definition.....                       | 228 |
| Communication Procedure.....                                    | 229 |
| Communication Command.....                                      | 230 |
| Communication Command – Error Code.....                         | 239 |
| Appendix 5 - Language Code Description.....                     | 241 |

Last update: Jan 18, 2019

| Version      | Update time | Update contents                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|--------------|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1.0.1.2<br>4 | 2018-08-24  | Ported from the Windows version, remove video related APIs, etc.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| 1.0.1.2<br>5 | 2018-11-02  | <ol style="list-style-type: none"> <li>1. Add a local device search interface (function 3-1).</li> <li>2. Add a video stream callback registration interface (function 5-3-4).</li> <li>3. Add a video stream open interface (function 6-1).</li> <li>4. Add a video stream to close the interface (6-2).</li> <li>5. Added video width acquisition interface (function 11-8-1).</li> <li>6. Add a video stream information setting interface (function 11-8-2).</li> <li>7. Add a video stream information acquisition interface (function 11-8-3).</li> </ol> |
| 1.0.1.2<br>6 | 2018-11-08  | <ol style="list-style-type: none"> <li>1. Added camera zoom control interface (function 11-3).</li> <li>2. Added camera focus control interface (function 11-3).</li> </ol>                                                                                                                                                                                                                                                                                                                                                                                     |

|              |            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|--------------|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1.0.1.2<br>7 | 2018-12-28 | <ol style="list-style-type: none"> <li>1. Update recovery device parameter interface (function 11-4-5)</li> <li>2. Added device network parameter interface (function 11-4-12)</li> <li>3. Added P2P server address interface (function 11-4-13)</li> <li>4. Added OSD control interface (function 11-4-14)</li> <li>5. Added Ftp setting interface (function 11-4-15)</li> <li>6. New device time interface (function 11-4-16)</li> <li>7. Added auxiliary output interface (function 11-12)</li> <li>8. Added P2P reconnection mechanism interface (function 11-13)</li> </ol> |
| 1.0.1.2<br>8 | 2019-01-10 | <ol style="list-style-type: none"> <li>1. Added device system time interface (function 11-4-14-2)</li> <li>2. Added device parameter interface (Function 11-2-6)</li> </ol>                                                                                                                                                                                                                                                                                                                                                                                                      |
| 1.0.1.2<br>9 | 2019-01-18 | <ol style="list-style-type: none"> <li>1. Update device network parameter interface support for p2p (11-4-10)</li> <li>2. Update the device interface to support p2p (11-3-3)</li> <li>3. Update recovery device interface support for p2p (11-4-5)</li> <li>4. Update device time interface support for p2p (11-4-14)</li> <li>5. Update OSD control interface support for p2p (11-4-12)</li> <li>6. Update Get device parameter interface support for p2p</li> </ol>                                                                                                           |

|  |  |                                                                            |
|--|--|----------------------------------------------------------------------------|
|  |  | (11-2-6)<br><br>7. Added video day and night switching interface (11-4-15) |
|--|--|----------------------------------------------------------------------------|

# 1 Brief introduction

## 1.1 About Our Product

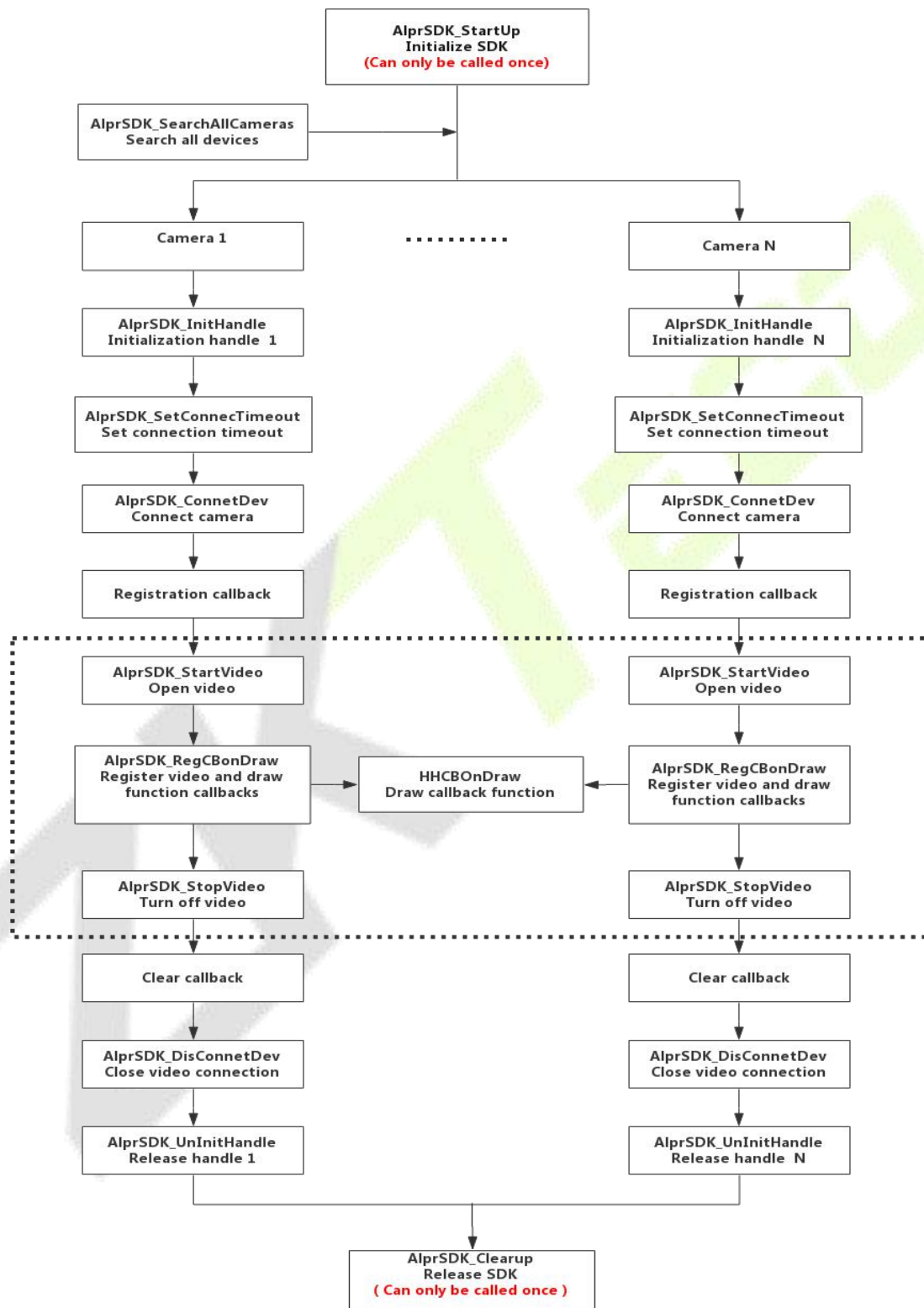
From the perspective of a developer, the key design objective of **AlprSDK** is compatibility and the ease of execution.

This Reference Manual contains the product development documentation for developers that describes the functions provided by the SDK and its related usage, which eases the development environment. In the following sections all the required information on how to integrate **AlprSDK** into a third-party application is provided.

## 1.2 Features

- **License Plate:** Gets the license plate, license plate image and the time to capture images by registering the license plate callback function.
- **Device Status:** Obtains the status of the online device, auxiliary camera, and the switch by registering the device status callback function.
- **Record:** Gets the real-time records by registering the record callback function.
- **Manual Capture:** Register manual capture callback. Call start capture function, can trigger callback.
- **Data management:** Support black and white list data management.
- **Remote control:** Supports remote opening function.
- **Display control:** Supports for transparent transmission: control of language and display

## 1.3 Programming Model

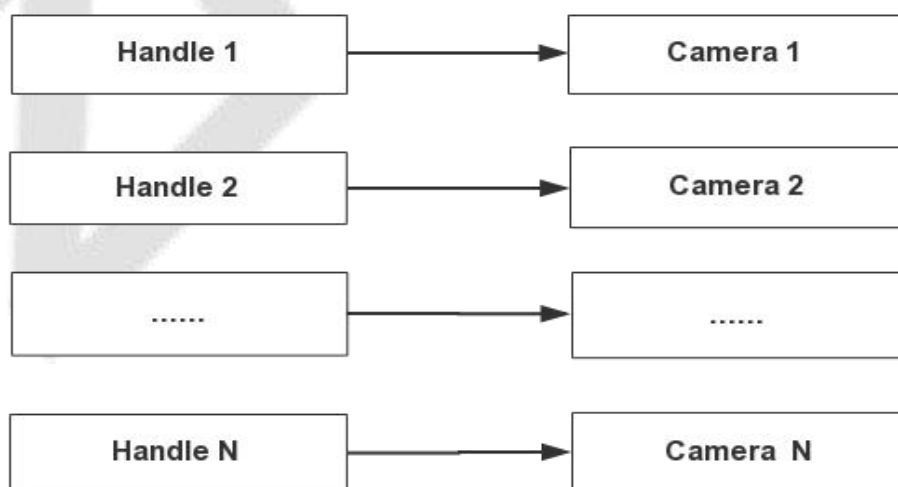


**Note:**

- License plate camera use: AlprSDK\_CreateRecogAllInfoTask (the license plate callback function).
- Parking camera use: AlprSDK\_CreateCarSpaceTask (parking callback function).
- Video-related interfaces are not supported on linux systems.

**Callback function considerations**

- SDK interface functions cannot be called in this type of function
- The total function call time cannot exceed 100ms
- There cannot be a sleep function

**1.4 About the value of Handle**

### Notes for Functions Containing Handle :

- Software management allocation, one handle id corresponds to one device.
- If multiple videos need to be played at the same time, handle has a value ranging from 0 to 18. if no video needs to be played, handle has a value ranging from 0 to 1024
- This function is called in a multithreaded concurrent environment. if it is the same Handle id, the upper layer needs lock protection.

## 2 AlprSDK Architecture

### 2.1 Supported Platform

| Operating System                      | SDK Package Path                                                                           | Demo Package Path                                                               |
|---------------------------------------|--------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
| Windows<br>32- or<br>64-bit<br>system | license plate business SDK<br>secondary development<br>package /SDK/ windows /<br>java/dll | License plate business SDK<br>secondary development<br>package/SDK/windows/java |
| Linux 32-<br>or 64-bit<br>system      | license plate business SDK<br>secondary development<br>package/SDK/windows/linux<br>/dll   | license plate business SDK<br>secondary development<br>package/SDK/linux/java   |

**Note:**

- Windows provides 32-bit SDK and 64-bit SDK, which is compatible with 64-bit systems and Video preview interface is not supported under 64-bit SDK. And the demo package provides C++ and C# source code.
- The SDK of Linux does not contain a video preview interface. But using the BS service architecture, the video preview plug-in can be called into the browser.
- This SDK document describes the commonly used function structure descriptions in detail. For other structure definitions, please refer to AlprSDK.h.

## 3 Brief introduction

Our AlprSDK\_JNI document includes all features that define the functions and related usage of the SDK and simplifies the development process.

### 3.1 Initialization of the SDK

This function contains the tasks of loading AlprSDK to the application process space and creating an instance.

| Parameter       | Description                                              |
|-----------------|----------------------------------------------------------|
| AlprSDK_Startup | Initialize the SDK and specify the path to the log file. |

### 3.2 Search for Local Devices

#### 3.2.1 Search for Local Devices

This function facilitates in searching all the available/connected Camera Devices

| Parameter                | Description                           |
|--------------------------|---------------------------------------|
| AlprSDK_SearchAllCameras | Initialize window handle.             |
| DevFindCallback          | (local) device search callback class. |

## 3.3 Initialization handle

### 3.3.1 Initialization of the Window Handle

Generally simplified as hWnd is a unique identifier assigned to each created window by Windows.

This function performs the task to initialize the Window handles based on the availability of the devices.

| Parameter                  | Description                                              |
|----------------------------|----------------------------------------------------------|
| AlprSDK_InitHandle         | Initialize window handle.                                |
| AlprSDK_InitVideoHandle    | Initialize window handle, supports window video playback |
| AlprSDK_InitVideoIntHandle | Initialize window handle.                                |

## 3.4 Device Connections

The following functions initiates the basic connection operations.

### 3.4.1 Set Connection Timeout

This function set the amount of time a connection waits to time out by using the Connect Timeout or Connection Timeout keywords in the connection string.

| Function                  | Description                                |
|---------------------------|--------------------------------------------|
| AlprSDK_SetConnectTimeout | This function sets the connection timeout. |

### 3.4.2 Connect to Device

This function obtains the available device information and connects all the searched devices.

| Function                           | Description                                         |
|------------------------------------|-----------------------------------------------------|
| <a href="#">AlprSDK_ConnectDev</a> | This function connects the available camera devices |

### 3.4.3 Send HeartBeat

This function triggers a periodic signal generated by the device to indicate normal operation.

Usually a heartbeat is sent between devices at a regular interval in the order of seconds.

| Function                              | Description                                                                    |
|---------------------------------------|--------------------------------------------------------------------------------|
| <a href="#">AlprSDK_SendHeartBeat</a> | This function sends the heartbeat signals to get the device connection status. |

## 3.5 Callback Functions

A callback function is a function that is passed as an argument to another function, to be "called back" at a later time.

### 3.5.1 License Plate Recognition

This function processes the License Plate Recognition callback function settings.

| Function                                       | Description                                        |
|------------------------------------------------|----------------------------------------------------|
| <a href="#">AlprSDK_CreateRecogAllInfoTask</a> | License plate callback function settings.          |
| <a href="#">AlprSDK_ClearRecogAllInfoTask</a>  | This function processes to clear the License Plate |

|                      |                                                       |
|----------------------|-------------------------------------------------------|
|                      | Callback function.                                    |
| RecogAllInfoCallback | License plate recognition information callback class. |

### 3.5.2 Capture Image

This function processes the capturing of the Vehicle images.

| Function                         | Description                                                      |
|----------------------------------|------------------------------------------------------------------|
| AlprSDK_CreateCaptureJpgCallback | Registered device grab callback                                  |
| AlprSDK_ClearCaptureJpgCallback  | Device grab callback clean-up                                    |
| DeviceCaptureCallback            | Image data capture and recall class                              |
| AlprSDK_CaptureJpg               | The device captures the image, and returns it through a callback |

### 3.5.3 Device Status Callback functions

The below functions processes to notify the status of the Device.

| Function                        | Description                       |
|---------------------------------|-----------------------------------|
| AlprSDK_CreateDevStatusCallback | Registered device status callback |
| AlprSDK_ClearDevStatusCallback  | Clear device status callback      |
| DevStatusCallback               | Device status callback class      |

### 3.5.4 Device Connection Status Callback functions

| Function                                         | Description                                      |
|--------------------------------------------------|--------------------------------------------------|
| <a href="#">AlprSDK_SetDevConnStatusCallBack</a> | Register device connection status callback       |
| <a href="#">DevConnStatusCallback</a>            | Device connection status callback delegate class |

### 3.5.5 Wiegand Card Data Callback functions

This function processes to get the Wiegand card data information

| Function                                          | Description                                 |
|---------------------------------------------------|---------------------------------------------|
| <a href="#">AlprSDK_CreateWiegandDataCallback</a> | Wiegand data callback                       |
| <a href="#">AlprSDK_ClearWiegandDataCallback</a>  | Clear the Wiegand card data callback.       |
| <a href="#">WiegandDataCallback</a>               | Get the Wiegand card number callback class. |

### 3.5.6 Video Callback functions

The following functions initializes the video and sets the essential video configurations

| Function                                     | Description                                 |
|----------------------------------------------|---------------------------------------------|
| <a href="#">AlprSDK_CreateEZStreamDataCB</a> | Register video stream callback              |
| <a href="#">AlprSDK_ClearEZStreamDataCB</a>  | Clear stream data callback (aware platform) |
| <a href="#">StreamDataCallBack</a>           | Video stream callback class.                |

### 3.6 Video connection

This function processes to power on and off the video.

| Function           | Description                    |
|--------------------|--------------------------------|
| AlprSDK_StartVideo | Turn on video stream callback  |
| AlprSDK_StopVideo  | Turn off video stream callback |

### 3.7 Close connection

This function processes to disconnect the Camera Devices.

| Function              | Description           |
|-----------------------|-----------------------|
| AlprSDK_DisConnectDev | Disconnect the device |

### 3.8 Release handle

This function processes to release the Device handles.

| Function             | Description     |
|----------------------|-----------------|
| AlprSDK_UnInitHandle | Release handle. |

### 3.9 Release SDK

This function processes to release from the SDK. And can only be called once

| Function        | Description                                 |
|-----------------|---------------------------------------------|
| AlprSDK_Cleanup | Close AlprSDK and release related resources |

## 3.10 Other Interface Functions

### 3.10.1 Recognition Area

The below functions initialize and set up the recognition area, and the virtual coil area.

| Function               | Description                             |
|------------------------|-----------------------------------------|
| AlprSDK_SetRoiEx       | Set the identification area (extension) |
| AlprSDK_GetRoiEx       | Get the identification area (extension) |
| AlprSDK_SetVirtualCoil | Set the virtual coil area               |
| AlprSDK_GetVirtualCoil | Get the virtual coil area               |

### 3.10.2 Device Information

| Function                  | Description                              |
|---------------------------|------------------------------------------|
| AlprSDK_GetDeviceFunList  | Get the features supported by the device |
| AlprSDK_GetDevLicenseInfo | Get device license information           |
| AlprSDK_GetDevVerInfo     | Get device version information           |
| AlprSDK_GetDevSN          | Get device encryption information        |
| AlprSDK_SetDevSN          | Set up device encryption information     |
| AlprSDK_GetDeviceParam    | Get device parameters                    |

### 3.10.3 Device Control Functions

The below functions process the Gateway control, Zoom control, Focus control, and the Device restart operations.

| Function             | Description        |
|----------------------|--------------------|
| AlprSDK_OpenGate     | Open the gate      |
| AlprSDK_ControlZoom  | Zoom control       |
| AlprSDK_ControlFocus | Focus control      |
| AlprSDK_RebootDevice | Restart the device |

### 3.10.4 Information Parameter Settings

Gate Parameter Settings:

| Function           | Description                                   |
|--------------------|-----------------------------------------------|
| AlprSDK_SetGateCfg | Set the configuration parameters of the gate. |
| AlprSDK_GetGateCfg | Obtain gate configuration parameters          |

Camera Identification Parameter Settings:

| Function           | Description                   |
|--------------------|-------------------------------|
| AlprSDK_SetAlprCfg | Set identification parameters |
| AlprSDK_GetAlprCfg | Get identification parameters |

|                         |                                          |
|-------------------------|------------------------------------------|
| AlprSDK_SetNAIprCfg     | Set identification parameters (Expand) . |
| AlprSDK_SetOfflineParam | Set Offline Parameters                   |
| AlprSDK_GetOfflineParam | Get offline parameters                   |

## Basic Parameters:

| Function             | Description          |
|----------------------|----------------------|
| AlprSDK_SetBaseParam | Set basic parameters |
| AlprSDK_GetBaseParam | Get basic parameters |

## SD Card Operation Settings:

| Function              | Description       |
|-----------------------|-------------------|
| AlprSDK_OperateSDCard | Operation SD card |

## Restore Device Parameters:

| Function             | Description               |
|----------------------|---------------------------|
| AlprSDK_ResetFactory | Restore device parameters |

## Device Information Settings:

| Function | Description |
|----------|-------------|
|----------|-------------|

|                    |                           |
|--------------------|---------------------------|
| AlprSDK_SetDevInfo | Set up device information |
| AlprSDK_GetDevInfo | Get device information    |

#### Motion Detection Settings:

| Function                 | Description                                     |
|--------------------------|-------------------------------------------------|
| AlprSDK_StartMDThrDetect | Start detection of movement detection threshold |
| AlprSDK_GetMDThreshold   | Start detection of movement detection threshold |

#### Country Code Settings:

| Function               | Description                                            |
|------------------------|--------------------------------------------------------|
| AlprSDK_SetCountryCode | Set up a country code which used in identify algorithm |

#### Image List Settings:

| Function             | Description                                                       |
|----------------------|-------------------------------------------------------------------|
| AlprSDK_GetImageList | Gets the current picture list (not supported for the time being). |

#### Device Network Parameter Settings:

| Function | Description |
|----------|-------------|
|----------|-------------|

|                         |                        |
|-------------------------|------------------------|
| AlprSDK_SetNetworkParam | Set network parameters |
| AlprSDK_GetNetworkParam | Get network parameters |

#### P2P Server Address Settings:

| Function                    | Description                |
|-----------------------------|----------------------------|
| AlprSDK_SetP2PServerAddress | Set the P2P server address |

#### OSD Control Settings:

| Function       | Description    |
|----------------|----------------|
| AlprSDK_SetOSD | Setting up OSD |
| AlprSDK_GetOSD | Get the OSD    |

#### FTP Settings:

| Function                | Description                             |
|-------------------------|-----------------------------------------|
| AlprSDK_SetFtpClientCfg | Set FtpClient configuration parameters  |
| AlprSDK_GetFtpClientCfg | Get FtpClient configuration parameters. |

#### Device Time Settings:

| Function           | Description            |
|--------------------|------------------------|
| AlprSDK_SetSysTime | Set device system time |
| AlprSDK_GetSysTime | Get device system time |

#### Video Parameter Settings:

| Function              | Description                                  |
|-----------------------|----------------------------------------------|
| AlprSDK_SetVideoParam | Set video day and night switching parameters |
| AlprSDK_GetVideoParam | Get video day and night switching parameters |

### 3.10.5 License Plate Information Management

| Function                   | Description                                                                                                |
|----------------------------|------------------------------------------------------------------------------------------------------------|
| AlprSDK_AddPlateListExt    | Additional license plate information.(if there is a duplicate license plate, it will be replaced directly) |
| AlprSDK_DelPlateListExt    | Delete license plate information                                                                           |
| AlprSDK_ExportPlateListExt | Export license plate information                                                                           |
| AlprSDK_ClearPlateListExt  | Clear white / black / fixed car list data                                                                  |

### 3.10.6 Data Delivery Management

| Function | Description |
|----------|-------------|
|----------|-------------|

|                            |                                             |
|----------------------------|---------------------------------------------|
| AlprSDK_GetDeviceDataCount | Gets the number of data bars for the device |
| AlprSDK_GetDeviceData      | Get device data                             |
| AlprSDK_SetDeviceData      | Send data to the device                     |
| AlprSDK_ClearDeviceData    | Clear a table                               |

### 3.10.7 RS485 Communication Settings

| Function                        | Description                                               |
|---------------------------------|-----------------------------------------------------------|
| AlprSDK_CommTransparentTransfer | RS485 transparent transmission                            |
| AlprSDK_SetDColorDLineSCNParam  | Set the default parameters for two-color dual-line screen |

### 3.10.8 Video Stream Settings

| Function               | Description                      |
|------------------------|----------------------------------|
| AlprSDK_GetVideoSize   | Get the video size               |
| AlprSDK_SetVideoConfig | Set up video stream information. |

### 3.10.9 Auxiliary Out and P2P Settings

| Function | Description |
|----------|-------------|
|----------|-------------|

|                             |                                                            |
|-----------------------------|------------------------------------------------------------|
| AlprSDK_OpenAuxOut          | Turn on auxiliary output                                   |
| AlprSDK_EnableP2PReconnect  | Turn on P2P reconnection (valid only for P2P connections)  |
| AlprSDK_DisableP2PReconnect | Turn off P2P reconnection (valid only for P2P connections) |

## 4 AlprSDK Functions

### 4.1 AlprSDK\_Startup

#### Function

```
public native static int AlprSDK_Startup(String path);
```

#### Description

Initialize the SDK and specify the path to the log file.

#### Parameters

| Parameter   | Description            |
|-------------|------------------------|
| String path | In: License file path. |

#### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

#### Remarks

Method parameter String path is to specify the location of the Log and self.pem files. You can specify the location other than the Android platform, but you must pass in the "" null character.

**Example: Specify location:**

```
int ret = -1;
ret = AlprSDK.AlprSDK_Startup("/storage/emulated/legacy/down/");

System.out.println("AlprSDK_Startup Ret = " + ret);
```

**Example: Does not specify a location:**

```
int ret = -1;
ret = AlprSDK.AlprSDK_Startup("");

// Note: you cannot pass in null
System.out.println("AlprSDK_Startup Ret = " + ret);
```

## 4.2 AlprSDK\_InitHandle

### Function

```
public native static int AlprSDK_InitHandle(int nHandleID);
```

### Description

Initialize window handle. However, video playback is not supported.

### Parameters

| Parameter     | Description           |
|---------------|-----------------------|
| int nHandleID | In: Number of handles |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code..

## 4.3 AlprSDK\_InitVideoHandle

### Function

```
public native static int AlprSDK_InitVideoHandle  
(  
    int nHandleID,  
    JFrame hVideoWnd  
);
```

### Description

Initialize window handle and support window video playback.

### Parameters

| Parameter                     | Description                        |
|-------------------------------|------------------------------------|
| <code>int nHandleID</code>    | In: Number of handles              |
| <code>JFrame hVideoWnd</code> | In: A window handle of type JFrame |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

#### Remarks

A handle can only be initialized once; at the same time, it can support up to 40 devices, that is, 40 screens.

can get the video on the form and play it on the JFrame form.

## 4.4 AlprSDK\_InitVideoIntHandle

#### Function

```
public native static int AlprSDK_InitVideoIntHandle
(
    int nHandleID,
    int hVideoWnd
);
```

#### Description

Initialize window handle.

#### Parameters

| Parameter     | Description           |
|---------------|-----------------------|
| int nHandleID | In: Number of handles |
| int hVideoWnd | In: An integer        |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Remarks

A handle can only be initialized once. Handle can be allocated by software, one handle is connected to a camera.

## 4.5 AlprSDK\_SearchAllCameras

### Function

```
public native static int AlprSDK_SearchAllCameras  
(  
    int nHandleID,  
    DevFindCallback devFindCb  
);
```

### Description

Search for local devices.

### Parameters

| Parameter | Description |
|-----------|-------------|
|-----------|-------------|

|                                        |                              |
|----------------------------------------|------------------------------|
| <code>int nHandleID</code>             | <b>In:</b> Number of handles |
| <code>DevFindCallback devFindCb</code> | <b>In:</b> Callback class    |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

### Remarks

You need to obtain root privileges, and this interface is temporarily not supported on the Android platform. You can only search for devices on the same network segment in the LAN.

### Example:

```
int ret = -1;
DevFindCallback devFindCb = new DevFindCallback();
ret = AlprSDK.AlprSDK_SearchAllCameras(0, devFindCb);
System.out.println("AlprSDK_SearchAllCameras ret = " + ret);
```

## 4.6 DevFindCallback

### Function

```
public class DevFindCallback {

    public short deviceType;
```

```

    public String devName;
    public String ip;
    public String macAddr;
    public int portWeb;
    public int portListen;
    public String subMask;
    public String gateway;
    public String multiAddr;
    public String dnsAddr;
    public int multiPort;
    public int channelNum;
    public int findCount;
    public long deviceID;
    public void CALLBACK() {
        System.out.println("CALLBACK");
        System.out.println("ip:" + ip);
    }
};

```

## Description

Local device search callback.

## Parameters

| Parameter               | Description                       |
|-------------------------|-----------------------------------|
| public short deviceType | Digital Video Server Machine Type |
| public String devName   | Device name                       |
| public String ip        | Device ip                         |
| public String macAddr   | Device Mac address                |
| public int portWeb      | Web port                          |
| public int portListen   | Listening port                    |

|                                      |                   |
|--------------------------------------|-------------------|
| <code>public String subMask</code>   | Submask           |
| <code>public String gateway</code>   | Gateway           |
| <code>public String multiAddr</code> | multicast address |
| <code>public String dnsAddr</code>   | DNS address       |
| <code>public int multiPort</code>    | Multicast port    |
| <code>public int channelNum</code>   | port number       |
| <code>public int findCount</code>    | Number of found   |
| <code>public long deviceID</code>    | Device id         |

## 4.7 AlprSDK\_SetConnectTimeout

### Function

```
public native static int AlprSDK_SetConnectTimeout
(
    int nHandleID,
    int nTimeout
);
```

### Description

Set connection timeout.

### Parameters

| Parameter                  | Description                          |
|----------------------------|--------------------------------------|
| <code>int nHandleID</code> | <b>In:</b> Number of Handles         |
| <code>int nTimeout</code>  | <b>In:</b> Timeout (in milliseconds) |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.8 AlprSDK\_ConnectDev

### Function

```
public native static int AlprSDK_ConnectDev
(
    int nHandleID,
    DEVINFO devInfo,
    int clientType
);
```

### Description

Connecting device.

### Parameters

| Parameter | Description |
|-----------|-------------|
|-----------|-------------|

|                              |                                    |
|------------------------------|------------------------------------|
| <code>int nHandleID</code>   | <b>In:</b> Number of Handles       |
| <code>DEVINFO devInfo</code> | <b>In:</b> Device link information |
| <code>int clientType</code>  | <b>In:</b> Connection Type         |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.9 AlprSDK\_SendHeartBeat

### Function

```
public native static int AlprSDK_SendHeartBeat(int nHandleID);
```

### Description

Send a heartbeat and trigger a device connection status callback.

### Parameters

| Parameter                  | Description                  |
|----------------------------|------------------------------|
| <code>int nHandleID</code> | <b>In:</b> Number of Handles |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Remarks:

The SDK already supports automatic heartbeat transmission. There is no need to manually call this API.

Calling this API will trigger device state and device connection state callback.

## 4.10 AlprSDK\_CreateRecogAllInfoTask

### Function

```
int AlprSDK_CreateRecogAllInfoTask
(
    int nHandleID,
    RecogAllInfoCallback callback
);
```

### Description

Create license plate recognition callback.

### Parameters

| Parameter | Description |
|-----------|-------------|
|-----------|-------------|

|                                            |                              |
|--------------------------------------------|------------------------------|
| <code>int nHandleID</code>                 | In: Number of Handles        |
| <code>RecogAllInfoCallback callback</code> | In: Callback delegate class. |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Remarks:

After calling this API, when the camera recognizes the license plate information, it will automatically call back the method in the callback object.

## 4.11 RecogAllInfoCallback

### Function

```
public class RecogAllInfoCallback {  
    private RECOG_ALL_INFO recogResult;  
    private IRecogAllInfoCallback recogAllInfo;  
    public RecogAllInfoCallback(IRecogAllInfoCallback recogAllInfo) {  
        recogResult = new RECOG_ALL_INFO();  
        this.recogAllInfo = recogAllInfo;  
    }  
    public void CALLBACK() {
```

```
if (recogAllInfo != null) {  
    recogAllInfo.recogAllInfoCallback(recogResult);  
}  
}  
};
```

### Description

License plate callback function .

### Parameters

| Parameter                                               | Description                |
|---------------------------------------------------------|----------------------------|
| <code>private RECOG_ALL_INFO recogResult</code>         | Callback class definition  |
| <code>private IRecogAllInfoCallback recogAllInfo</code> | Callback method definition |

### Remarks:

The member variable `recogResult` is used to receive the return value from the JNI callback.

### Example:

```
RecogAllInfoCallback recogAllCb = new RecogAllInfoCallback();  
ret = AlprSDK.AlprSDK_CreateRecogAllInfoTask(0, recogAllCb);  
System.out.println("AlprSDK_CreateRecogAllInfoTask ret = " + ret);
```

## 4.12 IRecogAllInfoCallback

### Function

```
public interface IRecogAllInfoCallback {
```

```
public void recogAllInfoCallback(RECOG_ALL_INFO recogAllInfo);  
};
```

### Description

License plate callback function .

## 4.13 AlprSDK\_ClearRecogAllInfoTask

### Function

```
public native static int AlprSDK_ClearRecogAllInfoTask(int nHandleID);
```

### Description

License plate callback function clean-up.

### Parameters

| Parameter     | Description       |
|---------------|-------------------|
| int nHandleID | Number of Handles |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code

## 4.14 AlprSDK\_CreateCaptureJpgCallback

### Function

```
public native static int AlprSDK_CreateCaptureJpgCallback
(
    int nHandleID,
    IDeviceCaptureCallback callback
);
```

### Description

Registered device grab callback.

### Parameters

| Parameter                      | Description                         |
|--------------------------------|-------------------------------------|
| int nHandleID                  | In: Number of Handles               |
| DeviceCaptureCallback callback | In: Capture callback delegate class |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

### Remarks:

After registering the callback using this interface, use the int AlprSDK\_CaptureJpg (int nHandleID) method to trigger the capture callback

**Example:**

How to grab a picture:

```
DeviceCaptureCallback deviceCaptureCallback = new DeviceCaptureCallback();
ret = AlprSDK.AlprSDK_CreateCaptureJpgCallback(0, deviceCaptureCallback);
System.out.println("AlprSDK_CreateCaptureJpgCallback ret = " + ret);

ret = AlprSDK.AlprSDK_CaptureJpg(0);
System.out.println("AlprSDK_CaptureJpg Ret = " + ret);
```

## 4.15 DeviceCaptureCallback

**Function**

```
public class DeviceCaptureCallback {

    private byte[] buf;
    private int len;

    private IDeviceCaptureCallback deviceCaptureCb;

    public DeviceCaptureCallback(IDeviceCaptureCallback deviceCaptureCb) {
        this.deviceCaptureCb = deviceCaptureCb;
    }

    public void CALLBACK() {
        if (deviceCaptureCb != null) {
            deviceCaptureCb.deviceCaptureCallback(buf, len);
        }
    }
}
```

```
    }  
    }  
};
```

### Description

Image data capture callback class

### Parameters

| Parameter                                                   | Description                           |
|-------------------------------------------------------------|---------------------------------------|
| <code>private byte[] buf</code>                             | Picture data                          |
| <code>private int len</code>                                | Picture data length                   |
| <code>private IDeviceCaptureCallback deviceCaptureCb</code> | Image data capture callback interface |

## 4.16 IDeviceCaptureCallback

### Function

```
public interface IDeviceCaptureCallback {  
    public void deviceCaptureCallback(byte[] buf, int len);  
};
```

### Description

Image data capture callback interface.

## 4.17 AlprSDK\_ClearCaptureJpgCallback

### Function

```
public native static int AlprSDK_ClearCaptureJpgCallback(int nHandleID);
```

### Description

Device grab callback clean-up.

### Parameters

| Parameter     | Description       |
|---------------|-------------------|
| int nHandleID | Number of Handles |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.18 AlprSDK\_CaptureJpg

## Function

```
public native static int AlprSDK_CaptureJpg(int nHandleID);
```

## Description

(You need to register a callback before calling) The device grabs the diagram, grabs the current picture directly, and returns it through a callback

## Parameters

| Parameter                  | Description           |
|----------------------------|-----------------------|
| <code>int nHandleID</code> | In: Number of Handles |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Remarks:

You need to register a callback before calling this interface.

Please use the `int AlprSDK_CreateCaptureJpgCallback(int nHandleID, DeviceCaptureCallback callback)` interface to register the callback.

## Example:

How to grab a picture:

```
DeviceCaptureCallback deviceCaptureCallback = new DeviceCaptureCallback();
ret = AlprSDK.AlprSDK_CreateCaptureJpgCallback(0, deviceCaptureCallback);
System.out.println("AlprSDK_CreateCaptureJpgCallback ret = " + ret);

ret = sdk.AlprSDK_CaptureJpg(0);
System.out.println("AlprSDK_CaptureJpg Ret = " + ret);
```

## 4.19 AlprSDK\_CreateDevStatusCallback

### Function

```
public native static int AlprSDK_CreateDevStatusCallback
(
    int nHandleID,
    DevStatusCallback callback
);
```

### Description

Registered device status callback.

### Parameters

| Parameter                  | Description           |
|----------------------------|-----------------------|
| int nHandleID              | In: Number of Handles |
| DevStatusCallback callback | In: callback function |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Remarks:

Call:

1. When the first connection is successful, the heartbeat state 0.
2. Firmware will continue to compare the current state, if the state is always the same, every 10 seconds callback, if the state is different, immediately callback.
3. After the server refuses to connect, the heartbeat state is-997.
4. After the network reconnects, the heartbeat state 0.
5. Triggers the callback after manually calling the callback.

## 4.20 DevStatusCallback

### Function

```
public class DevStatusCallback {  
    private DevStatus devStatus;  
    private IDevStatusCallback devStatusCb;  
    public DevStatusCallback(IDevStatusCallback devStatusCb) {  
        devStatus = new DevStatus();  
        this.devStatusCb = devStatusCb;  
    }  
    public void CALLBACK() {
```

```
        devStatusCb.devStatusCallback(devStatus);  
    }  
};
```

### Description

Device state callback class.

### Parameters

| Parameter                                           | Description                                         |
|-----------------------------------------------------|-----------------------------------------------------|
| <code>private DevStatus devStatus</code>            | Equipment status parameters                         |
| <code>private IDevStatusCallback devStatusCb</code> | <code>private IDevStatusCallback devStatusCb</code> |

### Remarks:

The member variable `devStatus` is used to receive the device status return value from the JNI callback.

## 4.21 IDevStatusCallback

### Function

```
public interface IDevConnStatusCallback {  
    public void devConnStatusCallback(int handleId, int status);  
}
```

### Description

State callback interface.

## 4.22 AlprSDK\_ClearDevStatusCallback

### Function

```
public native static int AlprSDK_ClearDevStatusCallback(int nHandleID);
```

### Description

Clear device status callback.

### Parameters

| Parameter     | Description       |
|---------------|-------------------|
| int nHandleID | Number of Handles |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.23 AlprSDK\_SetDevConnStatusCallBack

## Function

```
int AlprSDK_SetDevConnStatusCallBack  
(  
    int nHandleID,  
    DevConnStatusCallback callback  
);
```

## Description

Register device connection status callback.

## Parameters

| Parameter                      | Description                          |
|--------------------------------|--------------------------------------|
| int nHandleID                  | In: Number of Handles                |
| DevConnStatusCallback callback | In: Callback function delegate class |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Remarks:

Call:

1. When the first connection is successful, the state 0.
2. Firmware will continue to compare the current state, if the state is always the same, every 10 seconds

callback, if the state is different, immediately callback.

3. Callback after the server refuses to connect, state-997.
4. Callback after network reconnection, state 0

## 4.24 DevConnStatusCallback

### Function

```
public class DevConnStatusCallback {  
  
    private int handleId;  
    private int status;  
    private IDevConnStatusCallback devConnStatusCb;  
    public DevConnStatusCallback(IDevConnStatusCallback devConnStatusCb) {  
        this.devConnStatusCb = devConnStatusCb;  
    }  
    public void CALLBACK() {  
        if (devConnStatusCb != null) {  
            devConnStatusCb.devConnStatusCallback(handleId, status);  
        }  
    }  
};
```

### Description

Device Connection Status Callback Delegate Class.

**Parameters**

| Parameter                                                       | Description                                        |
|-----------------------------------------------------------------|----------------------------------------------------|
| <code>private int handleId</code>                               | The number of the handle                           |
| <code>private int status</code>                                 | Connection Status (0: Not Connected, 1: Connected) |
| <code>private IDevConnStatusCallback<br/>devConnStatusCb</code> | State callback function                            |

## 4.25 IDevConnStatusCallback

**Function**

```
public interface IDevConnStatusCallback {  
    public void devConnStatusCallback(int handleId, int status);  
}
```

**Description**

State callback interface.

## 4.26 AlprSDK\_CreateWiegandDataCallback

**Function**

```
public native static int AlprSDK_CreateWiegandDataCallback  
(  
    int nHandleID,  
    WiegandDataCallback callback  
);
```

**Description**

Create Wigan Card Data Callback.

**Parameters**

| Parameter                                 | Description                      |
|-------------------------------------------|----------------------------------|
| <code>int nHandleID</code>                | The number of the handle         |
| <code>WiegandDataCallback callback</code> | Callback function delegate class |

## 4.27 WiegandDataCallback

**Function**

```
public class WiegandDataCallback {  
  
    public WiegandData wiegandData;  
  
    public void CALLBACK() {  
        System.out.println("wiegandData:" + wiegandData.toString());  
    }  
};
```

**Description**

Wiegand card data callback.

**Parameters**

| Parameter                                   | Description     |
|---------------------------------------------|-----------------|
| <code>public WiegandData wiegandData</code> | Wigan card data |

## 4.28 AlprSDK\_ClearWiegandDataCallback

### Function

```
public native static int AlprSDK_ClearWiegandDataCallback(int nHandleID);
```

### Description

Clear the Wiegand card data callback.

### Parameters

| Parameter     | Description       |
|---------------|-------------------|
| int nHandleID | Number of Handles |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.29 AlprSDK\_CreateEZStreamDataCB

### Function

```
public native static int AlprSDK_CreateEZStreamDataCB  
(  
    int nHandleID,  
    StreamDataCallBack callback  
);
```

## Description

Register video stream callback.

## Parameters

| Parameter                   | Description           |
|-----------------------------|-----------------------|
| int nHandleID               | In: Number of Handles |
| StreamDataCallBack callback | Out: Callback class   |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Remarks:

After registering the callback using this interface, use the int AlprSDK\_StartVideo (int nHandleID); method to start the video.

Using the int AlprSDK\_StopVideo (int nHandleID); method to stop video.

## Example

How to grab a video stream:

```
StreamDataCallBack steamDateCb = new StreamDataCallBack();
ret = AlprSDK.AlprSDK_CreateEZStreamDataCB(0, steamDateCb);
System.out.println("AlprSDK_CreateEZStreamDataCB ret = " + ret);

ret = AlprSDK.AlprSDK_StartVideo(0);
System.out.println("AlprSDK_StartVideo ret = " + ret);

try {
    Thread.sleep(5000);
} catch (InterruptedException e) {
    e.printStackTrace();
}

ret = AlprSDK.AlprSDK_StopVideo(0);
System.out.println("AlprSDK_StopVideo ret = " + ret);
```

## 4.30StreamDataCallBack

### Function

```
public class StreamDataCallBack {  
    private byte[] bytes;  
    private int nBytesLen;  
    private IStreamDataCallBack streamCb;  
    public StreamDataCallBack(IStreamDataCallBack streamCb) {  
        this.streamCb = streamCb;  
        bytes = null;  
        nBytesLen = -1;  
    }  
    private void CALLBACK() {  
        if (streamCb != null) {  
            streamCb.streamDataCallback(bytes, nBytesLen);  
        }  
    }  
};
```

### Description

Video stream callback class.

### Parameters

| Parameter                           | Description                     |
|-------------------------------------|---------------------------------|
| private byte[] bytes                | Video stream data               |
| private int nBytesLen               | Data size                       |
| private IStreamDataCallBack streamC | Video stream callback interface |

## 4.31 IStreamDataCallBack

### Function

```
public interface IStreamDataCallBack {  
    public void streamDataCallback(byte[] bytes, int nBytesLen);  
}
```

### Description

Video stream callback interface.

## 4.32 AlprSDK\_ClearEZStreamDataCB

### Function

```
public native static int AlprSDK_ClearEZStreamDataCB(int nHandleID);
```

### Description

Clear stream data callback.

### Parameters

| Parameter     | Description       |
|---------------|-------------------|
| int nHandleID | Number of Handles |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.33 AlprSDK\_StartVideo

### Function

```
public native static int AlprSDK_StartVideo(int nHandleID);
```

### Description

Turn on video stream callback.

### Parameters

| Parameter     | Description       |
|---------------|-------------------|
| int nHandleID | Number of Handles |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Remarks

Before calling this function, you need to register a video stream callback, see the instructions for the `int AlprSDK_CreateEZStreamDataCB (int nHandleID, StreamDataCallBack callback)` method.

## 4.34 AlprSDK\_StopVideo

### Function

```
public native static int AlprSDK_StopVideo(int nHandleID);
```

### Description

Turn off video stream callback.

### Parameters

| Parameter                  | Description       |
|----------------------------|-------------------|
| <code>int nHandleID</code> | Number of Handles |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code

## Remarks

Before calling this function, you need to register a video stream callback, see the instructions for the `int AlprSDK_CreateEZStreamDataCB (int nHandleID, StreamDataCallBack callback)` method.

## 4.35 AlprSDK\_DisConnectDev

### Function

```
Private native static int AlprSDK_DisConnectDev(int nHandleID);
```

### Description

Disconnect the device.

### Parameters

| Parameter                  | Description       |
|----------------------------|-------------------|
| <code>int nHandleID</code> | Number of Handles |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.36 AlprSDK\_UnInitHandle

### Function

```
public native static int AlprSDK_UnInitHandle(int nHandleID);
```

### Description

Release handle.

### Parameters

| Parameter     | Description       |
|---------------|-------------------|
| int nHandleID | Number of Handles |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.37 AlprSDK\_Cleanup

## Function

```
public native static int AlprSDK_Cleanup();
```

## Description

Close AlprSDK and release related resources.

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.38 AlprSDK\_SetRoiEx

### Function

```
public native static int AlprSDK_SetRoiEx  
(  
    int nHandleID,  
    XPoint[] rgnRoi  
);
```

### Description

Set the identification area (extension).

### Parameters

| Parameter                     | Description                                                          |
|-------------------------------|----------------------------------------------------------------------|
| <code>int nHandleID;</code>   | <b>In:</b> Number of Handles                                         |
| <code>XPoint[] rgnRoi;</code> | <b>In:</b> A Quadrilateral region composed of four coordinate points |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.39 AlprSDK\_GetRoiEx

### Function

```
public native static int AlprSDK_GetRoiEx
(
    int    nHandleID,
    XPoint[] rgnRoi
);
```

### Description

Get the identification area (extension)

### Parameters

| Parameter       | Description                                                           |
|-----------------|-----------------------------------------------------------------------|
| int nHandleID   | <b>In:</b> Number of Handles                                          |
| XPoint[] rgnRoi | <b>Out:</b> A Quadrilateral region composed of four coordinate points |

### Returns

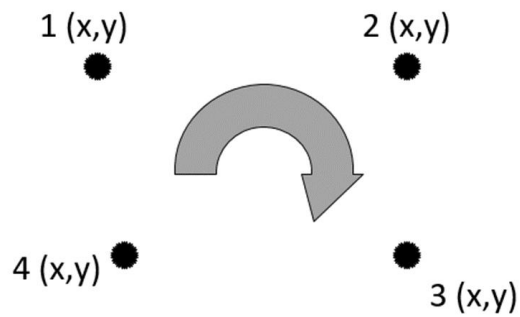
Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

**Remarks:**

The four points are ordered clockwise in the coordinate system



#### 4.40 AlprSDK\_SetVirtualCoil

**Function**

```
public native static int AlprSDK_SetVirtualCoil
(
    int    nHandleID,
    XPoint[] rgnRoi
);
```

**Description**

Set the virtual coil area.

**Parameters**

| Parameter                    | Description                                                          |
|------------------------------|----------------------------------------------------------------------|
| <code>int nHandleID</code>   | <b>In:</b> Number of Handles                                         |
| <code>XPoint[] rgnRoi</code> | <b>In:</b> A Quadrilateral region composed of four coordinate points |

### Returns

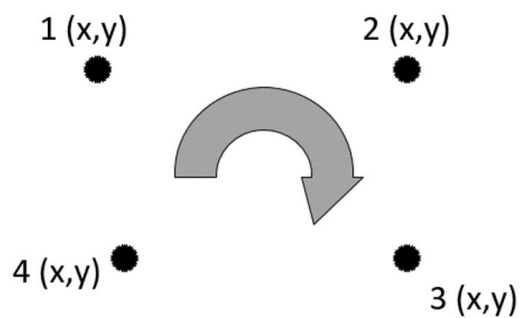
Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

### Remarks:

The four points are ordered clockwise in the coordinate system



## 4.41 AlprSDK\_GetVirtualCoil

## Function

```
public native static int AlprSDK_GetVirtualCoil  
(  
    int    nHandleID,  
    XPoint[]  rgnRoi  
);
```

## Description

Get the virtual coil area.

## Parameters

| Parameter        | Description                                                    |
|------------------|----------------------------------------------------------------|
| int nHandleID    | In: Number of Handles                                          |
| XPoint[]  rgnRoi | Out: A Quadrilateral region composed of four coordinate points |

## Returns

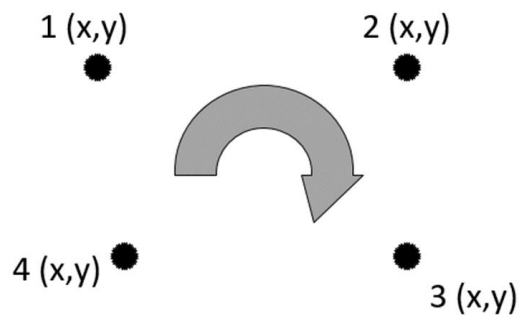
Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Remarks:

The four points are ordered clockwise in the coordinate system



## 4.42 AlprSDK\_GetDeviceFunList

### Function

```
public native static int AlprSDK_GetDeviceFunList
(
    int nHandleID,
    byte[] devFunList,
    int len
);
```

### Description

Get the features supported by the device.

### Parameters

| Parameter     | Description           |
|---------------|-----------------------|
| int nHandleID | In: Number of Handles |

|                   |                                                                                                                                                                                                                                                                                                                                                            |
|-------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| byte[] devFunList | <b>Out:</b> The upper layer allocates a buf of 256 bytes,<br>and the meaning of the value corresponding to each<br>byte: When the byte of the 0th byte is 1: the entry<br>record increases the admission information, and the<br>first byte is 1: the fleet mode, the second When the<br>byte is 1, the license plate list information is<br>expanded, etc |
| int len           | <b>In:</b> Upload fixed to 256                                                                                                                                                                                                                                                                                                                             |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Example:

```
byte[] devFunList = new byte[256];
ret = AlprSDK.AlprSDK_GetDeviceFunList(0, devFunList, devFunList.length);
System.out.println("AlprSDK_GetDeviceFunList Ret: " + ret);
for (int i = 0; i < devFunList.length; i++)
{
    System.out.print(Integer.toBinaryString(devFunList[i]));
    if (i != devFunList.length - 1)
    {
        System.out.print(",");
    }
}
System.out.println();
```

## 4.43 AlprSDK\_GetDevLicenseInfo

### Function

```
public native static int AlprSDK_GetDevLicenseInfo
(
    int      nHandleID,
    DevLicense devLicense
);
```

### Description

Get device license information.

### Parameters

| Parameter     |            | Description                             |
|---------------|------------|-----------------------------------------|
| int nHandleID |            | <b>In:</b> Number of Handles            |
| DevLicense    | devLicense | <b>Out:</b> License related information |

## 4.44 AlprSDK\_GetDevVerInfo

## Function

```
public native static int AlprSDK_GetDevVerInfo  
(  
    int    nHandleID,  
    DEV_VER_INFO devVerInfo  
);
```

## Description

Get device version information.

## Parameters

| Parameter               | Description                            |
|-------------------------|----------------------------------------|
| int    nHandleID        | <b>In:</b> Number of Handles           |
| DEV_VER_INFO devVerInfo | <b>Out:</b> Device version information |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.45 AlprSDK\_GetDevSN

## Function

```
public native static int AlprSDK_GetDevSN  
(  
    int    nHandleID,  
    DEVSN  devSN  
);
```

## Description

Get device encryption information.

## Parameters

| Parameter        | Description                        |
|------------------|------------------------------------|
| int    nHandleID | In: Number of Handles              |
| DEVSN  devSN     | Out: Device encryption information |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.46 AlprSDK\_SetDevSN

## Function

```
public native static int AlprSDK_SetDevSN  
(  
    int      nHandleID,  
    DEVSN    devSN  
);
```

## Description

Set up device encryption information.

## Parameters

| Parameter |           | Description                        |
|-----------|-----------|------------------------------------|
| int       | nHandleID | In: Number of Handles              |
| DEVSN     | devSN     | Out: Device encryption information |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.47 AlprSDK\_GetDeviceParam

## Function

```
public native static int AlprSDK_GetDeviceParam
(
    int    nHandleID,
    DeviceParam  deviceParam
);
```

## Description

Get device parameters.

## Parameters

| Parameter                | Description            |
|--------------------------|------------------------|
| int    nHandleID         | In: Number of Handles  |
| DeviceParam  deviceParam | Out: Device parameters |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.48 AlprSDK\_OpenGate

## Function

```
public native static int AlprSDK_OpenGate(int nHandleID);
```

### Description

Open the gate.

### Parameters

| Parameter     | Description           |
|---------------|-----------------------|
| int nHandleID | In: Number of Handles |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.49 AlprSDK\_ControlZoom

### Function

```
public native static int AlprSDK_ControlZoom  
(  
    int nHandleID,  
    ZoomCtlPara zoomCtlPara,
```

```
    int ifStop  
);
```

## Description

Zoom control.

## Parameters

| Parameter                            | Description                      |
|--------------------------------------|----------------------------------|
| <code>int nHandleID</code>           | In: Number of Handles            |
| <code>ZoomCtlPara zoomCtlPara</code> | In: Zoom control parameters      |
| <code>int ifStop</code>              | In: Whether to stop 0--no 1--yes |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Example

```
ZoomCtlPara zoomCtlPara = new ZoomCtlPara();  
for (int i = 0; i < 100; i++) {  
    zoomCtlPara.cmd = 0;  
    zoomCtlPara.stepSpeed = 1;  
    ret = AlprSDK.AlprSDK_ControlZoom(0, zoomCtlPara, 0);  
}
```

```
System.out.println("Emulate click down AlprSDK_ControlZoom ret = " + ret);
try {
    Thread.sleep(80);
} catch (InterruptedException e) {
    e.printStackTrace();
}
ret = AlprSDK.AlprSDK_ControlZoom(0, zoomCtlPara, 1);

try {
    Thread.sleep(80);
} catch (InterruptedException e) {
    e.printStackTrace();
}
System.out.println("Emulate click up AlprSDK_ControlZoom ret = " + ret);
}

for (int i = 0; i < 100; i++) {
    zoomCtlPara.cmd = 1;
    zoomCtlPara.stepSpeed = 1;
    ret = AlprSDK.AlprSDK_ControlZoom(0, zoomCtlPara, 0);

    System.out.println("Emulate click down AlprSDK_ControlZoom ret = " + ret);
    try {
        Thread.sleep(80);
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
    ret = AlprSDK.AlprSDK_ControlZoom(0, zoomCtlPara, 1);

    try {
        Thread.sleep(80);
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
    System.out.println("Emulate click up AlprSDK_ControlZoom ret = " + ret);
}
```

## 4.50 AlprSDK\_ControlFocus

### Function

```
public native static int AlprSDK_ControlFocus
(
    int      nHandleID,
    FocusCtlPara focusCtlPara,
    int      ifStop
);
```

### Description

Focus control.

### Parameters

| Parameter                 | Description                      |
|---------------------------|----------------------------------|
| int nHandleID             | In: Number of Handles            |
| FocusCtlPara focusCtlPara | In: Focus control parameters     |
| int ifStop                | In: Whether to stop 0--no 1--yes |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Example

Description: The following example simulates a mouse button click operation

```
FocusCtlPara focusCtlPara = new FocusCtlPara();
for (int i = 0; i < 100; i++)
{
    focusCtlPara.cmd = 0;
    focusCtlPara.stepSpeed = 1;
    ret = AlprSDK_ControlFocus(0, focusCtlPara, 0); // Simulate mouse button press

    try {
        Thread.sleep(30);
    } catch (InterruptedException e) {
        e.printStackTrace();
    }

    ret = AlprSDK_ControlFocus(0, focusCtlPara, 1); // Simulate mouse button
release
    System.out.println("AlprSDK_ControlFocus ret = " + ret);
}

for (int i = 0; i < 100; i++)
{
    focusCtlPara.cmd = 1;
    focusCtlPara.stepSpeed = 1;
    ret = AlprSDK_ControlFocus(0, focusCtlPara, 0); // Simulate mouse button press
    try {
        Thread.sleep(30);
    } catch (InterruptedException e) {
        e.printStackTrace();
    }

    ret = AlprSDK_ControlFocus(0, focusCtlPara, 1); // Simulate mouse button
release
    System.out.println("AlprSDK_ControlFocus ret = " + ret);
}.
```

## 4.51 AlprSDK\_RebootDevice

### Function

```
public native static int AlprSDK_RebootDevice(int nHandleID);
```

### Description

Restart the device.

### Parameters

| Parameter     | Description           |
|---------------|-----------------------|
| int nHandleID | In: Number of Handles |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.52 AlprSDK\_SetGateCfg

### Function

```
public native static int AlprSDK_SetGateCfg
(
    int      nHandleID,
    GATE_CONFIG gateCfg
);
```

## Description

Set the configuration parameters of the gate.

## Parameters

| Parameter           | Description                          |
|---------------------|--------------------------------------|
| int nHandleID       | In: Number of Handles                |
| GATE_CONFIG gateCfg | In: Configuration parameters of gate |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.53 AlprSDK\_GetGateCfg

## Function

```
public native static int AlprSDK_GetGateCfg
```

```
(  
    int nHandleID,  
    GATE_CONFIG gateCfg  
);
```

## Description

Obtain gate configuration parameters.

## Parameters

| Parameter                        | Description                          |
|----------------------------------|--------------------------------------|
| <code>int nHandleID</code>       | In: Number of Handles                |
| <code>GATE_CONFIG gateCfg</code> | In: Configuration parameters of gate |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.54 AlprSDK\_SetAlprCfg

## Function

```
public native static int AlprSDK_SetAlprCfg
(
    int      nHandleID,
    ALPR_CONFIG  alprCfg
);
```

## Description

Set identification parameters.

## Parameters

| Parameter            | Description                  |
|----------------------|------------------------------|
| int      nHandleID   | In: Number of Handles        |
| ALPR_CONFIG  alprCfg | In: Identification parameter |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.55 AlprSDK\_GetAlprCfg

## Function

```
public native static int AlprSDK_GetAlprCfg  
(  
    int      nHandleID,  
    ALPR_CONFIG alprCfg  
);
```

## Description

Get identification parameters.

## Parameters

| Parameter           | Description                  |
|---------------------|------------------------------|
| int      nHandleID  | In: Number of Handles        |
| ALPR_CONFIG alprCfg | In: Identification parameter |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.56 AlprSDK\_SetNAlprCfg

## Function

```
public native static int AlprSDK_SetNAlprCfg  
(  
    int    nHandleID,  
    NAlprCfg nAlprCfg  
);
```

## Description

Set identification parameters(Expand). Not available for Chinese version firmware.

## Parameters

| Parameter         | Description                  |
|-------------------|------------------------------|
| int    nHandleID  | In: Number of Handles        |
| NAlprCfg  alprCfg | In: Identification parameter |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Remarks

Support for European firmware only.

## 4.57 AlprSDK\_GetNAlprCfg

### Function

```
public native static int AlprSDK_GetNAlprCfg
(
    int    nHandleID,
    NAlprCfg  nAlprCfg
);
```

### Description

Get identification parameters.

### Parameters

| Parameter         | Description                  |
|-------------------|------------------------------|
| int    nHandleID  | In: Number of Handles        |
| NAlprCfg  alprCfg | In: Identification parameter |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

### Remarks

Support for European firmware only.

## 4.58 AlprSDK\_SetOfflineParam

### Function

```
public native static int AlprSDK_SetOfflineParam
(
    int    nHandleID,
    OfflineParam OfflineParam
);
```

### Description

Set Offline Parameters.

### Parameters

| Parameter                 | Description                  |
|---------------------------|------------------------------|
| int    nHandleID          | In: Number of Handles        |
| OfflineParam OfflineParam | In: Offline charge parameter |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.59 AlprSDK\_GetOfflineParam

### Function

```
public native static int AlprSDK_GetOfflineParam
(
    int    nHandleID,
    OfflineParam OfflineParam
);
```

### Description

Get offline parameters.

### Parameters

| Parameter                 | Description                  |
|---------------------------|------------------------------|
| int    nHandleID          | In: Number of Handles        |
| OfflineParam OfflineParam | In: Offline charge parameter |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.60 AlprSDK\_SetBaseParam

### Function

```
int AlprSDK_SetBaseParam
(
    int      nHandleID,
    BASE_PARAM baseParam
);
```

### Description

Set basic parameters.

### Parameters

| Parameter            | Description                                                                           |
|----------------------|---------------------------------------------------------------------------------------|
| int      nHandleID   | In: Number of Handles                                                                 |
| BASE_PARAM baseParam | In: Basic parameters (Whether the identification area and virtual coil are displayed) |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Structure

COUNTRY\_CODE

## Parameters

| Parameter | Description  |
|-----------|--------------|
| CA = 1    | Canada       |
| US = 1    | US           |
| KZ = 7    | Kazakhstan   |
| RU = 7    | Russia       |
| EG = 20   | Egypt        |
| ZA = 27   | South Africa |
| GR = 30   | Greece       |
| NL = 31   | Netherlands  |
| BE = 32   | Belgium      |
| FR = 33   | France       |
| ES = 34   | Spain        |
| IT = 39   | Italy        |
| UK = 44   | UK           |
| RO = 40   | Romania      |
| CH = 41   | Switzerland  |
| AT = 43   | Austria      |
| DK = 45   | Denmark      |

|         |             |
|---------|-------------|
| SE = 46 | Sweden      |
| NO = 47 | Norway      |
| PL = 48 | Poland      |
| PE = 51 | Peru        |
| MX = 52 | Mexico      |
| CU = 53 | Cuba        |
| AR = 54 | Argentina   |
| BR = 55 | Brazil      |
| CL = 56 | Chile       |
| CO = 57 | Colombia    |
| VE = 58 | Venezuela   |
| MY = 60 | Malaysia    |
| AU = 61 | Australia   |
| ID = 62 | Indonesia   |
| PH = 63 | Philippines |
| NZ = 64 | New Zealand |
| SG = 65 | Singapore   |
| TH = 66 | Thailand    |
| JP = 81 | Japan       |
| KR = 82 | Korea       |
| VN = 84 | Vietnam     |
| CN = 86 | China       |

|          |                |
|----------|----------------|
| TR = 90  | Turkey         |
| IND = 91 | India          |
| PK = 92  | Pakistan       |
| AF = 93  | Afghanistan    |
| LK = 94  | Sri Lanka      |
| MM = 95  | Myanmar        |
| IR = 98  | Iran           |
| MA = 210 | Morocco        |
| EH = 210 | Western Sahara |
| DZ = 213 | Algeria        |
| TN = 216 | Tunisia        |
| LY = 218 | Libya          |
| GM = 220 | Gambia         |
| SN = 221 | Senegal        |
| MR = 222 | Mauritania     |
| ML = 223 | Mali           |
| GN = 224 | Guinea         |
| CI = 225 | Côte d'Ivoire  |
| BF = 226 | Burkina Faso   |
| NE = 227 | Niger          |
| TG = 228 | Togo           |
| BJ = 229 | Benin          |

|          |                                     |
|----------|-------------------------------------|
| MU = 230 | Mauritius                           |
| LR = 231 | Liberia                             |
| SL = 232 | Sierra Leone                        |
| GH = 233 | Ghana                               |
| NG = 234 | Nigeria                             |
| TD = 235 | Chad                                |
| CF = 236 | Central African Republic            |
| CM = 237 | Cameroon                            |
| CV = 238 | Cape Verde                          |
| ST = 239 | Sao Tome and Principe               |
| GQ = 240 | Equatorial Guinea                   |
| GA = 241 | Gabon                               |
| CD = 242 | Congo                               |
| CG = 242 | Democratic Republic of the<br>Congo |
| AO = 244 | Angola                              |
| GW = 245 | Guinea-Bissau                       |
| SD = 249 | Sudan                               |
| RW = 250 | Rwanda                              |
| ET = 251 | Ethiopia                            |
| SO = 252 | Somalia                             |
| DJ = 253 | Kyrgyzstan                          |

|          |            |
|----------|------------|
| KE = 254 | Kenya      |
| TZ = 255 | Tanzania   |
| UG = 256 | Uganda     |
| BI = 257 | Burundi    |
| MZ = 258 | Mozambique |
| ZM = 260 | Zambia     |
| MG = 261 | Madagascar |
| RE = 262 | Reunion    |
| ZW = 263 | Zimbabwe   |
| NA = 264 | Namibia    |
| MW = 265 | Malawi     |
| LS = 266 | Lesotho    |
| BW = 267 | Botswana   |
| SZ = 268 | Swaziland  |
| KM = 269 | Comoros    |
| YT = 269 | Mayotte    |
| ER = 291 | Eritrea    |
| AW = 297 | Aruba      |
| HU = 336 | Hungary    |
| DE = 349 | Germany    |
| PT = 351 | Portugal   |
| LU = 352 | Luxembourg |

|           |                        |
|-----------|------------------------|
| IE = 353  | Ireland                |
| IS = 354  | Iceland                |
| AL = 355  | Albania                |
| MT = 356  | Malta                  |
| CYP = 357 | Cyprus                 |
| FI = 358  | Finland                |
| BG = 359  | Bulgaria               |
| LT = 370  | Lithuania              |
| LV = 371  | Latvia                 |
| EE = 372  | Estonia                |
| MD = 373  | Moldova                |
| AM = 374  | Armenia                |
| BY = 375  | Belarus                |
| AD = 376  | Andorra                |
| UA = 380  | Ukraine                |
| RS = 381  | Serbia                 |
| ME = 382  | Montenegro             |
| HR = 385  | Croatia                |
| SI = 386  | Slovenia               |
| BA = 387  | Bosnia and Herzegovina |
| MK = 389  | Macedonia              |
| VA = 396  | Vatican                |

|          |                      |
|----------|----------------------|
| CZ = 420 | Czech Republic       |
| SK = 421 | Slovakia             |
| BZ = 501 | Belize               |
| GT = 502 | Guatemala            |
| SV = 503 | El Salvador          |
| HN = 504 | Honduras             |
| NI = 505 | Nicaragua            |
| CR = 506 | Costa Rica           |
| PA = 507 | Panama               |
| HT = 509 | Haiti                |
| GP = 590 | Guadeloupe           |
| BO = 591 | Bolivia              |
| GY = 592 | Guyana               |
| EC = 593 | Ecuador              |
| GF = 594 | French Guiana        |
| PY = 595 | Paraguay             |
| MQ = 596 | Martinique           |
| SR = 597 | Suriname             |
| UY = 598 | Uruguay              |
| AN = 599 | Netherlands Antilles |
| GU = 671 | Guam                 |
| TL = 670 | East Timor           |

|          |                   |
|----------|-------------------|
| BN = 673 | Brunei Darussalam |
| NR = 674 | Nauru             |
| PG = 675 | Papua New Guinea  |
| TO = 676 | Tonga             |
| SB = 677 | Solomon Islands   |
| VU = 678 | Vanuatu           |
| FJ = 679 | Fiji              |
| CK = 682 | Cook Islands      |
| WS = 685 | Samoa             |
| KI = 686 | Kiribati          |
| NC = 687 | New Caledonia     |
| TV = 688 | Tuvalu            |
| PF = 689 | French Polynesia  |
| FM = 691 | Micronesia        |
| MH = 692 | Marshall Islands  |
| KP = 850 | North Korea       |
| HK = 852 | Hong Kong         |
| MO = 853 | Macau             |
| KH = 855 | Cambodia          |
| LA = 856 | Laos              |
| BD = 880 | Bangladesh        |
| TW = 886 | Taiwan            |

|          |                      |
|----------|----------------------|
| MV = 961 | Maldives             |
| LB = 962 | Lebanon              |
| JO = 963 | Jordan               |
| SY = 964 | Syria                |
| IQ = 965 | Iraq                 |
| KW = 966 | Kuwait               |
| SA = 967 | Saudi Arabia         |
| YE = 968 | Yemen                |
| OM = 969 | Oman                 |
| PS = 970 | Palestine            |
| AE = 972 | United Arab Emirates |
| IL = 973 | Israel               |
| BH = 974 | Bahrain              |
| QA = 975 | Qatar                |
| BT = 976 | Bhutan               |
| MN = 977 | Mongolia             |
| NP = 978 | Nepal                |
| TJ = 992 | Tajikistan           |
| TM = 993 | Turkmenistan         |
| AZ = 994 | Azerbaijan           |
| GE = 995 | Georgia              |
| KG = 996 | Djibouti             |

|           |                        |
|-----------|------------------------|
| UZ = 998  | Uzbekistan             |
| BB = 1809 | Barbados               |
| BS = 1809 | Bahamas                |
| PR = 1809 | Puerto Rico            |
| DO = 1809 | Dominican Republic     |
| GD = 1809 | Grenada                |
| LC = 1809 | Saint Lucia            |
| VC = 1809 | Saint Vincent          |
| TT = 1809 | Trinidad and Tobago    |
| JM = 1809 | Jamaica                |
| VG = 1809 | British Virgin Islands |

## 4.61 AlprSDK\_GetBaseParam

### Function

```
public native static int AlprSDK_GetBaseParam
(
    int      nHandleID,
    BASE_PARAM baseParam
);
```

## Description

Get basic parameters.

## Parameters

| Parameter                 | Description                                                                           |
|---------------------------|---------------------------------------------------------------------------------------|
| int      nHandleID        | In: Number of Handles                                                                 |
| BASE_PARAM      baseParam | In: Basic parameters (Whether the identification area and virtual coil are displayed) |

## Returns

Successful:    It returns 0.

Unsuccessful:    Returns non-zero Error Code.

Error Code:    See [Appendix 1](#) for Error Code.

## 4.62 AlprSDK\_OperateSDCard

### Function

```
public native static int AlprSDK_OperateSDCard
(
    int      nHandleID,
    int      nOpType,
```

```
    SDCardCapacity    sdCardCapacity  
);
```

## Description

Operation SD card.

## Parameters

| Parameter                         | Description                                                                     |
|-----------------------------------|---------------------------------------------------------------------------------|
| int     nHandleID                 | In: Number of Handles                                                           |
| int     nOpType                   | In: Mount SD card, 1--unload SD card, 2-format SD card, 3--get SD card capacity |
| SDCardCapacity     sdCardCapacity | In: SD card capacity information                                                |

## Returns

Successful:     It returns 0.

Unsuccessful:     Returns non-zero Error Code.

Error Code:     See [Appendix 1](#) for Error Code.

## 4.63 AlprSDK\_ResetFactory

## Function

```
public native static int AlprSDK_ResetFactory
(
    int nHandleID,
    int resetType
);
```

## Description

Restore device parameters.

## Parameters

| Parameter     | Description                                                                                                                          |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------|
| int nHandleID | In: Number of Handles                                                                                                                |
| int resetType | In: Restart Type.<br><br>0: Recovery Technical Parameters,<br><br>1: Recovery Network Parameters,<br><br>2: Recovery All Parameters. |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.64 AlprSDK\_SetDevInfo

### Function

```
public native static int AlprSDK_SetDevInfo
(
    int      nHandleID,
    DEVINFO  devInfo
);
```

### Description

Set up device information.

### Parameters

| Parameter       | Description        |
|-----------------|--------------------|
| int nHandleID   | Number of Handles  |
| DEVINFO devInfo | Device information |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.65 AlprSDK\_GetDevInfo

### Function

```
public native static int AlprSDK_GetDevInfo  
(  
    int    nHandleID,  
    DEVINFO devInfo  
);
```

### Description

Get device information.

### Parameters

| Parameter        | Description             |
|------------------|-------------------------|
| int    nHandleID | In: Number of Handles   |
| DEVINFO devInfo  | out: Device Information |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.66 AlprSDK\_StartMDThrDetect

### Function

```
public native static int AlprSDK_StartMDThrDetect(int nHandleID);
```

### Description

Start detection of movement detection threshold.

### Parameters

| Parameter     | Description            |
|---------------|------------------------|
| int nHandleID | In: Number of Handles. |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.67 AlprSDK\_GetMDThreshold

### Function

```
public native static int AlprSDK_GetMDThreshold(int nHandleID);
```

### Description

Start detection of movement detection threshold.

### Parameters

| Parameter     | Description            |
|---------------|------------------------|
| int nHandleID | In: Number of Handles. |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.68 AlprSDK\_GetNetworkParam

### Function

```
public native static int AlprSDK_GetNetworkParam  
(  
    int nHandleID,
```

```
    NetworkParam networkParam  
);
```

## Description

Get network parameters.

## Parameters

| Parameter                 | Description            |
|---------------------------|------------------------|
| int nHandleID             | In: Number of Handles. |
| NetworkParam networkParam | In: Network parameter  |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.69 AlprSDK\_SetCountryCode

## Function

```
public native static int AlprSDK_SetCountryCode
(
    int nHandleID,
    int nCountryCode
);
```

## Description

Set up a country code which used in identify algorithm.

## Parameters

| Parameter                     | Description                                            |
|-------------------------------|--------------------------------------------------------|
| <code>int nHandleID</code>    | In: Number of Handles.                                 |
| <code>int nCountryCode</code> | In: Country code, see the following table COUNTRY_CODE |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Remarks

It is not recommended and is not valid in most cases. If you want to switch countries, use the upgrade package tool.

## Structure

COUNTRY\_CODE

## Description

Country Code

## Parameters

| Parameter | Description  |
|-----------|--------------|
| CA=1      | Canada       |
| US=1      | US           |
| KZ=7      | Kazakhstan   |
| RU=7      | Russia       |
| EG=20     | Egypt        |
| ZA=27     | South Africa |
| GR=30     | Greece       |
| NL=31     | Netherlands  |
| BE=32     | Belgium      |
| FR=33     | France       |
| ES=34     | Spain        |
| IT=39     | Italy        |
| UK=44     | UK           |

|       |             |
|-------|-------------|
| RO=40 | Romania     |
| CH=41 | Switzerland |
| AT=43 | Austria     |
| DK=45 | Denmark     |
| SE=46 | Sweden      |
| NO=47 | Norway      |
| PL=48 | Poland      |
| PE=51 | Peru        |
| MX=52 | Mexico      |
| CU=53 | Cuba        |
| AR=54 | Argentina   |
| BR=55 | Brazil      |
| CL=56 | Chile       |
| CO=57 | Colombia    |
| VE=58 | Venezuela   |
| MY=60 | Malaysia    |
| AU=61 | Australia   |
| ID=62 | Indonesia   |
| PH=63 | Philippines |
| NZ=64 | New Zealand |
| SG=65 | Singapore   |
| TH=66 | Thailand    |

|        |                |
|--------|----------------|
| JP=81  | Japan          |
| KR=82  | Korea          |
| VN=84  | Vietnam        |
| CN=86  | China          |
| TR=90  | Turkey         |
| IND=91 | India          |
| PK=92  | Pakistan       |
| AF=93  | Afghanistan    |
| LK=94  | Sri Lanka      |
| MM=95  | Myanmar        |
| IR=98  | Iran           |
| MA=210 | Morocco        |
| EH=210 | Western Sahara |
| DZ=213 | Algeria        |
| TN=216 | Tunisia        |
| LY=218 | Libya          |
| GM=220 | Gambia         |
| SN=221 | Senegal        |
| MR=222 | Mauritania     |
| ML=223 | Mali           |
| GN=224 | Guinea         |
| CI=225 | Côte d'Ivoire  |

|        |                                  |
|--------|----------------------------------|
| BF=226 | Burkina Faso                     |
| NE=227 | Niger                            |
| TG=228 | Togo                             |
| BJ=229 | Benin                            |
| MU=230 | Mauritius                        |
| LR=231 | Liberia                          |
| SL=232 | Sierra Leone                     |
| GH=233 | Ghana                            |
| NG=234 | Nigeria                          |
| TD=235 | Chad                             |
| CF=236 | Central African Republic         |
| CM=237 | Cameroon                         |
| CV=238 | Cape Verde                       |
| ST=239 | Sao Tome and Principe            |
| GQ=240 | Equatorial Guinea                |
| GA=241 | Gabon                            |
| CD=242 | Congo                            |
| CG=242 | Democratic Republic of the Congo |
| AO=244 | Angola                           |
| GW=245 | Guinea-Bissau                    |
| SD=249 | Sudan                            |
| RW=250 | Rwanda                           |

|        |            |
|--------|------------|
| ET=251 | Ethiopia   |
| SO=252 | Somalia    |
| DJ=253 | Kyrgyzstan |
| KE=254 | Kenya      |
| TZ=255 | Tanzania   |
| UG=256 | Uganda     |
| BI=257 | Burundi    |
| MZ=258 | Mozambique |
| ZM=260 | Zambia     |
| MG=261 | Madagascar |
| RE=262 | Reunion    |
| ZW=263 | Zimbabwe   |
| NA=264 | Namibia    |
| MW=265 | Malawi     |
| LS=266 | Lesotho    |
| BW=267 | Botswana   |
| SZ=268 | Swaziland  |
| KM=269 | Comoros    |
| YT=269 | Mayotte    |
| ER=291 | Eritrea    |
| AW=297 | Aruba      |
| HU=336 | Hungary    |

|         |            |
|---------|------------|
| DE=349  | Germany    |
| PT=351  | Portugal   |
| LU=352  | Luxembourg |
| IE=353  | Ireland    |
| IS=354  | Iceland    |
| AL=355  | Albania    |
| MT=356  | Malta      |
| CYP=357 | Cyprus     |
| FI=358  | Finland    |
| BG=359  | Bulgaria   |
| LT=370  | Lithuania  |
| LV=371  | Latvia     |
| EE=372  | Estonia    |
| MD=373  | Moldova    |
| AM=374  | Armenia    |
| BY=375  | Belarus    |
| AD=376  | Andorra    |
| UA=380  | Ukraine    |
| RS=381  | Serbia     |
| ME=382  | Montenegro |
| HR=385  | Croatia    |
| SI=386  | Slovenia   |

|        |                        |
|--------|------------------------|
| BA=387 | Bosnia and Herzegovina |
| MK=389 | Macedonia              |
| VA=396 | Vatican                |
| CZ=420 | Czech Republic         |
| SK=421 | Slovakia               |
| BZ=501 | Belize                 |
| GT=502 | Guatemala              |
| SV=503 | El Salvador            |
| HN=504 | Honduras               |
| NI=505 | Nicaragua              |
| CR=506 | Costa Rica             |
| PA=507 | Panama                 |
| HT=509 | Haiti                  |
| GP=590 | Guadeloupe             |
| BO=591 | Bolivia                |
| GY=592 | Guyana                 |
| EC=593 | Ecuador                |
| GF=594 | French Guiana          |
| PY=595 | Paraguay               |
| MQ=596 | Martinique             |
| SR=597 | Suriname               |
| UY=598 | Uruguay                |

|        |                      |
|--------|----------------------|
| AN=599 | Netherlands Antilles |
| GU=671 | Guam                 |
| TL=670 | East Timor           |
| BN=673 | Brunei Darussalam    |
| NR=674 | Nauru                |
| PG=675 | Papua New Guinea     |
| TO=676 | Tonga                |
| SB=677 | Solomon Islands      |
| VU=678 | Vanuatu              |
| FJ=679 | Fiji                 |
| CK=682 | Cook Islands         |
| WS=685 | Samoa                |
| KI=686 | Kiribati             |
| NC=687 | New Caledonia        |
| TV=688 | Tuvalu               |
| PF=689 | French Polynesia     |
| FM=691 | Micronesia           |
| MH=692 | Marshall Islands     |
| KP=850 | North Korea          |
| HK=852 | Hong Kong            |
| MO=853 | Macau                |
| KH=855 | Cambodia             |

|        |                      |
|--------|----------------------|
| LA=856 | Laos                 |
| BD=880 | Bangladesh           |
| TW=886 | Taiwan               |
| MV=961 | Maldives             |
| LB=962 | Lebanon              |
| JO=963 | Jordan               |
| SY=964 | Syria                |
| IQ=965 | Iraq                 |
| KW=966 | Kuwait               |
| SA=967 | Saudi Arabia         |
| YE=968 | Yemen                |
| OM=969 | Oman                 |
| PS=970 | Palestine            |
| AE=972 | United Arab Emirates |
| IL=973 | Israel               |
| BH=974 | Bahrain              |
| QA=975 | Qatar                |
| BT=976 | Bhutan               |
| MN=977 | Mongolia             |
| NP=978 | Nepal                |
| TJ=992 | Tajikistan           |
| TM=993 | Turkmenistan         |

|         |                        |
|---------|------------------------|
| AZ=994  | Azerbaijan             |
| GE=995  | Georgia                |
| KG=996  | Djibouti               |
| UZ=998  | Uzbekistan             |
| BB=1809 | Barbados               |
| BS=1809 | Bahamas                |
| PR=1809 | Puerto Rico            |
| DO=1809 | Dominican Republic     |
| GD=1809 | Grenada                |
| LC=1809 | Saint Lucia            |
| VC=1809 | Saint Vincent          |
| TT=1809 | Trinidad and Tobago    |
| JM=1809 | Jamaica                |
| VG=1809 | British Virgin Islands |

## 4.70 AlprSDK\_GetImageList

### Function

```
public native static int AlprSDK_GetImageList(int nHandleID);
```

### Description

Gets the current picture list (not supported for the time being).

#### Parameters

| Parameter     | Description            |
|---------------|------------------------|
| int nHandleID | In: Number of Handles. |

#### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.71 AlprSDK\_SetNetworkParam

#### Function

```
public native static int AlprSDK_SetNetworkParam
(
    int nHandleID,
    NetworkParam networkParam
);
```

## Description

Set network parameters.

## Parameters

| Parameter                              | Description            |
|----------------------------------------|------------------------|
| <code>int nHandleID</code>             | In: Number of Handles. |
| <code>NetworkParam networkParam</code> | In: Network parameter  |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Function

`NetworkParam`

## Description

Network parameter.

## Parameters

| Parameter | Description |
|-----------|-------------|
|-----------|-------------|

|                          |                                                                                                                  |
|--------------------------|------------------------------------------------------------------------------------------------------------------|
| Int NetInterface:        | Network interface 1-10MBase-T; 2-10Base-T full duplex;<br>3-100Base-TX; 4-100M full duplex; 5-10M/100M adaptive. |
| String DVRIP:            | IP address                                                                                                       |
| String DVRIPMask:        | Subnetmask                                                                                                       |
| String GatewayIpAddr:    | Gateway address                                                                                                  |
| String MACAddr:          | Physical address                                                                                                 |
| Short DvrPort:           | Communication port, default 8000                                                                                 |
| Short HttpPortNo:        | Http port                                                                                                        |
| Short HttpsPort:         | Https port                                                                                                       |
| Short RtspPort:          | Rtsp port                                                                                                        |
| Short RTMPPort:          | Rtmp port                                                                                                        |
| Short MulticastPort:     | Multicast port                                                                                                   |
| Int AlarmHostIpPort:     | Alarm host port                                                                                                  |
| String MulticastIpAddr:  | Multicast address                                                                                                |
| String DnsServer1IpAddr: | IP address of domain name server 1.                                                                              |
| String DnsServer2IpAddr: | IP address of domain name server 2.                                                                              |
| Int DefaultRoute:        | Default route, 0-struEtherNet[0], 1-struEtherNet[1]                                                              |
| Int NetworkCardNum:      | Number of NICs that the device can actually configure.                                                           |
| String AlarmHostIpAddr:  | Alarm host IP address                                                                                            |
| String SnmpHostIp:       | Self-trapped host IP address description, supporting IPv4 IPv6 and domain name description                       |
| Int SnmpCount:           | The number of transmissions                                                                                      |

|                       |                         |
|-----------------------|-------------------------|
| Int SnmpInterval:     | Send time interval      |
| Boolean EnableUPNP:   | Whether to enable UPNP  |
| Boolean EnableSnmp:   | Whether to enable Snmp  |
| Boolean UseDhcp:      | Whether to enable Dhcp  |
| Boolean EnablePPPOE:  | Whether to enable PPPoE |
| String PPPoEUser:     | PPPoE username          |
| String PPPoEIP:       | PPPoE IP address        |
| String PPPoEPassword: | PPPoE password          |

## 4.72 AlprSDK\_SetP2PServerAddress

### Function

```
public native static int AlprSDK_SetP2PServerAddress(
    String strP2PServerAddress;
);
```

### Description

Set the P2P server address.

### Parameters

| Parameter | Description |
|-----------|-------------|
|-----------|-------------|

|                                  |                                                     |
|----------------------------------|-----------------------------------------------------|
| <code>strP2PServerAddress</code> | <b>In:</b> The address, which can be a domain name. |
|----------------------------------|-----------------------------------------------------|

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.73 AlprSDK\_SetOSD

### Function

```
public native static int AlprSDK_SetOSD
(
    int    nHandleID,
    OSD_CTRL osdCtrl
);
```

### Description

Setting up OSD.

### Parameters

| Parameter | Description |
|-----------|-------------|
|-----------|-------------|

|                               |                                                     |
|-------------------------------|-----------------------------------------------------|
| <code>int nHandleID</code>    | <b>In:</b> Number of Handles.                       |
| <code>OSD_CTRL osdCtrl</code> | <b>In:</b> The address, which can be a domain name. |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Example

```
OSD_CTRL modifiedOsdCtrl = (OSD_CTRL) restoreOsdCtrl.clone();
modifiedOsdCtrl.stTitleOSD.bShow = 1;
modifiedOsdCtrl.stTitleOSD.x = 48;
modifiedOsdCtrl.stTitleOSD.y = 48;
modifiedOsdCtrl.strTitle = "ZKTco123";
System.out.println("Set data: " + modifiedOsdCtrl);
ret = AlprSDK.AlprSDK_SetOSD(0, modifiedOsdCtrl);
```

## Structure

`OSD_CTRL`

## Description

Video control: the perceptual device only supports setting the title `strTitle` and whether to display the title.

## Parameters

| Parameter                            | Description                  |
|--------------------------------------|------------------------------|
| <code>OSD_PARAM stDateOSD:</code>    | Video date display location  |
| <code>OSD_PARAM stTimeOSD:</code>    | Video time display position  |
| <code>OSD_PARAM stWeekOSD:</code>    | Video week display position  |
| <code>OSD_PARAM stBitrateOSD:</code> | Video resolution display bit |
| <code>OSD_PARAM stTitleOSD:</code>   | Video title display position |
| <code>String strTitle:</code>        | video title                  |

## 4.74 AlprSDK\_GetOSD

### Function

```
public native static int AlprSDK_GetOSD
(
    int    nHandleID,
    OSD_CTRL osdCtrl
);
```

### Description

Get the OSD.

#### Parameters

| Parameter                     | Description             |
|-------------------------------|-------------------------|
| int nHandleID                 | In: Number of Handles   |
| <code>OSD_CTRL</code> osdCtrl | Out: Video Control Mode |

#### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.75 AlprSDK\_SetFtpClientCfg

#### Function

```
public native static int AlprSDK_SetFtpClientCfg
(
    int nHandleID,
    FtpClientCfg ftpClientCfg
);
```

## Description

Set FtpClient configuration parameters.

## Parameters

| Parameter                              | Description                 |
|----------------------------------------|-----------------------------|
| int nHandleID                          | In: Number of Handles       |
| <code>FtpClientCfg</code> ftpClientCfg | In: Configuration parameter |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.76 AlprSDK\_GetFtpClientCfg

### Function

```
public native static int AlprSDK_GetFtpClientCfg
(
    int nHandleID,
    FtpClientCfg ftpClientCfg
);
```

## Description

Get FtpClient configuration parameters.

## Parameters

| Parameter                              | Description                 |
|----------------------------------------|-----------------------------|
| int nHandleID                          | In: Number of Handles       |
| <code>FtpClientCfg</code> ftpClientCfg | In: Configuration parameter |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Structure

`FtpClientCfg`

## Description

FtpClient configuration parameters.

## Parameters

| Parameter          | Description              |
|--------------------|--------------------------|
| boolean enable:    | Whether Ftp is enabled   |
| boolean pasvMode:  | Remote port              |
| short remotePort:  | Passive mode connection. |
| String remoteHost: | Remote IP                |
| String userName:   | Name of user             |
| String passWord:   | User Password            |
| String uploadPath: | Upload path              |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See Appendix 1 for Error Code.

## 4.77 AlprSDK\_SetSysTime

### Function

```
public native static int AlprSDK_SetSysTime
```

```
(  
    int    nHandleID,  
    Time   time  
);
```

## Description

Set device system time.

## Parameters

| Parameter        | Description           |
|------------------|-----------------------|
| int    nHandleID | In: Number of Handles |
| Time   time      | In: Time              |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Structure

Time

## Description

Time.

## Parameters

| Parameter     | Description                    |
|---------------|--------------------------------|
| short year:   | Year (current year minus 2000) |
| short month:  | Month                          |
| short day:    | Day                            |
| short hour:   | Hour                           |
| short minute: | Minute                         |
| short second: | Second                         |

## 4.78 AlprSDK\_GetSysTime

### Function

```
public native static int AlprSDK_GetSysTime
(
    int    nHandleID,
    Time    time
);
```

### Description

Get device system time.

## Parameters

| Parameter         | Description           |
|-------------------|-----------------------|
| int     nHandleID | In: Number of Handles |
| Time    time      | In: Time              |

## Returns

Successful:    It returns 0.

Unsuccessful:    Returns non-zero Error Code.

Error Code:    See [Appendix 1](#) for Error Code.

## Structure

Time

## Description

Time.

## Parameters

| Parameter    | Description                    |
|--------------|--------------------------------|
| short year:  | Year (current year minus 2000) |
| short month: | Month                          |
| short day:   | Day                            |

|               |        |
|---------------|--------|
| short hour:   | Hour   |
| short minute: | Minute |
| short second: | Second |

## 4.79 AlprSDK\_SetVideoParam

### Function

```
public native static int AlprSDK_SetVideoParam
(
    int      nHandleID,
    VideoParam videoParam
);
```

### Description

Set video day and night switching parameters.

### Parameters

| Parameter             | Description                                  |
|-----------------------|----------------------------------------------|
| int nHandleID         | In: Number of Handles                        |
| VideoParam videoParam | In: Video switching parameters day and night |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.80 AlprSDK\_GetVideoParam

### Function

```
public native static int AlprSDK_GetVideoParam
(
    int      nHandleID,
    VideoParam videoParam
);
```

### Description

Get video day and night switching parameters.

### Parameters

| Parameter | Description |
|-----------|-------------|
|-----------|-------------|

|                                                 |                                              |
|-------------------------------------------------|----------------------------------------------|
| <code>int</code> <code>nHandleID</code>         | In: Number of Handles                        |
| <code>VideoParam</code> <code>videoParam</code> | In: Video switching parameters day and night |

## Returns

Successful:    It returns 0.

Unsuccessful:    Returns non-zero Error Code.

Error Code:    See [Appendix 1](#) for Error Code.

## 4.81 AlprSDK\_AddPlateListExt

### Function

```
public native static int AlprSDK_AddPlateListExt
(
    int nHandleID,
    int t,
    PLATE_INFO_EXT[] plateList,
    int count
);
```

### Description

Additional license plate information (if there is a duplicate license plate, it will be replaced directly).

### Parameters

| Parameter                  | Description                                                                                     |
|----------------------------|-------------------------------------------------------------------------------------------------|
| int nHandleID              | In: Number of Handles                                                                           |
| int t                      | In: license plate type, 0: whitelist, 1: blacklist, 2: fixed car list                           |
| PLATE_INFO_EXT[] plateList | In: An array of license plate information, which will store the license plate data to be added. |
| int count                  | In: count number of license plate data                                                          |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.82 AlprSDK\_DelplateListExt

### Function

```
public native static int AlprSDK_DelPlateListExt
(
    int nHandleID,
    int t,
```

```
    PLATE_INFO_EXT[] plateList,  
        int count  
);
```

### Description

Delete license plate information.

### Parameters

| Parameter                  | Description                                                                                     |
|----------------------------|-------------------------------------------------------------------------------------------------|
| int nHandleID              | In: Number of Handles                                                                           |
| int t                      | In: license plate type, 0: whitelist, 1: blacklist, 2: fixed car list                           |
| PLATE_INFO_EXT[] plateList | In: An array of license plate information, which will store the license plate data to be added. |
| int count                  | In: Number of license plate data                                                                |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.83 AlprSDK\_ClearPlateListExt

### Function

```
public native static int AlprSDK_ClearPlateListExt
(
    int nHandleID,
    int t
);
```

### Description

Clear white / black / fixed car list data.

### Parameters

| Parameter     | Description                                                           |
|---------------|-----------------------------------------------------------------------|
| int nHandleID | In: Number of Handles                                                 |
| int t         | In: license plate type, 0: whitelist, 1: blacklist, 2: fixed car list |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.84 AlprSDK\_ExportPlateListExt

### Function

```
public native static int AlprSDK_ExportPlateListExt
(
    int      nHandleID,
    int      t,
    PLATE_INFO_EXT[] plateList,
    Integer  count
);
```

### Description

Export license plate information (extension) .

### Parameters

| Parameter                   | Description                                                                   |
|-----------------------------|-------------------------------------------------------------------------------|
| int nHandleID               | In: Number of Handles                                                         |
| int t                       | In: license plate type, 0: whitelist, 1: blacklist, 2: fixed car lists        |
| PLATE_INFO_EXT[] plateList, | In: An array of license plate information that stores the data to be imported |
| Integer count               | In: count number of License plate data                                        |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

#### Remark

Support only for African firmware

## 4.85 AlprSDK\_GetDeviceDataCount

#### Function

```
public native static int AlprSDK_GetDeviceDataCount
(
    int    nHandleID,
    int    tType,
    int    offset,
    Integer count
);
```

#### Description

Gets the number of data bars for the device.

#### Parameters

| Parameter | Description |
|-----------|-------------|
|-----------|-------------|

|                    |                                                                                                                                   |
|--------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| int      nHandleID | In: Number of Handles                                                                                                             |
| int      tType     | In: table type (3 - Temporary vehicle charging rules, 4 - General charging rules, 5 - Entry and exit records, 7 charging records) |
| int      offset    | In: offset starts from an ID                                                                                                      |
| Integer    count   | In: count number of data                                                                                                          |

## Returns

Successful:    It returns 0.

Unsuccessful:    Returns non-zero Error Code.

Error Code:    See [Appendix 1](#) for Error Code.

## 4.86AlprSDK\_GetDeviceData

### Function

```
public native static int AlprSDK_GetDeviceData
(
    int      nHandleID,
    int      tType,
    char[]    buf,
    int      offset,
    integer    count
);
```

## Description

Get device data.

## Parameters

| Parameter        | Description                                                                                                                              |
|------------------|------------------------------------------------------------------------------------------------------------------------------------------|
| int    nHandleID | <b>In:</b> Number of Handles                                                                                                             |
| int    tType     | <b>In:</b> table type (3 - Temporary vehicle charging rules, 4 - General charging rules, 5 - Entry and exit records, 7 charging records) |
| char[] buf       | <b>Out:</b> Data                                                                                                                         |
| int    offset    | <b>In:</b> starts from an ID                                                                                                             |
| integer    count | <b>In/Out:</b> The number of records expected to be returned / the number of data actually returned                                      |

## Returns

Successful:    It returns 0.

Unsuccessful:    Returns non-zero Error Code.

Error Code:    See [Appendix 1](#) for Error Code.

## 4.87 AlprSDK\_SetDeviceData

### Function

```
public native static int AlprSDK_SetDeviceData
(
    int    nHandleID,
    int    type,
    char[] buf,
    int    count
);
```

### Description

Send data to the device.

### Parameters

| Parameter     | Description                                                                                                                                                                                           |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| int nHandleID | In: Number of Handles                                                                                                                                                                                 |
| int type      | In: table type (3 - Temporary vehicle charging rules, 4 - General charging rules)                                                                                                                     |
| char[] buf    | In: Data (need to construct a string of a specific format, please be sure to use the XX Container class provided by the SDK to construct, the demo below the call will give a specific usage example) |

|           |                    |
|-----------|--------------------|
| int count | In: Number of data |
|-----------|--------------------|

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Example

```
TempCarChargeRuleContainer container = new TempCarChargeRuleContainer();

// The secondary cycle is to build 10 tests in batches. The mock data, writing method is not standard for
reference only.

for (int i = 0; i < 10; i++) {
    TempCarChargeRule tempCarChargeRule = new TempCarChargeRule();
    tempCarChargeRule.ID = i;
    tempCarChargeRule.FreeMinutes = 10;
    tempCarChargeRule.IncludeFreeMinFlag = 1;
    tempCarChargeRule.CrosstimeSpitFlag = 0;
    tempCarChargeRule.MaxFeePerDay = 100.0f;

    for (int j = 0; j < 16; j++) {
        tempCarChargeRule.GenChargeRuleIDs[j] = j;
    }
    container.add(tempCarChargeRule);
}

System.out.println("container:" + container);

ret = AlprSDK.AlprSDK_SetDeviceData(0, 3, container.simpleSerialization(),
container.size());
System.out.println("AlprSDK_SetDeviceData ret = " + ret);
```

```
GenChargeRuleContainer genChargeRuleContainer = new GenChargeRuleContainer();
```

```
// The following two loops are designed to bulk build 20 tests using mock data, writing that is not
```

standard for reference only

```
for (int i = 0; i < 15; i++) {
    TimezoneRule tmpTimezoneRule = new TimezoneRule();
    tmpTimezoneRule.StartTime = 830;
    tmpTimezoneRule.EndTime = 900;
    tmpTimezoneRule.PayPerViewFlag = 0;
    tmpTimezoneRule.UnitTimeMin = 30;
    tmpTimezoneRule.UnitTimeCharge = 2;
    tmpTimezoneRule.MaxCharge = 50;
    GenChargeRule tmpGenChargeRule = new GenChargeRule(i, tmpTimezoneRule);
    genChargeRuleContainer.add(tmpGenChargeRule);
}

for (int i = 15; i < 20; i++) {
    PeriodRule tmpPeriodRule = new PeriodRule();
    tmpPeriodRule.Min = 30;
    tmpPeriodRule.PayPerViewFlag = 0;
    tmpPeriodRule.UnitTimeMin = 30;
    tmpPeriodRule.UnitTimeCharge = 2;
    tmpPeriodRule.MaxCharge = 50;
    GenChargeRule tmpGenChargeRule = new GenChargeRule(i, tmpPeriodRule);
    genChargeRuleContainer.add(tmpGenChargeRule);
}

System.out.println("container:" + new
String(genChargeRuleContainer.simpleSerialization()));
ret = AlprSDK.AlprSDK_SetDeviceData(0, 4,
genChargeRuleContainer.simpleSerialization(),
genChargeRuleContainer.size());
System.out.println("AlprSDK_SetDeviceData ret:" + ret);
```

## 4.88 AlprSDK\_ClearDeviceData

### Function

```
public native static int AlprSDK_ClearDeviceData  
(  
    int nHandleID,  
    int tType  
);
```

### Description

Clear a table.

### Parameters

| Parameter     | Description                                                                                                                       |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------|
| int nHandleID | In: Number of Handles                                                                                                             |
| int tType     | In: table type (3 - Temporary vehicle charging rules, 4 - General charging rules, 5 - Entry and exit records, 7 charging records) |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.89 AlprSDK\_CommTransparentTransfer

### Function

```
public native static int AlprSDK_CommTransparentTransfer
(
    int      nHandleID,
    byte[]    data,
    int      len
);
```

### Description

RS485 transparent transmission.

### Parameters

| Parameter     | Description           |
|---------------|-----------------------|
| int nHandleID | In: Number of Handles |
| byte[] data   | In: Data              |
| int len       | In: Data length       |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Remarks

The length cannot exceed 255, and the software must be called online to be valid (the parameter clientType must be 0 or 1 when connecting to the device)

## Example

If you want to set up a real-time display of "welcome" to the two-color screen, the following code:

```
char[] dataChars = new char[50];

String rawData =
"AA420000AE1A0023090001BBB6D3ADB9E2C1D9240B0000323031382D30352D3238F535";

for (int i = 0; i < rawData.length() / 2; i++) {
    dataChars[i] = (char) (0xFF & Integer.parseInt(rawData.substring(i * 2, i *
2 + 2), 16));
    System.out.println("The" + i + " " + (int) dataChars[i]);
}
ret = AlprSDK.AlprSDK_CommTransparentTransfer(0, dataChars, rawData.length() /
2);
System.out.println("AlprSDK_CommTransparentTransfer ret = " + ret);
```

For the format of the rawData in the above example, see Chapter 3.5 of the Communication Protocol between the camera and the dual-color screen (voice and display) in the appendix.

## 4.90 AlprSDK\_SetDColorDLineSCNParam

### Function

```
public native static int AlprSDK_SetDColorDLineSCNParam
```

```
(  
    int nHandleID,  
    DScreenDLine screenParam  
);
```

## Description

Set the default parameters for two-color dual-line screen.

## Parameters

| Parameter                | Description                                |
|--------------------------|--------------------------------------------|
| int nHandleID            | In: Number of Handles                      |
| DScreenDLine screenParam | In: Two-color dual-line screen information |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Remarks

The software must be called online to be valid (the parameter clientType must be 0 or 1 when connecting to the device)

## 4.91 AlprSDK\_GetVideoSize

### Function

```
public native static int AlprSDK_GetVideoSize
(
    int    nHandleID,
    Integer width,
    Integer hight
);
```

### Description

Get the video size.

### Parameters

| Parameter     | Description           |
|---------------|-----------------------|
| int nHandleID | In: Number of Handles |
| Integer width | Out: Video Weight     |
| Integer hight | Out: Video Height     |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Remarks

In P2P connection mode, AlprSDK.AlprSDK\_StartVideo () must be called to get the size of the video.

## 4.92 AlprSDK\_SetVideoConfig

### Function

```
public native static int AlprSDK_SetVideoConfig  
(  
    int nHandleID,  
    ALPR_VIDEO_CONFIG videoCfg  
);
```

### Description

Set up video stream information.

### Parameters

| Parameter                  | Description                                       |
|----------------------------|---------------------------------------------------|
| int nHandleID              | <b>In:</b> Number of Handles                      |
| ALPR_VIDEO_CONFIG videoCfg | <b>In:</b> Video stream configuration information |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.93 AlprSDK\_GetVideoConfig

### Function

```
public native static int AlprSDK_GetVideoConfig
(
    int      nHandleID,
    ALPR_VIDEO_CONFIG videoCfg
);
```

### Description

Get video stream information.

### Parameters

| Parameter                     | Description                                |
|-------------------------------|--------------------------------------------|
| int      nHandleID            | In: Number of Handles                      |
| ALPR_VIDEO_CONFIG    videoCfg | In: Video stream configuration information |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.94 AlprSDK\_OpenAuxOut

### Function

```
public native static int AlprSDK_OpenAuxOut
(
    int    nHandleID ,
    int    index,
    long   time
);
```

### Description

Turn on auxiliary output.

### Parameters

| Parameter        | Description                                       |
|------------------|---------------------------------------------------|
| int    nHandleID | In: Number of Handles                             |
| int    index     | In: Auxiliary output number (values 1, 2, 3, ...) |

|                        |                                                                 |
|------------------------|-----------------------------------------------------------------|
| <code>long time</code> | <b>In:</b> Auxiliary output duration (values 0, 2, 3, ..., 255) |
|------------------------|-----------------------------------------------------------------|

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## 4.95 AlprSDK\_EnableP2PReconnect

### Function

```
public native static int AlprSDK_EnableP2PReconnect(int nHandleID);
```

### Description

Turn on P2P reconnection (valid only for P2P connections).

### Parameters

| Parameter                  | Description                  |
|----------------------------|------------------------------|
| <code>int nHandleID</code> | <b>In:</b> Number of Handles |

### Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

#### Remarks

When the reconnect function is enabled, the user's following interface is called.

The precondition for AlprSDK\_DisConnectDev, AlprSDK\_StopVideo is to turn off the P2P reconnection function, otherwise the call will fail.

The call timing of this interface should be before AlprSDK\_ConnectDev, AlprSDK\_StartVideo.

## 4.96 AlprSDK\_DisableP2PReconnect

#### Function

```
public native static int AlprSDK_DisableP2PReconnect(int nHandleID);
```

#### Description

Turn off P2P reconnection (valid only for P2P connections).

#### Parameters

| Parameter     | Description           |
|---------------|-----------------------|
| int nHandleID | In: Number of Handles |

## Returns

Successful: It returns 0.

Unsuccessful: Returns non-zero Error Code.

Error Code: See [Appendix 1](#) for Error Code.

## Remarks

When calling this interface, if the SDK is reconnecting, wait until the SDK reconnects and returns.

## 5 Structure definition

### 5.1 PLATERESULT

#### PLATERESULT Class Description

```
public class PLATERESULT {  
    public String license;  
    public int letterCount;  
    public float confidence;  
    public XRect rect;  
    public byte plateColor;  
    public boolean doublePlates;  
    public byte direction;  
    public byte bFakePlate;  
    public int typeResult;  
  
    public PLATERESULT() {  
        license = new String();  
        letterCount = -1;  
        confidence = -1;  
        rect = new XRect();  
        plateColor = -1;  
        doublePlates = false;  
        direction = -1;  
    }  
    .....  
};
```

#### Description

The result structure of the recognized license plate

#### Parameters

| Parameter                         | Description                                                                                                          |
|-----------------------------------|----------------------------------------------------------------------------------------------------------------------|
| public String license             | <b>Out:</b> License plate number character                                                                           |
| public int letterCount            | <b>Out:</b> Number of characters of license plate number                                                             |
| public float confidence           | The credibility of the license plate, the value range [1,100].<br>The smaller the value, the higher the credibility. |
| public <a href="#">XRect</a> rect | License plate location information                                                                                   |
| public byte plateColor            | License plate color:<br>0-black;20-green;30-blue;40-red;50-yellow;255-white                                          |
| public boolean doublePlates       | Is there a double license plate                                                                                      |
| public byte direction             | For movement direction, please refer to MotionDir for values.                                                        |
| public byte bFakePlate            | False license plate                                                                                                  |
| public int typeResult             | Vehicle Type Identified by Equipment End: Large and Small Vehicles                                                   |

## 5.2DEVINFO

### DEVINFO Structure Description

```
public class DEVINFO implements Cloneable {  
    public String ipAddr;  
    public String devUid;  
    public short ifOpenP2p;  
    public short ul6port;  
    public String userName;  
    public String password;  
    public String picturesSavePath;  
    public short alprPort;  
    public short alprPullPort;  
    public short lprDevType;  
    public long pullHandle;  
  
    public DEVINFO() {  
        ipAddr = new String();  
        devUid = new String();  
        userName = new String();  
        password = new String();  
        picturesSavePath = new String();  
    }  
    .....  
};
```

### Description

Camera device parameter definition.

### Parameters

| Parameter            | Description |
|----------------------|-------------|
| public String ipAddr | IP address  |

|                                             |                                                                                                                           |
|---------------------------------------------|---------------------------------------------------------------------------------------------------------------------------|
| <code>public String devUid</code>           | Device serial number                                                                                                      |
| <code>public short ifOpenP2p</code>         | Whether to use P2P connection method                                                                                      |
| <code>public short u16port</code>           | Port number                                                                                                               |
| <code>public String userName</code>         | Username                                                                                                                  |
| <code>public String password</code>         | Password                                                                                                                  |
| <code>public String picturesSavePath</code> | Picture local save path                                                                                                   |
| <code>public short alprPort</code>          | Alpr port number                                                                                                          |
| <code>public short alprPullPor</code>       | Pull port number                                                                                                          |
| <code>public short lprDevType</code>        | Device Type:<br>0-Unknown Device, 1-Dragon Device, 2-Sense Device                                                         |
| <code>public long pullHandle</code>         | <b>pullsdk handle:</b> This sdk only manages its opening and closing, other software directly calls the pullsdk interface |

## 5.3 ALPR\_VIDEO\_CONFIG

### ALPR\_VIDEO\_CONFIG Class Description

```
public class ALPR_VIDEO_CONFIG {
    public int nEncFormat;
    public String toString() {
        StringBuffer retBuf = new StringBuffer();
        retBuf.append("{");
        retBuf.append("nEncFormat:").append(nEncFormat);
    }
}
```

```
        retBuf.append("}");  
        return retBuf.toString();  
    }  
};
```

### Description

Definition of video coding mode

### Parameters

| Parameter                          | Description                                                      |
|------------------------------------|------------------------------------------------------------------|
| <code>public int nEncFormat</code> | Only two are supported for the time being:<br><br>0:1080p, 3:D1. |

## 5.4 DEV\_VER\_INFO

### DEV\_VER\_INFO Class Description

```
public class DEV_VER_INFO {  
  
    public String algorithm_Ver;  
    public String firm_Ver;  
    public String adk_Ver;  
  
    .....  
  
};
```

### Description

Device version information definition.

#### Parameters

| Parameter                                | Description        |
|------------------------------------------|--------------------|
| <code>public String algorithm_Ver</code> | Algorithm Version. |
| <code>public String firm_Ver</code>      | Firmware Version   |
| <code>public String adk_Ver</code>       | Sdk version        |

## 5.5 ALPR\_CONFIG

#### ALPR\_CONFIG Class Description

```
public class ALPR_CONFIG implements Cloneable {  
  
    public byte installDistID;  
    public byte recogNewEmbassyLic;  
    public byte enableMotionDectect;  
    public short defaultProvinceID;  
    public byte laneCount;  
    public byte ifAirportVehicle;  
    public byte ifAgriculturalVehicle;  
    public byte outputStableResult;  
    public short maxRecogCount;  
    public short outputInterval;  
    public byte triggerMode;  
    public byte recogMode;  
    public byte uploadMode;  
    public byte outputMode;  
    public byte recogEmbassyLic;  
    public byte recogDoubleLic;  
}
```

```

    public byte recogSameLicOneTime;
    public byte recogOverlayChars;
    public byte recogOverlayRect;
    public byte samePosFilter;
    public byte virtualCoilTrigger;
    public byte filterStrip;
    public short countryCode;
    public short alarmMaxRecogTime;
    public float fRecogThreshold;

    .....
};

```

### Parameters

| Parameter                         | Description                                                                                     |
|-----------------------------------|-------------------------------------------------------------------------------------------------|
| public byte installDistID         | by distance ID<br>0-3.5m below<br>1-3.5m-4.5m<br>2-4.5m-5m<br>3-5 -6 meters<br>4-6 meters above |
| public byte recogNewEmbassyLic    | Support for the new embassy license plate                                                       |
| public byte enableMotionDectect   | Start motion detection                                                                          |
| public short defaultProvinceID    | The default province, the value is tPLATE_SYM_ID                                                |
| public byte laneCount             | Currently only supports 1                                                                       |
| public byte ifAirportVehicle      | Does it support civil aviation license plates?                                                  |
| public byte ifAgriculturalVehicle | Support for license plates of agricultural vehicles                                             |

|                                              |                                                                                                                      |
|----------------------------------------------|----------------------------------------------------------------------------------------------------------------------|
| <code>public byte outputStableResult</code>  | Stable identification trigger                                                                                        |
| <code>public short maxRecogCount</code>      | maximum number of recognitions                                                                                       |
| <code>public short outputInterval</code>     | Same license plate entry interval in seconds                                                                         |
| <code>public byte triggerMode</code>         | Trigger mode, the value is :<br><br>0-Video Trigger, 1-Ground Trigger, 2-Mixed Trigger                               |
| <code>public byte recogMode</code>           | Recognition mode, the value is:<br><br>0-single frame mode, 1-multi-frame preferred mode                             |
| <code>public byte uploadMode</code>          | Upload mode, the value is:<br><br>0-- FTP upload,<br><br>1-SDK upload,<br><br>2-JSON upload,<br><br>3-U disk storage |
| <code>public byte outputMode</code>          | Input mode, the value is:<br><br>0-all output,<br><br>1-filter vehicles leaving,<br><br>2-filter vehicles leaving    |
| <code>public byte recogEmbassyLic</code>     | Is it the embassy license plate                                                                                      |
| <code>public byte recogDoubleLic</code>      | Whether it is a double deck license plate                                                                            |
| <code>public byte recogSameLicOneTime</code> | 1: The same license plate is only output once, 0 has<br><br>no such limit                                            |
| <code>public byte recogOverlayChars</code>   | Overlay OSD characters                                                                                               |

|                                             |                                                                                                                                      |
|---------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| <code>public byte recogOverlayRect</code>   | Overlay OSD Rectangular Box                                                                                                          |
| <code>public byte samePosFilter</code>      | Is the license plate output at the same position,<br>1--no output, 0--output                                                         |
| <code>public byte virtualCoilTrigger</code> | Virtual coil division 1--Use virtual coil, 0--Do not use<br>virtual coil                                                             |
| <code>public byte filterStrip</code>        | Whether to enable stripe filtering, 1--enabled, 0--not<br>enabled                                                                    |
| <code>public short countryCode</code>       | Country code                                                                                                                         |
| <code>public short alarmMaxRecogTime</code> | Maximum recognition time after ground trigger, in<br>seconds.                                                                        |
| <code>public float fRecogThreshold</code>   | Identifies the threshold, which is [0, 1000.0f]. The<br>smaller the value, the higher the confidence. The<br>default value is 100.0f |

## 5.6 OSD\_CTRL

### OSD\_CTRL Class Description

```
public class OSD_CTRL implements Cloneable {

    public OSD_PARAM stDateOSD;
    public OSD_PARAM stTimeOSD;
    public OSD_PARAM stWeekOSD;
```

```
public OSD_PARAM stBitrateOSD;
public OSD_PARAM stTitleOSD;
public String strTitle;

public OSD_CTRL() {
    stDateOSD = new OSD_PARAM();
    stTimeOSD = new OSD_PARAM();
    stWeekOSD = new OSD_PARAM();
    stBitrateOSD = new OSD_PARAM();
    stTitleOSD = new OSD_PARAM();
    strTitle = "";
}

.....
};
```

### Description

Video control mode: The sensing device only supports setting the title strTitle and whether to display the title or not

### Parameters

| Parameter                     | Description                       |
|-------------------------------|-----------------------------------|
| public OSD_PARAM stDateOSD    | Video date display location       |
| public OSD_PARAM stTimeOSD    | Video time display position       |
| public OSD_PARAM stWeekOSD    | Video week display position       |
| public OSD_PARAM stBitrateOSD | Video resolution display position |
| public OSD_PARAM stTitleOSD   | Video title display location      |
| public String strTitle        | video title                       |

## 5.7BASE\_PARAM

### BASE\_PARAM Class Description

```
public class BASE_PARAM implements Cloneable {
    public byte rgnArea;
    public byte virtualCoil;
    public byte lprAccuracyMatch;
    public byte imageSwitchType;
    public byte fillLightBrightness;
    public byte auxiliaryCameraFlag;
    public String sMasterCamIP;
    public byte ledCharacterEncoding;
    public byte onlyShowLic;
    public short nCountryCode;
    public short nStartTime;
    public short nEndTime;
    public short nPhotosensitiveThreshold;
    public short nOfflineLevel;

    public BASE_PARAM() {
        sMasterCamIP = new String();
    }

    .....
};
```

### Parameters

| Parameter               | Description                                                        |
|-------------------------|--------------------------------------------------------------------|
| public byte rgnArea     | Whether to display the recognition area, 0: no display, 1: display |
| public byte virtualCoil | Whether to display virtual coil, 0: no display, 1:                 |

|                                                    |                                                                                                                                                           |
|----------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                                    | display                                                                                                                                                   |
| <code>public byte lprAccuracyMatch</code>          | License plate matching accuracy                                                                                                                           |
| <code>public byte imageSwitchType</code>           | The switching type of image parameters during the day and at night,<br><br>0-photosensitive threshold switching,<br><br>1-time period switching           |
| <code>public byte fillLightBrightness</code>       | fill light brightness [0-100], 0: turn off the fill light,<br><br>other: fill light brightness value                                                      |
| <code>public byte auxiliaryCameraFlag</code>       | Whether it is a secondary camera,<br><br>0: host, 1: auxiliary camera                                                                                     |
| <code>public String sMasterCamIP</code>            | main camera IP address                                                                                                                                    |
| <code>public byte ledCharacterEncoding</code>      | Display character encoding                                                                                                                                |
| <code>public byte onlyShowLic</code>               | Does the display only show license plates, 0: other, 1:<br><br>only send license plates                                                                   |
| <code>public short nCountryCode</code>             | country code, reference enumeration<br><br><a href="#">COUNTRY_CODE_E</a>                                                                                 |
| <code>public short nStartTime</code>               | Switch to daytime image parameters at 6:00-->600                                                                                                          |
| <code>public short nEndTime</code>                 | Switch to night image parameters at 18:00-->1800                                                                                                          |
| <code>public short nPhotosensitiveThreshold</code> | Photosensitive threshold [0-1024], which is greater<br><br>than this value to switch to the night parameter,<br><br>otherwise switch to daytime parameter |

|                                       |                                                                                                                                                                                                          |
|---------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre>public short nOfflineLevel</pre> | <p>Offline level,</p> <p>0-turn off offline function,</p> <p>1-save records to SD card,</p> <p>2-save records to FLASH,</p> <p>3-save pictures to SD card,</p> <p>255- turn on all offline functions</p> |
|---------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

## 5.8 PLATE\_INFO\_EXT

### PLATE\_INFO\_EXT Class Description

```
public class PLATE_INFO_EXT {

    public String plate;
    public XDate fromDate;
    public XDate toDate;
    public String name;
    public String cardNo;
    public short plateColor;

    public PLATE_INFO_EXT() {
        plate = new String();
        fromDate = new XDate();
        toDate = new XDate();
        name = new String();
        cardNo = new String();
    }

    public PLATE_INFO_EXT(String plate, XDate fromDate, XDate toDate,
        String name, String cardNo, short plateColor) {
        this.plate = plate;
        this.fromDate = fromDate;
        this.toDate = toDate;
    }
}
```

```
        this.name = name;
        this.cardNo = cardNo;
        this.plateColor = plateColor;
    }

    .....
};
```

## Description

The list of license plate structures used in the country.

## Parameters

| Parameter                            | Description                                                              |
|--------------------------------------|--------------------------------------------------------------------------|
| <code>public String plate</code>     | license plate                                                            |
| <code>public XDate fromDate</code>   | date commenced                                                           |
| <code>public XDate toDate</code>     | End date                                                                 |
| <code>public String name</code>      | Name of owner                                                            |
| <code>public String cardNo</code>    | License plate number                                                     |
| <code>public short plateColor</code> | License plate<br>color,0--black,20--green,30--bule,50--yellow,255--white |

## 5.9GATE\_CONFIG

### GATE\_CONFIG Class Description

```
public class GATE_CONFIG implements Cloneable {
    public int workMode;
```

```
        public int latency;  
        public int inverseSignal;  
  
        .....  
};
```

### Description

Gate parameter definition.

### Parameters

| Parameter                | Description                                                |
|--------------------------|------------------------------------------------------------|
| pubilc int WorkMode      | Operation mode of road gate                                |
| pubilc int Latency       | Delay in operation of gate                                 |
| public int inverseSignal | Whether or not to reverse the signal. The value is 1 or 0. |

## 5.10 DSCREENDLINE

### DScreenDLine Class Description

```
public class DScreenDLine {  
  
    public int countryCode;  
    public byte firstLineColor;  
    public byte secondLineColor;  
    public byte moveDirection1;  
    public byte moveDirection2;  
    public int moveSpeed;  
}
```

```

public int showTimer;
public int volume;
public String firstLinedef;
public boolean timeSync;
public SysTime curTimer;

public DScreenDLine() {
    curTimer = new SysTime();
}

.....
};

```

### Description

Definition of Two-color and Two-line Screen Information.

### Parameters

| Parameter                   | Description                                                                   |
|-----------------------------|-------------------------------------------------------------------------------|
| public int countryCode      | Country code                                                                  |
| public byte firstLineColor  | The first row shows colors (0: red, 1: green)                                 |
| public byte secondLineColor | The second row shows color (0:red,1:green)                                    |
| public byte moveDirection1  | First row scroll direction (0: from left to right, 1: from right to left)     |
| public byte moveDirection2  | Scroll direction of second row (0: from left to right, 1: from right to left) |
| public int moveSpeed        | Moving speed (0-100)                                                          |
| public int showTimer        | Display duration (0-100s)                                                     |
| public int volume           | Volume (0-31)                                                                 |

|                            |                                                |
|----------------------------|------------------------------------------------|
| public String firstLinedef | The first line displays information by default |
| public boolean timeSync    | Time synchronization                           |
| public SysTime curTimer    | Rcurrent time                                  |

## 5.11 WIEGANDDATA

### WiegandData Class Description

```
public class WiegandData {
    public long wiegandCardNo;

    .....

};
```

### Description

Wigan card data.

### Parameters

| Parameter                 | Description       |
|---------------------------|-------------------|
| public long wiegandCardNo | Wigan card number |

## 5.12 ZOOMCTLPARA

### ZoomCtlPara Function Description

```
public class ZoomCtlPara {
    public int cmd;
    public int stepSpeed;
```

```
.....  
};
```

### Description

Zoom control parameters

### Parameters

| Parameter            | Description                                                                                                              |
|----------------------|--------------------------------------------------------------------------------------------------------------------------|
| public int cmd       | Zoom control command :<br><br>0-focal length zooms in at stepSpeed,<br><br>1-focal length is reduced at speed stepSpeed. |
| public int stepSpeed | Step size, range: 1-16                                                                                                   |

## 5.13 FOCUSCTL PARA

### FocusCtlPara Function Description

```
public class FocusCtlPara {  
  
    public int cmd;  
    public int stepSpeed;  
    .....  
};
```

### Description

Focus control parameters.

## Parameters

| Parameter            | Description                                                                                                  |
|----------------------|--------------------------------------------------------------------------------------------------------------|
| public int cmd       | Focus Control Command:<br><br>0-Focus Forward at Speed stepSpeed,<br><br>1-Focus Backward at Speed stepSpeed |
| public int stepSpeed | Step size, range: 1-16                                                                                       |

## 5.14 JPG\_BYTES

### Jpg\_Bytes Class Description

```
public class JPG_BYTES {  
  
    public String time;  
    public int nBytesLen;  
    public byte[] jpgBytes;  
    public JPG_BYTES() {  
        time = new String();  
        nBytesLen = -1;  
        jpgBytes = null;  
    }  
  
    .....  
};
```

### Description

Capture JPG picture information definition.

## Parameters

| Parameter                           | Description                                                                                                    |
|-------------------------------------|----------------------------------------------------------------------------------------------------------------|
| <code>public String time</code>     | The time when the picture was captured is<br>20161012163417050, i.e. 16: 34: 17: 050 ms on<br>October 12, 2016 |
| <code>public int nBytesLen</code>   | Jpg data length                                                                                                |
| <code>public byte[] jpgBytes</code> | Jpg data can be directly saved as jpg files.                                                                   |

## 5.15 XRECT

### XRect Class Description

```
public class XRect implements Cloneable {  
  
    public int left;  
    public int right;  
    public int top;  
    public int bottom;  
  
    public XRect() {  
        left = -1;  
        right = -1;  
        top = -1;  
        bottom = -1;  
    }  
  
    .....  
};
```

### Description

Coordinate information. the current resolution is based on 1024\*720.

#### Parameters

| Parameter               | Description |
|-------------------------|-------------|
| <code>int left</code>   | left        |
| <code>int right</code>  | right       |
| <code>int top</code>    | top         |
| <code>int bottom</code> | bottom      |

## 5.16DEVLICENSE

#### DevLicense Class Description

```
public class DevLicense {  
    public int licenseType;  
    public int licenseStatus;  
  
    .....  
};
```

#### Description

Device license information

#### Parameters

| Parameter | Description |
|-----------|-------------|
|-----------|-------------|

|                                       |                                                                                                                                                                                                                                                          |
|---------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>public int licenseType</code>   | 0: old, 1: new                                                                                                                                                                                                                                           |
| <code>public int licenseStatus</code> | <div>License Status</div> <div>0: Authorization success</div> <div>1: License file matching is incorrect</div> <div>2: License expired</div> <div>3: Algorithm initialization failed or no license</div> <div>4: Encrypted chip unlicensed content</div> |

5.17 SYS\_TIME

SYS\_TIME Function Description

```
public class SysTime {
    public byte year;
    public byte month;
    public byte day;
    public byte week;
    public byte hour;
    public byte minute;
    public byte second;

    public SysTime() {
    }

    public SysTime(byte year, byte month, byte day, byte week, byte hour, byte
minute, byte second) {
        this.year = year;
        this.month = month;
        this.day = day;
```

```

        this.week = week;
        this.hour = hour;
        this.minute = minute;
        this.second = second;
    }

    .....

};

```

### Description

System time definition.

### Parameters

| Parameter             | Description                           |
|-----------------------|---------------------------------------|
| public byte    year   | Year, excluding the current year 2000 |
| public byte    month  | moon                                  |
| public byte    day    | Day                                   |
| public byte    week   | Week                                  |
| public byte    hour   | Hour                                  |
| public byte    minute | Minute                                |
| public byte    second | Seconds                               |

## 5.18 OFFLINEPARAM

### OfflineParam Class Description

```

public class OfflineParam implements Cloneable{
    public String serverIP;
    public int serverPort;
    public int parkID;
}

```

```
public byte recordIsCover;
public byte parkInOutFlag;
public short monthcarAlarmDays;
public byte recognitionAccuracy;
public byte recordMatchAccuracy;
public byte monthCarToTempcarFlag;
public byte monthCarOpenType;
public byte tempCarOpenType;
public float minCharge;
public byte tempCarForbiddenFlag;
public int syncTimeFromMaster;
public byte onlineFlag;
public byte oneChannelMode;
public long oneChannelWaitTime;
public long normalModeWaitTime;
public byte minChargeFlag;
public short displayRefreshInterval;
public String propertyLogo;
public byte nScreenType;
public byte nZeroQuickOut;
public OfflineParam() {
    serverIP = new String();
    propertyLogo = new String();
}
.....
};
```

## Parameters

| Parameter                 | Description                                                                   |
|---------------------------|-------------------------------------------------------------------------------|
| public String serverIP    | Primary server IP address                                                     |
| public int serverPort     | Primary server port                                                           |
| public int parkID         | parking lot number                                                            |
| public byte recordIsCover | Record (outbound and charge records) coverage, 1: coverage,<br>0: no coverage |

|                                                |                                                                                                   |
|------------------------------------------------|---------------------------------------------------------------------------------------------------|
| <code>public byte parkInOutFlag</code>         | Logo of depot entrance, 0-depot entrance, 1-depot exit                                            |
| <code>public short monthcarAlarmDays</code>    | Fixed vehicle warning days                                                                        |
| <code>public byte recognitionAccuracy</code>   | Fixed vehicle matching accuracy, 99: exact match                                                  |
| <code>public byte recordMatchAccuracy</code>   | record matching precision, 99: exact match                                                        |
| <code>public byte monthCarToTempcarFlag</code> | Enable fixed car to temporary car,<br>0-not enabled, 1-enabled                                    |
| <code>public byte monthCarOpenType</code>      | Fixed vehicle brake opening mode,<br>0-manual brake opening, 1-automatic brake opening            |
| <code>public byte tempCarOpenType</code>       | Temporary car brake opening mode, 0-manual brake<br>opening, 1-automatic brake opening            |
| <code>public float minCharge</code>            | Minimum charge                                                                                    |
| <code>public byte tempCarForbiddenFlag</code>  | Temporary car is forbidden to enter the venue, 0: Admission<br>is allowed, 1: No entry is allowed |
| <code>public int syncTimeFromMaster</code>     | Time point 23:00-->2300 -->0x08fc                                                                 |
| <code>public byte onlineFlag</code>            | Whether the display is in online mode, 0-offline, 1-online                                        |
| <code>public byte oneChannelMode</code>        | Whether to enable single channel mode, 0: not enable single<br>channel, 1: single channel mode    |
| <code>public long oneChannelWaitTime</code>    | Single channel repeat license plate waiting time in seconds                                       |
| <code>public long normalModeWaitTime</code>    | Normal mode repeats license plate waiting time in seconds                                         |
| <code>public byte minChargeFlag</code>         | Whether to enable the minimum charge, 0: not enabled, 1:<br>enabled                               |

|                                                      |                                                                                                                             |
|------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| <code>public short<br/>displayRefreshInterval</code> | The refresh interval of the offline display content, in seconds                                                             |
| <code>public String propertyLogo</code>              | The corporate logo displayed by default on the offline display                                                              |
| <code>public byte nScreenType</code>                 | Display screen type,<br><br>0-monochrome screen,<br><br>1-dual color dual-line screen,<br><br>2-dual color four-line screen |
| <code>public byte nZeroQuickOut</code>               | Charge 0 yuan, whether to play quickly                                                                                      |

## 5.19 SDCARDCAPACITY

### SDCardCapacity Function Description

```
public class SDCardCapacity {  
  
    public int nTotalCapacity;  
    public int nRemainCapacity;  
    .....  
};
```

#### Description

SD card capacity information definition.

#### Parameters

| Parameter                               | Description                 |
|-----------------------------------------|-----------------------------|
| <code>public int nTotalCapacity</code>  | Total capacity in megabytes |
| <code>public int nRemainCapacity</code> | Remaining space, in MB      |

## 5.20 NETWORKPARAM

### NetworkParam Function Description

```
public class NetworkParam implements Cloneable {  
  
    public String DVRIP;  
    public String DVRIPMask;  
    public String GatewayIpAddr;  
    public String MACAddr;  
    public String DnsServer1IpAddr;  
    public String DnsServer2IpAddr;  
    .....  
}
```

#### Description

Network configuration parameters.

#### Parameters

| Parameter                                   | Description      |
|---------------------------------------------|------------------|
| <code>public String DVRIP</code>            | IP address       |
| <code>public String DVRIPMask</code>        | Subnet mask      |
| <code>public String GatewayIpAddr</code>    | Gateway address  |
| <code>public String MACAdd</code>           | Physical address |
| <code>public String DnsServer1IpAddr</code> | DNS address 1    |
| <code>public String DnsServer2IpAddr</code> | DNS address 2    |

## 5.21 VIDEOPARAM

### VideoParam Function Description

```
public class VideoParam implements Cloneable {  
  
    public int Mode;  
    public int DayStart;  
    public int DayEnd;  
    .....  
}
```

## Description

Video parameter interface.

## Parameters

| Parameter           | Description                                                                          |
|---------------------|--------------------------------------------------------------------------------------|
| public int Mode     | Mode, 0- External Trigger; 1-Automatic; 2- Color; 3- black and white; 4- time period |
| public int DayStart | Daytime period start time (in seconds)=hour * 3600 + min * 60 + sec)                 |
| public int DayEnd   | Daytime period start time (in seconds)=hour * 3600 + min * 60 + sec)                 |

## 5.22DEVICEPARAM

### DeviceParam Function Description

```
public class DeviceParam implements Cloneable {  
    public String softwareVer;  
    public String hardwareVer;  
    public String serialNum;  
    public String dvrName;  
    public byte dvrType;
```

```
public byte alarmInPortNum;
public WebApiVer webApiVer;

public DeviceParam() {
    webApiVer = new WebApiVer();
}
.....
}
```

### Description

Equipment information.

### Parameters

| Parameter                               | Description                              |
|-----------------------------------------|------------------------------------------|
| <code>public String softwareVer</code>  | Software version number (master version) |
| <code>public String hardwareVer</code>  | Hardware version number                  |
| <code>public String serialNum</code>    | serial number                            |
| <code>public String dvrName</code>      | DVR name                                 |
| <code>public byte dvrType</code>        | DVR type. 1 - DVR, 2 - NVR, 3 - DVS/IPC  |
| <code>public byte alarmInPortNum</code> | Number of DVR Alarm Inputs               |
| <code>public WebApiVer webApiVer</code> | Web API Information                      |

## 5.23 WERAPIVER

### WebApiVer Function Description

```
public class WebApiVer implements Cloneable {
    public String standard;
```

```
        public String build;  
        .....  
    }
```

### Description

Web API information.

### Parameters

| Parameter              | Description      |
|------------------------|------------------|
| public String standard | Version number   |
| public String build    | Compilation date |

## 5.24 RECOG\_ALL\_INFO

### RECOG\_ALL\_INFO Structure Description

```
public class RECOG_ALL_INFO {  
    public LICENSE_PLATE PlateInfo;  
    public JPG_BYTES JpgBytes;  
    public RECOG_ALL_INFO() {  
        PlateInfo = new LICENSE_PLATE();  
        JpgBytes = new JPG_BYTES();  
    }  
    .....  
};
```

### Description

License Plate Recognition Information Definition.

### Parameters

| Parameter                      | Description                                      |
|--------------------------------|--------------------------------------------------|
| public LICENSE_PLATE PlateInfo | <b>Out:</b> Recognized license plate information |

|                                        |                                             |
|----------------------------------------|---------------------------------------------|
| <code>public JPG_BYTES JpgBytes</code> | <b>Out:</b> Capture JPG picture information |
|----------------------------------------|---------------------------------------------|

## 5.25DEVINFO

### DEVINFO Structure Description

```
public class DEVINFO implements Cloneable {
    public String ipAddr;
    public String devUid;
    public short ifOpenP2p;
    public short ul6port;
    public String userName;
    public String password;
    public String picturesSavePath;
    public short alprPort;
    public short alprPullPort;
    public short lprDevType;
    public long pullHandle;

    public DEVINFO() {
        ipAddr = new String();
        devUid = new String();
        userName = new String();
        password = new String();
        picturesSavePath = new String();
    }
    .....
}
```

### Description

Camera device parameter definition.

**Parameters**

| Parameter                                   | Description                                                                                                               |
|---------------------------------------------|---------------------------------------------------------------------------------------------------------------------------|
| <code>public String ipAddr</code>           | IP address                                                                                                                |
| <code>public String devUid</code>           | Device serial number                                                                                                      |
| <code>public short ifOpenP2p</code>         | Whether to use P2P connection method                                                                                      |
| <code>public short ul6port</code>           | Port number                                                                                                               |
| <code>public String userName</code>         | Username                                                                                                                  |
| <code>public String password</code>         | Password                                                                                                                  |
| <code>public String picturesSavePath</code> | Picture local save path                                                                                                   |
| <code>public short alprPort</code>          | Alpr port number                                                                                                          |
| <code>public short alprPullPort</code>      | Pull port number                                                                                                          |
| <code>public short lprDevType</code>        | Camera device type: 0-Unknown Device, 1-Dragon Device, 2-Sense Device                                                     |
| <code>public long pullHandle</code>         | <b>pullsdk handle:</b> This sdk only manages its opening and closing, other software directly calls the pullsdk interface |

## 5.26 DEVSTATUS

### DevStatus Class Description

```
public class DevStatus {  
    public AuxCameraStatus[] auxCameraStatus;  
    public int heartBeatStatus;  
    public byte gateSwitchStatus;  
    public byte gateRunStatus;  
    public short feelStatus;  
    public short photosensitiveValue;  
    public DevStatus() {  
        auxCameraStatus = new AuxCameraStatus[2];  
        for (int i = 0; i < 2; ++i) {  
            auxCameraStatus[i] = new AuxCameraStatus();  
        }  
        heartBeatStatus = -1;  
        gateSwitchStatus = -1;  
        gateRunStatus = -1;  
        feelStatus = -1;  
        photosensitiveValue = -1;  
    }  
    .....  
};
```

### Description

device status.

### Parameters

| Parameter                                   | Description             |
|---------------------------------------------|-------------------------|
| public AuxCameraStatus[]<br>auxCameraStatus | Secondary camera status |
| public int heartBeatStatus                  | Heartbeat state         |
| public byte gateSwitchStatus                | State of gate switch    |
| public byte gateRunStatus                   | Running state of gate   |

|                                  |                        |
|----------------------------------|------------------------|
| public short feelStatus          | Ground sensation state |
| public short photosensitiveValue | Photosensitive signal  |

## 5.27DEFBRIGHTNESS

### DefBrightness Class Description

```
public class DefBrightness {  
    public short defSensitiveValue;  
    public byte defBrightness;  
    .....  
};
```

#### Description

device status.

#### Parameters

| Parameter                      | Description                                                                                          |
|--------------------------------|------------------------------------------------------------------------------------------------------|
| public short defSensitiveValue | The default photosensitive lower limit value, above which the supplementary light will be turned on. |
| public byte defBrightness      | The brightness value when the supplementary light is turned on is in the range of [0-100]            |

## 5.28CHARGELOG

### ChargeLog Class Description

```
public class ChargeLog {  
    public int id;  
    public int inID;
```

```

    public int outID;
    public float receivableAmount;
    public float discountAmount;
    public float actualAmount;
    public int carTypeID;
    public int chargeTime;
    public int inSrcID;
    public String inSrcAddr;
    public ChargeLog() {
    }
    .....
};

```

### Description

Charge record definition.

### Parameters

| Parameter                     | Description                                                                                                                                                                                                                     |
|-------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| public int id                 | Record serial number                                                                                                                                                                                                            |
| public int inID               | Admission record number                                                                                                                                                                                                         |
| public int outID              | Admission record number                                                                                                                                                                                                         |
| public float receivableAmount | Amount receivable                                                                                                                                                                                                               |
| public float discountAmount   | Preferential amount                                                                                                                                                                                                             |
| public float actualAmount     | Paid in amount                                                                                                                                                                                                                  |
| public int carTypeID          | Car type number                                                                                                                                                                                                                 |
| public int chargeTime         | Charge time,<br>$(t \rightarrow tm\_year - 100) * 12 * 31 + ((t \rightarrow tm\_mon) * 31) + t \rightarrow tm\_mday - 1) * (24 * 60 * 60) + (t \rightarrow tm\_hour * 60 + t \rightarrow tm\_min) * 60 + t \rightarrow tm\_sec$ |
| public int inSrcID            | Admission ID                                                                                                                                                                                                                    |
| public String inSrcAddr       | Admission IP address                                                                                                                                                                                                            |

## 5.29 OSD\_PARAM

### OSD\_PARAM Class Description

```
public class OSD_PARAM implements Cloneable {  
    public int bShow;  
    public short x;  
    public short y;  
  
    public OSD_PARAM() {  
        bShow = 0;  
        x = 0;  
        y = 0;  
    }  
  
    .....  
};
```

#### Description

Video display position

#### Parameters

| Parameter        | Description                                                                                                                 |
|------------------|-----------------------------------------------------------------------------------------------------------------------------|
| public int bShow | 0: Close 1: Display                                                                                                         |
| public short x   | The displayed position, ranging from 0 to 704, is independent of the resolution of the image, and x must be a multiple of 4 |
| public short y   | The value range is 0-576                                                                                                    |

## 5.30 FPCLIENTCFG

### FtpClientCfg Class Description

```
public class FtpClientCfg implements Cloneable {

    public boolean enable;
    public boolean pasvMode;
    public short remotePort;
    public String remoteHost;
    public String userName;
    public String passWord;
    public String uploadPath;

    public FtpClientCfg() {
        enable = false;
        pasvMode = false;
        remotePort = 0;
        userName = "";
        passWord = "";
        uploadPath = "";
    }
    .....
};
```

### Description

Ftp Client configuration parameters.

### Parameters

| Parameter                | Description                        |
|--------------------------|------------------------------------|
| public boolean enable    | Whether to enable Ftp              |
| public boolean pasvMode  | Whether to connect in passive mode |
| public short remotePort  | Remote port                        |
| public String remoteHost | Remote IP                          |
| public String userName   | User name                          |
| public String passWord   | Password                           |
| public String uploadPath | Upload path                        |

## 5.31 RS485\_CONFIG

### RS485\_CONFIG Class Description

```
public class RS485_CONFIG {  
    public byte rS485WorkModeID;  
    public byte bitrateID;  
    public byte databitID;  
    public byte checksumID;  
    public int stopbitID;  
    public byte rS485ScreenID;  
    public short enableWhitelist;  
    public short enableBlacklist;  
    public String whitelistBannerA;  
    public String whitelistBannerB;  
    public String blacklistBannerA;  
    public String blacklistBannerB;  
  
    public RS485_CONFIG() {  
        whitelistBannerA = new String();  
        whitelistBannerB = new String();  
        blacklistBannerA = new String();  
        blacklistBannerB = new String();  
    }  
    .....  
};
```

#### Description

485 configuration parameters.

#### Parameters

| Parameter                   | Description                                                                                                    |
|-----------------------------|----------------------------------------------------------------------------------------------------------------|
| public byte rS485WorkModeID | 485 working mode:<br>0-system LED display,<br>1-offline identification result output,<br>2-offline LED display |
| public byte bitrateID       | Please refer to the help document for bit rate ID and value meaning.                                           |

|                                             |                                                                                                             |
|---------------------------------------------|-------------------------------------------------------------------------------------------------------------|
| <code>public byte databitID</code>          | Data bit ID,<br>0-5 valid data bits,<br>1-6 valid data bits,<br>2-7 valid data bits,<br>3-8 valid data bits |
| <code>public byte checksumID</code>         | Checksum ID,<br>0-no check,<br>1-odd check,<br>2-even check,<br>3-mark check,<br>4-space check              |
| <code>public int stopbitID</code>           | Stop bit ID,<br>0-- no stop bit,<br>1-1 stop bit,<br>2-2 stop bit                                           |
| <code>public byte rS485ScreenID</code>      | RS485 screen model, currently only 0                                                                        |
| <code>public short enableWhitelist</code>   | Whether to enable whitelisting                                                                              |
| <code>public short enableBlacklist</code>   | Whether to enable blacklisting                                                                              |
| <code>public String whitelistBannerA</code> | Whitelist character 1, maximum length is 24                                                                 |
| <code>public String whitelistBannerB</code> | White list character 2, maximum length is 24                                                                |
| <code>public String blacklistBannerA</code> | Blacklist character 1, maximum length is 24                                                                 |
| <code>public String blacklistBannerB</code> | Blacklist character 2, maximum length is 24                                                                 |

## 5.32RS485\_DATA

### RS485\_DATA Class Description

```
public class RS485_DATA {  
    public byte destAddr;  
    public byte command;  
    public byte rollingCode;
```

```
public byte devType;
public short dataLen;
public char[] dataBuf;

.....

};
```

### Description

RS485 transparent transmission data.

### Parameters

| Parameter               | Description                                                |
|-------------------------|------------------------------------------------------------|
| public byte destAddr    | 485 Address, 0-Voice and Display, 1-Gateway                |
| public byte command     | Command                                                    |
| public byte rollingCode | Rolling code, 1-new packet, 0-duplicate packet             |
| public byte devType     | Type of equipment, 1-display screen, 2-voice board, 3-gate |
| public short dataLen    | Data Length, Length of DataBuf                             |
| public char[] dataBuf   | Data                                                       |

## 5.33AUXCAMERASTATUS

### AuxCameaStatus Class Description

```
public class AuxCameraStatus {
    public long ip;
    public int status;
    .....
};
```

### Description

Auxiliary camera status.

### Parameters

| Parameter         | Description                                   |
|-------------------|-----------------------------------------------|
| public long ip    | Auxiliary camera IP                           |
| public int status | Secondary camera status: linked, disconnected |

5.34DEVSN

DEVSN Class Description

```
public class DEVSN {
    public String szDevSN;
    public String szEncContent;
    public long szLen;
    public String szLicense;
    .....
};
```

Description

Device encryption information.

Parameters

| Parameter                  | Description                                |
|----------------------------|--------------------------------------------|
| public String szDevSN      | serial number                              |
| public String szEncContent | Encrypt the contents of the chip           |
| public long szLen          | Permissible length                         |
| public String szLicense    | Licensed content, memory allocated outside |

### 5.35NALPRCFG

NAlprCfg **Class Description**

```
public class NAlprCfg implements Cloneable {
    public int region;
    public int[] country;
    public int minimumheight;
    public int maximumheight;
    public int mode;
    public int capturecount;
    public int timeout;
    public NAlprCfg() {
        country = new int[3];
        for (int i = 0; i < 3; i++) {
            country[i] = -1;
        }
    }
    .....
};
```

**Description**

Identification parameter class.

**Parameters**

| Parameter            | Description                                                                                                          |
|----------------------|----------------------------------------------------------------------------------------------------------------------|
| public int region    | Region<br>1: Europe and Morocco<br>2: South and Central America<br>3: North and Central America<br>4: Asia 5: Africa |
| public int[] country | Country code, please refer to the help document for specific values, up to three                                     |

|                                       |                                                                                                                                |
|---------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| <code>public int minimumheight</code> | The lowest height (15-50), in pixels, prevents the algorithm from starting recognition when the car is very far or very close. |
| <code>public int maximumheight</code> | The lowest height (15-50), in pixels, prevents the algorithm from starting recognition when the car is very far or very close. |
| <code>public int mode</code>          | Working mode: 1: Trigger 2: Free-Flow                                                                                          |
| <code>public int capturecount</code>  | Number of captured photos (1-10), the algorithm will capture the set number of photos to identify when identifying.            |
| <code>public int timeout</code>       | Timeout, in milliseconds                                                                                                       |

## 5.36COUNTRY\_CODE

### COUNTRY\_CODE Class Description

#### Parameters

| Parameter | Description  |
|-----------|--------------|
| CA = 1    | Canada       |
| US = 1    | US           |
| KZ = 7    | Kazakhstan   |
| RU = 7    | Russia       |
| EG = 20   | Egypt        |
| ZA = 27   | South Africa |
| GR = 30   | Greece       |
| NL = 31   | Netherlands  |
| BE = 32   | Belgium      |

|         |             |
|---------|-------------|
| FR = 33 | France      |
| ES = 34 | Spain       |
| IT = 39 | Italy       |
| UK = 44 | UK          |
| RO = 40 | Romania     |
| CH = 41 | Switzerland |
| AT = 43 | Austria     |
| DK = 45 | Denmark     |
| SE = 46 | Sweden      |
| NO = 47 | Norway      |
| PL = 48 | Poland      |
| PE = 51 | Peru        |
| MX = 52 | Mexico      |
| CU = 53 | Cuba        |
| AR = 54 | Argentina   |
| BR = 55 | Brazil      |
| CL = 56 | Chile       |
| CO = 57 | Colombia    |
| VE = 58 | Venezuela   |
| MY = 60 | Malaysia    |
| AU = 61 | Australia   |
| ID = 62 | Indonesia   |

|          |                |
|----------|----------------|
| PH = 63  | Philippines    |
| NZ = 64  | New Zealand    |
| SG = 65  | Singapore      |
| TH = 66  | Thailand       |
| JP = 81  | Japan          |
| KR = 82  | Korea          |
| VN = 84  | Vietnam        |
| CN = 86  | China          |
| TR = 90  | Turkey         |
| IND = 91 | India          |
| PK = 92  | Pakistan       |
| AF = 93  | Afghanistan    |
| LK = 94  | Sri Lanka      |
| MM = 95  | Myanmar        |
| IR = 98  | Iran           |
| MA = 210 | Morocco        |
| EH = 210 | Western Sahara |
| DZ = 213 | Algeria        |
| TN = 216 | Tunisia        |
| LY = 218 | Libya          |
| GM = 220 | Gambia         |
| SN = 221 | Senegal        |

|          |                            |
|----------|----------------------------|
| MR = 222 | Mauritania                 |
| ML = 223 | Mali                       |
| GN = 224 | Guinea                     |
| CI = 225 | Côte d'Ivoire              |
| BF = 226 | Burkina Faso               |
| NE = 227 | Niger                      |
| TG = 228 | Togo                       |
| BJ = 229 | Benin                      |
| MU = 230 | Mauritius                  |
| LR = 231 | Liberia                    |
| SL = 232 | Sierra Leone               |
| GH = 233 | Ghana                      |
| NG = 234 | Nigeria                    |
| TD = 235 | Chad                       |
| CF = 236 | Central African Republic   |
| CM = 237 | Cameroon                   |
| CV = 238 | Cape Verde                 |
| ST = 239 | Sao Tome and Principe      |
| GQ = 240 | Equatorial Guinea          |
| GA = 241 | Gabon                      |
| CD = 242 | Congo                      |
| CG = 242 | Democratic Republic of the |

|          |               |
|----------|---------------|
|          | Congo         |
| AO = 244 | Angola        |
| GW = 245 | Guinea-Bissau |
| SD = 249 | Sudan         |
| RW = 250 | Rwanda        |
| ET = 251 | Ethiopia      |
| SO = 252 | Somalia       |
| DJ = 253 | Kyrgyzstan    |
| KE = 254 | Kenya         |
| TZ = 255 | Tanzania      |
| UG = 256 | Uganda        |
| BI = 257 | Burundi       |
| MZ = 258 | Mozambique    |
| ZM = 260 | Zambia        |
| MG = 261 | Madagascar    |
| RE = 262 | Reunion       |
| ZW = 263 | Zimbabwe      |
| NA = 264 | Namibia       |
| MW = 265 | Malawi        |
| LS = 266 | Lesotho       |
| BW = 267 | Botswana      |
| SZ = 268 | Swaziland     |

|           |            |
|-----------|------------|
| KM = 269  | Comoros    |
| YT = 269  | Mayotte    |
| ER = 291  | Eritrea    |
| AW = 297  | Aruba      |
| HU = 336  | Hungary    |
| DE = 349  | Germany    |
| PT = 351  | Portugal   |
| LU = 352  | Luxembourg |
| IE = 353  | Ireland    |
| IS = 354  | Iceland    |
| AL = 355  | Albania    |
| MT = 356  | Malta      |
| CYP = 357 | Cyprus     |
| FI = 358  | Finland    |
| BG = 359  | Bulgaria   |
| LT = 370  | Lithuania  |
| LV = 371  | Latvia     |
| EE = 372  | Estonia    |
| MD = 373  | Moldova    |
| AM = 374  | Armenia    |
| BY = 375  | Belarus    |
| AD = 376  | Andorra    |

|          |                        |
|----------|------------------------|
| UA = 380 | Ukraine                |
| RS = 381 | Serbia                 |
| ME = 382 | Montenegro             |
| HR = 385 | Croatia                |
| SI = 386 | Slovenia               |
| BA = 387 | Bosnia and Herzegovina |
| MK = 389 | Macedonia              |
| VA = 396 | Vatican                |
| CZ = 420 | Czech Republic         |
| SK = 421 | Slovakia               |
| BZ = 501 | Belize                 |
| GT = 502 | Guatemala              |
| SV = 503 | El Salvador            |
| HN = 504 | Honduras               |
| NI = 505 | Nicaragua              |
| CR = 506 | Costa Rica             |
| PA = 507 | Panama                 |
| HT = 509 | Haiti                  |
| GP = 590 | Guadeloupe             |
| BO = 591 | Bolivia                |
| GY = 592 | Guyana                 |
| EC = 593 | Ecuador                |

|          |                      |
|----------|----------------------|
| GF = 594 | French Guiana        |
| PY = 595 | Paraguay             |
| MQ = 596 | Martinique           |
| SR = 597 | Suriname             |
| UY = 598 | Uruguay              |
| AN = 599 | Netherlands Antilles |
| GU = 671 | Guam                 |
| TL = 670 | East Timor           |
| BN = 673 | Brunei Darussalam    |
| NR = 674 | Nauru                |
| PG = 675 | Papua New Guinea     |
| TO = 676 | Tonga                |
| SB = 677 | Solomon Islands      |
| VU = 678 | Vanuatu              |
| FJ = 679 | Fiji                 |
| CK = 682 | Cook Islands         |
| WS = 685 | Samoa                |
| KI = 686 | Kiribati             |
| NC = 687 | New Caledonia        |
| TV = 688 | Tuvalu               |
| PF = 689 | French Polynesia     |
| FM = 691 | Micronesia           |

|          |                      |
|----------|----------------------|
| MH = 692 | Marshall Islands     |
| KP = 850 | North Korea          |
| HK = 852 | Hong Kong            |
| MO = 853 | Macau                |
| KH = 855 | Cambodia             |
| LA = 856 | Laos                 |
| BD = 880 | Bangladesh           |
| TW = 886 | Taiwan               |
| MV = 961 | Maldives             |
| LB = 962 | Lebanon              |
| JO = 963 | Jordan               |
| SY = 964 | Syria                |
| IQ = 965 | Iraq                 |
| KW = 966 | Kuwait               |
| SA = 967 | Saudi Arabia         |
| YE = 968 | Yemen                |
| OM = 969 | Oman                 |
| PS = 970 | Palestine            |
| AE = 972 | United Arab Emirates |
| IL = 973 | Israel               |
| BH = 974 | Bahrain              |
| QA = 975 | Qatar                |

|           |                        |
|-----------|------------------------|
| BT = 976  | Bhutan                 |
| MN = 977  | Mongolia               |
| NP = 978  | Nepal                  |
| TJ = 992  | Tajikistan             |
| TM = 993  | Turkmenistan           |
| AZ = 994  | Azerbaijan             |
| GE = 995  | Georgia                |
| KG = 996  | Djibouti               |
| UZ = 998  | Uzbekistan             |
| BB = 1809 | Barbados               |
| BS = 1809 | Bahamas                |
| PR = 1809 | Puerto Rico            |
| DO = 1809 | Dominican Republic     |
| GD = 1809 | Grenada                |
| LC = 1809 | Saint Lucia            |
| VC = 1809 | Saint Vincent          |
| TT = 1809 | Trinidad and Tobago    |
| JM = 1809 | Jamaica                |
| VG = 1809 | British Virgin Islands |

## 5.37XPOINT

### XPOINT Class Description

```
public class XPoint implements Cloneable {  
  
    public int x;  
    public int y;  
    .....  
};
```

#### Description

coordinate position.

#### Parameters

| Parameter    | Description        |
|--------------|--------------------|
| public int x | X-axis coordinates |
| public int y | Y-axis coordinates |

## 5.38LICENSE\_PLATE

### LICENSE\_PLATE Class Description

```
public class LICENSE_PLATE {  
    public String createPicTime;  
    public long processTime;  
    public int plateNum;  
    public PLATERESULT[] pPlate;  
  
    public LICENSE_PLATE() {
```

```

        createPicTime = new String();
        processTime = -1;
        plateNum = -1;
        pPlate = new PLATERESULT[4];
        for (int i = 0; i < 4; i++) {
            pPlate[i] = new PLATERESULT();
        }
    }

    .....
};

```

## Description

Identification of License Plate Information Definition.

## Parameters

| Parameter                   | Description                                                                                      |
|-----------------------------|--------------------------------------------------------------------------------------------------|
| public String createPicTime | <b>Out:</b> trigger time. format: 20161012163417050, i.e. 16: 34: 17: 050 ms on October 12, 2016 |
| public long processTime     | <b>Out:</b> Processing Time of Current Picture                                                   |
| public int plateNum         | <b>Out:</b> Number of plates currently recognized                                                |
| public PLATERESULT[] pPlate | You can identify up to 4 license plates at a time.                                               |

5.39XDATE

XDate Class Description

```
public class XDate {

    public short nYear;
    public byte nMonth;
    public byte nDay;

    public XDate() {
    }

    public XDate(short nYear, byte nMonth, byte nDay) {
        this.nYear = nYear;
        this.nMonth = nMonth;
        this.nDay = nDay;
    }

    .....
};
```

Description

Date.

Parameters

| Parameter          | Description         |
|--------------------|---------------------|
| public short nYear | Time format: 2016   |
| public byte nMonth | Values 1,2, ..., 12 |
| public byte nDay   | Values 1,2, ..., 31 |

## 6 Appendixes

### Appendix 1 – Error Code

| Error type definition           | Error code | Remarks                                  |
|---------------------------------|------------|------------------------------------------|
| OS_NoErr = (OS_Error)0          | 0          | Successful operation                     |
| OS_BadURLFormat=(OS_Error)-1000 | -1000      | error code                               |
| OS_NotEnoughSpace               | -999       | Not enough space                         |
| OS_CannotOpenSocket             | -998       | Failed to create socket                  |
| OS_CannotConnect                | -997       | Device disconnected                      |
| OS_InvalidArgument              | -996       | Invalid data                             |
| OS_CannotSendData               | -995       | Failed to send data                      |
| OS_TRYAGAIN                     | -994       | The server is too busy, please try again |
| OS_OperationErr                 | -993       | Function call error                      |
| OS_CannotOpenChn                | -992       | Failed to open channel                   |
| OS_VideoNotOpen                 | -991       | Video channel is not open                |
| OS_ConnectExist                 | -990       | Connection already exists                |
| OS_PullCannotConn               | -989       | Pull cannot be connected                 |
| OS_NotEnoughMem                 | -988       | Not enough storage                       |
| OS_FileReadErr                  | -987       | File read error                          |
| OS_FileWriteErr                 | -986       | File write error                         |

|                          |      |                                                        |
|--------------------------|------|--------------------------------------------------------|
| OS_CannotRecvData        | -985 | Read data timeout                                      |
| OS_SetTimeFail           | -984 | Set time failed                                        |
| OS_DevCannotSupport      | -983 | The device does not support this command or interface. |
| OS_ExceedDevCapacity     | -982 | Exceeding device storage capacity                      |
| OS_PlaySoundFail         | -981 | Failed to play sound                                   |
| OS_StarVoiceFail         | -980 | Failed to start intercom                               |
| OS_SendVoiceFail         | -979 | Failed to send voice                                   |
| OS_InputAudioFail        | -978 | Input Audio defeat                                     |
| OS_DevNotSupportP2P      | -977 | The device does not support P2P                        |
| OS_JniError              | -976 | JNI error                                              |
| OS_JsonParseError        | -975 | Json parsing error                                     |
| OS_InitThirdSDKFaild     | -974 | Calling the third-party SDK failed                     |
| OS_GetHandleIDFail       | -973 | HandleID has been used up                              |
| OS_DateFormatErr         | -972 | Error in date format, illegal date format              |
| OS_InvalidDate           | -971 | Invalid date (start date is greater than end date)     |
| OS_InvalidData           | -970 | Invalid Data                                           |
| OS_VideoChannelException | -969 | Video channel anomaly                                  |

## Appendix 2– Voice board and display communication protocol

The content in this document is mainly used for the serial communication protocol between the camera and the expansion board, and follows the original embedded master-slave RS485 protocol standard.

### Note:

|                |   |
|----------------|---|
| Display Screen | 1 |
| Voice board    | 2 |
| Device address |   |
| Display screen | 0 |
| Voice board    | 0 |

### Communication Data Format Definition

|         | Start  | Target  | Command | Total length | Parameters and | CRC16       |
|---------|--------|---------|---------|--------------|----------------|-------------|
|         | tag    | Address | Word    | of Data      | Data           | Calibration |
| Content | 0xaa   |         |         |              |                |             |
| Length  | 1 Byte | 1 Byte  | 1 Byte  | 2 Bytes      | N Bytes        | 2 Bytes     |

**Start tag:** Fixed to 0xaa, binary is 1011010, on the circuit as a neat peak, valley and valley waveform, easy to distinguish from interference data.

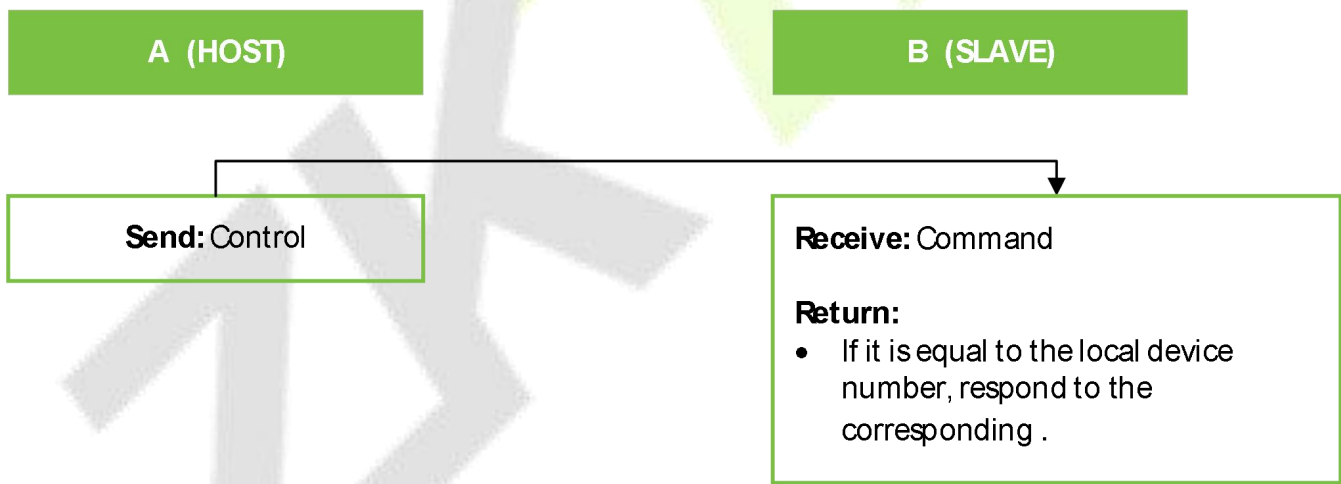
**Target address:** Refers to the device ID number received by this data stream (i.e., the 485 address of the slave). In principle, the ID number of all devices in a network or system cannot be repeated; the address of the broadcast command is 0xff, and all received are executed.

**Total length of data:** The sum of the parameters and data length of this communication.

**CRC calibration:** The result value of the CRC16 calculation for content from the beginning of the destination address to the end of the parameters and data.

Communication Procedure

Host polling slave



Communication Command

Host sends command format

|           |                |              |                      |                     |                   |
|-----------|----------------|--------------|----------------------|---------------------|-------------------|
| Start Tag | Target Address | Command Word | Total length of Data | Parameters and Data | CRC16 Calibration |
|-----------|----------------|--------------|----------------------|---------------------|-------------------|

|         |        |        |        |         |        |         |
|---------|--------|--------|--------|---------|--------|---------|
| Content | 0xaa   |        | 5      | 1       | 0      |         |
| length  | 1 Byte | 1 Byte | 1 Byte | 2 Bytes | N Byte | 2 Bytes |

**Command Format:**

The Parameters and the Data format are as follows,

| Rolling code | Equipment type | data   |
|--------------|----------------|--------|
| 1 Byte       | 1 Byte         | N Byte |

## Instruction Slave Type and Data Type Definition

| Name    | Equipment Type                           | Device Address                          | Rolling Code                         |
|---------|------------------------------------------|-----------------------------------------|--------------------------------------|
| Content | Display : 1<br>Voice board: 2<br>Gate: 3 | Display: 0<br>Voice board: 0<br>Gate: 1 | 1: New packet<br>0: Duplicate packet |
| Length  | 1 Byte                                   | 1 Byte                                  | 1 Byte                               |

**Display Screen:**

In the data content, the Chinese character is GB2313, the character is ASCII, and the characters are 0D and 0A. If there are 0D or 0A characters, the front and back contents are displayed on the line. If the content exceeds the display width, it will be displayed in scroll mode. For example, if the data is 0D 0A, the uplink content is empty, and all data is displayed on the downlink.

**Test case:****Voice Board**

```
AA 00 05 04 00 01 02 30 31 d7 5c //broadcast"0","1"
```

**Display Screen**

```
AA 00 05 05 00 01 01 41 42 43 39 98 //Send a new package when the content ABC.
```

## Appendix 3 - Voice Segment Definition Description

Voice segment code [segment code is ASCLL code]

| Voice content                               | Segment code |
|---------------------------------------------|--------------|
| 0                                           | 0            |
| 1                                           | 1            |
| 2                                           | 2            |
| 3                                           | 3            |
| 4                                           | 4            |
| 5                                           | 5            |
| 6                                           | 6            |
| 7                                           | 7            |
| 8                                           | 8            |
| 9                                           | 9            |
| ten                                         | A            |
| hundred                                     | B            |
| drop                                        | C            |
| Please charge                               | D            |
| yuan                                        | E            |
| Pro-car, please press the card              | F            |
| Please pay for the card in the sensor area. | G            |
| welcome                                     | H            |

|                                                       |   |
|-------------------------------------------------------|---|
| The parking space is full                             | I |
| Card issuing machine lacks card                       | J |
| The system is processing                              | K |
| Please wait                                           | L |
| Invalid card                                          | M |
| This card has expired                                 | N |
| This card has entered the market                      | O |
| This card has no entry record                         | P |
| This card is about to expire, please recharge in time | Q |
| This card has not expired                             | R |
| This card has been reported lost                      | S |
| Validity period                                       | T |
| month                                                 | U |
| day                                                   | V |
| hour                                                  | W |
| Minute                                                | X |
| welcome                                               | Y |
| I wish you a safe journey.                            | Z |
| Please insert the card into the recycling card mouth  | a |
| sorry                                                 | b |

|                                                     |   |
|-----------------------------------------------------|---|
| Please charge first and then play                   | c |
| Balance                                             | d |
| thousand                                            | e |
| Please contact your administrator                   | f |
| Please pay the fee first and then enter the market. | g |
| week                                                | h |
| Please keep the parking card                        | i |
| day                                                 | j |
| year                                                | k |
| Ten thousand20161127                                | l |

## Appendix 4 - RS485 Communication Protocol

### RS485 Communication Protocol between Camera and Two-color screen (voice and display)

The content in this document is mainly used for the 485-communication protocol between the camera and the display and voice, and follows the original embedded master-slave RS485 protocol standard.

#### Note:

|                                  |    |
|----------------------------------|----|
| Display and Voice Device Address | 66 |
| Host Address                     | 0  |
| Display Screen                   |    |
| The First row of the Data Type   | 35 |
| The Second row of the Data Type  | 36 |

### Communication Data Format Definition

|         | Start Tag | Target Address | Serial Number | Command Word | Total length of Data | Parameters and data | CRC16 Check |
|---------|-----------|----------------|---------------|--------------|----------------------|---------------------|-------------|
| content | 0xaa      |                |               |              |                      |                     |             |
| length  | 1 Byte    | 1 Byte         | 2 Bytes       | 1 Byte       | 2 Bytes              | N Bytes             | 2 Bytes     |

**Start tag:** Fixed to 0xaa, binary is 1011010, on the circuit as a neat peak, valley and valley waveform, easy to distinguish from interference data.

**Serial number:** When the host computer sends data, it automatically increments by 1, and the lower computer responds with the same serial number.

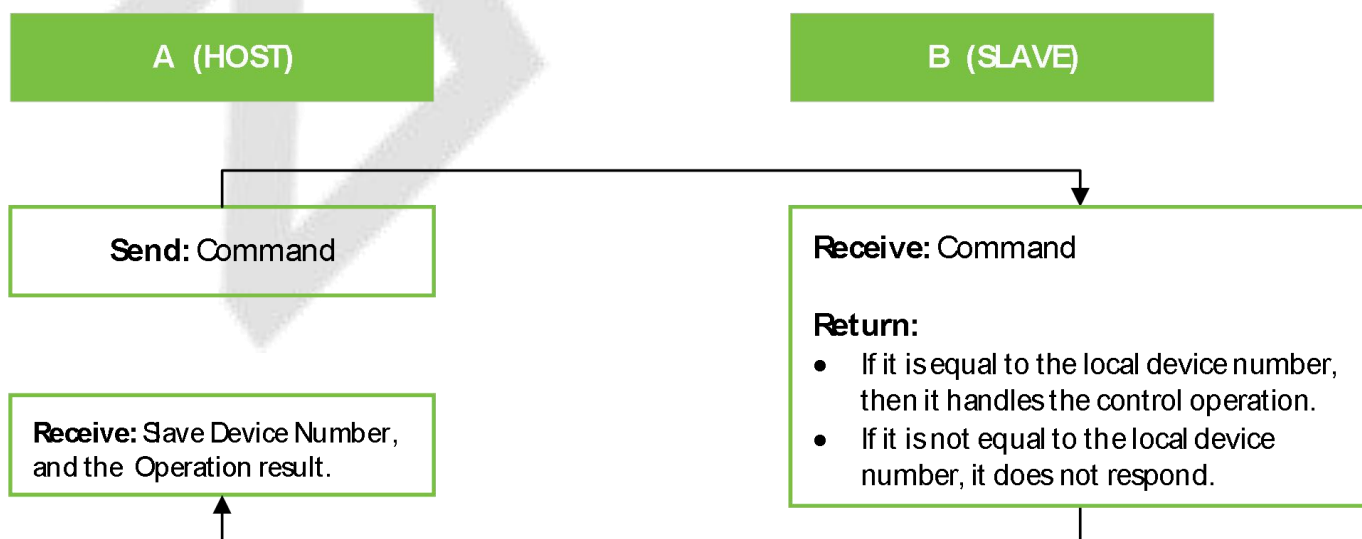
**Target address:** Refers to the device ID number received by this data stream (that is, the 485 address of the slave). In principle, the ID number of all devices in a network or system cannot be repeated; the address of the broadcast command is 0xff, and all received are executed;

**Total length of data:** The sum of the parameters and data length of this communication.

**CRC check:** The result value of CRC16 calculation for the content from the "target address" to the end of "parameters and data".

## Communication Procedure

Host-Slave Control:



## Communication Command

### Setting Country Code 4

|         | Start Tag | Target Address | Serial Number | Command Word | Total length of Data | Parameters and Data | CRC16 Check |
|---------|-----------|----------------|---------------|--------------|----------------------|---------------------|-------------|
| content | 0xaa      | 66             | 1             | 170          |                      | National code       |             |
| length  | 1 Byte    | 1 Byte         | 2 Bytes       | 1 Byte       | 2 Bytes              | N Bytes             | 2 Bytes     |

#### Command Format:

Cmd=170, datalen=2, data[0]=National code.

### Command – 171: Instruction setting data moving direction

|         | Start tag | target address | serial number | Command word | Total length of data | Parameters and data | CRC16 check |
|---------|-----------|----------------|---------------|--------------|----------------------|---------------------|-------------|
| Content | 0xaa      | 66             | 1             | 171          |                      |                     |             |
| Length  | 1 Byte    | 1 Byte         | 2 Bytes       | 1 Byte       | 2 Bytes              | N Bytes             | 2 Bytes     |

#### Command Format:

Cmd=171, datalen=1, data[0]=moving direction.

- The direction of movement of the first line: lower four, 1-4 (up, down, left and right).
- The moving direction of the second line: high four, 1-4 (up, down, left and right).

## Command - 172 Command setting data movement speed

|         | Start Tag | Target Address | Serial Number | Command Word | Total length of Data | Parameters and Data | CRC16 Check |
|---------|-----------|----------------|---------------|--------------|----------------------|---------------------|-------------|
| content | 0xaa      | 66             | 1             | 172          |                      |                     |             |
| length  | 1 Byte    | 1 Byte         | 2 Bytes       | 1 Byte       | 2 Bytes              | N Bytes             | 2 Bytes     |

### Command Format:

Cmd=172, datalen=1, data[0]=moving speed.

**Movement speed:** In milliseconds (50-1000).

## Command – 173

The command sets the default display data (the second line cannot be set; the default is the clock)

|         | Start Tag | Target Address | Serial Number | Command word | Total length of data | Parameters and data | CRC16 Check |
|---------|-----------|----------------|---------------|--------------|----------------------|---------------------|-------------|
| content | 0xaa      | 66             | 1             | 173          |                      |                     |             |
| length  | 1 Byte    | 1 Byte         | 2 Bytes       | 1 Byte       | 2 Bytes              | N Bytes             | 2 Bytes     |

| Data 1 type | Data 1 length | Data 1 content | ... | Data N type | Data N length | Data N content |
|-------------|---------------|----------------|-----|-------------|---------------|----------------|
| 1 Byte      | 2 Bytes       | Data 1 length  | ... | 1 Byte      | 2 Bytes       | Data N length  |

### Command Format:

Cmd=173, datalen=N, data[0]=data.

The Parameters and the Data format are as follows,

|         | Data type              | Data length | Data    | Data type              | Data length | Data    |
|---------|------------------------|-------------|---------|------------------------|-------------|---------|
| Content | First row of data (35) | N           | DATA    | First row of data (36) | N           | DATA    |
| Length  | 1 Byte                 | 2 Bytes     | N Bytes | 1 Byte                 | 2 Bytes     | N Bytes |

**DATA Content:** The first byte, the color of the data (0: red, 1: green), followed by the content that needs to be displayed.

### Command - 174 instructions to send real-time displayed data.

| Start Tag | Target Address | Serial Number | Command Word | Total length of Data | Parameters and Data | CRC16 Check |
|-----------|----------------|---------------|--------------|----------------------|---------------------|-------------|
|-----------|----------------|---------------|--------------|----------------------|---------------------|-------------|

|         |        |        |         |        |         |         |         |
|---------|--------|--------|---------|--------|---------|---------|---------|
| Content | 0xaa   | 66     | 1       | 174    |         |         |         |
| Length  | 1 Byte | 1 Byte | 2 Bytes | 1 Byte | 2 Bytes | N Bytes | 2 Bytes |

| Data 1 type | Data 1 length | Data 1 content | ... | Data N type | Data N length | Data N content |
|-------------|---------------|----------------|-----|-------------|---------------|----------------|
| 1 Byte      | 2 Bytes       | Data 1 length  | ... | 1 Byte      | 2 Bytes       | Data N length  |

#### Command Format:

Cmd=174, datalen=N, data[0]=data.

The Parameters and the Data format are as follows,

|         | Data type              | Data length | Data    | Data type               | Data length | Data    |
|---------|------------------------|-------------|---------|-------------------------|-------------|---------|
| Content | First row of data (35) | N           | DATA    | Second row of data (36) | N           | DATA    |
| Length  | 1 Byte                 | 2 Bytes     | N Bytes | 1 Byte                  | 2 Bytes     | N Bytes |

**DATA Content:** The first byte, the color of the data (0: red, 1: green), followed by the content that needs to be displayed.

**Command – 175: Command to set the volume of the voice board**

|         | Start Tag | Target Address | Serial Number | Command Word | Total length of Data | Parameters and Data | CRC16 Check |
|---------|-----------|----------------|---------------|--------------|----------------------|---------------------|-------------|
| Content | 0xaa      | 66             | 1             | 175          |                      |                     |             |
| Length  | 1 Byte    | 1 Byte         | 2 Bytes       | 1 Byte       | 2 Bytes              | N Bytes             | 2 Bytes     |

**Command Format:**

Cmd=175, Datalen=1, Data[0]=volume, Volume size: 0-31.

**Command – 176: Command to interrupt the last voice broadcast**

|         | Start tag | Target Address | Serial Number | Command Word | Total length of Data | Parameters and Data | CRC16 Check |
|---------|-----------|----------------|---------------|--------------|----------------------|---------------------|-------------|
| Content | 0xaa      | 66             | 1             | 176          |                      |                     |             |
| Length  | 1 Byte    | 1 Byte         | 2 Bytes       | 1 Byte       | 2 Bytes              | N Byte              | 2 Bytes     |

**Command Format:**

Cmd=176, datalen=1, data[0]=

**Command – 177: Command to send voice data.**

|  | Start Tag | Target | Serial | Command | Total | Parameters | CRC16 |
|--|-----------|--------|--------|---------|-------|------------|-------|
|--|-----------|--------|--------|---------|-------|------------|-------|

|         |        | Address | Number  | Word   | length of Data | and Data | Check   |
|---------|--------|---------|---------|--------|----------------|----------|---------|
| content | 0xaa   | 66      | 1       | 177    |                | data     |         |
| length  | 1 Byte | 1 Byte  | 2 Bytes | 1 Byte | 2 Bytes        | N Bytes  | 2 Bytes |

**Command Format:**

Cmd=177, datalen=N, data=voice data.

**Command – 178: Command to set the time.**

|         | Start Tag | Target Address | Serial Number | Command Word | Total length of Data | Parameters and Data | CRC16 Check |
|---------|-----------|----------------|---------------|--------------|----------------------|---------------------|-------------|
| Content | 0xaa      | 66             | 1             | 178          |                      | Data                |             |
| Length  | 1 Byte    | 1 Byte         | 2 Bytes       | 1 Byte       | 2 Bytes              | N Bytes             | 2 Bytes     |

**Command Format:**

Cmd=178, datalen=N, data=time.

**Time Format :** Seconds, days, months, years, time and day are divided by 0, and the data is as follows: 21 32 14 00 21 09 17.

**Command – 179: Command to Query Version Number**

|         | Start Tag | Target Address | Serial Number | Command Word | Total length of Data | Parameters and Data | CRC16 Check |
|---------|-----------|----------------|---------------|--------------|----------------------|---------------------|-------------|
| Content | 0xaa      | 66             | 1             | 179          |                      | data                |             |
| Length  | 1 Byte    | 1 Byte         | 2 Bytes       | 1 Byte       | 2 Bytes              | N Bytes             | 2 Bytes     |

**Command Format:**

Cmd=179, datalen=N, data=0。

**Command – 180: Command to Save Default Parameter Values**

|         | Start Tag | Target Address | Serial Number | Command Word | Total length of Data | Parameters and Data | CRC16 Check |
|---------|-----------|----------------|---------------|--------------|----------------------|---------------------|-------------|
| Content | 0xaa      | 66             | 1             | 180          |                      | data                |             |
| Length  | 1 Byte    | 1 Byte         | 2 Bytes       | 1 Byte       | 2 Bytes              | N Byte              | 2 Bytes     |

**Command Format:**

Cmd=180, datalen=N, data=0

**Command – 181: Command to set the display duration**

|         | Start Tag | Target Address | Serial Number | Command Word | Total length of Data | Parameters and Data | CRC16 Check |
|---------|-----------|----------------|---------------|--------------|----------------------|---------------------|-------------|
| Content | 0xaa      | 66             | 1             | 181          |                      | data                |             |
| Length  | 1 Byte    | 1 Byte         | 2 Bytes       | 1 Byte       | 2 Bytes              | N Bytes             | 2 Bytes     |



**Command Format:**

Cmd=181, datalen=N, data=0,duration: 5-255second.

**Command – 200: Instructions executed successfully**

|         | Start Tag | Target Address | Serial number | Command Word | Total length of Data | Parameters and Data | CRC16 Check |
|---------|-----------|----------------|---------------|--------------|----------------------|---------------------|-------------|
| Content | 0xaa      | 0              | 1             | 200          |                      |                     |             |
| Length  | 1 Byte    | 1 Byte         | 2 Bytes       | 1 Byte       | 2 Bytes              | N Bytes             | 2 Bytes     |

**Command Format:**

Cmd=200, datalen=1, data=0.

**Destination address:** The address of the display or voice.

**The slave receives the get version command reply version information to the host:**

Cmd=200, datalen=Version number length, data=version number.

**Command – 201: Instruction execution failed**

|        | Start Tag | Target Address | Serial Number | Command word | Total length of Data | Parameters and Data | CRC16 Check |
|--------|-----------|----------------|---------------|--------------|----------------------|---------------------|-------------|
| Conten | 0xaa      | 0              | 1             | 201          | 1                    | Error Code          |             |

|        |        |        |         |        |         |        |         |
|--------|--------|--------|---------|--------|---------|--------|---------|
| t      |        |        |         |        |         |        |         |
| Length | 1 Byte | 1 Byte | 2 Bytes | 1 Byte | 2 Bytes | 1 Byte | 2 Bytes |

**Command Format:**

Cmd=201, datalen=1, data[0]=Error code.

**Target address:** The address of the display screen or voice.

**Communication Command – Error Code**

| Error Code | Description                                                        |
|------------|--------------------------------------------------------------------|
| -1         | Command failed to send                                             |
| -2         | Command did not respond                                            |
| -3         | Insufficient cache required                                        |
| -4         | Decompression failed                                               |
| -5         | The length of the read data is incorrect.                          |
| -6         | The length of the decompression does not match the expected length |
| -7         | Command repetition                                                 |
| -8         | Connection not authorized                                          |
| -9         | CRC check failed                                                   |
| -10        | Data API cannot be parsed                                          |
| -11        | Parameter error                                                    |

|     |                              |
|-----|------------------------------|
| -12 | Command execution error      |
| -13 | Without this command         |
| -14 | Communication password error |

Appendix 5 - Language Code Description

| Code | Description |
|------|-------------|
| 86   | Chinese     |
| 66   | Thai        |
| 54   | Spanish     |
| 97   | Arabic      |
| 62   | Indonesian  |
|      | Chinese     |



**ZKTeco Biometrics India Pvt Ltd.**

B-1, Khata No. 31, Devasandra Industrial Estate,  
Whitefield Main Road, Beside Phoenix mall,  
Mahadevapura, Bengaluru-560048, Karnataka, India.

Phone : 080 6828 1342

Mail : [sales@zkteco.in](mailto:sales@zkteco.in)

Website : [www.zkteco.in](http://www.zkteco.in)

