

# Communication Protocol

## License Plate Recognition Central Server

Date: March 2020

Version: 3.2.3

English

Thank you for choosing our product. Please read the instructions carefully before operation. Follow these instructions to ensure that the

Copyright © 2020 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 Headquarters

**Address** ZKTeco Industrial Park, No. 26, 188 Industrial Road,  
Tangxia Town, Dongguan, China.

**Phone** +86 769 - 82109991

**Fax** +86 755 - 89602394

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

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

## About the Manual

This manual introduces the operations of License Plate Recognition Central Server Communication Protocol.

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



## Revision Record

| Reviser | Revised Date | Version | Page | Revision Contents                                                              | Effective Date |
|---------|--------------|---------|------|--------------------------------------------------------------------------------|----------------|
| Carl    | 2018-9-10    | V3.2.1  |      | Draft                                                                          |                |
| Carl    | 2018-11-14   | V3.2.2  |      | Add black and white list issued<br>Add upload records<br>Add parameters issued |                |
| Carl    | 2019-11-26   | V3.2.3  |      | Add capture, auxiliary input status<br>upload                                  |                |
|         |              |         |      |                                                                                |                |
|         |              |         |      |                                                                                |                |

# Table of Contents

|                                                         |           |
|---------------------------------------------------------|-----------|
| <b>1 COMMUNICATION INITIALIZATION</b>                   | <b>8</b>  |
| 1.1 PARAMETER DESCRIPTION                               | 11        |
| <b>2 REGISTER</b>                                       | <b>12</b> |
| 2.1 PARAMETER DESCRIPTION                               | 14        |
| 2.2 DEVICETYPE                                          | 14        |
| <b>3 DOWNLOAD CONFIGURATION PARAMETERS</b>              | <b>17</b> |
| <b>4 DOWNLOAD CACHE COMMAND</b>                         | <b>19</b> |
| 4.1 REMOTE CONTROL GATE COMMAND                         | 20        |
| 4.2 REMOTE CONTROL LED/VOICE COMMAND                    | 21        |
| 4.2.1 REMOTE CONTROL VOICE COMMAND                      | 22        |
| 4.2.2 REMOTE CONTROL LED COMMAND                        | 22        |
| 4.3 REMOTE CONTROL LCD COMMAND                          | 24        |
| 4.4 SYNCHRONOUS SERVER TIME                             | 24        |
| 4.4.1 PARAMETER DESCRIPTION:                            | 25        |
| 4.5 ADD CARS TO BLACKLIST, WHITELIST, AND FIXEDLIST     | 25        |
| 4.6 REMOVE CARS FROM BLACKLIST, WHITELIST AND FIXEDLIST | 26        |
| 4.7 DISTRIBUTED PARAMETER                               | 26        |
| 4.8 REMOTE MANUAL CAPTURE                               | 27        |
| <b>5 UPLOAD REAL-TIME LICENSE PLATE INFORMATION</b>     | <b>28</b> |
| 5.1 PARAMETER DESCRIPTION:                              | 29        |
| <b>6 UPLOAD IO PORT STATUS</b>                          | <b>31</b> |
| 6.1 PARAMETER DESCRIPTION:                              | 32        |
| <b>7 UPLOAD LIVE EVENTS</b>                             | <b>34</b> |

|                                                                                     |           |
|-------------------------------------------------------------------------------------|-----------|
| 7.1 PARAMETER DESCRIPTION.....                                                      | 36        |
| <b>8 UPLOAD REAL-TIME LICENSE PLATE INFORMATION AND RETURN CONTROL COMMAND.....</b> | <b>37</b> |
| 8.1 PARAMETER DESCRIPTION:.....                                                     | 40        |
| <b>APPENDIX.....</b>                                                                | <b>41</b> |
| APPENDIX 1.....                                                                     | 41        |
| APPENDIX 2.....                                                                     | 42        |

## 1 Communication Initialization

To check and register the device to the server in order to initiate the communication.

### Function

This connection is required for the device to communicate with the server and to share the data.

And this data is compatible with the company Push standards.

#### 1. Request

At first, register the device if it is not registered. And if the device is registered, then download the configuration parameters.

```
GET /iclock/cdata?SN=2602011070363&options=all&DeviceType=park HTTP/1.1
```

```
Host: 192.168.18.188:8080
```

```
User-Agent: iClock Proxy/1.09
```

**Connection: starting**

```
Accept: application/push
```

```
Accept-Charset: UTF-8
```

```
Accept-Language: zh-CN
```

**Remarks:**

Message body will be empty, as this request is to establish the communication.

**2. Response**

If the Device is already registered, Server will respond with the **registry details**, and the **RegistryCode**.

HTTP/1.1 200 OK

Server: C#demo/1.1

Date: Sat, 28 Dec 2013 09:14:04 GMT

Content-Type: application/push;charset=UTF-8

Content-Length: 218

**Connection: keep-alive**

registry=ok

RegistryCode=RndOT0MUMI

ServerVersion=3.1.1

ServerName=ADMS

PushVersion=3.2.0

ErrorDelay=60

RequestDelay=5

TransTimes=00:00 14:00

TransInterval=1

TransTables=User Transaction

Realtime=1

SessionID=h2kgbpcm2x

## 1.1 Parameter Description

| Attributes    | Default                          | Description                                                                  |
|---------------|----------------------------------|------------------------------------------------------------------------------|
| ServerVersion | 3.0.1                            | Server version number                                                        |
| ServerName    | ADMS                             | Server name                                                                  |
| PushVersion   | 3.0.1                            | PUSH version number supported by the server                                  |
| PushDataUrl   | /iclock/cdata?SN= <SN>           | Upload data interface URL                                                    |
| GetRequestUrl | /iclock/getrequest?SN= <SN><br>> | Request for the device command interface URL on the server.                  |
| DeviceCmdUrl  | /iclock/devcmd?SN= <SN>          | The device returns the interface URL for the result of executing the command |
| RTStatusUrl   | /iclock/rtstate?SN= <SN>         | Returns the Live Status of the Interface URL.                                |
| RTLogUrl      | /iclock/rtlog?SN= <SN>           | Interface URL for transmitting live records                                  |
| BGDataUrl     | /iclock/BGData?SN= <SN>          | Interface URL for transmitting background validation data                    |
| PhotoUrl      | /iclock/photo?SN= <SN>           | Interface URL for transmitting field photos                                  |
| ErrorDelay    | 60                               | Retry wait time (s) in case of server error                                  |
| RequestDelay  | 5                                | Request for the device command interval on the server in seconds.            |
| TransTimes    | ["00:00"/t"14:00"]               | Periodically check the time to transfer new data                             |
| TransInterval | 1                                | Checks if there is any new data to be transferred (in minutes)               |
| TransTables   | ["User"/t"Transaction"]          | Checks for the new data and uploads the data sheet                           |
| Realtime      | 1                                | The identification of new data uploaded in real time.                        |

|            |               |                                                                                                                               |
|------------|---------------|-------------------------------------------------------------------------------------------------------------------------------|
| SessionID  | 23dhsie93023k | It is the PUSH communication's session ID, which is required for the upcoming device requests in order compute the new Token. |
| TimeoutSec | 5             | It sets the network timeout to 10 seconds by default if the firmware is not configured.                                       |

## 2 Register

Registration is to obtain the verification code from the server, and the following code is the communication request sent to the server and server responds to the request.

### Function:

Automatic registration is convenient for customers for their minimum - security requirements.

### 1. Request

POST /iclock/registry?SN=2602011070363 HTTP/1.1

Cookie: token=eaa52f71d2c549f9801028a66a732b59

Host: 192.168.18.188:8080

User-Agent: iClock Proxy/1.09

**Connection: starting**

Accept: application/push

Accept-Charset: UTF-8

Accept-Language: zh-CN

Content-Type: application/push;charset=UTF-8

Content-Language: zh-CN

Content-Length: 720

DeviceType=park,~DeviceName=Alpr,FirmVer=AC Ver 5.7.6.0001 Dec 25

2013,PushVersion=3.2.3,CommType=ethernet,MaxPackageSize=2048000, InOutState=1

## 2.1 Parameter Description

Set the below input parameters on the request function for the device to get registered on the server.

| Field Name                    | Examples | Description                                                                                                                                          |
|-------------------------------|----------|------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Uniform Parameter</b>      |          |                                                                                                                                                      |
| DeviceType                    | Park     | It is the device type code, which specifies the main function of the device. The different <a href="#">DeviceType</a> are listed in the below table. |
| ~DeviceName                   | Alpr     | Name of the Device.                                                                                                                                  |
| FirmVer                       | 5.2.8    | Firmware version number of the Device.                                                                                                               |
| PushVersion                   | 3.2.3    | PUSH SDK version number of the Device.                                                                                                               |
| CommType                      | Ethernet | Communication Mode : Ethernet, WiFi, GPRS, 3G etc.                                                                                                   |
| MaxPackageSize                | 2048000  | Maximum packet size for the PUSH communication                                                                                                       |
| <b>Parking lot Parameters</b> |          |                                                                                                                                                      |
| InOutState                    | 1        | Parking Area - Entrance or Exit; 0: Entrance, 1: Exit                                                                                                |

## 2.2 DeviceType

The Code name defined for each **DeviceType** distinguishes the main functions of the device, such as Attendance Machine, Access Control Machine, Consumer Machine, and more.

The following are the types of devices and its associated code's that are currently available.

| Device Type           | Code | Description                              |
|-----------------------|------|------------------------------------------|
| Access Control Device | acc  | Access control device for access control |

| Device Type                | Code   | Description                                                          |
|----------------------------|--------|----------------------------------------------------------------------|
| Park                       | park   | Parking lot master control device                                    |
| T&A Device                 | att    | Employees Time and Attendance computing device.                      |
| Contactless Payment Device | pos    | Contact less Point of Sale Terminal device                           |
| Fingerprint Lock           | lock   | Hotel Lock, Wireless Lock, Smart Lock device                         |
| Guard Patrol Device        | patrol | A device that checks in at a fixed location and at a specified time. |

## 2. Response

Once the device is registered, the server sends the **OK** response along with the **registry code**.

HTTP/1.1 200 OK

Server: C#demo/1.1

Date: Sat, 28 Dec 2013 10:07:12 GMT

Content-Type: application/push;charset=UTF-8

Content-Length: 23

**Connection: keep-alive**

registrycode=1Ru8sNR0iw

**Remarks:**

The server generates a 10-bit random string **RegistryCode**, which is a combination of 26 characters and numbers.



### 3 Download Configuration Parameters

#### Function:

To download the communication configuration parameters from the server.

#### 1. Request

Request sent to the server for retrieving the communication configuration parameters from the server.

```
ET /iclock/push?SN=2602011070363 HTTP/1.1
```

```
Cookie: token=ffe2be42b2bf3bfffef73de804c483112,timestamp=2013-12-28-09-44-28
```

```
Host: 192.168.18.188:8080
```

```
User-Agent: iClock Proxy/1.09
```

**Connection: starting**

```
Accept: application/push
```

```
Accept-Charset: UTF-8
```

```
Accept-Language: zh-CN
```

#### 2. Response

```
HTTP/1.1 200 OK
```

Server: C#demo/1.1

Date: Sat, 28 Dec 2013 10:01:20 GMT

Content-Type: application/push;charset=UTF-8

Content-Length: 182

**Connection: keep-alive**

ServerVersion=3.1.1

ServerName=ADMS

PushVersion=3.2.0

ErrorDelay=60

RequestDelay=5

TransTimes=00:00 14:00

TransInterval= 1

TransTables=User Transaction

Realtime=1

SessionID=d8y8o75hgn

## 4 Download Cache Command

### Function:

To get the cache data from the server.

Cache data refers to the data stored in the queue cache by the server.

### 1. Request

Request sent to download the cache data from the server.

```
GET /iclock/getrequest?SN=2602011070363 HTTP/1.1
```

```
Cookie: token=51ac250e5c7cd97bcb2a8e3703b1f5c7
```

```
Host: 192.168.18.188:8080
```

```
User-Agent: iClock Proxy/1.09
```

**Connection: starting**

```
Accept: application/push
```

```
Accept-Charset: UTF-8
```

```
Accept-Language: zh-CN
```

### 2. Response

If the device does not contain any data in the server cache queue, the server responds **OK** else it returns the **Cache data**.

HTTP/1.1 200 OK

Server: C#demo/1.1

Date: Sat, 28 Dec 2013 10:48:49 GMT

Content-Type: application/push;charset=UTF-8

Content-Length: 84

**Connection: keep-alive**

C:1:CONTROL DEVICE

```
{"Auth":{"Username":"admin","Password":"123456"},"Cmd":"OpenGate","Data":""}
```

**Package data format:**

C:1:CONTROL DEVICE

```
{"Auth":{"Username":"admin","Password":"123456"},"Cmd":"OpenGate","Data":""}
```

**Remarks:**

Send the cache data in the following format: **C:\${CmdID}: Specific data.**

## 4.1 Remote Control Gate Command

C:1:CONTROL DEVICE

```
{"Auth":{"Username":"admin","Password":"123456"},"Cmd":"OpenGate","Data":""}
```

This "OpenGate" syntax need to be verified

## 4.2 Remote Control Led/Voice Command

C:1:CONTROL DEVICE

```
{"Auth":{"Username":"admin","Password":"123456"},"Cmd":"CtrLedVoice","Data":"base64<org>"}
```

### Remarks:

"<org>" is the voice display board protocol data, which requires software to encapsulate the protocol and assemble the data.

## 4.2.1 Remote Control Voice Command

C:1:CONTROL DEVICE

```
{"Auth":{"Username":"admin","Password":"123456"},"Cmd":"CtrlVoice","Data":"base64<voice list>"}
```

For example:

CONTROL DEVICE

```
{"Auth":{"Username":"admin","Password":"123456"},"Cmd":"CtrlVoice","Data":"YgA="}
```

### Remarks:

"<voice list>" – For Voice List, refer to the description of the voice segment list given in the "AlprSDK interface.pdf" appendix where two voice segments account for two bytes.

## 4.2.2 Remote Control Led Command

C:1:CONTROL DEVICE

```
{"Auth":{"Username":"admin","Password":"123456"},"Cmd":"CtrlLed","Data":{"FirstLineColor":"<color value>","FirstLineCenter":1,"FirstLineData":"<led content>","SecondLineColor":"<color value>","SecondLineCenter":1,"SecondLineData":"<led content>","ThirdLineColor":"<color value>","ThirdLineCenter":1,"ThirdLineData":"<led content>","FourthLineColor":"<color value>","FourthLineCenter":1,"FourthLineData":"<led content>"}}
```

### Example

C:1:CONTROL DEVICE

```
{"Auth":{"Username":"admin","Password":"123456"},"Cmd":"CtrlLed","Data":{"FirstLineColor":"red","FirstLineCenter":1,"FirstLineData":"YB12345 Welcome",  
"SecondLineColor":"green","SecondLineCenter":1,"SecondLineData":"xxx","ThirdLineColor":"","ThirdLineCenter":1,"ThirdLineData":"","FourthLineColor":"","FourthLineCenter":1,"FourthLineData":""}}
```

### Remarks

This function controls even the display color, alignment, display content and more.

**Color value:** Supports red, green only.

### Led content:

Currently the domestics only supports gb2312 standard characters, which needs to be unified into the issued utf-8 code, such as "Welcome to Guangdong B12345".

Utf-8 code is used uniformly. and Egyptian license plate with @ [like @+ license plate].

### Note:

The above indicates the encoding format supported by the display screen.

If the encoding format is incorrect, the display screen will display distorted codes.

## 4.3 Remote Control LCD Command

C:1:CONTROL DEVICE <TBD>

### Remarks:

This function is temporarily unavailable.

## 4.4 Synchronous Server Time

C:1:CONTROL DEVICE

```
{"Auth":{"Username":"admin","Password":"123456"},"Cmd":"SyncTime","Data":""}
```

The device receives the command and initiates the request:

GET /iclock/rtdata?SN=3383154200002&type=time HTTP/1.1

Cookie: token=af65a75608cf5b80fbb3b48f0b4df95a

Host: 192.168.213.17:8088

.....

Return:

HTTP/1.1 200 OK

Server: Apache-Coyote/1.1

Content-Type: text/plain;charset=UTF-8

Content-Length: 33

Date: Wed, 11 Jan 2017 01:30:03 GMT

DateTime=583050990,ServerTZ=+0800

#### 4.4.1 Parameter Description:

| Parameter | Description                                                                                                        |
|-----------|--------------------------------------------------------------------------------------------------------------------|
| DateTime  | The value is Greenwich time and the DateTime is converted to seconds using the <a href="#">[Appendix 2]</a> method |
| ServerTZ  | The time zone of the server (the device and the server may not be in the same time zone)                           |

#### 4.5 Add Cars to Blacklist, Whitelist, and Fixedlist

##### Add Cars to Blacklist entry

```
C:5:DATA UPDATE blacklist {"Data":[{"car_number":"AG5FT69","start_time":"2017-09-15","end_time":"2018-12-15"}]}
```

##### Add Cars to Whitelist entry

```
C:5:DATA UPDATE whitelist {"Data":[{"car_number":"AG5FT68","start_time":"2017-09-15","end_time":"2018-12-15"}]}
```

##### Add Cars to Fixedlist entry

```
C:5:DATA UPDATE fixedlist {"Data":[{"car_number":"AG5FT66","start_time":"2017-09-15","end_time":"2018-12-15"}]}
```

##### Remarks:

The license plate number is in utf-8 code, which is compatible with the following format.

```
C:5:DATA UPDATE Table name car_number=AG5FT69/tstart_time=2017-09-15/tend_time=2018-12-15
```

## 4.6 Remove Cars from Blacklist, Whitelist and Fixedlist

### Remove Cars from Blacklist entry

```
C:5:DATA DELETE blacklist {"Data":{"car_number":"AG5FT69"}}
```

### Remove Cars from Whitelist entry

```
C:5:DATA DELETE whitelist {"Data":{"car_number":"AG5FT68"}}
```

### Remove Cars from Fixedlist entry

```
C:5:DATA DELETE fixedlist {"Data":{"car_number":"AG5FT66"}}
```

### Deleting all the entries from the table

```
C:5:DATA DELETE Table name *
```

### Remarks:

The license plate number is in utf-8 code, which is compatible with the following format.

```
C:5:DATA DELETE Table name car_number=AG5FT69
```

## 4.7 Distributed Parameter

```
C:11:SET OPTIONS {"OfflineAutoOpenGate":1}
```

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

|                        |                                                                                                             |
|------------------------|-------------------------------------------------------------------------------------------------------------|
| Offline Auto Open Gate | Automatically verifies and opens the gate in offline, for the white listed vehicles and the fixed vehicles. |
|------------------------|-------------------------------------------------------------------------------------------------------------|

## 4.8 Remote Manual Capture

Currently, not available in the Mainstream Firmware

C:1:CONTROL DEVICE

```
{"Auth":{"Username":"admin","Password":"123456"},"Cmd":"ManualCapture","Data":""}
```

## 5 Upload Real-time License Plate Information

### Function:

Uploads the real-time license plate information.

### Remarks:

Real-time license plate information is pushed asynchronously.

#### 1. Request

Virtual table name: **lpr\_rtdata**

Request posted to the server with the below Real Time License Plate data.

```
POST /iclock/cdata?SN=2602011070363&table=lpr_rtdata HTTP/1.1
```

```
Host: 192.168.18.188:8080
```

```
User-Agent: iClock Proxy/1.09
```

**Connection: starting**

```
Accept: application/push
```

```
Accept-Charset: UTF-8
```

```
Accept-Language: zh-CN
```

```
Content-Type: application/push;charset=UTF-8
```

Content-Language: zh-CN

Content-Length: 98

```
{  
  
  "Seqno":101  
  
  "Time":"20180719135258778",  
  
  "ProcessTime":584,  
  
  "PlateNum":3,  
  
  "Plates":[{"LicensePlate":"AG5FT69","LetterCount":7,"Confidence":0,"Left":964,"Right":1180,"Top":795,  
    "Bottom":852,"PlateColor":50,"DoublePlate":0,"Direction":0,"CountryCode":0}, {"LicensePlate":"AYYPR0  
1","LetterCount":7,"Confidence":0,"Left":972,"Right":1187,"Top":717,"Bottom":775,"PlateColor":30,"Do  
ublePlate":0,"Direction":0,"CountryCode":0}, {"LicensePlate":"APZXD8S","LetterCount":7,"Confidence":  
0,"Left":972,"Right":1190,"Top":641,"Bottom":699,"PlateColor":30,"DoublePlate":0,"Direction":0,"Coun  
tryCode":0}],  
  
  "PlatesPhoto":"base64(binary)"  
}
```

## 5.1 Parameter Description:

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

|             |                                                                         |
|-------------|-------------------------------------------------------------------------|
| Seqno       | Package number                                                          |
| Time        | Recognition time                                                        |
| ProcessTime | Algorithm identification time (milliseconds)                            |
| PlateNum    | The license plate number                                                |
| Plates      | License plate information (utf-8 code is used for license plate number) |
| PlatesPhoto | The license plate images                                                |

## 6 Upload IO Port Status

Currently, Mainstream Firmware has not released this feature

### Function:

Upload Input/output port status to the server.

### Remarks:

Currently only two auxiliary inputs are supported.

#### 1. Request

Virtual table name: **lpr\_iostate**

POST /iclock/cdata?SN=2602011070363&table=lpr\_iostate HTTP/1.1

Host: 192.168.18.188:8080

User-Agent: iClock Proxy/1.09

**Connection: starting**

Accept: application/push

Accept-Charset: UTF-8

Accept-Language: zh-CN

Content-Type: application/push;charset=UTF-8

Content-Language: zh-CN

Content-Length: 98

```
{
  "Seqno":101
  "Time":"20180719135258778",
  "AuxInputNum":4,
  "AuxInputs":[{"ID":1, "Status":0}, {"ID":2, "Status":0}, {"ID":3, "Status":0}, {"ID":4, "Status":0}],
}
```

## 6.1 Parameter Description:

| Parameter   | Description              |
|-------------|--------------------------|
| Seqno       | Package number           |
| Time        | Recognition time         |
| AuxInputNum | Auxiliary input quantity |
| AuxInputs   | Auxiliary input state    |

| InputPosition     | ID | Status                     | Description                                                                                                         |
|-------------------|----|----------------------------|---------------------------------------------------------------------------------------------------------------------|
| AuxInput 1        | 1  | 1: Have Input; 0: No Input |                                                                                                                     |
| AuxInput 2        | 2  | 1: Have Input; 0: No Input |                                                                                                                     |
| Open to limit IO  | 3  | 1: Have Input; 0: No Input | When the Open to limit IO and Close to limit IO is 0, it indicates that the camera has not connected to the barrier |
| Close to limit IO | 4  | 1: Have Input; 0: No Input |                                                                                                                     |

|  |  |  |       |
|--|--|--|-------|
|  |  |  | gate. |
|--|--|--|-------|

| GateStatus |                                                     |
|------------|-----------------------------------------------------|
| Switch     | 3: Open to limit; 0: Close to limit; 255: No status |
| Running    | Reserve                                             |

## 7 Upload Live Events

Currently, Mainstream Firmware has not released this feature.

### Function:

Upload device live events to the server.

### Remarks:

Real-time upload when online; Free time after the Internet automatically uploaded.

#### 1. Request

```
POST /iclock/cdata?SN=2602011070363&table=rtlog HTTP/1.1
```

```
Host: 192.168.18.188:8080
```

```
User-Agent: iClock Proxy/1.09
```

**Connection: starting**

```
Accept: application/push
```

```
Accept-Charset: UTF-8
```

```
Accept-Language: zh-CN
```

```
Content-Type: application/push;charset=UTF-8
```

```
Content-Language: zh-CN
```

```
Content-Length: 98
```

```
{  
  
  "id":123456,  
  
  "time":"2018-11-14 14:59:18",  
  
  "car_number":"AG5FT69",  
  
  "card_no":327568,  
  
  "inout_state":0,  
  
  "event_addr":"192.168.107.186",  
  
  "event_type":101,  
  
  "photo_path":"20181113104939155-00-w386-t284-AG5FT69.jpg",  
  
  "is_leave":0,  
  
  "is_upload_flag":4,  
  
  "in_id":312,  
  
  "in_addr":"192.168.107.185",  
  
  "photo_data":"base64(binary)"  
}
```

## 7.1 Parameter Description

| Field            | Implication                                                                                                |
|------------------|------------------------------------------------------------------------------------------------------------|
| Id               | Record id                                                                                                  |
| time             | Current time                                                                                               |
| car_number       | License plate number of the vehicle                                                                        |
| card_no          | Wiegand card number                                                                                        |
| inout_state      | Incoming and outgoing status: 0-represents incoming, 1-represents outgoing                                 |
| event_addr       | The IP address at which the event occurs                                                                   |
| event_type       | Event code, see <a href="#">Appendix 1</a>                                                                 |
| photo_path       | Photo path                                                                                                 |
| is_leave         | Is the vehicle available (for temporary vehicle charges)                                                   |
| is_upload_flag   | Event status indicator (for temporary vehicle charges)                                                     |
| in_id            | In the case of an exit record, the entry id number (for temporary vehicle charges) is known by this field. |
| in_addr          | If it is an exit record, the entry IP will be known (for temporary vehicle charges) through this field.    |
| push_upload_flag | Push upload success flag, 1: uploaded, 0: not uploaded                                                     |
| photo_data       | Image data                                                                                                 |

## 8 Upload Real-time License Plate Information and Return Control Command

Currently, Mainstream Firmware has not released this feature.

### Function:

Upload real-time license plate information and return control command.

### Remarks:

Real-time license plate information is pushed asynchronously to the server.

#### 1. Request

Virtual table name: **lpr\_rtdata\_return\_ctrl**

```
POST /iclock/cdata?SN=2602011070363&table=lpr_rtdata_return_ctrl HTTP/1.1
```

```
Host: 192.168.18.188:8080
```

```
User-Agent: iClock Proxy/1.09
```

**Connection: starting**

```
Accept: application/push
```

```
Accept-Charset: UTF-8
```

```
Accept-Language: zh-CN
```

```
Content-Type: application/push;charset=UTF-8
```

Content-Language: zh-CN

Content-Length: 98

```
{  
  
  "Seqno":101  
  
  "Time":"20180719135258778",  
  
  "ProcessTime":584,  
  
  "PlateNum":3,  
  
  "Plates":[{"LicensePlate":"G5FT69","LetterCount":7,"Confidence":0,"Left":964,"Right":1180,"Top":795,"  
Bottom":852,"PlateColor":50,"DoublePlate":0,"Direction":0,"CountryCode":0},{"LicensePlate":"YYPR01",  
"LetterCount":7,"Confidence":0,"Left":972,"Right":1187,"Top":717,"Bottom":775,"PlateColor":30,"DoublePlate":0,"Direction":0,"CountryCode":0},  
{"LicensePlate":"PZXD8S","LetterCount":7,"Confidence":0,"Left":972,"Right":1190,"Top":641,"Bottom":699,"PlateColor":30,"DoublePlate":0,"Direction":0,"CountryCode":0}],  
  
  "PlatesPhoto":"base64(binary)"  
}
```

Return:

CONTROL DEVICE {"Auth":{"Username":"admin","Password":"123456"},"Cmd":"OpenGate","Data":""}/t

**CONTROL DEVICE**

```
{"Auth":{"Username":"admin","Password":"123456"},"Cmd":"CtrlVoice","Data":"base64<voice list>"}/t
```

**CONTROL DEVICE**

```
{"Auth":{"Username":"admin","Password":"123456"},"Cmd":"CtrlLed","Data":{"FirstLineColor":"<color value>","FirstLineCenter":1,"FirstLineData":"<led content>","SecondLineColor":"<color value>","SecondLineCenter":1,"SecondLineData":"<led content>","ThirdLineColor":"<color value>","ThirdLineCenter":1,"ThirdLineData":"<led content>","FourthLineColor":"<color value>","FourthLineCenter":1,"FourthLineData":"<led content>"}}}
```

## 8.1 Parameter Description:

| Parameter                | Description                                                     |
|--------------------------|-----------------------------------------------------------------|
| Seqno                    | Package number                                                  |
| Time                     | Recognition time                                                |
| ProcessTime              | Algorithm identification time (milliseconds)                    |
| PlateNum                 | The license plate number                                        |
| Plates                   | License plate information (utf-8 code for license plate number) |
| PlatesPhoto              | The license plate images                                        |
| Return command reference | 1.4.2 remote control led/ voice command                         |

Follow-up planning: Subscribe to release data function, subscribe to upload event record, subscribe to upload license plate recognition result, and the subscription requires password verification.

## Appendix

### Appendix 1

| Event Code                       | Description                                         |
|----------------------------------|-----------------------------------------------------|
| <b>Note:</b> Events start at 100 |                                                     |
| 101                              | Whitelist release events                            |
| 102                              | Blacklist rejection event                           |
| 103                              | Temporary vehicle release event                     |
| 104                              | Temporary vehicle rejection event                   |
| 105                              | White list transfer temporary car incident          |
| 107                              | Temporary vehicle does not match entry record event |
| 108                              | Minimum charges for temporary vehicles              |
| 109                              | Fixed vehicle release event                         |
| 110                              | Fixed vehicle to temporary vehicle event            |
| 111                              | Fixed vehicle expiration event                      |

## Appendix 2

ZKTeco algorithm: The algorithm that converts years, months, days, hours, minutes and seconds into seconds.

```
unsigned int OldEncodeTime(int year, int mon, int day, int hour, int min, int sec)
{
    unsigned int tt;

    tt = ((year - 2000) * 12 * 31 + ((mon - 1) * 31) + day - 1) * (24 * 60 * 60)
    +(hour * 60 + min) * 60 + sec;

    return tt;
}
```

ZKTeco Industrial Park, No. 26, 188 Industrial Road,

Tangxia Town, Dongguan, China.

Phone : +86 769 - 82109991

Fax : +86 755 - 89602394



Copyright © 2020 ZKTECO CO., LTD. All Rights Reserved.