API of Conn modem

Instructions

Catalogue

1.screeshot cmd:GetScreenshot

2.screenshot by camera GetScreenshotByCamera

3.screenshot by android full size ScreenshotBySize

4.screen switch by manual SwitchScreen

5.screen switch schedule task (different with previous one)TaskKeepingScreenOn

6.Query screen switch schedule task

7.set brightness by manual SetBrightness

8.auto brightness SetAutoBrightness

9.set brightness schedule task TaskSettingBrightness

10.query brightness schedule task GetTaskToSetBrightness

1. Set volume by manual SetVolume

12.set volume schedule task

13.query volume schedule task GetTaskToSetVolume

14.get ntp server and timezone GetNtpAndTimezone

15.query gps data

16.Set sync SetSync

17.query sync settings GetSync

18.query hardware status

19.reboot

20.reboot on time

21.query reboot schedule

22.upgrade app

23.get device information

24.query auto brightness information GetAutoBrightness

25.uninstall APP by package name

26.advanced settings SetAdvancedConfig

27.backup

28.restore

29.set scoring system

30.set conn config

31.get scoring result

32.config hired status corresponding to IO high or low level

33.Query IO high/ low config when hired status

sendCommand interface

Url:"/v1/cms/conn/sendCommand"

method post

instructions：

When add deferred:true in the body of request, means offline sending

Theoretically, all commands support offline sending, but there is meaningless for some commands to do like this, such as some query commands.

We provide following suggestions for whether providing checkbox for offline operation:

When command types are TaskKeepingScreenOn、TaskSettingVolume、TaskSettingBrightness, support offline sending.

Need token or not: Need

Screenshot cmd:GetScreenshot

web -> server

{

"cardId": "y30-999-09998",

"\_type": "GetScreenshot"

}

server->device

{

"id":"5e8d6cf0e3fd984c2ceb076d"

"\_type":"GetScreenshot"

}

device->web（back end return the data to the front end as it is）

{

"\_type": "Screenshot",

"cardId": "e16-b19-40470",

"commandId": "5e8d6cf0e3fd984c2ceb076d",

"screenshot": "/9j/4AAQSkZJRgABAQAAAQABAAD/2wBDAAYEBQYFBAYGBQYHBwYIChAKCgkJChQODwwQFxQYGBcUFhYaHSUfGhsjHBYWICwgIyYnKSopGR8tMC0oMCUoKSj/2wBDAQcHBwoIChMKChMoGhYaKCgoKCgoKCgoKCgoKCgoKCgoKCgoKCgoKCgo",

"\_cardId": "e16-b19-40470"

}

Screenshot by camera GetScreenshotByCamera

web -> server

{

"\_type": "GetScreenshotByCamera",

"cardId": "e16-b19-40470"

}

server->device

{

"id":"5e8d6cf0e3fd984c2ceb076d",

"type":"GetScreenshotByCamera",

"sentTo":"joey-cardsystem"

}

device ->web back end return the data to the front end as it is

{

"cardId": "e16-b19-40470",

"commandId": "5e8d6d93e3fd984c2ceb076e",

"screenshot": "" , //character string or null

"\_cardId": "e16-b19-40470",

"\_type": "DataCallback"

}

Screenshot by android fullsize ScreenshotBySize

web -> server

{

"cardId": "e16-b19-40470",

"height": -1,

"quality": 100,

"scale": 100,

"width": -1,

"\_type": "ScreenshotBySize"

}

server->devices

{

"id":"5e8d6cf0e3fd984c2ceb076d",

"cardId":"e16-b19-40470",

"quality":100,

"scale":100,

"width":-1,

"height":-1,

"sendTo":"joey-cardsystem"

}

devices -> web

{

"cardId": "e16-b19-40470",

"commandId": "5e8d6f67e3fd984c2ceb076f",

"image": "/9j/4AAQSkZJRgABAQAAAQABAAD/2wBDAAEBAQEBAQEBAQEBAQ",

"\_cardId": "e16-b19-40470",

"\_type": "DataCallback",

}

Screen switch by manual SwitchScreen

web->server

{

"cardId": "e16-b19-40470",

"turnOn": true,

"\_type": "SwitchScreen"

}

server ->device

{

"\_type":"SwitchScreen",

"turnOn":true,"schedules":[],

"id":"5e8d720fe3fd984c2ceb0770"

}

devices ->web

{

"cardId": "e16-b19-40470",

"commandId": "5e8d720fe3fd984c2ceb0770",

"\_cardId": "e16-b19-40470",

"\_type": "Success"

}

Screen switch schedule task(different with previous one)TaskKeepingScreenOn

web->server

{

"taskId":"5e8d720fe3fd984c2ceb0770",

"cardId": "e16-b19-40470",

"\_type": "TaskKeepingScreenOn",

}

server->device

{

"\_type": "TaskKeepingScreenOn",

"task": {

"lng": "zh-CN",

"id": "5baae51f46ebb9d4223465f7",

"createBy": "check",

"createDate": "2018-09-26T01:47:11.408Z",

"dateCreated": "2018-09-26T01:47:11.408Z",

"schedules": [

{

"lng": "zh-CN",

"monthFilter": [],

"weekFilter": [],

"filterType": "None",

"endTime": "17:30",

"startTime": "17:05",

"timeType": "Range",

"endDate": null,

"startDate": null,

"dateType": "All"

}

],

"\_\_v": 0,

"\_user": "check",

"\_role": "539eaaedb6e1232a1566d9c3",

"\_department": "539eaaedb6e1232a1566d9c2",

"\_company": "alahover",

"name": "6",

"\_id": "5baae51f46ebb9d4223465f7"

},

"id": "5e869bae0db159693cc6cc24"

}

device ->web

{

"cardId": "e16-b19-40470",

"commandId": "5e8d740ee3fd984c2ceb0772",

"\_cardId": "e16-b19-40470",

"\_type": "Success"

}

Query screen switch schedule task

web->server

{

"cardId": "e16-b19-40470",

"\_type": "GetTaskToKeepScreenOn"

}

server->device

{"\_type":"GetTaskToKeepScreenOn","id":"5e8d79a7e3fd984c2ceb0773"}

device ->web

{

"\_type": "DataCallback",

"cardId": "e16-b19-40470",

"commandId": "5e8d79a7e3fd984c2ceb0773",

"task": {

"createBy": "test",

"createDate": "2020-04-03T06:35:36.913Z",

"name": "1111",

"schedules": [

{

"dateType": "Range",

"endDate": "2020-04-30",

"endTime": "18:00",

"filterType": "Week",

"monthFilter": [],

"startDate": "2020-04-03",

"startTime": "00:00",

"timeType": "Range",

"weekFilter": [

0,

1,

2,

3

]

}

]

},

"\_cardId": "e16-b19-40470"

}

Set brightness by manual SetBrightness

web ->server

{

"brightness": 8,

"cardId": "e16-b19-40470",

"\_type": "SetBrightness"

}

server->device

{

"\_type":"SetBrightness",

"brightness":8,

"schedules":[],

"id":"5e8d7c6fe3fd984c2ceb0783"

}

device ->web

{

"\_type": "Success",

"cardId": "e16-b19-40470",

"commandId": "5e8d7c6fe3fd984c2ceb0783",

"\_cardId": "e16-b19-40470"

}

Auto brightness SetAutoBrightness

web->server

{

"cardId": "e16-b19-40470",

"minBrightness": 1,

"sensitivity": 50,

"\_type": "SetAutoBrightness"

}

server->device

{

"\_type":"SetAutoBrightness",

"sensitivity":50,

"minBrightness":1,

"sendTo":"joey-cardsystem",

"id":"5e8d7d46e3fd984c2ceb0784"

}

device->web

{

"\_type": "Success",

"cardId": "e16-b19-40470",

"commandId": "5e8d7d46e3fd984c2ceb0784",

"\_cardId": "e16-b19-40470"

}

Set brightness schedule task TaskSettingBrightness

web->server

{

"taskId":"5e8d720fe3fd984c2ceb0770",

"cardId": "e16-b19-40470",

"\_type": "TaskKeepingScreenOn",

}

server->device

{"\_type":"TaskSettingBrightness",

"task":{

"lng":"zh-CN",

"id":"5e89bdf18283968413a4503d",

"createBy":"test",

"createDate":"2020-04-05T11:16:01.626Z",

"dateCreated":"2020-04-05T11:16:01.626Z",

"items":[

{

"id":"5e89bdf18283968413a4503e",

"schedules":[

{"lng":"zh-CN",

"monthFilter":[],

"weekFilter":[0,1,2,3,4,5],

"filterType":"Week",

"endTime":"18:30",

"startTime":null,

"timeType":"Range",

"endDate":"2020-04-22",

"startDate":"2020-04-06",

"dateType":"Range"}],

"\_id":"5e89bdf18283968413a4503e",

"brightness":5

}],

"\_\_v":0,

"\_user":"test",

"\_role":"5e673567908de9e019fbf231",

"\_department":"5e673567908de9e019fbf230",

"\_company":"test",

"name":"qwert",

"defaultBrightness":5,

"\_id":"5e89bdf18283968413a4503d"

},

"id":"5e8d7edbe3fd984c2ceb0785"}

​

device->web

{

"\_type": "Success",

"cardId": "e16-b19-40470",

"commandId": "5e8d7edbe3fd984c2ceb0785",

"\_cardId": "e16-b19-40470"

}

Query brightness schedule task GetTaskToSetBrightness

web->server

{

"cardId": "e16-b19-40470",

"\_type": "GetTaskToSetBrightness

}

server->device

{"\_type":"GetTaskToSetBrightness","id":"5e8d7f71e3fd984c2ceb0786"}

device -> web

​

Set Volume by manual SetVolume

web->server

{

"cardId": "e16-b19-40470",

"volume": 10,

"\_type": "SetVolume"

}

server->device

{"\_type":"SetVolume","volume":10,"id":"5e8d80d8e3fd984c2ceb079c"}

device -> web

{

"\_type": "Success",

"cardId": "e16-b19-40470",

"commandId": "5e8d80d8e3fd984c2ceb079c",

"\_cardId": "e16-b19-40470"

}

Set volume schedule task

web->server

{

"cardId":"e16-b19-40470",

"\_type":"TaskSettingVolume",

"taskId":"5e857783721111c038246212"

}

server->device

{

"\_type":"TaskSettingVolume",

"task":{

"lng":"zh-CN",

"id":"5e857783721111c038246211",

"createBy":"test",

"createDate":"2020-04-02T05:26:27.787Z",

"dateCreated":"2020-04-02T05:26:27.787Z",

"items":[{

"id":"5e857783721111c038246212",

"schedules":[

{

"lng":"zh-CN",

"monthFilter":[],

"weekFilter":[],

"filterType":"None",

"endTime":null,

"startTime":null,

"timeType":"All",

"endDate":null,

"startDate":null,

"dateType":"All"

}],

"\_id":"5e857783721111c038246212",

"volume":5

}],

"\_\_v":0,

"\_user":"test",

"\_role":"5e673567908de9e019fbf231",

"\_department":"5e673567908de9e019fbf230",

"\_company":"test",

"name":"tt1111",

"defaultVolume":10,

"\_id":"5e857783721111c038246211"},

"id":"5e8d81efe3fd984c2ceb079d"}

device-> web

{

"\_type": "Success",

"cardId": "e16-b19-40470",

"commandId": "5e8d81efe3fd984c2ceb079d",

"\_cardId": "e16-b19-40470"

}

Query volume schedule task GetTaskToSetVolume

web->server

{

"cardId": "e16-b19-40470",

"\_type": "GetTaskToSetVolume

}

server->device

{"\_type":"GetTaskToSetVolume","id":"5e8d83f0e3fd984c2ceb079e"}

device -> web

{

"\_type": "DataCallback",

"cardId": "e16-b19-40470",

"commandId": "5e8d83f0e3fd984c2ceb079e",

"task": {

"createBy": "test",

"createDate": "2020-04-02T05:26:27.787Z",

"defaultVolume": 10,

"items": [

{

"schedules": [

{

"dateType": "All",

"endDate": null,

"endTime": null,

"filterType": "None",

"monthFilter": [],

"startDate": null,

"startTime": null,

"timeType": "All",

"weekFilter": []

}

],

"volume": 5

}

],

"name": "tt1111",

"volume": 5

},

"\_cardId": "e16-b19-40470"

}

Get ntp server and timezone GetNtpAndTimezone

web ->server

{

"cardId": "e16-b19-40470",

"\_type": "GetNtpAndTimezone"

}

server->device

{"\_type":"GetNtpAndTimezone","id":"5e8d849de3fd984c2ceb079f"}

device->web

{

"\_type": "DataCallback",

"cardId": "e16-b19-40470",

"commandId": "5e8d849de3fd984c2ceb079f",

"language": "en",

"locationFeedback": 2,

"locked": false,

"now": "Apr 8, 2020 4:00:28 PM",

"ntpServer": "ntp1.aliyun.com",

"timezone": "Asia/Shanghai",

"\_cardId": "e16-b19-40470"

}

Query GPS data

web->server

{

"cardId": "e16-b19-40470",

"\_type": "GetGpsInfo"

}

server -> device

{"\_type":"GetGpsInfo","id":"5e8d85d1e3fd984c2ceb07a0"}

device ->web

{

"\_type": "DataCallback",

"cardId": "e16-b19-40470",

"commandId": "5e8d8712e3fd984c2ceb07a1",

"lat": 31.237386666666666,

"lng": 121.32845,

"satelliteNumber": 18,

"speed": 3.2638535,

"\_cardId": "e16-b19-40470"

}

Set sync SetSync

web ->server

//Type 1

{

"brightness": "none",

"cardId": "e16-b19-40470",

"checkNtpTime": 10,

"delaySync": "",

"identificationCode": "xixun",

"screenSwitch": "none",

"time": "serial",

"volume": "none",

"\_type": "SetSync"

}

//type 2

{

"brightness": "none",

"cardId": "e16-b19-40470",

"checkNtpTime": 10,

"delaySync": "",

"identificationCode": "xixun",

"screenSwitch": "none",

"time": "gps",

"volume": "none",

"\_type": "SetSync"

}

//type 3

{

"brightness": "none",

"cardId": "e16-b19-40470",

"checkNtpTime": 10,

"delaySync": "",

"identificationCode": "xixun",

"screenSwitch": "none",

"time": "ntp",

"volume": "none",

"\_type": "SetSync",

}

server->device

{

"\_type":"SetSync",

"identificationCode":"xixun",

"delaySync":null,

"checkNtpTime":10,

"screenSwitch":"none",

"volume":"none",

"brightness":"none",

"time":"serial",

"sendTo":"joey-cardsystem",

"id":"5e8d8736e3fd984c2ceb07a2"

}

device->web

{

"\_type": "Success",

"cardId": "e16-b19-40470",

"commandId": "5e8d8736e3fd984c2ceb07a2",

"\_cardId": "e16-b19-40470"

}

Query sync settings GetSync

web->server

{

"cardId": "e16-b19-40470",

"\_type": "GetSync"

}

server->device

{

"\_type":"GetSync",

"sendTo":"joey-cardsystem",

"id":"5e8d89b8e3fd984c2ceb07a7"

}

device->web

{

"\_type": "DataCallback",

"cardId": "e16-b19-40470",

"commandId": "5e8d89b8e3fd984c2ceb07a7",

"brightness": "NONE",

"checkNtpTime": 10,

"delaySync": 0,

"identificationCode": "xixun",

"lastSynchronousTime": 1586333951000,

"screenSwitch": "NONE",

"time": "NTP",

"volume": "NONE",

"\_cardId": "e16-b19-40470"

}

Query hardware status

web->server

{

cardId: "e16-b19-40470",

\_type: "GetFpgaInfo"

}

server->device

{"\_type":"GetFpgaInfo","id":"5e8d8a4ee3fd984c2ceb07a8"}

device ->web

{

"\_type": "FpgaInfoCallback",

"cardId": "e16-b19-40470",

"commandId": "5e8d8a4ee3fd984c2ceb07a8",

"list": [

{

"cardVoltage": "0.0V",

"doorOpened": "Close",

"externalVoltage1": "0.0V",

"externalVoltage2": "0.0V",

"height": "4",

"humidity": "0.0%",

"smoke": "Alarm",

"temperature": "0.0℃",

"version": "af21",

"width": "3",

"x": "2",

"y": "1"

}

],

"\_cardId": "e16-b19-40470"

}

Reboot

web ->server

{

"cardId": "e16-b19-40470",

"\_type": "Reboot"

}

server->device

{"\_type":"Reboot","id":"5e8d8bafe3fd984c2ceb07a9"}

device ->web

{

"\_type": "Success",

"cardId": "e16-b19-40470",

"commandId": "5e8d8ccee3fd984c2ceb07b7",

"\_cardId": "e16-b19-40470"

}

Reboot on time

web ->server

{

"cardId": "e16-b19-40470",

"time": "00:00",

"\_type": "SetScheduleToReboot"

}

server->device

{

"\_type":"SetScheduleToReboot",

"time":"00:00",

"sendTo":"joey-cardsystem",

"id":"5e8d8ccee3fd984c2ceb07b7"}

device -> web

{

"\_type": "Success",

"cardId": "e16-b19-40470",

"commandId": "5e8d8ccee3fd984c2ceb07b7",

"\_cardId": "e16-b19-40470"

}

Query reboot schedule

web ->server

{

"cardId": "e16-b19-40470",

"\_type": "GetScheduleToReboot"

}

server->device

{

"\_type":"GetScheduleToReboot",

"sendTo":"joey-cardsystem",

"id":"5e8d9217e3fd984c2ceb089f"}

device ->web

{

"\_type": "DataCallback",

"time": "00:00",

"commandId": "5e8d9217e3fd984c2ceb089f",

"cardId": "e16-b19-40470",

"\_cardId": "e16-b19-40470"

}

Upgrade app

web ->server

{

"apkId": "5e86f392721111c0382463ac",

"apkName": "Display-516.zip",

"cardId": "y30-999-09998",

"\_type": "UpdateApp",

}

server->device

{

"\_type":"UpdateApp",

"apkId":"5e86f392721111c0382463ac",

"apkName":"Display-516.zip",

"id":"5e8d8f09e3fd984c2ceb07d2"

}

device -> web

{

"\_type": "Success",

"cardId": "e16-b19-40470",

"commandId": "5e8d8ccee3fd984c2ceb07b7",

"\_cardId": "e16-b19-40470"

}

Get device information

web->server

{

"\_type": "GetCardInfo",

"cardId": "e16-b19-40470"

}

server->device

{"\_type":"GetCardInfo","id":"5e8d906ce3fd984c2ceb089e"}

device -> web

{

"\_type": "CardInfo",

"cardId": "e16-b19-40470",

"commandId": "5e8d906ce3fd984c2ceb089e",

"card": {

"alias": "e16-470",

"asu": 0,

"autoBrightness": false,

"basicAppVersion": "1.4.5.3",

"basicAppVersionCode": 16,

"brightness": 8,

"companyId": "test",

"connVersion": "10.2.7TM",

"connVersionCode": 127,

"currentProgramId": null,

"currentProgramName": "unknown",

"diskSize": 0,

"displayVersion": "v5.1.6",

"displayVersionCode": 4,

"height": 32,

"humidity": -1,

"lat": 31.237225,

"ledsetVersion": "5.1.8.5",

"ledsetVersionCode": 526,

"lng": 121.32818,

"locked": false,

"lora": "unknown",

"netType": "WIFI",

"orderVideoCode": 0,

"orderVideoVersion": "",

"playerVersion": "unknown",

"playerVersionCode": -1,

"rssi": -32,

"screenStatus": "on",

"sim": {

"deviceId": "86985603113945",

"networkCountryIso": "",

"number": null,

"simCountryIso": "",

"simOperatorName": "",

"simSerialNumber": null,

"simState": 1,

"subscriberId": null

},

"starterVersion": "starter179",

"starterVersionCode": 179,

"taxiAppVersion": "2.0.7.19",

"taxiAppVersionCode": 3,

"temperature": 31,

"updateVersion": "7.1",

"updateVersionCode": 19,

"volume": 10,

"width": 64,

"\_company": "test"

},

"\_cardId": "e16-b19-40470"

}

Query auto brightness information GetAutoBrightness

web->server

{

cardId: "e16-b19-40470",

\_type: "GetAutoBrightness"

}

​

server->device

{

"id":"asdasdasd",

"\_type":"asdsadasd",

"sendTo":"joye-cardsystme"

}

device-web

{

"\_type": "DataCallback",

"sensitivity": 100,

"minBrightness": 1,

"commandId": "5e9e9ae73ff04e581a942d5a",

"cardId": "e16-b19-40470",

"\_cardId": "e16-b19-40470"

}

Uninstall APP by package name

web ->server

{"\_type":"UninstallApp",

"packageName":"com.xixun.display",

"cardId":"e16-b19-40470"

}

server->device

{

"\_type":"UninstallApp",

"packageName":"com.xixun.display",

"id":"5e9e96bd3ff04e581a942d57"}

device -> web

{

"\_type": "Success",

"cardId": "e16-b19-40470",

"commandId": "5e8d8ccee3fd984c2ceb07b7",

"\_cardId": "e16-b19-40470"

}

​

Advanced configuration

\* web -> server

```json

{

"cardId": "m60-518-00021",

"companyId": "alahover", //can be empty

"realtimeURL": "m2mled.net", //can be empty

"serverURL": "m2mled.net", //can be empty

"usbProgramPwd": "111", //can be empty

"\_type": "SetAdvancedConfig"

}

```

\* server->device

```json

{

"id":"5e9e96bd3ff04e581a942d57",

"\_type":"SetAdvancedConfig",

"deferred":false,

"companyId":"test", //can be empty

"realtimeUrl":"192.168.8.88:2345", //can be empty

"serverURL":"m2mled.net", //can be empty

"usbProgramPwd":"11111" //can be empty

}

```

\* device -> web

```json

{

"cardId": "y30-120-01093",

"commandId": "5f8faccfe69a4c78241f1740",

"\_cardId": "y30-120-01093",

"\_type": "Success"

}

```

backup

web ->server

{

"cardId": "y10-a18-00005",

"\_type": "PostCardSystemConfig"

}

device->web

{

"cardId":"y1099900003",

"\_type":"Received",

"commandId":"5fb63698d82f06cc1ab3796d"}

​

restore

web->server

{

"cardId": "y60-620-40359",

"downloadURL": "5fb6369ed82f06cc1ab3798d",

"\_type": "DownloadCardSystemConfig,

}

device->web

{

"cardId":"y10-999-00003",

"\_type":"Received",

"commandId":"5fb637a9d82f06cc1ab37d81"}

​

​

​

Set scoring system

web->server

​

{

"cardId":"",

"stars":5,

"\_type":"EvaluationStars"

}

​

device-> web

{

"cardId": "y30-120-01093",

"\_type": "DataCallback",

"result": true,

"commandId": "6046d1922d2a66a474bcc09a",

"success": true,

"\_cardId": "y30-120-01093"

}

Set conn configuration

web ->server

{

​

"cardId": "l10-a17-60003",

"ntpServer": "http://aliyun.com",

"timezone": "As/shanghai",

"\_type": "SetConnectionConfig"

}

​

device->web

​

​

​

​

Get scoring result

web -> server

{

"cardId":"y30-120-01093",

"\_type":"GetStars"

}

device->web

​

{

"cardId": "y30-120-01093",

"\_type": "DataCallback",

"result": 5,

"commandId": "6046d1dd2d2a66a474bcc7f5",

"success": true,

"\_cardId": "y30-120-01093"

}

Set hired status corresponding to IO high/low level

web ->server

{

"cardId":"y30-120-01093",

"\_type":"SetHighForBusy",

"busyState":0 //0:low;1high

}

device->web

{

"\_type": "DataCallback",

"commandId":"98e8d3cd47fad6ce8e3f7b8d42cb4d9",

"success": true

}

​

Query IO high/ low config under hired status

web->server

{

"cardId":"y30-120-01093",

"\_type": "GetStateForBusy",

}

server->device

{

"\_type": "DataCallback",

"commandId": "98e8d3cd47fad6ce8e3f7b8d42cb4d9b",

"success": true,

"result": 1

}