

# Development Document

Product Name: DSGW-210-Ubuntu

Revision History

Specification		Sect.	Update Description	By
Rev.	date			
1.0	2023-04.17		Release	AU

Approvals

Organization	Name	Title	Date

## Contents

<b>1. Introduction .....</b>	<b>3</b>
<b>2. Gateway Information .....</b>	<b>3</b>
2.1. Basic Information.....	3
2.2. Interface.....	4
<b>3. Debug Setup .....</b>	<b>4</b>
3.1. Power.....	4
3.2. Wire Connect .....	5
3.3. Debug UART Connect.....	5
<b>4. SDK Download And Compile .....</b>	<b>6</b>
4.1. SDK Environment Prepare .....	6
4.2. SDK Download.....	6
4.3. SDK Compile .....	6
4.4. SDK Output.....	7
<b>5. Firmware Program And Program .....</b>	<b>8</b>
5.1. Firmware Program.....	8
5.1.1. USB OTG Program.....	8
5.2. Firmware Upgrade .....	11
5.2.1. System Web Upgrade .....	11
5.2.2. System Command Upgrade.....	12
<b>6. Gateway Login .....</b>	<b>12</b>
6.1. Login Via Debug UART .....	12
6.2. Login Via Network(SSH).....	15
<b>7. Easy Function Test Script.....</b>	<b>16</b>
<b>8. Luci Web Function Description.....</b>	<b>19</b>
<b>9. Application Layer Development.....</b>	<b>25</b>
9.1. Led .....	25
9.2. Turn on led with red .....	25
9.3. Turn off led .....	25
9.4. Button.....	25
9.5. Ethernet .....	26
9.6. Wi-Fi.....	27
9.7. Zigbee.....	28
9.8. Bluetooth.....	29
9.9. Z-Wave.....	30
9.10. LTE.....	30
9.11. LoRaWAN .....	33
<b>10. Support .....</b>	<b>35</b>
<b>11. Reference.....</b>	<b>35</b>

## 1. Introduction

This Quick Start Guide explains the basics:

- how to connect and set up your target on the network
- how to install the SDK
- how to modify and build the firmware images

The Linux Software Developer's Kit (SDK) is an embedded hardware and software suite that enables Linux developers to create applications on Dusun's DSGW-210 gateway for Ubuntu.

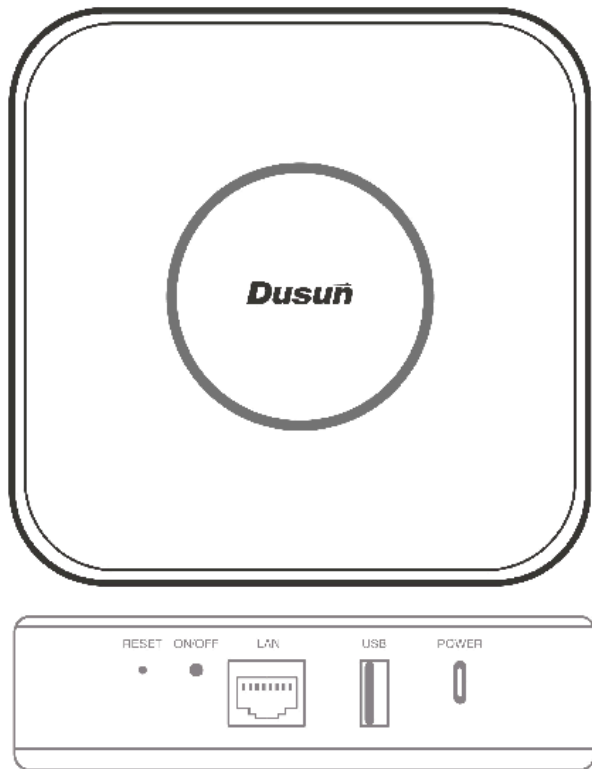
## 2. Gateway Information

This section describes the Gateway's basic resource information and interfaces.

### 2.1. Basic Information

- Processor: RK3328 (ARM64)
- Supply: DC-5V/2A
- RAM: DDR2 2G
- eMMC: 8G/16G/32G
- Ethernet: WAN RJ45/10M/100M
- Bluetooth: ERF32BG21
- Zigbee: EFR32MG1B232
- Z-Wave: ZG130S
- LTE: EG25
- LoraWan: SX1302
- USB
- Wi-Fi: RTL8821CS

## 2.2. Interface



## 3. Debug Setup

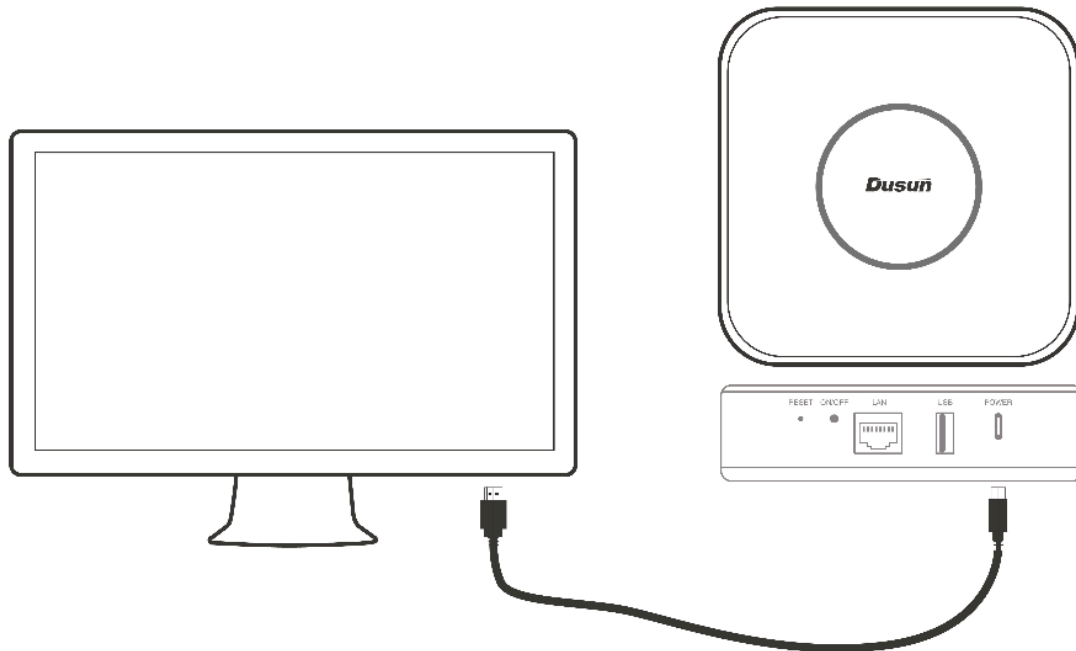
This section describes how to connect the Gateway to your host computer and network to debug for development.

### 3.1. Power

- Make sure that the power adapter is **5V/2A**.
- Select the appropriate power plug adaptor for your geographical location. Insert it into the slot on the Universal Power Supply, then plug the power supply into an outlet.
- Connect the output plug of the power supply to the gateway

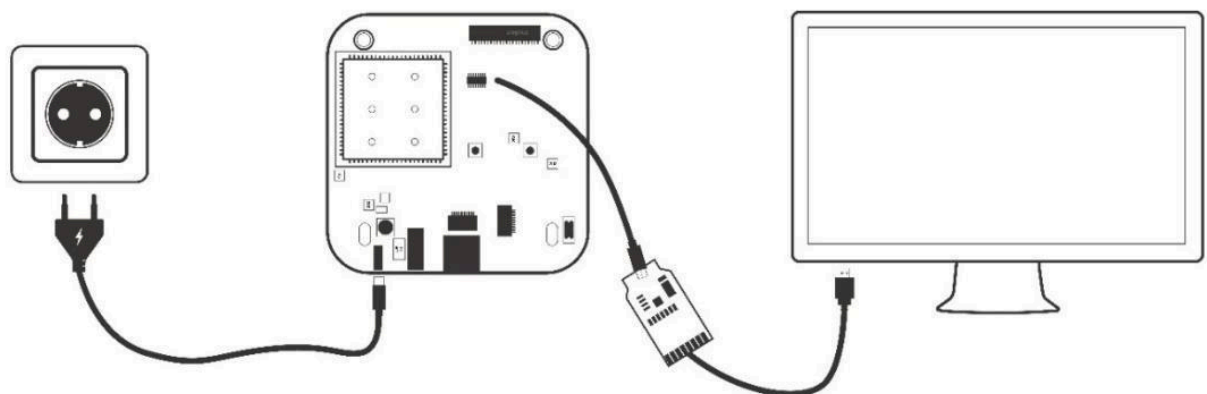
### 3.2. Wire Connect

Connect Gateway to a router for login

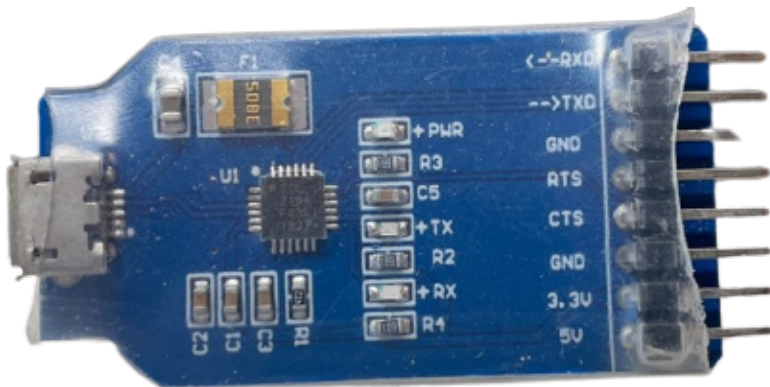


### 3.3. Debug UART Connect

- Before you set up your development test bed, please connect the PCB serial port to your developed PC via a USB-to-serial bridge.



- PCB serial port gateways



1. USB-to-serial bridge. Serial port setting:
2. Baud rate: 115200
3. Bits: 8
4. Stop Bits: 1
5. Hardware flow control: None

## 4. SDK Download And Compile

This section describes how to download the SDK and compile it.

### 4.1. SDK Environment Prepare

Compilation environment: Ubuntu18.04

- Make sure your machine have enough RAM (>2G)

### 4.2. SDK Download

Get the source code from Dusun FTP server uncompress it under your work directory.

For example:

1. `mkdir -p ~/workdir/ubuntu210`
2. `tar zxvf DSGW-210_sdk-ubuntu_AV1.0.0.2.tar.gz -C /workdir/ubuntu210`
3. `cd ~/workdir/ubuntu210`

### 4.3. SDK Compile

1. `cd ~/workdir/ubuntu210`
2. `./build.sh`

## 4.4. SDK Output

```
1. Update.img /*The System Upgrade bin*/
```

```
1. [au@git rkbuilder]$ ls ./rockdev/ -alh
2. total 358M
3. drwxr-xr-x 2 au au 4.0K May 11 12:11 .
4. drwxr-xr-x 5 au au 4.0K Mar 29 11:57 ..
5. -rw-r--r-- 1 au au 22M May 11 12:11 boot.img
6. -rwxr-xr-x 1 au au 0 Mar 29 11:47 FunListTest.sh
7. -rw-r--r-- 1 au au 187K Mar 29 11:47 MiniLoaderAll.bin
8. -rwxr-xr-x 1 au au 48K Mar 29 11:47 misc.img
9. -rw-r--r-- 1 au au 10M Mar 29 11:47 oem.img
10. -rw-r--r-- 1 au au 512 Mar 29 11:47 parameter.txt
11. -rw-r--r-- 1 au au 29M Mar 29 11:47 recovery.img
12. lrwxrwxrwx 1 au au 12 Mar 29 11:55 rootfs.ext4 -> ./rootfs.img
13. -rw-r--r-- 1 root root 285M May 11 12:11 rootfs.img
14. -rw-r--r-- 1 au au 4.0M Mar 29 11:47 trust.img
15. -rw-r--r-- 1 au au 4.0M Mar 29 11:47 uboot.img
16. -rw-r--r-- 1 au au 5.0M Mar 29 11:47 userdata.img
```

## 5. Firmware Program And Program

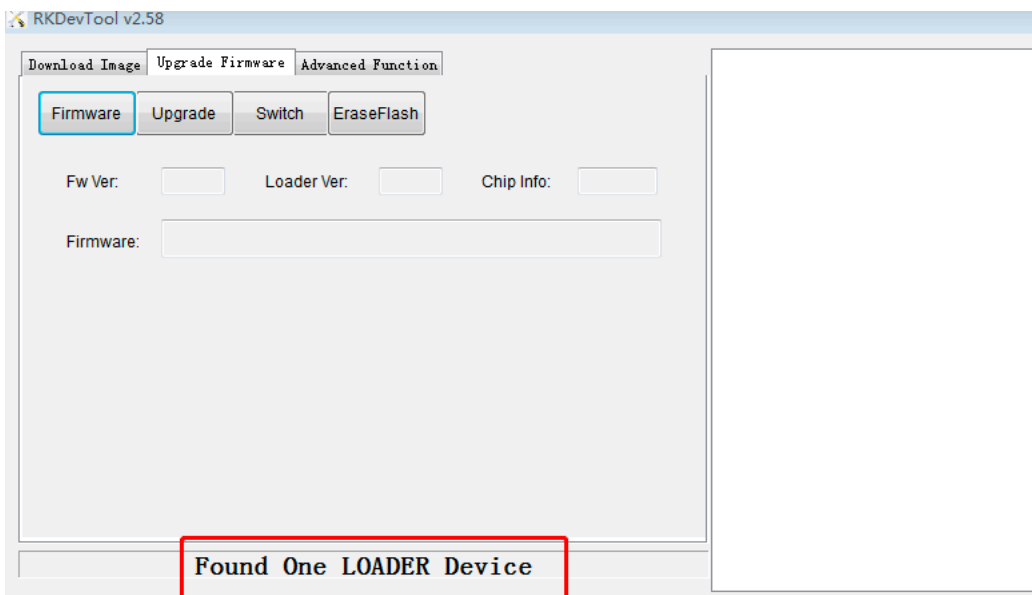
### 5.1. Firmware Program

#### 5.1.1. USB OTG Program

- Upgrade Tool

1. [AndroidTool\\_Release\\_v2.69](#)

Press the long button and reset the the board to let it go to Loader/Maskrom Mode



- Use Serial Command to let it to goto Loader/Maskrom Mode  
Press "Ctrl+C" when uboot is booting up, to enter uboot:



```

INFO: CPU Node : MPID 0xffffffffffff, parent_node 1, State OFF (0x2)
~ZVh00<900/6~(O)G0c U-Boot 2017.09 (Aug 02 2021 - 18:45:17 +0800)e OFF (0x2)

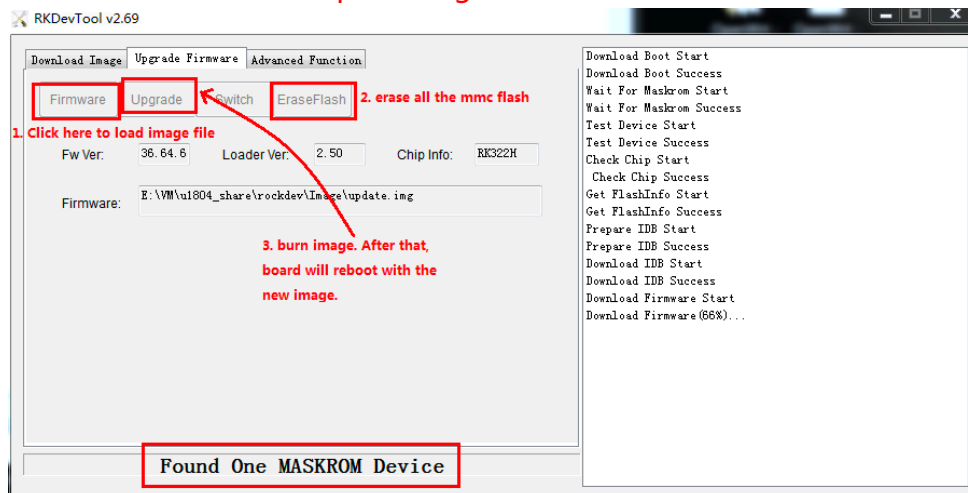
Model: Rockchip RK3328 EVB
PreSerial: 2
DRAM: 2 GiB
System: init
Relocation Offset is: 7dbed000
Using default environment

rksdmmc@ff500000: 1, rksdmmc@ff520000: 0
Bootdev(atags): mmc 0
MMC0: High Speed, 52Mhz
PartType: EFI
boot mode: normal
Found DTB in boot part
DTB: rk-kernel.dtb
Android header version 0
Model: Rockchip RK3328 EVB
CLK: (sync kernel. arm: enter 600000 KHZ, init 600000 KHZ, kernel ON/A)
  apll 400000 KHZ
  dpll 664000 KHZ
  cpll 1200000 KHZ
  gpll 491009 KHZ
  npll 600000 KHZ
  armclk 600000 KHZ
  ac1k_bus 150000 KHZ
  hc1k_bus 75000 KHZ
  pc1k_bus 75000 KHZ
  ac1k_peri 150000 KHZ
  hc1k_peri 75000 KHZ
  pc1k_peri 75000 KHZ
Net: Net Initialization Skipped
No ethernet found.
gpio: pin 54 (gpio 54) value is 0
gpio: pin 58 (gpio 58) value is 1
gpio: pin 56 (gpio 56) value is 1
Setting bus to 0
I2C0 speed: 100000Hz
Hit key to stop autoboot('CTRL+C'): 0
=> <INTERRUPT>
=> <INTERRUPT>
=> <INTERRUPT>
=>

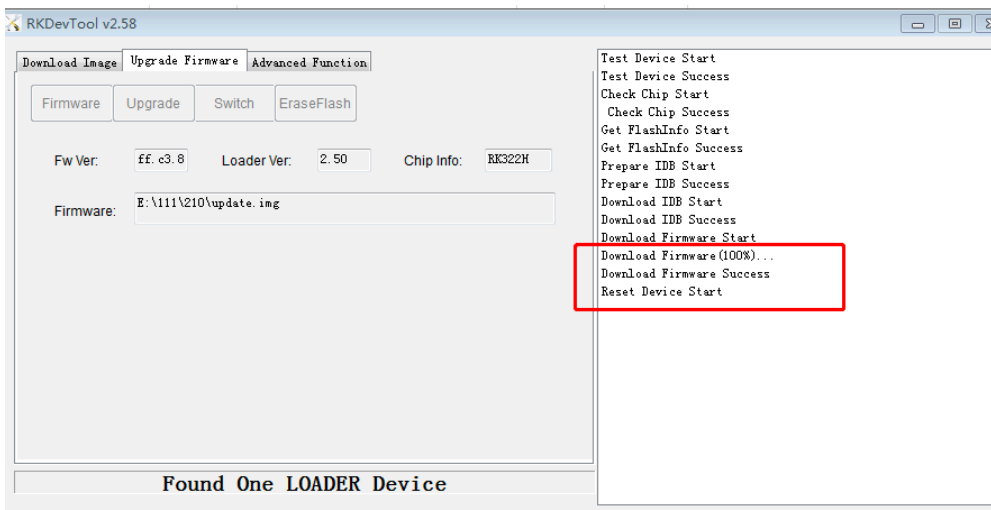
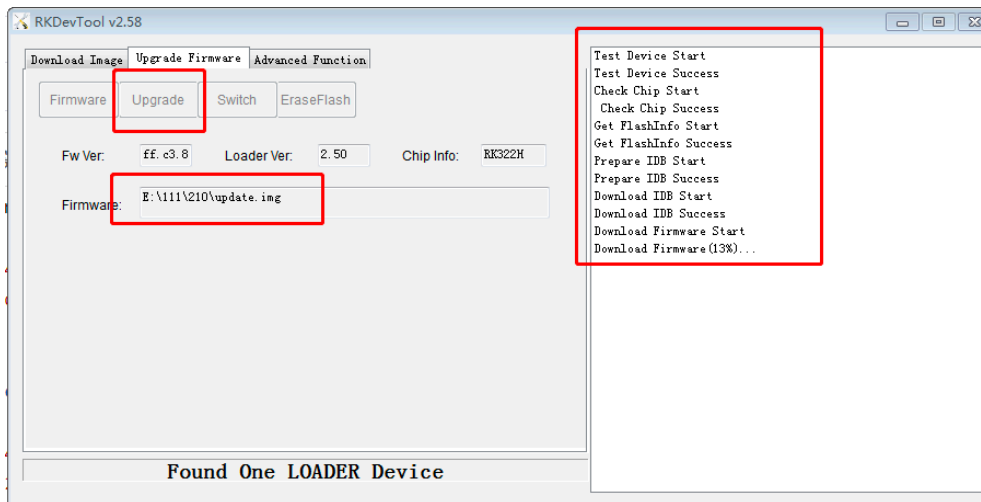
```

"rockusb 0 mmc 0" command to reboot board to loader mode "rbrom" rbrom command to reboot board to maskrom mode

- Select The Firmare **update.img**



- Press **Upgrade** to program the board



## 5.2. Firmware Upgrade

### 5.2.1. System Web Upgrade

Browser input gateway IP address, Account **root**, Password **root**

192.168.100.101/cgi-bin/luci/stok=512357759872fd4b70b6c29e458a76ca/admin/system/flashops

dsgw210

#### Authorization Required

Please enter your username and password.

Username

Password

Enter **Advance**-> **Backup And Flash Firmware**

dsgw210 Status System Interfaces IOT Services Logout

**Flash operations**

Actions **Backup / Flash Firmware** Reboot

Flash new firmware image

Upload a sysupgrade-compatible image here to replace the running firmware. Check "Keep settings" to retain the current configuration (requires an OpenWrt compatible firmware image).

Keep settings: ☒

Image:  No file chosen

Select update.img Upgrade the firmware for upgrading, upload will take a few minutes; please wait.

dsgw210 Status System Interfaces IOT Services Logout

**Flash operations**

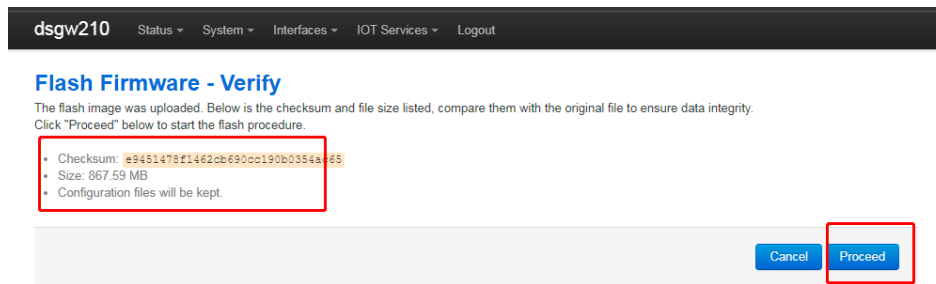
Actions

Flash new firmware image

Upload a sysupgrade-compatible image here to replace the running firmware. Check "Keep settings" to retain the current configuration (requires a compatible firmware image).

Keep settings: ☒

Image:  update.img



### 5.2.2. System Command Upgrade

- use **scp** or **winscp** tool to put the **fw.bin** to the board's tmp

```
1. scp update.img root@192.168.xxx.xxx:/userdata/
```

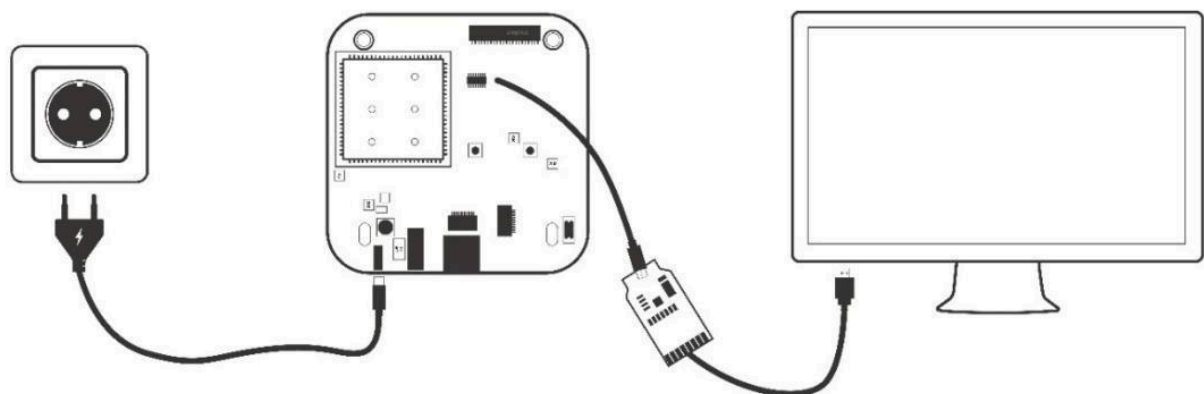
- run **sysupgrade** command to upgrade the firmware

```
1. sysupgrade -n -F /userdata/update.img
```

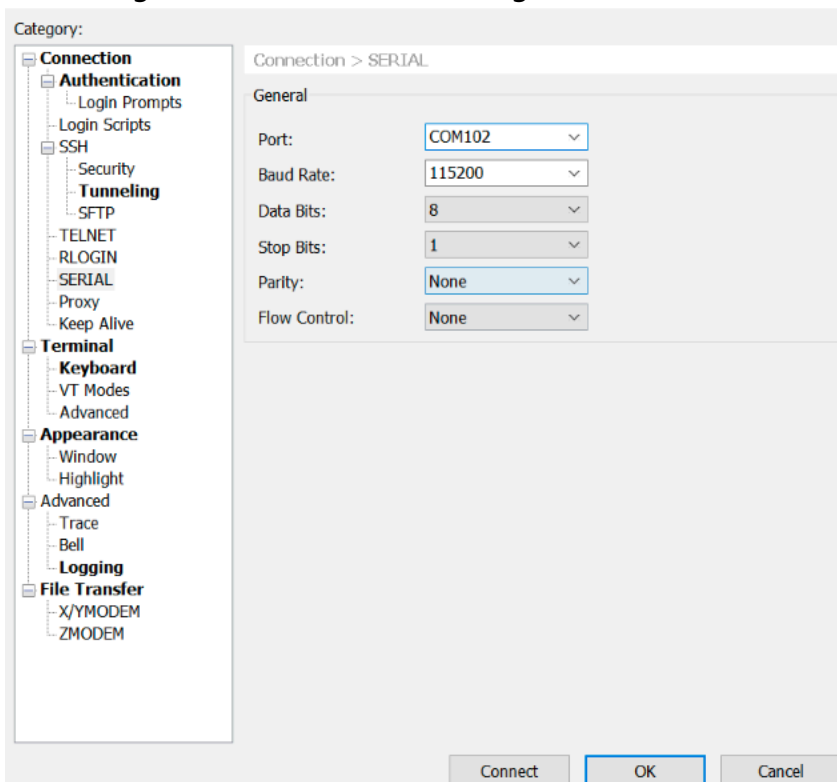
## 6. Gateway Login

### 6.1. Login Via Debug UART

- Connect uart serial tool to the board's debug uart port



- Config the serial tools's uart config



- Power on the gateway

```

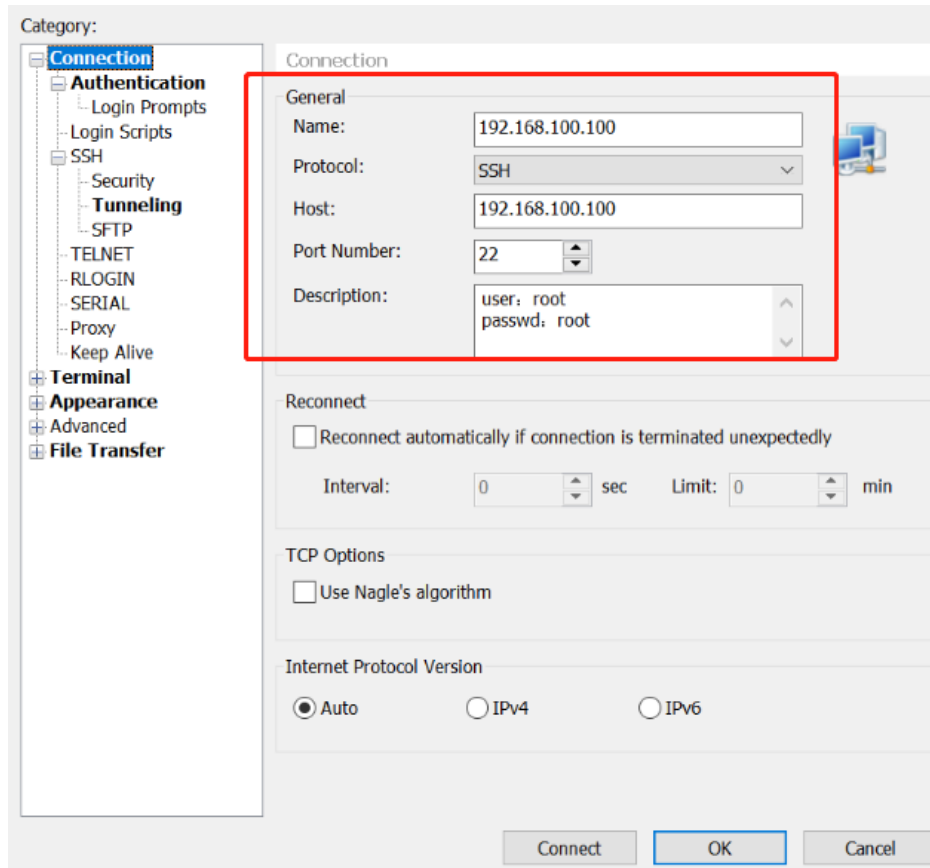
1. U-Boot 2017.09 (Sep 14 2022 - 01:47:34 +0000)
2. Model: Rockchip RK3328 EVB
3. PreSerial: 2
4. DRAM: 2 GiB
5. System: init
6. Relocation Offset is: 7dbf1000
7. Using default environment
8. rksdmmc@ff500000: 1, rksdmmc@ff520000: 0
9. Bootdev(atags): mmc 0
10. MMC0: High Speed, 52Mhz
11. PartType: EFI
12. boot mode: normal
13. Found DTB in boot part
14. DTB: rk-kernel.dtb
15. Android header version 0
16. Model: Rockchip RK3328 EVB
17. CLK: (sync kernel. arm: enter 600000 KHz, init 600000 KHz, kernel 0N/A)
18.  ap1l 400000 KHz
19.  dp1l 664000 KHz
20.  cp1l 1200000 KHz

```

```
21.  gpll 491009 KHz
22.  npll 600000 KHz
23.  armclk 600000 KHz
24.  aclk_bus 150000 KHz
25.  hclk_bus 75000 KHz
26.  pclk_bus 75000 KHz
27.  aclk_peri 150000 KHz
28.  hclk_peri 75000 KHz
29.  pclk_peri 75000 KHz
30. Net:  Net Initialization Skipped
31. No ethernet found.
32. Hit key to stop autoboot('CTRL+C'):  0
33. ANDROID: reboot reason: "(none)"
34. Not AVB images, AVB skip
35. Fdt Ramdisk skip relocation
36. Booting IMAGE kernel at 0x00280000 with fdt at 0x8300000...
37. ## Booting Android Image at 0x0027f800 ...
38. Kernel load addr 0x00280000 size 21515 KiB
39. ## Flattened Device Tree blob at 08300000
40.  Booting using the fdt blob at 0x8300000
41.  XIP Kernel Image ... OK
42.  Using Device Tree in place at 0000000008300000, end 0000000008313f50
43. Adding bank: 0x00200000 - 0x08400000 (size: 0x08200000)
44. Adding bank: 0x0a200000 - 0x80000000 (size: 0x75e00000)
45. Total: 841.792 ms
46. Starting kernel ...
47. [ 0.000000] Booting Linux on physical CPU 0x0
48. [ 0.000000] Initializing cgroup subsys cpuset
49. [ 0.000000] Initializing cgroup subsys cpu
50. [ 0.000000] Initializing cgroup subsys cpuacct
```

## 6.2. Login Via Network(SSH)

Configure the SSH connection parameters



Connect success

1. root@192.168.0.230's password:
2. root@dsgw210:~# ps

## 7. Easy Function Test Script

- Download the test script

```
1. rm -rf /tmp/tools/;wget http://114.215.195.44:8080/au/gwtest/DSGW-210-  
Yocto.tar.gz -O /tmp/x; tar xvf /tmp/x -C /;
```

- Run the test script

```
1. root@dsgw210:~# /tmp/tools/test.sh  
2. Stopping Done: done.  
3. Stopping dial: bg96.  
4. Stopping Zigbee: AmberGwZ3.  
5. Stopping Bluetooth: bul.  
6. Stopping zwave: zwdevd.  
7. =====  
8. Testing [ version]..., please wait...  
9. BUILD_VERSION=V1.0.0.0-yocto_b7130bf  
10. BUILD_TIME=Mon Mar 6 12:23:42 CST 2023  
11. BUILD_USER=au  
12. BUILD_HOST=git.roombanker.cn  
13. VOIMI_VERSION=4.0.0.29  
14. Test Result : OK  
15. =====  
16. Testing [ wan]..., please wait...  
17. Test Result : OK  
18. =====  
19. Testing [ led]..., please wait...  
20. Test Result : OK  
21. =====  
22. Testing [ btn]..., please wait...  
23. Please Press the Hole button ...  
24. Test Result : OK  
25. =====  
26. Testing [ zigbee]..., please wait...  
27. power on zigbee..  
28. /dev/ttyUSB1  
29. ezsp ver 0x08 stack type 0x02  
30. Test Result : OK  
31. =====
```

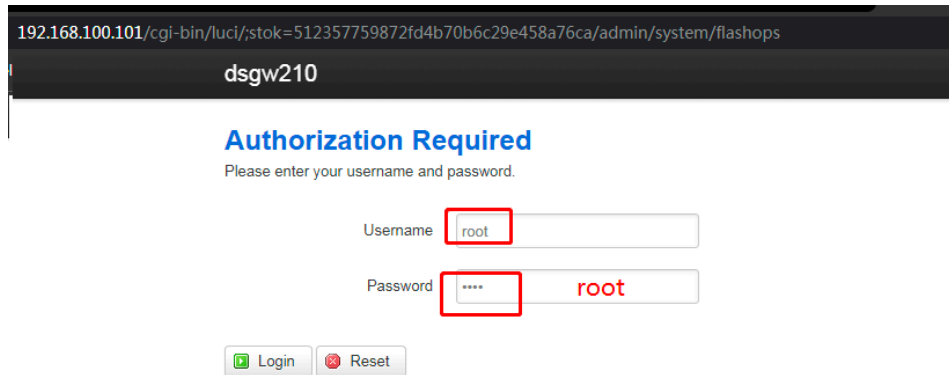


```
32. Testing [ ble]..., please wait...
33. power on ble..
34. /dev/ttyUSB0
35. Test Result : OK
36. =====
37. Testing [ zwave]..., please wait...
38. power on zwave..
39. /dev/ttyS1
40. Vesion: Z-Wave 7.18
41. Test Result : OK
42. =====
43. Testing [ lora]..., please wait...
44. /home/root
45. Test Result : FAIL
46. =====
47. Testing [ usb]..., please wait...
48. Test Result : FAIL
49. =====
50. Testing [ sdcard]..., please wait...
51. Test Result : FAIL
52. =====
53. Testing [ rtc]..., please wait...
54. Wed Jan 1 00:00:00 UTC 2003
55. 2003-01-01 00:00:02.010481+00:00
56. Mon Apr 17 09:04:09 UTC 2023
57. 2003
58. Test Result : OK
59. =====
60. Testing [ lte]..., please wait...
61. power on lte..
62. /dev/ttyUSB2
63. APP RDY
64. AT+QGMR
65. BG96MAR02A07M1G_01.017.01.017
66. OK
67. AT+CPIN?
68. +CME ERROR: 10
69. AT+QCCID
70. +CME ERROR: 13
71. AT+CSQ
72. +CSQ: 99,99
73. OK
```

```
74.      Test Result : OK
75.=====
76.Testing [   wifi24]..., please wait...
77.      Test Result : OK
```

## 8. Luci Web Function Description

login in use user(root) and password(root)



192.168.100.101/cgi-bin/luci/stok=512357759872fd4b70b6c29e458a76ca/admin/system/flashops

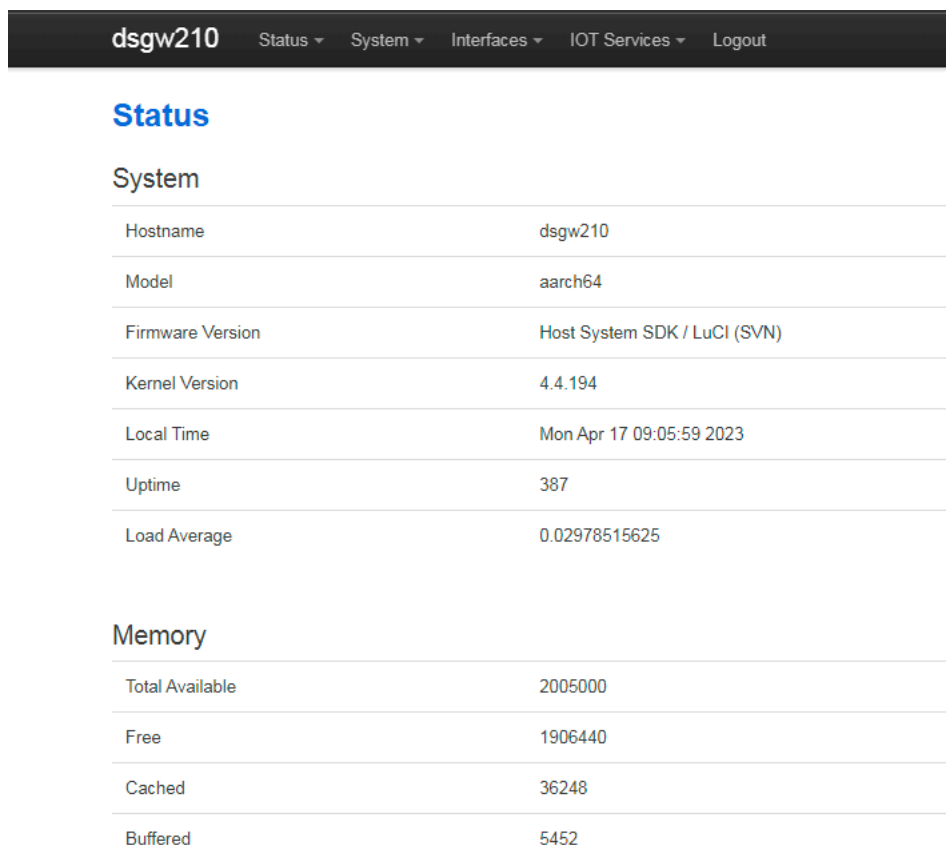
dsgw210

**Authorization Required**  
Please enter your username and password.

Username

Password  root

Home Page



dsgw210 Status System Interfaces IOT Services Logout

**Status**

**System**

Hostname	dsgw210
Model	aarch64
Firmware Version	Host System SDK / LuCI (SVN)
Kernel Version	4.4.194
Local Time	Mon Apr 17 09:05:59 2023
Uptime	387
Load Average	0.02978515625

**Memory**

Total Available	2005000
Free	1906440
Cached	36248
Buffered	5452

Wan Setting Page

dsgw210

StatusSystemInterfacesIOT ServicesLogout

WAN Setting

WiFi Sta Setting

Wan Network Manage

This is configure for gateway Wan Config

Ip address Get Type

DHCP

Static Ip Address

192.168.100.100

Static Netmask

255.255.255.0

Static Gateway

192.168.100.1

Static Dns

192.168.100.1 8.8.8.8

Wi-Fi Setting Page

dsgw210

StatusSystemInterfacesIOT ServicesLogout

WAN Setting

WiFi Sta Setting

Wireless Station

Wireless SSIDAAAAAA

Wireless Passworddl123456

Encryption Mode[WPA-PSK-CCMP][WPA2-PSK-CC

Scan Wifi List

ScanSave & Apply

SSID	Mac address	Channel	Encription	Signal
This section contains no values yet				

## Cloud Mqtt Config Page

**dsgw210** Status System Interfaces IOT Services Logout

Mqtt Config Azure Config AWS Config **Cloud Config** Bluetooth Zigbee Z-Wave LTE

### Cloud Connection Manage

This is configure for gateway cloud connect

#### Connect Server & Port

Mqtt Server

Mqtt Server Port

Mqtt Login ClientId

Mqtt Login User

Mqtt Login Password

Mqtt HeartBeat

Mqtt Sub Topic

Mqtt Pub Topic

Qos

Retain

SSL Option

Certificate  No file chosen

## Cloud Azure Config Pag

**dsgw210** Status System Interfaces IOT Services Logout

Mqtt Config **Azure Config** AWS Config AWS Lora Cloud ChipStack Things Stack

### Azure Manage

This is configure for Azure Connect String

#### Azure Config

AzureConnectionString

## Cloud AWS Mqtt Config Page

**dsgw210** Status ▾ System ▾ Interfaces ▾ IOT Services ▾ Logout

Mqtt Config Azure Config **AWS Config** AWS Lora Cloud ChipStack Things Stack

### AWS Manage

This is configure for aws config

#### AWS Config

AWS Server

AWS Port

AWS ClientId

AWS Root Ca

Choose File

No file chosen

AWS Cert

Choose File

No file chosen

AWS Private Key

Choose File

No file chosen

AWS Subscribe Topic

AWS Publish Topic

## System Setting Page

**dsgw210** Status ▾ System ▾ Interfaces ▾ IOT Services ▾ Logout

**System**  
Here you can configure the basic system settings like its hostname or the timezone.

### System Properties

General settings **Language and Style**

Local Time

Mon Apr 17 09:09:18 2023

Sync with browser

Hostname

Timezone

## System Administrator Setting page

**dsgw210** Status System Interfaces IOT Services Logout

**Router Password**  
Changes the administrator password

System  
Administration  
Backup / Flash Firmware  
Reboot

Password  
Confirmation

## System Upgrade Page

**dsgw210** Status System Interfaces IOT Services Logout

**Flash operation**

Actions

System  
Administration  
Backup / Flash Firmware  
Reboot

**Flash new firmware image**  
Upload a sysupgrade-compatible image here to replace the running firmware. Check "Keep settings" to retain the compatible firmware image).

Keep settings:  
☒

Image:  
Choose File No file chosen Flash image...

## Reboot Page

**dsgw210** Status System Interfaces IOT Services Logout

**System Reboot**  
Reboots the operating system

Perform reboot

System  
Administration  
Backup / Flash Firmware  
Reboot

Logout Page

dsgw210

Status ▾System ▾Interfaces ▾IOT Services ▾Logout

Status

System

Hostname	dsgw210
Model	aarch64
Firmware Version	Host System SDK / LuCI (SVN)
Kernel Version	4.4.194
Local Time	Mon Apr 17 09:11:28 2023
Uptime	716
Load Average	0

Memory

Total Available	2005000
Free	1906152
Cached	36984
Buffered	5620



## 9. Application Layer Development

### 9.1. Led

This board has six leds can be controlled by software, they are led0 ~ led5.

### 9.2. Turn on led with red

```
1. echo 15,0,0 > /sys/class/leds/led0/rgb_value
```

### 9.3. Turn off led

```
1. echo 0,0,0 > /sys/class/leds/led0/rgb_value
```

### 9.4. Button

This board has one button that can be used by the software. When pressed, the system will auto call the button script in the [/etc/rc.button/battery\\_power\\_switch](#)

Here are two incoming parameters:

- SEEN: this is the time, unit seconds
- ACTION: this is the action, it will pressed or release

See, the example has one function in the button script

Long pressed 3 seconds to close the battery power supply:

```
1. #!/bin/sh
2.
3. [ "$ACTION" = "pressed" ] && {
4. logger "battery_power_switch pressed, SEEN:$SEEN"
5. exit 0
6. }
7.
8.
9. [ ! "$ACTION" = "released" ] && {
```

```
10. exit 0
11. }
12.
13. logger "battery_power_switch released, SEEN:$SEEN"
14.
15.
16. [ $SEEN -gt 3 ] && {
17. logger "battery switch ing"
18. val=`i2cget -f -y 0 0x6a 0x09`
19. ret=$?
20. [ "$ret" = "0" ] && {
21. if [ "$val" = "0x20" ]; then #
22. logger "open battery !!"
23. i2cset -f -y 0 0x6a 0x09 0x44
24. else
25. logger "close battery !!"
26. i2cset -f -y 0 0x6a 0x09 0x20
27. fi
28. }
29. exit 0
30. }
```

## 9.5. Ethernet

This Gateway has two Internet ports:

wan port eth0

see the config

```
1. root@dsgw210:~# cat /etc/network/interfaces
2. auto lo
3. iface lo inet loopback
4.
5. iface wlan1 inet static
6. address 192.168.66.1
7. netmask 255.255.255.0
8.
9. auto wlan0
10. iface wlan0 inet dhcp
11. wireless_mode managed
12. wireless_essid any
13. wpa-driver wext
14. wpa-conf /etc/wpa_supplicant.conf
```

```
15.metric 1
16.
17.auto eth0
18.iface eth0 inet dhcp
```

## 9.6. Wi-Fi

This Gateway has a 2.4g/5.0g radio, it worked at sta mode when startup see the default config:

```
1. root@dsgw210:~# cat /etc/network/interfaces
2. auto lo
3. iface lo inet loopback
4.
5. iface wlan1 inet static
6. address 192.168.66.1
7. netmask 255.255.255.0
8.
9. auto wlan0
10. iface wlan0 inet dhcp
11. wireless_mode managed
12. wireless_essid any
13. wpa-driver wext
14. wpa-conf /etc/wpa_supplicant.conf
15. metric 1
16.
17. auto eth0
18. iface eth0 inet dhcp
```

```
1. root@dsgw210:~# cat /etc/wpa_supplicant.conf
2. ctrl_interface=/var/run/wpa_supplicant
3. ctrl_interface_group=0
4. update_config=1
5.
6. network={
7. key_mgmt=NONE
8. }
```

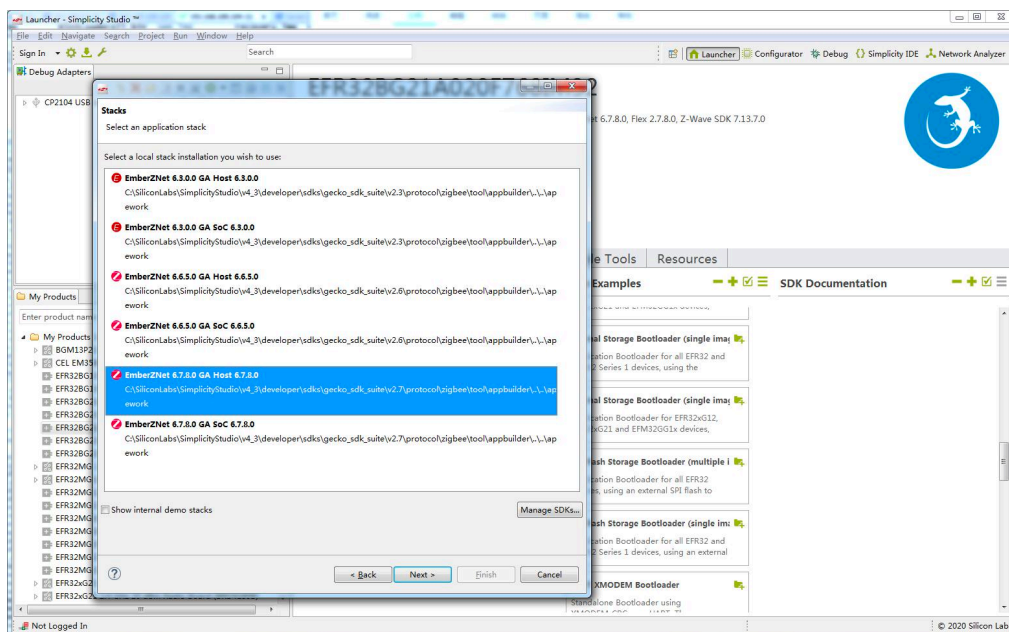
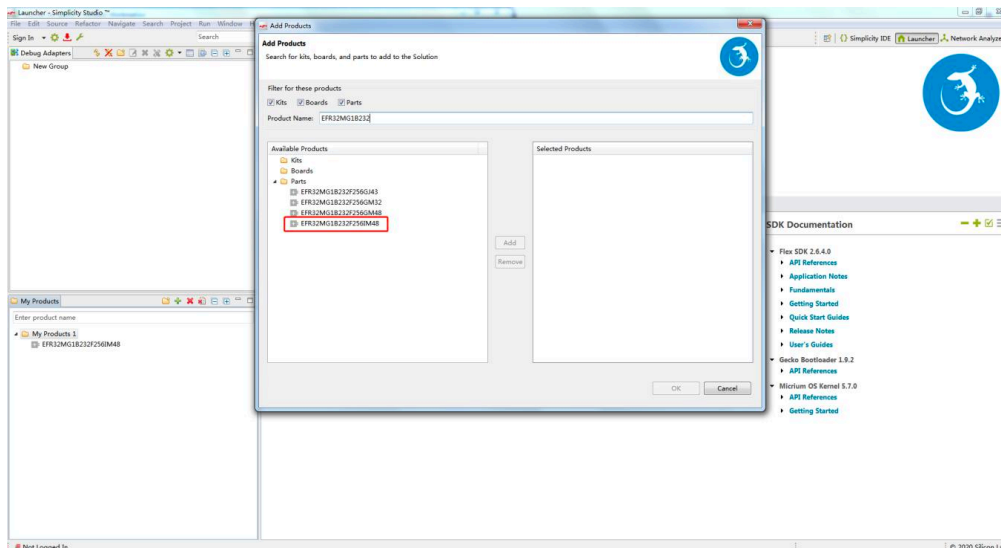
## 9.7. Zigbee

This Gateway supports Zigbee Module EFR32MG1B232 or EFR32MG21

### Host Development Demo Example

#### NCP Development

User can obtain the zigbee module's NCP program in simplicity studio, the module number is **EFR32MG1B232F256G**



For detailed information to flash image to the zigbee module, please refer to document **ZIGBEE MODULE FLASH FIRMWARE v1.0**

For SDK to develop program inGateway, please refer to document API Reference for EmberZNet PC Host. It can be found in the ss5's directory of

[C:\SiliconLabs\SimplicityStudio\v4\\_3\developer\sdk\gecko\\_sdk\\_suite\v2.7\protocol\zigbee\documentation](C:\SiliconLabs\SimplicityStudio\v4_3\developer\sdk\gecko_sdk_suite\v2.7\protocol\zigbee\documentation)

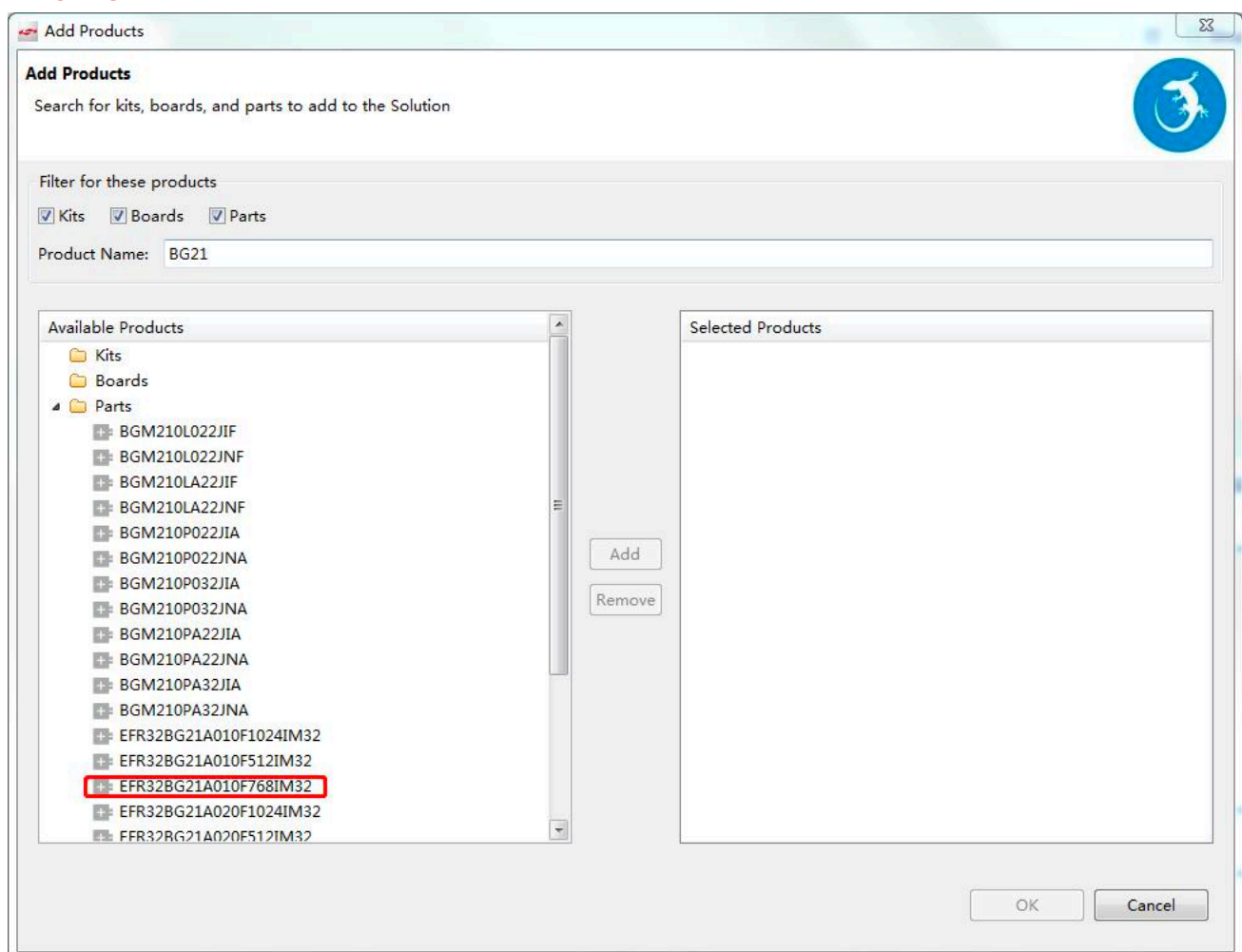
## 9.8. Bluetooth

ThisGateway supports Bluetooth Module **EFR32BG21**

### Host Development Demo Example

#### NCP Development

User can obtained the ble module's NCP program in simplicity studio, the module number is **ERF32BG21**

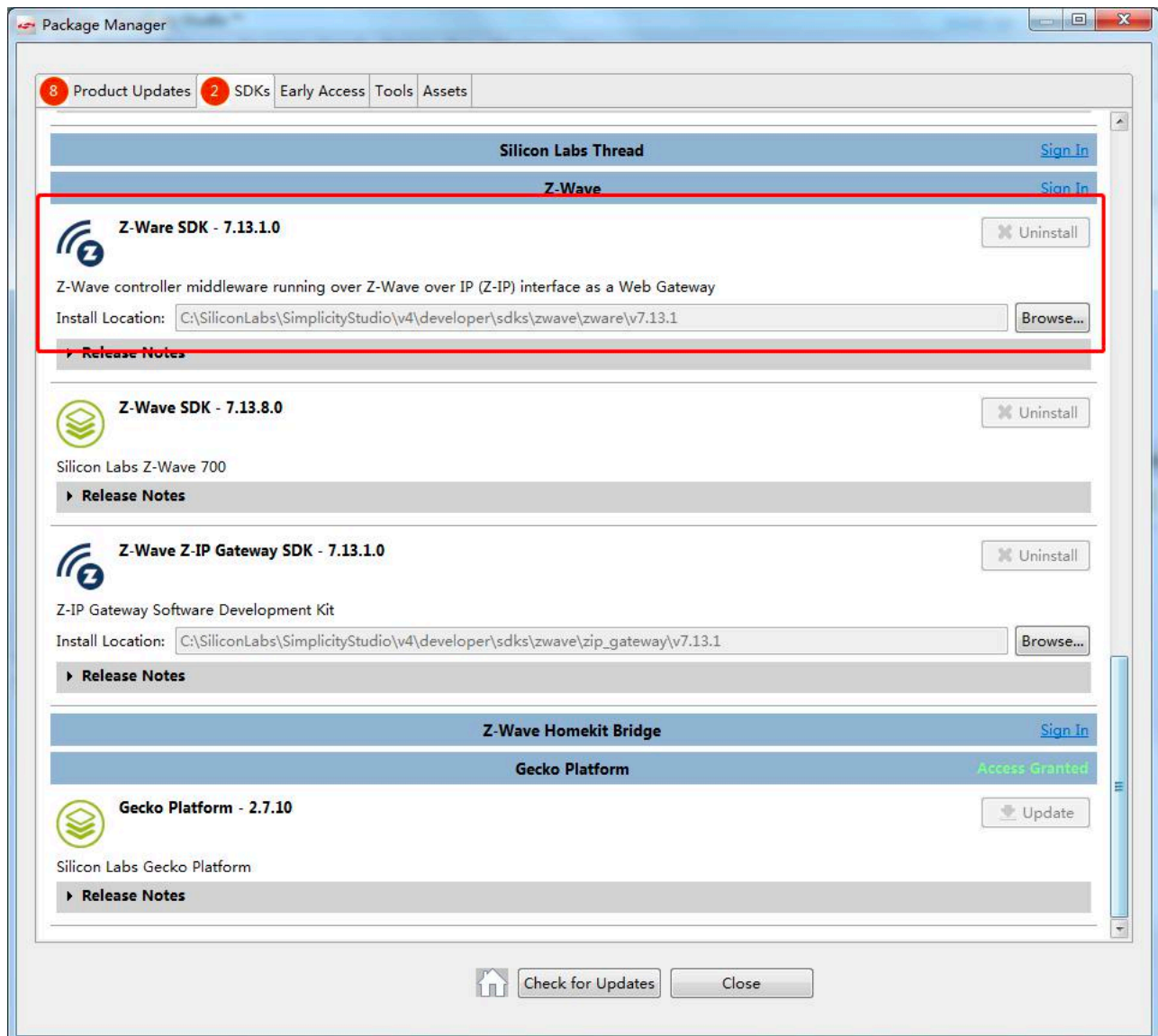


## 9.9. Z-Wave

ThisGateway supports Z-Wave Module **ZG130S**

### Host Development Demo Example

#### NCP Development



Find the correct Z-Wave module in simplicity studio, then follow the same guide in section 10

## 9.10. LTE

ThisGateway supports LTE Module **BG96** or other Module

We use the pppd to dial and manage the bg96.

Here is some config and steps about the lte module

- pppd dial scripts

```
1. root@dsgw210:~# ls /etc/ppp/peers/* -alh
2. -rwxr-xr-x 1 root root 366 Apr 7 07:25 /etc/ppp/peers/dial
3. -rwxr-xr-x 1 root root 163 Apr 7 07:25 /etc/ppp/peers/mo_3gmodule.dial
4. -rwxr-xr-x 1 root root 1.1K Jan 4 2021 /etc/ppp/peers/provider
```

- Do the pppd call in the foreground

Stop the demo program.

```
1. /etc/init.d/done stop;/etc/init.d/dial stop;
```

Modify the pppd to debug mode

```
1. root@dsgw210:~# cat /etc/ppp/options
2. debug // remove #, open the debug mode
3. nodetach // add this
4. #logfile /dev/null // comment this line
5. noipdefault
```

open the lte module

```
1. echo 0 > /sys/class/leds/ltepwr/brightness
2. echo 0 > /sys/class/leds/lterst/brightness
3. echo 0 > /sys/class/leds/lterf/brightness
4. sleep 1
5. echo 1 > /sys/class/leds/ltepwr/brightness
6. echo 1 > /sys/class/leds/lterst/brightness
7. echo 1 > /sys/class/leds/lterf/brightness
8. sleep 15
```

## pppd dial

```
1. root@dsgw210:~# pppd call dial
2. timeout set to 3 seconds
3. send (ate0^M)
4. expect (OK)
5. ^M
6. OK
7. -- got it
8.
9. send (at^M)
10. expect (OK)
11. ^M
12. ^M
13. OK
14. -- got it
15.
16. send (AT+CSQ^M)
17. expect (OK)
18. ^M
19. ^M
20. +CSQ: 99,99^M
21. ^M
22. OK
23. -- got it
24.
25. send (AT+COPS?^M)
26. expect (OK)
27. ^M
28. ^M
29. +COPS: 0^M
30. ^M
31. OK
32. -- got it
33.
34. send (AT+CREG?^M)
35. expect (OK)
36. ^M
37. ^M
38. +CREG: 0,0^M
39. ^M
40. OK
```



```

41. -- got it
42.
43. send (AT+CEREG?^M)
44. expect (OK)
45. ^M
46. ^M
47. +CEREG: 0,0^M
48. ^M
49. OK
50. -- got it
51.
52. send (AT+CPIN?^M)
53. expect (READY)

```

## 9.11. LoRaWAN

Stop the run lorawan demo

```
1. /etc/init.d/done stop; /etc/init.d/lora stop
```

Run the LoRa packet forward

```

1. root@dsgw210:/usr/bin# cd /usr/bin/lora/
2. root@dsgw210:/usr/bin/lora# ./lora_pkt_fwd -c ./global_conf.json
3. *** Packet Forwarder ***
4. Version: 2.1.0
5. *** SX1302 HAL library version info ***
6. Version: 2.1.0;
7. ***
8. INFO: Little endian host
9. INFO: found configuration file ./global_conf.json, parsing it
10. INFO: ./global_conf.json does contain a JSON object named SX130x_conf, parsing
    SX1302 parameters
11. INFO: com_type SPI, com_path /dev/spidev32766.0, lorawan_public 1, clksrc 0,
    full_duplex 0
12. INFO: antenna_gain 0 dBi
13. INFO: Configuring legacy timestamp
14. INFO: no configuration for SX1261
15. INFO: Configuring Tx Gain LUT for rf_chain 0 with 16 indexes for sx1250
16. INFO: radio 0 enabled (type SX1250), center frequency 471400000, RSSI offset -
    207.000000, tx enabled 1, single input mode 1

```

```
17.INFO: radio 1 enabled (type SX1250), center frequency 475000000, RSSI offset -
207.000000, tx enabled 0, single input mode 1
18.INFO: Lora multi-SF channel 0> radio 0, IF -300000 Hz, 125 kHz bw, SF 5 to 12
19.INFO: Lora multi-SF channel 1> radio 0, IF -100000 Hz, 125 kHz bw, SF 5 to 12
20.INFO: Lora multi-SF channel 2> radio 0, IF 100000 Hz, 125 kHz bw, SF 5 to 12
21.INFO: Lora multi-SF channel 3> radio 0, IF 300000 Hz, 125 kHz bw, SF 5 to 12
22.INFO: Lora multi-SF channel 4> radio 1, IF -300000 Hz, 125 kHz bw, SF 5 to 12
23.INFO: Lora multi-SF channel 5> radio 1, IF -100000 Hz, 125 kHz bw, SF 5 to 12
24.INFO: Lora multi-SF channel 6> radio 1, IF 100000 Hz, 125 kHz bw, SF 5 to 12
25.INFO: Lora multi-SF channel 7> radio 1, IF 300000 Hz, 125 kHz bw, SF 5 to 12
26.INFO: Lora std channel> radio 1, IF -200000 Hz, 250000 Hz bw, SF 7, Explicit
header
27.INFO: FSK channel> radio 1, IF 300000 Hz, 125000 Hz bw, 50000 bps datarate
28.INFO: ./global_conf.json does contain a JSON object named gateway_conf, parsing
gateway parameters
29.INFO: gateway MAC address is configured to AA555A0000000000
30.INFO: server hostname or IP address is configured to "localhost"
31.INFO: upstream port is configured to "1730"
32.INFO: downstream port is configured to "1730"
33.INFO: downstream keep-alive interval is configured to 10 seconds
34.INFO: statistics display interval is configured to 30 seconds
35.INFO: upstream PUSH_DATA time-out is configured to 100 ms
36.INFO: packets received with a valid CRC will be forwarded
37.INFO: packets received with a CRC error will NOT be forwarded
38.INFO: packets received with no CRC will NOT be forwarded
39.INFO: GPS serial port path is configured to "/dev/ttyS0"
40.INFO: Reference latitude is configured to 0.000000 deg
41.INFO: Reference longitude is configured to 0.000000 deg
42.INFO: Reference altitude is configured to 0 meters
43.INFO: Beacons period is configured to 0 seconds
44.INFO: Beacons signal will be emitted at 869525000 Hz
45.INFO: Beacons datarate is set to SF9
46.INFO: Beacons modulation bandwidth is set to 125000Hz
47.INFO: Beacons TX power is set to 14dBm
48.INFO: Beacons information descriptor is set to 0
49.INFO: ./global_conf.json does contain a JSON object named debug_conf, parsing
debug parameters
50.INFO: got 2 debug reference payload
51.INFO: reference payload ID 0 is 0xCAFE1234
52.INFO: reference payload ID 1 is 0xCAFE2345
53.INFO: setting debug log file name to loragw_hal.log
54.INFO: [main] TTY port /dev/ttyS0 open for GPS synchronization
```

55. Opening SPI communication interface

## 10. Support

Please contact our Sales and FAE colleagues for more information

## 11. Reference