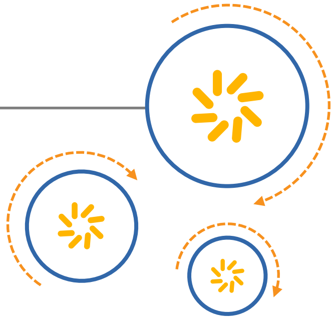
Qualcomm Technologies, Inc.



Qualcomm® Snapdragon Flight™

User Guide

80-P8822-1 Rev. A

October 26, 2016

Qualcomm Technologies, Inc.
5775 Morehouse Drive
San Diego, CA 92121
U.S.A.

.

Revision history

Revision	Date	Description
A	October 2016	Initial release

Contents

Figures

Tables

1 Introduction

This document is the user guide for Snapdragon Flight software.

1.1 References

The references listed in Table 1-1 provide additional information about topics discussed in this document.

Table 1-1 References

Title	DCN or URL
Qualcomm® Snapdragon Flight™ Developer Guide	80-P8822-2
Snapdragon Flight™ Documentation on GitHub	GitHub link
Snapdragon Flight™ Information on Qualcomm Developer Network	QDN link

2 Package Components

2.1 Snapdragon Flight development kit

- Snapdragon Flight CCA
- Snapdragon Flight Serial Console adapter and serial cable
- 4K camera module
- Optic flow module
- 5 V DC power supply and power cord
- Micro-USB cable (for ADB)
- Serial cable

2.2 Components not provided

- Wi-Fi antennas
- GPS antennas
- USB (2.0 and/or 3.0) cables

NOTE: USB 3.0 cables should be sourced as micro-A to Std A cables, with no RX/TX swap (or straight through).

The Amphenol cable works for the design.

MCN: 364-53840-4010 CABLE ASSY, USB 3.0 MICRO-A PLUG TO STD-A RCPT 1M ROHS AMPHENOL CORP (29587) RUB30-0049

3 Setup and Operation of Development Kit

3.1 Orientation

The development platform provides the Snapdragon Flight, APM, console adapter, and cable.

Figure 3-1 shows the Snapdragon Flight board with the enclosure. The Android Debug Bridge (ADB) uUSB cable is connected to the Snapdragon Flight board directly and the serial port and power are connected via separate adapters

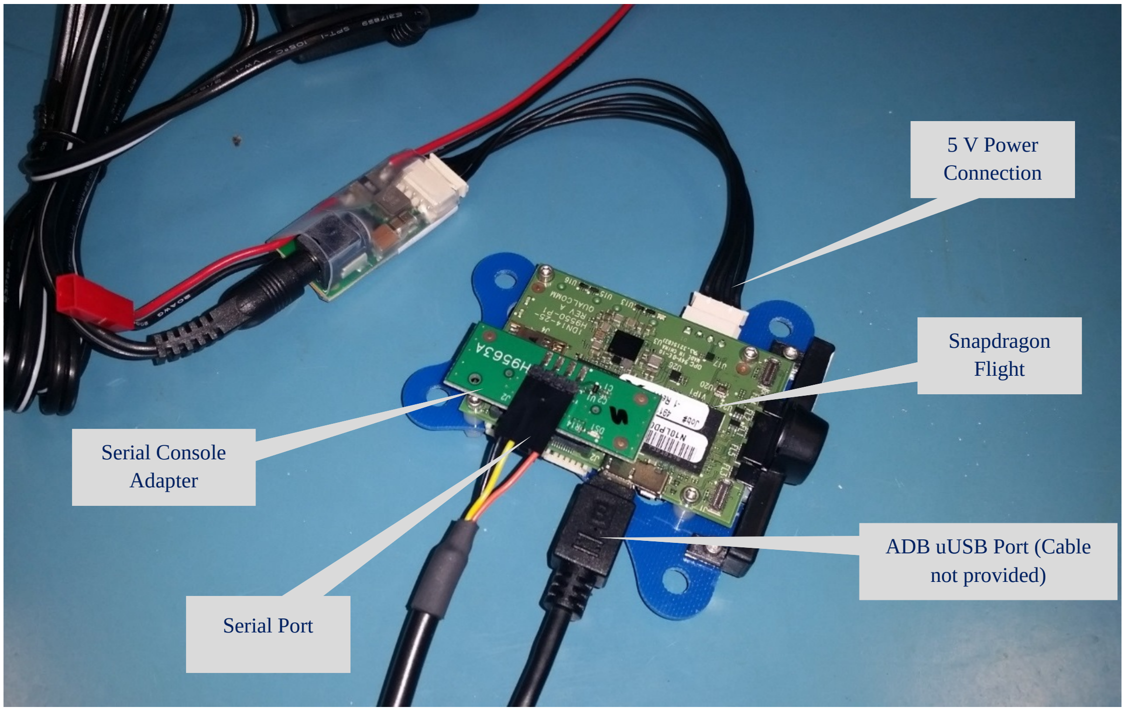


Figure 3-1 Snapdragon Flight board

3.2 Assembly

3.2.1 Connecting Snapdragon Flight and console adapter

Snapdragon Flight provides a mating connection with the console adapter. Align the connectors between the two boards and press firmly to mate the boards, as shown in Figure 3-2.

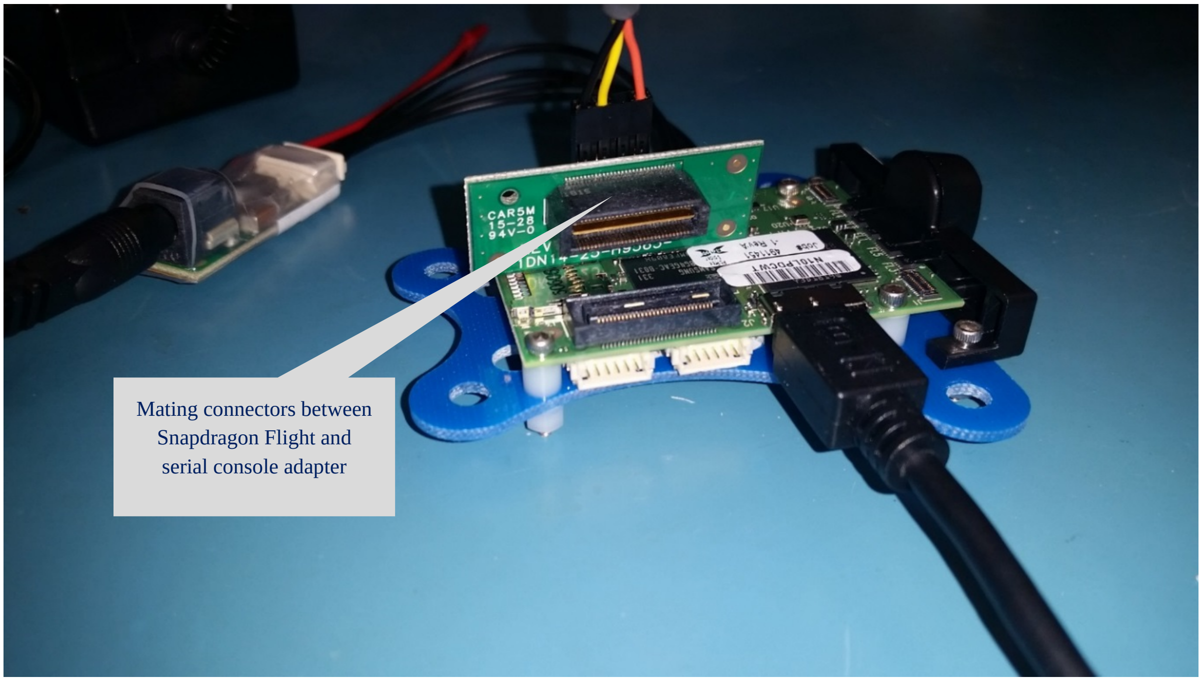

Mating connectors between Snapdragon Flight and serial console adapter

Figure 3-2 Snapdragon Flight with the console adapter

3.2.2 Connect antenna cables

Antennas are connected for the Wi-Fi and GPS components. The Wi-Fi module supports 2 x 2 MIMO (Multiple Input Multiple Output) and provides two antenna connectors. In addition, GPS provides an antenna connector.

Figure 3-3 shows the Snapdragon Flight with the antenna cables extending from the enclosure.

Wi-Fi antennas supporting a range of 2.4 GHz to 5.8 GHz can then be connected to the SMA connectors at the end of the Wi-Fi cables. A GPS antenna can be connected to the GPS cable.

To connect the µCoax to SMA cables to the Snapdragon Flight, J7 and J8 are designated for Wi Fi and J6 is used for GPS, as shown in Figure 3-3.

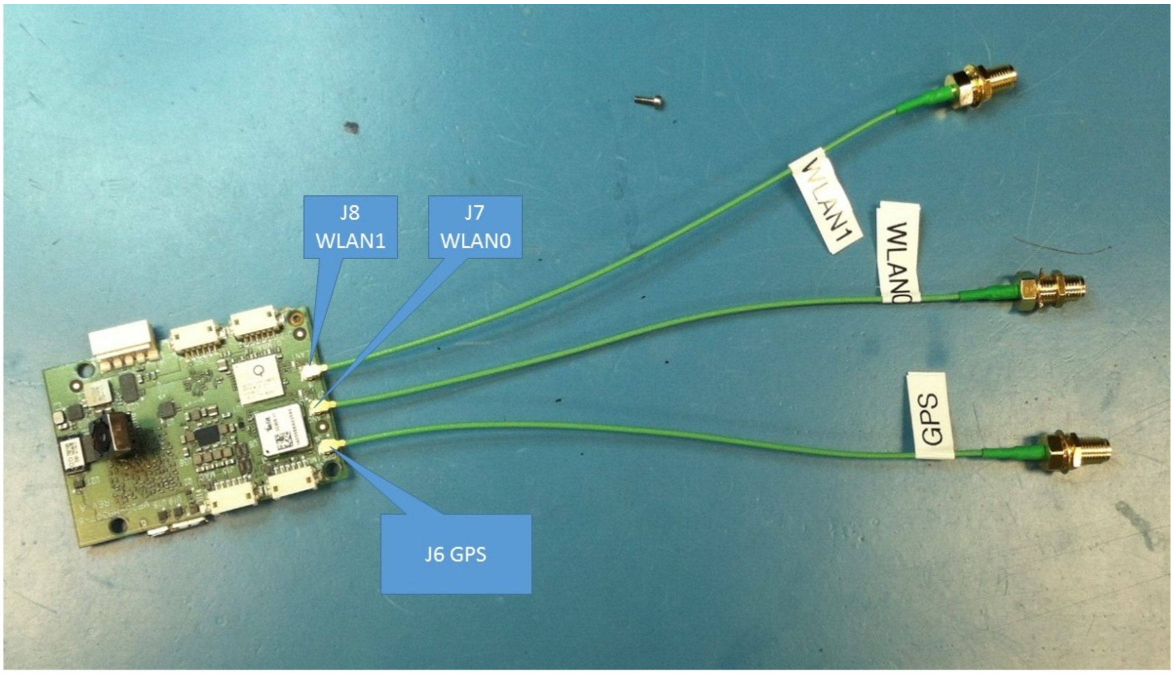


Figure 3-3 Snapdragon Flight mating antenna connectors

3.2.3 Connect ADB and serial cables

Complete the assembly by connecting the serial cable to the board and the µUSB ADB cable to the Snapdragon Flight. Connect the other end of the USB cables to a development host machine.

3.3 Power on

Connect the power adapter to the board. The board boots automatically after connecting the 5V power adapter.

3.4 Communicating with the target

Communication with the target can be facilitated with serial or ADB cable or over Wi-Fi.

3.4.1 Serial cable

The serial cable provides root access to the Linaro OS running on the target. The serial connection to the board is indicated in Figure 3-1.

- From the Linux host:
  - □ Choose the right /dev/ttyUSBx and start the serial console:

```
host> serial /dev/ttyUSB0 115200
```

- From the Windows host:
  - □ Once you identify the serial port (using Windows Device manager), use the serial application (putty, secureCRT, or mobaXterm etc.,) you choose to connect to the port.
  - □ The baud rate is 115200.

3.4.2 ADB

The instructions and commands in the following sections can be run using the serial interface or via the ADB. The ADB connection to the board is indicated in Figure 3-1.

To install ADB on an Ubuntu Linux distribution, use the following instructions.

To install adb:

```
apt-get install android-tools-adb android-tools-fastboot
```

If you do not have the proper repositories:

```
add-apt-repository ppa:phablet-team/tools && sudo apt-get update
```

To list ADB devices:

```
adb devices
```

To run the ADB shell and open bash, type the following:

```
adb shell
bash
```

For ADB installation instructions on other operating systems:

http://lifehacker.com/the-easiest-way-to-install-androids-adb-and-fastboot-to-1586992378

For more information on ADB:

http://developer.android.com/tools/help/adb.html

3.4.3 Wi-Fi

The system can be configured as an Access Point (AP) or Station (STA).

3.4.3.1 AP mode

The software AP (SoftAP) mode is facilitated with the host access point daemon (hostapd). Hostapd is a user space software access point capable of turning normal network interface cards into access points and authentication servers.

Enable AP mode

To configure the system for AP mode:

1. Run the following on the command line (via serial interface):

```
/usr/local/qr-linux/wificonfig.sh -s softap
```

2. Reboot the system.

Configure AP Mode

By default, the AP mode is configured for 802.11n, WPA2 authentication, and 2.4 GHz. The SSID is autogenerated with the installation of the platform image and the passphrase is set to *password*.

The hostapd configuration file is located at /etc/hostapd.conf on the target.

```
interface=wlan0
driver=nl80211
ieee80211d=1
ieee80211n=1
#
#
# Uncomment *one* of the following:
#
# 2.4Ghz setup
hw_mode=g
channel=2 # some channel in 2.4Ghz band
#
# or
#
# 5Ghz setup
#hw_mode=a
#channel=40 # some channel in 5Ghz band
#
#
ssid=Atlanticus_1952
macaddr_acl=0
ignore_broadcast_ssid=0
wpa=2
wpa_passphrase=password
wpa_key_mgmt=WPA-PSK
wpa_pairwise=TKIP CCMP
rsn_pairwise=CCMP
```

The file can be modified to suit the desired configuration. After changing the configuration, reboot the system for the changes to take effect.

See the following websites for information on common configurations of hostapd:

https://wiki.gentoo.org/wiki/Hostapd#Sample_configurations

http://linuxwireless.org/en/users/Documentation/hostapd/#Common_Options

Once the system is booted, the wlan0 IP address is listed as 192.168.1.1, which identifies the devices as the gateway.

```
> ifconfig wlan0

wlan0 Link encap:Ethernet HWaddr 00:03:7f:20:64:06
inet addr:192.168.1.1 Bcast:192.168.1.255 Mask:255.255.255.0
UP BROADCAST MULTICAST MTU:1500 Metric:1
RX packets:9 errors:0 dropped:0 overruns:0 frame:0
TX packets:12 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:1128 (1.1 KB) TX bytes:155 (155.0 B)
```

Set up the corresponding station client

From a client device, scan for the AP SSID and provide the correct authentication to correlate with the configuration of the AP. This then connects and obtains an IP address.

3.4.3.2 Station mode

Enable station mode

To configure the system for STA mode:

1. Run the following on the command line (via serial interface):

   ```
   /usr/local/qr-linux/wificonfig.sh -s station
   ```

2. Reboot the system.

Configure station mode

To change the SSID or AP configuration:

1. Edit the supplicant configuration from the command line (via the serial interface).

   For documentation details on the supplicant, see http://w1.fi/wpa_supplicant/.

2. Edit **/etc/wpa_supplicant/wpa_supplicant.conf** to configure the SSID and security options based on the corresponding AP to which the client is to connect.

3. For secure WPA2 personal configuration, specify the SSID and passkey in the following configuration. The SSIDName is the SSID to be used and XXXXXX is the pass-phrase of the AP.

```
network={
    ssid="SSIDName"
    proto=RSN
    key_mgmt=WPA-PSK
    pairwise=CCMP TKIP
    group=CCMP TKIP
    psk="XXXXXX"
}
```

4. For a non-secure AP, specify the SSID in the following configuration block. The SSIDName is the SSID of the AP.

```
network={
    ssid="SSIDName"
    proto=RSN
    key_mgmt=NONE
    pairwise=CCMP TKIP
    group=CCMP TKIP
}
```

NOTE: Due to a known limitation, an adjustment to the configuration requires the system to be rebooted to take effect.

Reboot the system when making a configuration change:

1. Launch the utility app from the command line:
   ```
   > wpa_cli (/sbin/wpa_cli)
   ```
2. Make the modified wpa_supplicant file active by typing:
   ```
   > reconfigure
   ```
3. Click OK.
4. To list available networks:
   ```
   list_networks
   ```
5. To select a network:
   ```
   select_network (index no)
   ```
6. Quit
   ```
   >quit
   ```

If necessary, run additional commands to enable or disable Wi-Fi and to obtain the status and IP address of the Wi-Fi connection.

- To enable Wi-Fi:
  ```
  > ifup wlan0
  ```
- To disable Wi-Fi
  ```
  > ifdown wlan0
  ```
- To check Wi-Fi status
  ```
  > ifconfig wlan0
  ```

3.4.4 SSH over Wi-Fi

From a host machine with route access to the Snapdragon Flight Wi-Fi network, a secure shell (SSH) connection can be established to interface with the Snapdragon Flight platform. From the serial interface, obtain the IP address of the system by executing ifconfig wlan0 on the command line; use this process to set up the SSH connection. Enter the password "linaro" when prompted.

```
$ ssh linaro@<Target IP Address>
linaro@<Target IP Address>'s password: ******
Welcome to Linaro 14.04 (GNU/Linux 3.4.0-som8064-const armv7l)
linaro@linaro-developer:~$
```

Where <Target IP Address> is the Wi-Fi IP address of the target Snapdragon Flight.

To continue session as root, use the pseudo user command.

```
su
Password: <enter root password>
```

NOTE: The password must be set the first time if it has not been set before. From the serial console or adb shell, set the root password with the following command:

```
> passwd
```

3.5 Administration

3.5.1 Provisioning Wi-Fi MAC address

When the device is first provisioned with a new image, it auto-generates a MAC address for Wi-Fi. However it is advised to use the MAC address that was specifically assigned for the board.

The Snapdragon Flight Kit comes assembled with a bottom mounted plate. On the underside of this plate is a label that includes a WLAN and BT MAC address. The “WLAN MAC” address can be used in the provisioning of the Wi-Fi MAC address.

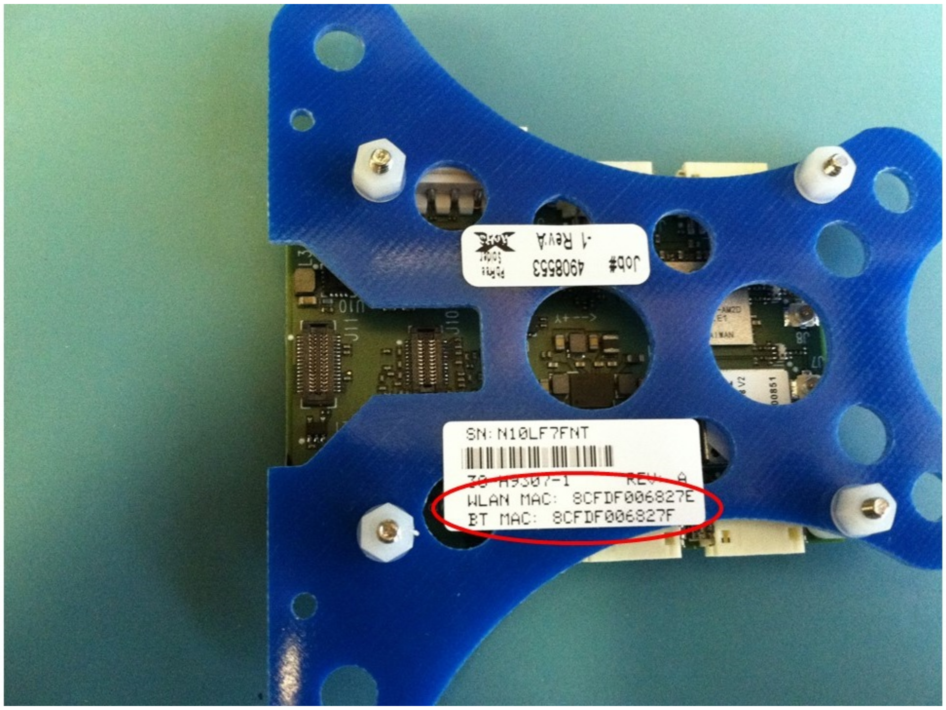


Figure 3-4 Snapdragon Flight with MAC address label

To change the MAC address of the WLAN module:

1. Obtain a valid MAC address for the device. Use MAC address on the label (as available).
2. Execute the following on the command line (via adb or serial interface)
   - ifdown wlan0
   - rm /etc/udev/rules.d/70-persistent-net.rules
   - /usr/local/qr-linux/qrl-config-macaddr.sh -i wlan -m <MAC>
   - reboot

NOTE: When specifying the MAC address, be sure to separate each octet with the ":" symbol. For example, 00:A0:C6:EC:5A:94.

4 System Features

4.1 Camera

4.1.1 Live camera recording

qcamvid is a command line application for:

- Capturing live H.264 video for recording at 30/60/120 fps
- Generating a First Person View (FPV) stream at 30 fps

4.1.1.1 Capturing live video for recording

Use the qcamvid app as described below to record live video using the hi-resolution camera.

```
usage: qcamvid [options]

  -c <input>      camera
                     hires: for imx
                     optic: for ov7251 (640x480 resolution only)
  -o <file-name>  video output file name [video.h264]
  -s              write video to stdout (disables file output)
  -r <res-name>   video resolution [1080p]
                    4k:    3840x2160 30fps
                    1080p: 1920x1080 30fps
                    1080p60: 1920x1080 60fps
                    720p:  1280x720 30fps
                    720p120: 1280x720 120fps
                    vga:   640x480 30fps
                    vga120:   640x480 120fps
  -f <mode>       focus mode [off]
                    off:  no autofocus enabled
                    auto: continuous autofocus
  -m <op-mode>    operation mode [both]
                    both:   both fpv and video recording
                    fpv:    fpv only
                    record: video recording only
  -b <bitrate>    bitrate in Mbps [40]
  -t <duration>   video capture duration in seconds [10]
  -n <count>      take count snapshot(s) [0]
                    max count supported: 20
  -l              enable logging on stderr
  -v <level>      syslog level [0]
```

```
                    0: silent
                    1: error
                    2: info
                    3: debug
  -h              print this message
```

From the serial console or adb shell, enter the following on the command line to capture a live recording and save it to a file:

```
qcamvid -t <duration>
```

4.1.1.2 Resolution

Use the -r option to specify the resolution of the recorded file. The default is 1080p. The following are possible resolution values:

- 1080p
- 4k
- 720p
- VGA
- 1080p60
- 720p120
- vga120

For example, `qcamvid -r 4k`.

The resolutions with fps sets the camera in High Frame Rate mode (with respective fps values). Resolutions without the fps mentioned are in the normal fps mode (30 fps).

4.1.1.3 Output file

By default, if not specified otherwise, the output file name is created in the current working directory and is named video.h264.

To specify a different location and/or filename, use the following -o switch:

```
qcamvid -o <desired path/file name>
```

4.1.1.4 Viewing the recorded file

1. Copy the output video file (default video.h264) from the device to the host machine using scp or adb:

   ```
   adb pull /path/to/video.h264
   ```

2. Open the video file in the VLC media player.

4.1.2 Generating a live FPV stream

When the qcamvid application is started, an FPV stream is automatically generated. The system sends a video stream when a client connects to the system using RTSP.

4.1.2.1 Configuration file

There is a configuration file located here: /etc/fpv.cfg. The file specifies the main parameters of the application. An example config file is shown here.

```
# camera preview width
width=1280
# camera preview height
height=720
# jpeg quality
quality=59
# the h264 bitrate in bps while rave = 1 (rave off)
bitrate=2000000
# debug level – 0: nodebug, 1: some, 2: most
debug=2
# on which network device shall rtsp server do streaming
net=wlan0
# stream jpeg or h264 - 0: jpeg, 1: h264
type=1
# hwacc - 1: use hardware omx  encoder, 0: use openh264 software encoder
hwacc=1
# dump - 0: no dump, 1: dump yuv file, 2: dump jpeg file
dump=0

###### For rave(dynamic video rate adaptation feature)parameters ##########
# window length, unit is ms
win_length=200
alpha=0.5
beta=4.0
gamma=0.6
Td=3
Sd=5
theta1=1.2
theta2=10.0
delta=1.2
min_Td=-100
# 0: rave on, 1: rave off
rave=0
cycle=4
rab_ratio=3
retrans_thres=1.3
fast_win_len=10
qvga_retrans_thres=1.4
qvga_rab_ratio=4
#match with the code: thres variable
#rate_report_thres=0.9
```

4.1.2.2 Starting FPV

FPV is automatically started when qcamvid is started. FPV can also be started without video recording using the FPV-streamer-app application.

4.1.2.3 Stopping FPV

FPV stops when qcamvid stops. There is no separate command to stop FPV.

4.1.2.4 Connecting to the FPV stream

A client using a video player (such as VLC Player) can use the following URL to connect to the video stream generated by the FPV app:

```
rtsp://<IP Address of system>/fpvview
```

4.1.2.5 Recommended setting changes for the VLC Player

The following settings are recommended for the VLC Player for best results. VLC Player Settings can be accessed via:

Tools → Preferences → Input/Codecs → Network

Change the “Default caching policy” to “Lowest Latency”

4.1.2.6 Dynamic video rate adaptation (rave)

Dynamic video rate adaptation is a feature wherein the encoded video stream’s bitrate, fps, and resolution are set dynamically based on the varying Wi-Fi bandwidth and channel quality, prior to streaming via RTSP. This helps minimizes latency and broken picture issues.

This feature can be enabled/disabled by setting the rave parameter in the file fpv.cfg.

4.1.2.7 Known issues

- No fps information is seen in the output H264 video. When video is played in the VLC player, it runs it only at 25 fps.
- Continuous Auto Focus is disabled due to lack of proper camera mounting on the Snapdragon Flight board. Vibrations caused by the AF lens movement can damage the camera if not properly mounted.
- Running the qcamvid app at 4K resolution results in degraded performance as thermal limitations are invoked.
- VLC drops frames if the network caching is set to 10 ms. The workaround is to set the VLC’s predefined Lowest Latency setting at the expense of increased lag. Alternately, use the MPlayer for FPV viewing with the *nocache* option.
- Image/video for 720p120: 1280x720 at 120fps will have reduced FOV due to a sensor limitation. The sensor lacks native support for 720p at 120 fps, but can achieve it by reducing FOV.

4.1.2.8 Debug logging with verbosity levels

The Qcamvid verbosity option (syslog level) displays debugging information based on the levels.

- Level 0 (default) – Silent; no information is displayed.
- Level 1 – Only KPI (key performance indices) information such as fps and streaming Mbps is displayed.
- Level 2 – KPI and frame information are displayed.
- Level 3 – KPI, frame information, functions calls, errors, and stack traces are displayed.

All information is prepended with a timestamp in the seconds.microseconds format (for example, 12345.987654).

The KPI displayed in levels 1 and up consists of the following:

- fps – Number of frames received from the camera in the last second; this is common to both the recorded file and FPV.
- Mbps – Bitrate of FPV stream transmitted over the configured interface.

Frame information displayed in levels 2 and up has the type of stream (fpv, 4K, or optic flow) prepended.

- [FPV]
- [4K]
- [OF]

All logging goes to /var/log/syslog. Adding the -l option to the qcamvid command line prints it on stdout also.

4.1.3 Camera test application

The system provides the camera-test application to capture image streams provided by the HiRes and optical flow cameras.

To run this application, log in to the device as a root (using serial, adb shell, or SSH). This application is written using the camera API. See *Qualcomm Snapdragon Flight Developer Guide* (80-P8822-2) for further details on the camera API.

```
root@linaro-developer:/# camera-test -h
Camera API test application
usage: camera-test [options]
  -t <duration>   capture duration in seconds [10]
  -d              dump frames
  -i              info mode
                    - print camera capabilities
                    - streaming will not be started
  -f <type>       camera type
                    - hires
                    - optic
                    - left
                    - stereo
  -p <size>       Set resolution for preview frame
                    - 4k             ( imx sensor only )
                    - 1080p          ( imx sensor only )
                    - 720p           ( imx sensor only )
                    - VGA            ( Max resolution of optic flow and 
left sensor )
                    - QVGA           ( 320x240 )
                    - stereoVGA      ( 1280x480 : Stereo only - Max 
resolution )
                    - stereoQVGA     ( 640x240  : Stereo only )
  -v <size>       Set resolution for video frame
                    - 4k             ( imx sensor only )
                    - 1080p          ( imx sensor only )
                    - 720p           ( imx sensor only )
                    - VGA            ( Max resolution of optic flow and 
left sensor )
                    - QVGA           ( 320x240 )
                    - stereoVGA      ( 1280x480 : Stereo only - Max 
resolution )
                    - stereoQVGA     ( 640x240  : Stereo only )
                    - disable        ( do not start video stream )
  -n              take a picture with  max resolution of camera ( disabled 
by default)
                  $camera-test -f <type> -i to find max picture size
  -s <size>       take picture at set resolution ( disabled by default)
                    - 4k             ( imx sensor only )
                    - 1080p          ( imx sensor only )
                    - 720p           ( imx sensor only )
                    - VGA            ( Max resolution of optic flow and 
left sensor )
                    - QVGA           ( 320x240 )
                    - stereoVGA      ( 1280x480 : Stereo only - Max 
resolution )
                    - stereoQVGA     ( 640x240  : Stereo only )
  -e <value>      set exposure control (only for ov7251)
                     min - 0
                     max - 65535
  -g <value>      set gain value (only for ov7251)
                     min - 0
                     max - 255
  -r < value>     set fps value      (Enter supported fps for requested 
resolution)
                    -  30 (default)
                    -  60
                    -  90
  -o <value>      Output format
                     0 : YUV format (default)
                     1 : RAW format (default of optic)
  -j <value>      Snapshot format
                     jpeg: JPEG format (default)
                     raw: Full-size MIPI raw format
  -V <level>      syslog level [0]
                    0: silent
                    1: error
                    2: info
                    3: debug
-S <MASK>         Enable stats log
                    0x00:  STATS_NO_LOG , ( Default )
                    0x01:  STATS_AEC_LOG_MASK  (1 << 0)
                    0x02:  STATS_AWB_LOG_MASK  (1 << 1)
                    0x04:  STATS_AF_LOG_MASK   (1 << 2)
                    0x08:  STATS_ASD_LOG_MASK  (1 << 3)
                    0x10:  STATS_AFD_LOG_MASK  (1 << 4)
                    0x1F:  STATS_ALL_LOG
  -u <value>       focus mode [3]
                    0: auto
                    1: infinity
                    2: macro
                    3: continuous-video
                    4: continuous-picture
                    5: manual
  -h              print this message
```

4.1.3.1 Capturing images with camera selection

Capture image using hires camera

Issue the following command to capture YUV images using the HiRes camera.

```
camera-test –d –t <duration in seconds> -f hires –o 0
```

Capture images using optical flow camera

- YUV images
  - Issue the following command to capture YUV images using the optical flow camera.
    ```
    camera-test –d –t <duration in seconds> -f optic –o 0
    ```
- RAW images
  - Issue the following command to capture RAW images using the optical flow camera.
    ```
    camera-test –d –t <duration in seconds> -f optic –o 1
    ```
- Capture images using stereo camera
  - Issue the following command to capture YUV images using the stereo camera.
    ```
    camera-test –d –t <duration in seconds> -f stereo –o 0
    ```

NOTE: If you select stereo, it enables both left and right sensor. The final output is a merged image from left and right sensor.

Default/Max resolution is stereo VGA (1280x480), i.e., two VGA size images (L+R) combined

4.1.3.2 fps configuration

The fps optic can be set using –r option. For instance, use the below command to set the optic flow sensor to 60 fps.

```
camera-test -d -f optic –o 0 -r 60
```

The fps setting works for all sensors.

NOTE: All fps settings do not work for all resolutions. Example: For hires camera at 4k, the maximum fps is 30.

4.1.3.3 Preview, video, and snapshot resolution

The preview, video, and snapshot resolution can be set using the –p, -v, and -s options respectively. If these options are not set, it uses the default configuration for that sensor.

For instance, set the preview frame resolution to 4k while using the default video frame resolution at 720p.

```
camera-test –d –t <duration in seconds> -f optic –o 0 –p 4k
```

You can opt to only have preview frames by disabling video frames.

```
camera-test –d –t <duration in seconds> -f optic –o 0 –p 4k  –v disable
```

4.1.3.4 Exposure and gain setting

Manual exposure is enabled for option flow and stereo sensor (ov7251 only). Exposure can be set using –e option and gain can be set using –g option.

Some configuration examples include:

```
camera-test -d -f optic -g <GAIN_VALUE>
camera-test –d –f optic –o 1 –e < EXPOSURE VALUE > -g < GAIN VALUE >
camera-test –d –f stereo –e < EXPOSURE VALUE > -g < GAIN VALUE >
```

NOTE: The maximum value for exposure is 65535.

Effect of the exposure setting varies for different fps settings.

Saturation in exposure setting can occur much before 65535.

4.1.3.5 Take MIPI RAW snapshot

Use the following command to take a Bayer MIPI RAW snapshot using this application:

```
$ camera-test –t 1 –n –j raw
```

This creates a file with RAW extension that can be opened using the Chromatix™ tool for image quality tuning.

4.1.3.6 View captured images

If you have access to the ffmpeg utility, use it to convert the YUV file to a JPEG off-target.

1. Pull a YUV file from the camera directory containing the YUV files to a host machine.
2. Run the ffmpeg utility to convert the image to JPEG format.

   ```
   ffmpeg -s 3840x2160 -pix_fmt nv21 -i P_C512_0218.yuv P_C512_0218.jpg
   ```

4.1.3.7 Live snapshot

The live snapshot feature can be tested using –n or –s option in camera-test application.

```
camera-test -f hires -o 1 -d -t <duration> -p 1080p -v 720p –n
```

The command above tests the snapshot feature. The -n option enables snapshot mode at maximum resolution supported by the sensor. The command above does the following:

- Starts preview frames at 1080p
- Starts video frames at 720p
- While preview and video stream is on, it enables snapshot stream after 2 seconds.
- After getting a callback on snapshot stream. It saves the YUV frames.
- It encodes the YUV frames and saves a JPEG image.
- It continues collecting images for the specified duration.

The snapshot resolution can also be set using the –s option.

```
camera-test  -f  hires  -o 1  -d  -t < duration> -p 4k -v 720p –s 1080p
```

Live snapshot using qcamvid application

Use –n option in qcamvid for snapshot

-n option takes number of snapshots as a parameter.

For example, the following command takes two snapshots during the recording session

```
    $ qcamvid –n 2
```

4.1.3.7.1 Known issues

- When taking a snapshot from the qcamvid application with 4K recording enabled, only low resolution is supported. For all other resolutions, it takes the maximum resolution supported by the sensor.
- The recorded video and FPV stream with 1080p60, 720p120, and VGA120 resolution options are zoomed in.

4.1.3.8 3A stats logging

3A stats logging can be enabled using the –S option.

camera-test -f hires -S 0x2

Notes

- 3A stats logging messages are stored in syslog.
- Stats are provided by the mm-qcamera-daemon application and must be enabled with the logging level “critical”.
  - To enable:
    - i stop mm-qcamera
    - ii mm-qcamera-daemon –v 1
    - iii camera-test -f hires –S 0x02 (On another adb shell)
    - iv cat /var/log/syslog (to see the logs)

- The following is a list of log types:
  - □ 0x00: STATS_NO_LOG , (Default)
  - □ 0x01: STATS_AEC_LOG_MASK (1 << 0)
  - □ 0x02: STATS_AWB_LOG_MASK (1 << 1)
  - □ 0x04: STATS_AF_LOG_MASK (1 << 2)
  - □ 0x08: STATS_ASD_LOG_MASK (1 << 3)
  - □ 0x10: STATS_AFD_LOG_MASK (1 << 4)
  - □ 0x1F: STATS_ALL_LOG
- Multiple masks may be combined. For instance, run the following to view both AEC (0x01) and AWB (0x02):

```
camera-test –f hires –S 0x03
```

4.1.3.9 Focus mode selection

Focus mode can be selected using the –u option. The focus mode is a continuous video focusing mode by default.

```
camera-test –n –u 1
```

This command creates a snapshot with focus to infinity. “-u 3” creates a snapshot with a continuous video focus mode.

4.1.3.10 Known issues

- Run camera-test with the “-d” option enabled.
- The current camera-test application version does not support 4K resolution snapshots. Use the qcamvid app instead.
- Only the optic flow sensor supports the RAW output format.
- The stereo camera supports only the YUV output format.
- Stereo images may appear washed out and saturated under certain lighting conditions as exposure and gain settings do not always get applied.
- Focus mode has been tested on IMX135. Specified lens position cannot be specified for manual focus mode yet.
- The actuator can be moved with a hardcoded lens position for IMX214. AF tuning is to be done for IMX214.

4.1.4 Video encoder test application

The system provides an H.264 video encoder test application.

To use the encoder test application:

1. Copy YUV files to the file system.
2. Edit the configuration. Provide the path to files and resolutions in the config file (i.e., 2160p, 1080p, VGA, and QCIF).

3. Run the encoder application from the command line.
4. The output file can be played on the PC using a VLC Player.
5. Initiate the encoder from the command line:

```
mm-venc-omx-test ENCODE <cfg_file> 1
```

   The <cfg_file> provides configuration details that include the source and output file locations, resolution of input/output, type of codec, profiles to be used, etc.

   The following is a sample config file for H.264 encoding:

```
# 4K input/output with H264 baseline
FrameWidth = 3840
FrameHeight = 2160
OutputFrameWidth = 3840
OutputFrameHeight = 2160
Codec = H264
Profile = H264_BASELINE
FPS = 30
NumFrames = 1000
InBufferCount = 9
OutBufferCount = 5
InFile = /tmp/4K.yuv
OutFile = /tmp/4K.h264
CABAC = 1
Deblock = 1
```

4.1.4.1 Video encoder configuration values

Table 4-1 Video encoder configuration values

Codec	Profile	Level	Max resolution	FPS	Bitrate (Mbps)
H.264	Baseline	5.1	1920 x 1080	30	40
H.264	Baseline	5.1	3840 x 2160	30	60
H.264	Baseline	5.1	3840 x 2160	30	100
H.264	Main	5.1	1920 x 1080	30	40
H.264	Main	5.1	3840 x 2160	30	60
H.264	Main	5.1	3840 x 2160	30	100
H.264	High	5.1	1920 x 1080	30	40
H.264	High	5.1	3840 x 2160	30	60
H.264	High	5.1	3840 x 2160	30	100
MPEG-4	Simple	8	1920 x 1080	30	40
MPEG-4	ASP	5	1920 x 1080	30	40
H.263	0	70	864 x 480	30	2

4.1.4.2 Verified test encoder combinations

Table 4-2 Verified test encoder combinations

Codec	Profile	Max resolution
H.264	Baseline	1920 x 1080
H.264	Baseline	3840 x 2160
H.264	Main	1920 x 1080
H.264	Main	3840 x 2160
H.264	High	1920 x 1080
H.264	High	3840 x 2160

4.1.5 Concurrency

Video recording, FPV, and optical flow are available for concurrent usage.

4.1.5.1 Scenarios

Table 4-3 lists the scenarios that are available for concurrency.

Table 4-3 Concurrency scenarios

Scenario	Video recording	FPV	Optical flow
1	1080p	720p	VGA
2	4K	720p	VGA
3	1080p	VGA	VGA
4	4K	VGA	VGA

4.1.5.2 Running a concurrency scenario

1. Start the qcamvid app as described in Section 4.1.1.1.
   a. Set up the video recording parameters as described in Section 4.1.1.1.
2. When qcamvid is started, FPV will also be started automatically.
3. Set up the FPV parameters as described in Section 4.1.2.
4. Start the optical flow captures as described in Section 4.1.3.
5. Select the optical flow camera by specifying the “-f optic” option.

4.2 Software update

The software update feature refers to support for updating the system via update packages (zip files). This is useful for providing maintenance releases when devices are already deployed in the field and for pushing new features, security fixes, or bug fixes.

4.2.1 Reboot2fastboot

This tool sets the mode that the device should be rebooted into. There are three possible modes:

- bootloader – Reboots into Fastboot mode (stays in LK) and allows flashing via fastboot.
- recovery – Boots the recovery bootable image. The bootable image runs the recovery binary then reboots into Linux.
- edl – Reboots into Emergency Download mode (stays in PBL) and allows flashing via QFIL.

The system goes to fastboot if no option is given, and -h shows the help message.

For example:

```
reboot2fastboot recovery
```

Use the reboot tool from Linux to reboot into Linux while in Recovery mode.

4.2.2 Starting an update

To start a software update, perform the following:

1. Push the update zip file to the device rootfs, for example /tmp/update.zip.

   ```
   % adb push eagle8074-ota.zip /tmp/eagle8074-ota.zip
   ```

2. Run the install-update script:

   ```
   % adb shell install-update /tmp/eagle8074-ota.zip
   ```

The device reboots into Recovery mode to apply the update. It then reboots back to the normal system, installs all the deb packages, and reboots one last time to conclude the process. No action can be performed during this time. LEDs flash green to indicate that the update is underway and flash red when any error occurs.

4.3 Factory reset

Factory reset restores the board state to its original state, when it was first flashed in the factory or by the developer. It removes any user modifications from the system, including installed packages, modified configuration files, or installed update packages. The effect is that the eMMC has the same contents after the factory reset as when it was first flashed via USB.

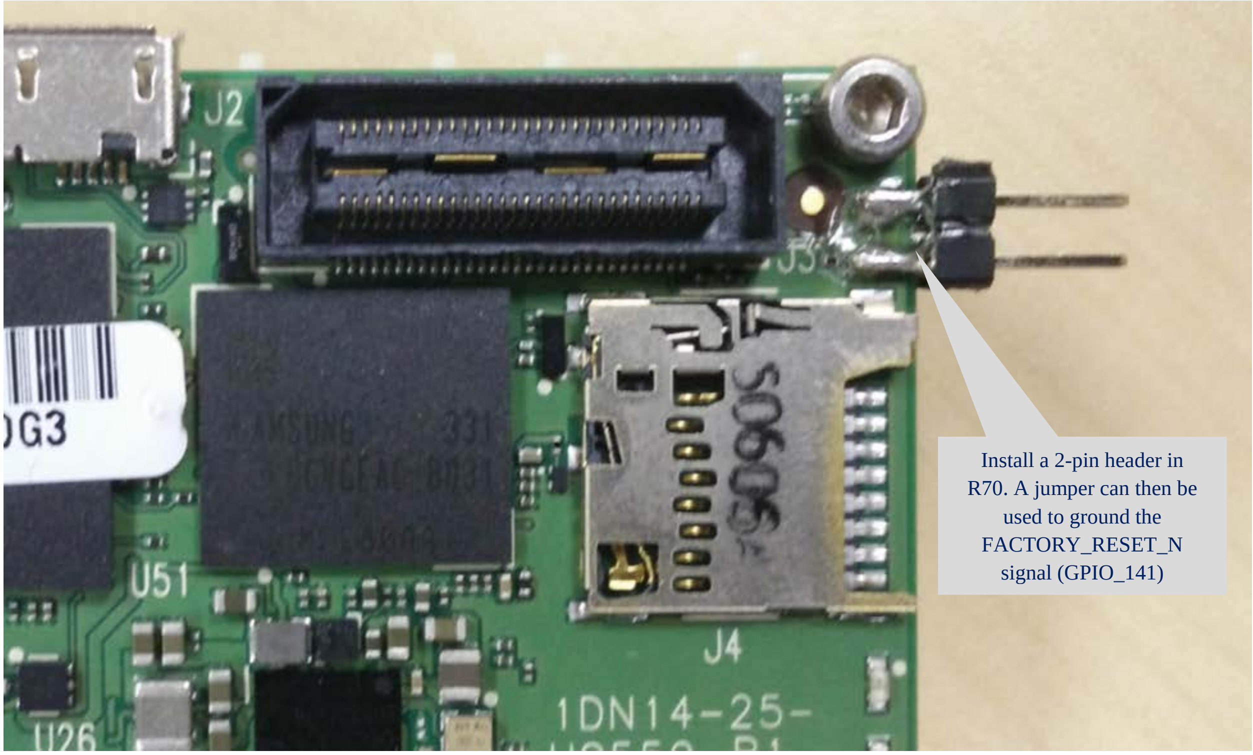
The factory reset can be initiated in two ways:

- From the command shell on the device, run the following command to initiate the factory reset process. This reinstalls all images back to a known state.

  ```
  $ factory-reset
  ```

- By pulling GPIO_141 to GND during bootup.

This GPIO is only enabled in the Snapdragon Flight P2 hardware. A jumper should be added as a rework. Power off the board, connect the jumper, then power it up. LK detects this GPIO and boots directly to Recovery mode to perform the factory data reset procedure.



Install a 2-pin header in R70. A jumper can then be used to ground the FACTORY_RESET_N signal (GPIO_141)

Figure 4-1 Snapdragon Flight P2 board with 2-pin header for factory reset jumper

5 Additional Information

5.1 Snapdragon Flight documentation on GitHub

See https://github.com/ATLFlight/ATLFlightDocs/ for additional information.

5.2 Snapdragon Flight information on QDN

See https://developer.qualcomm.com/hardware/snapdragon-flight for additional information.