EC2x&AG35-Quecopen ACDB Parameters Modifying and Importing

LTE Module Series

Rev.EC2x&AG35_Quecopen_Modifying_and_Importing_ACDB_Parameters_V1.0_Preliminary

Date: 2018-01-30

Status: Preliminary

Our aim is to provide customers with timely and comprehensive service. For any assistance, please contact our company headquarters:

Quectel Wireless Solutions Co., Ltd.
7th Floor, Hongye Building, No.1801 Hongmei Road, Xuhui District, Shanghai 200233, China
Tel: +86 21 5108 6236
Email: info@quectel.com

Or our local office. For more information, please visit:
http://www.quectel.com/support/sales.htm

For technical support, or to report documentation errors, please visit:
http://www.quectel.com/support/technical.htm
Or email to: support@quectel.com

GENERAL NOTES

QUECTEL OFFERS THE INFORMATION AS A SERVICE TO ITS CUSTOMERS. THE INFORMATION PROVIDED IS BASED UPON CUSTOMERS' REQUIREMENTS. QUECTEL MAKES EVERY EFFORT TO ENSURE THE QUALITY OF THE INFORMATION IT MAKES AVAILABLE. QUECTEL DOES NOT MAKE ANY WARRANTY AS TO THE INFORMATION CONTAINED HEREIN, AND DOES NOT ACCEPT ANY LIABILITY FOR ANY INJURY, LOSS OR DAMAGE OF ANY KIND INCURRED BY USE OF OR RELIANCE UPON THE INFORMATION. ALL INFORMATION SUPPLIED HEREIN IS SUBJECT TO CHANGE WITHOUT PRIOR NOTICE.

COPYRIGHT

About the Document

History

Revision	Date	Author	Description
1.0	2018-01-30	Grady QUAN	Initial

Contents

Figure Index

1 ACBD and QACT

ACDB is called as Audio Calibration Database, which can adjust ADSP audio parameter based on the interfaces provided by underlying driver. As Figure 1 shows, there are currently 7 ACDB files available for use. The workspaceFile.qwsp in the Figure is a project file for the QACT tool to open the ACDB file. QACT is the audio parameter adjustment tool provided by Qualcomm to calibrate ACDB.

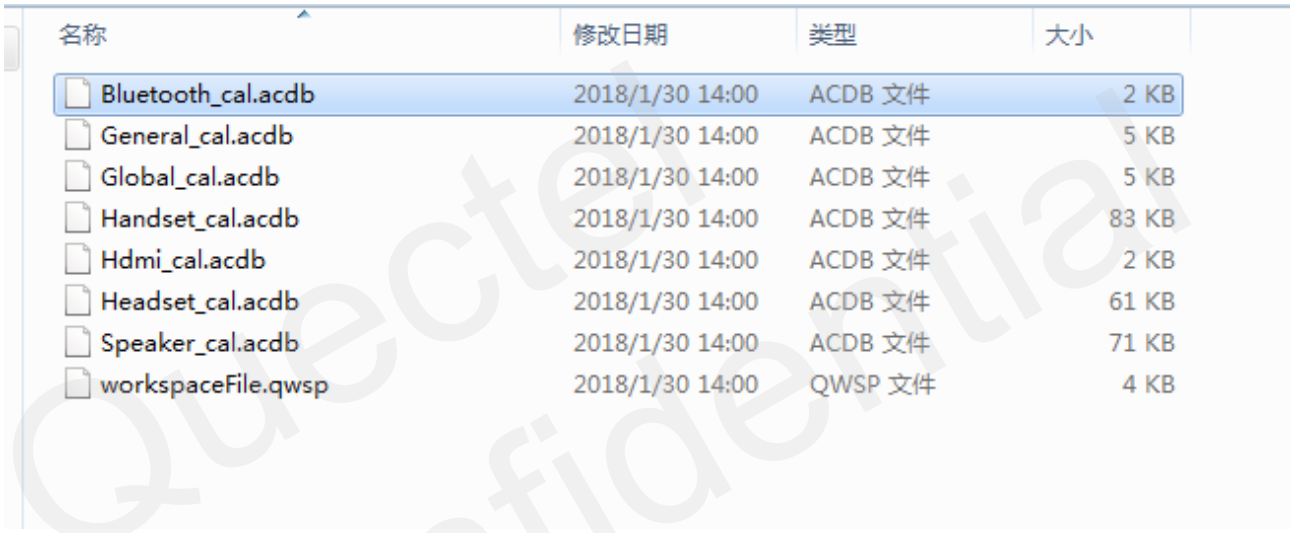

Figure 1: ACDB File

ACDB file is loaded into memory in the process of alsaucm_test. If the process is not enabled, QACT tool can't use online calibration mode.

2 How to Modify and Save ACDB File

QACT supports offline calibration mode and online calibration mode. Offline mode is to open the ACDB file with QACT, save the ACDB file after adjusting parameter in offline mode, and compile into the module. Online calibration mode supports two types of calibration, one is DSP calibration mode that can adjust DSP parameter in real time (currently only is available under Voice). Another one realizes calibration through modifying the calibration parameters in Linux side memory. After adjusting the parameters, please switch **AT+QAUDMOD** to make parameters effect. Parameters will not take effect when the module reboot.

2.1. Offline Mode

2.1.1. Saving ACDB File to Local

In offline mode，the ACDB file is in the module, export the ACDB file from the module first. Then turn on the device, open QPST, load DM port and open QACT to connect the device. As Figure 2 shows:

Figure 2: QACT Online Connection

After connecting the device, click ‘Save As’ button in the top left corner, the interface for saving ACDB file appears as shown in Figure 3. Select the path and click “OK” to save the ACDB file and project file to local.

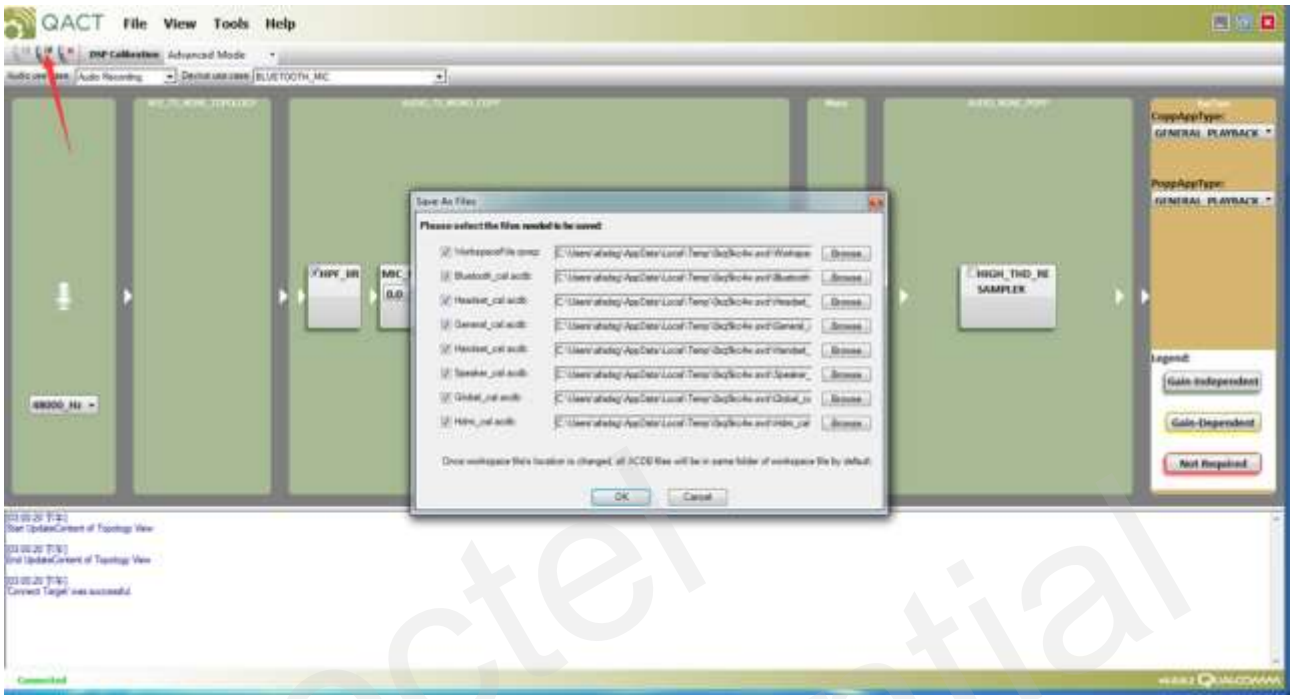

Figure 3: Saving ACDB File

2.1.2. Modifying and Saving the Local ACDB File

Turn on QACT tool again, choose offline mode and select the local workspaceFile.qwsp file, as shown in Figure 4.

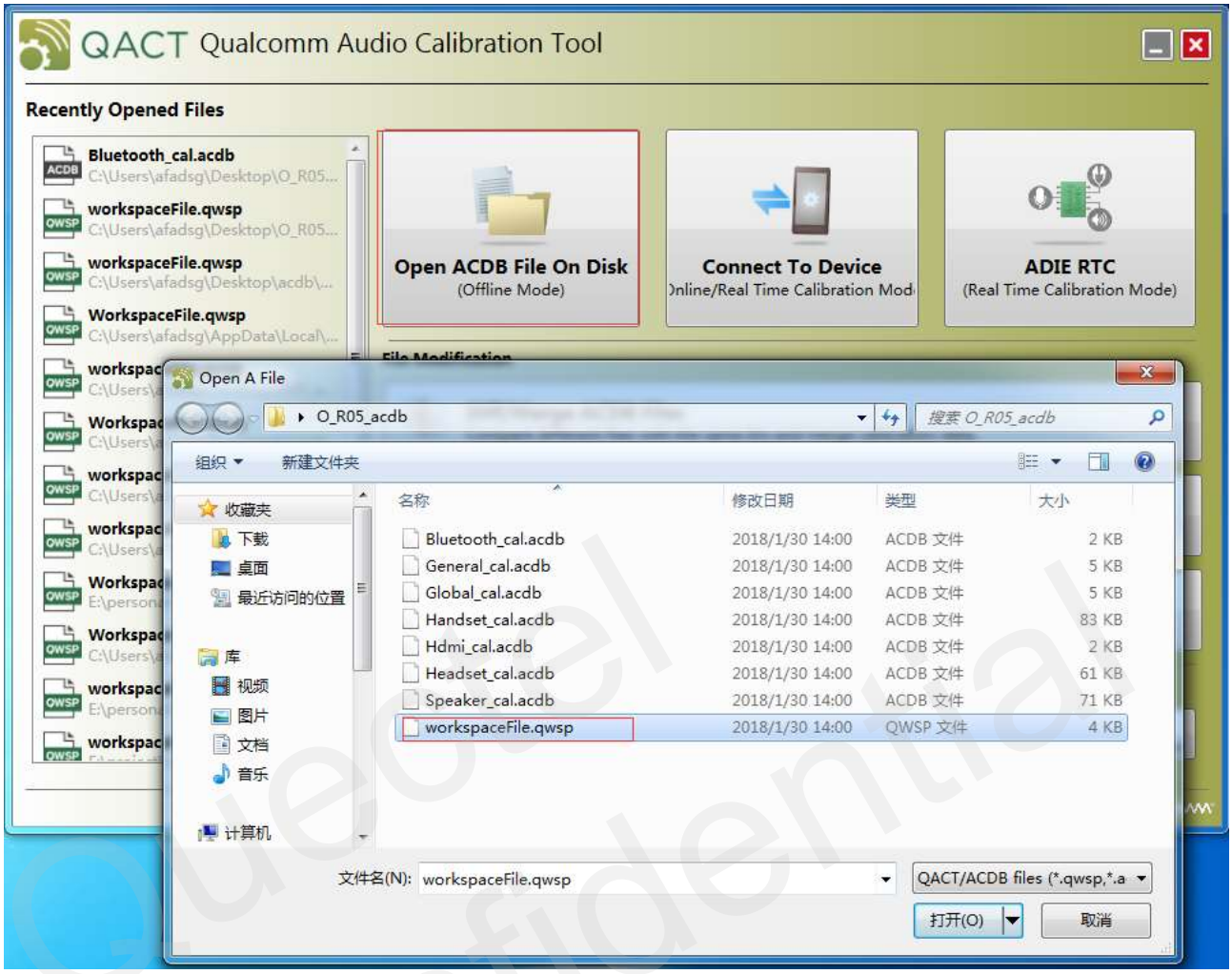

Figure 4: Opening ACDB File under Offline Mode

After turning on the file, there are 3 options under the “Audio use case” in the top left corner, they are “Audio Recording, Audio Playback and Voice” respectively. Audio Recording and Audio Playback are mainly for interphone using as shown in Figure 5. Figure 6 shows the “Device use case” under the “Audio use case”, Select the corresponding “Device use case” according to the current audio mode (Set by **AT+QAUDMOD**). Take Audio Playback as an example. If **AT+QAUDMOD** is set to 0, “Device use case”should select HANDSET_SPKR, as shown in Figure 7.

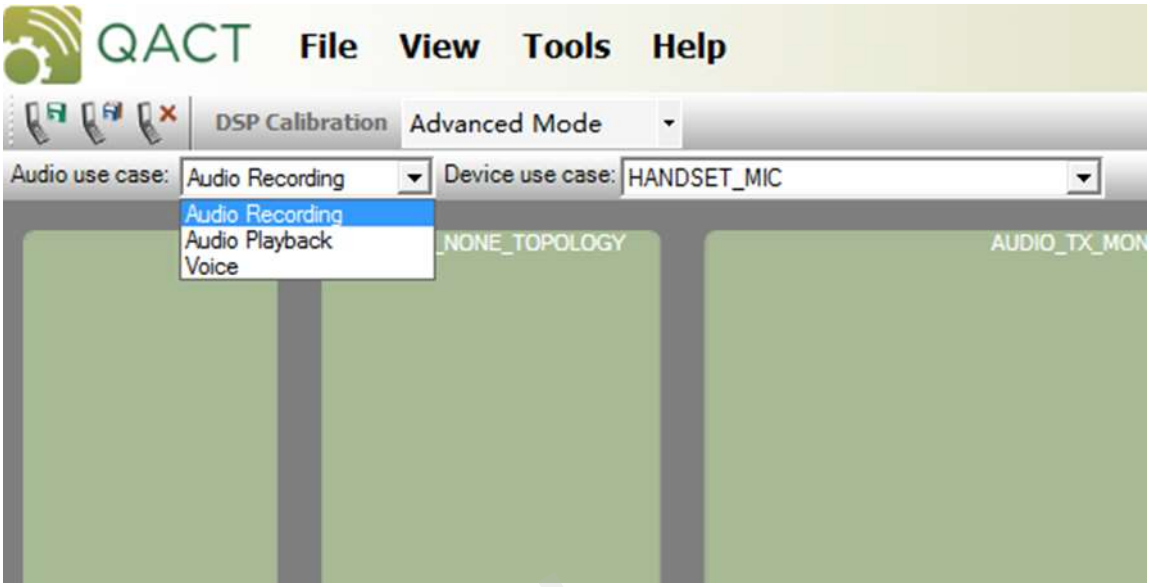

Figure 5: Audio Use Case

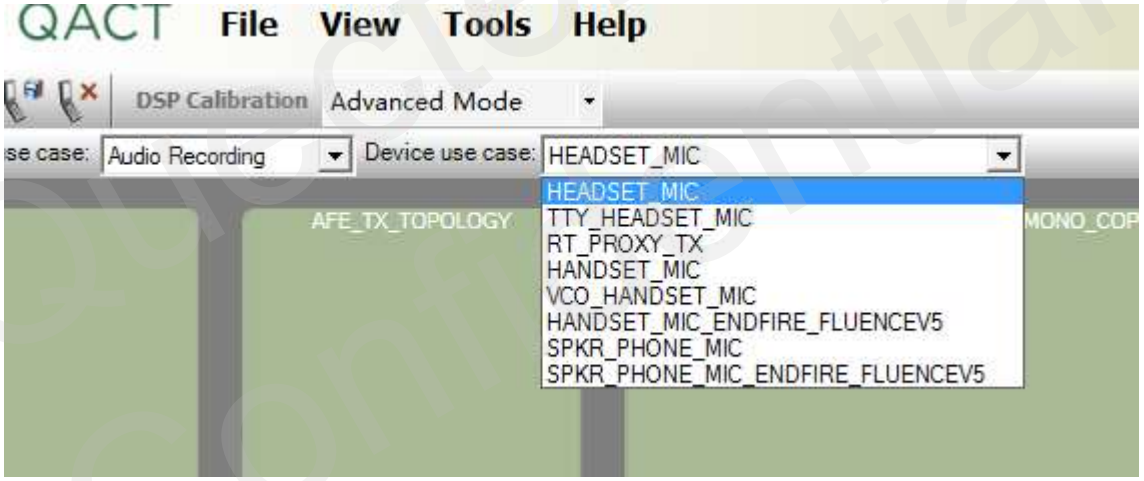

Figure 6: Device Use Case

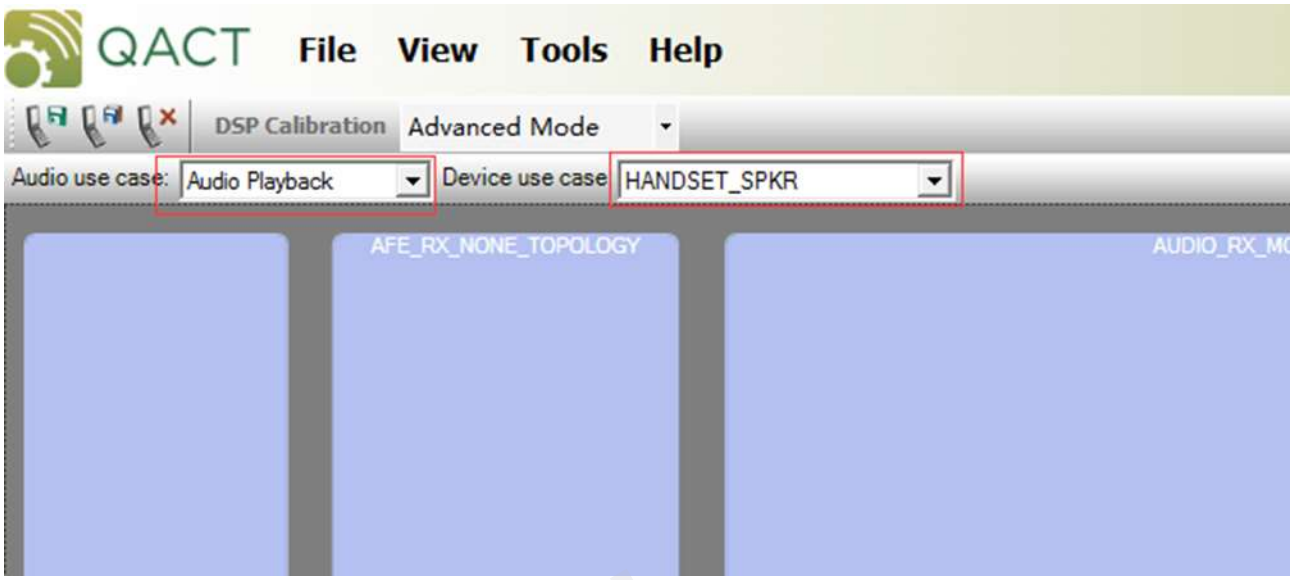

Figure 7: HANDSET SPKR

Here comes to modifying the parameters of each module in the corresponding mode, take CODEC_GAIN under HANDSET SPKR as an example. There are two ways to modify the value of this module. One is to directly modify in the drop-down box shown in Figure 8. Another one way is to double-click the module, modify it as shown in Figure 9, then click 'Set to ACDB' button. After the modification, click 'Save' or 'Save As' button in the upper left corner to save ACBD file.

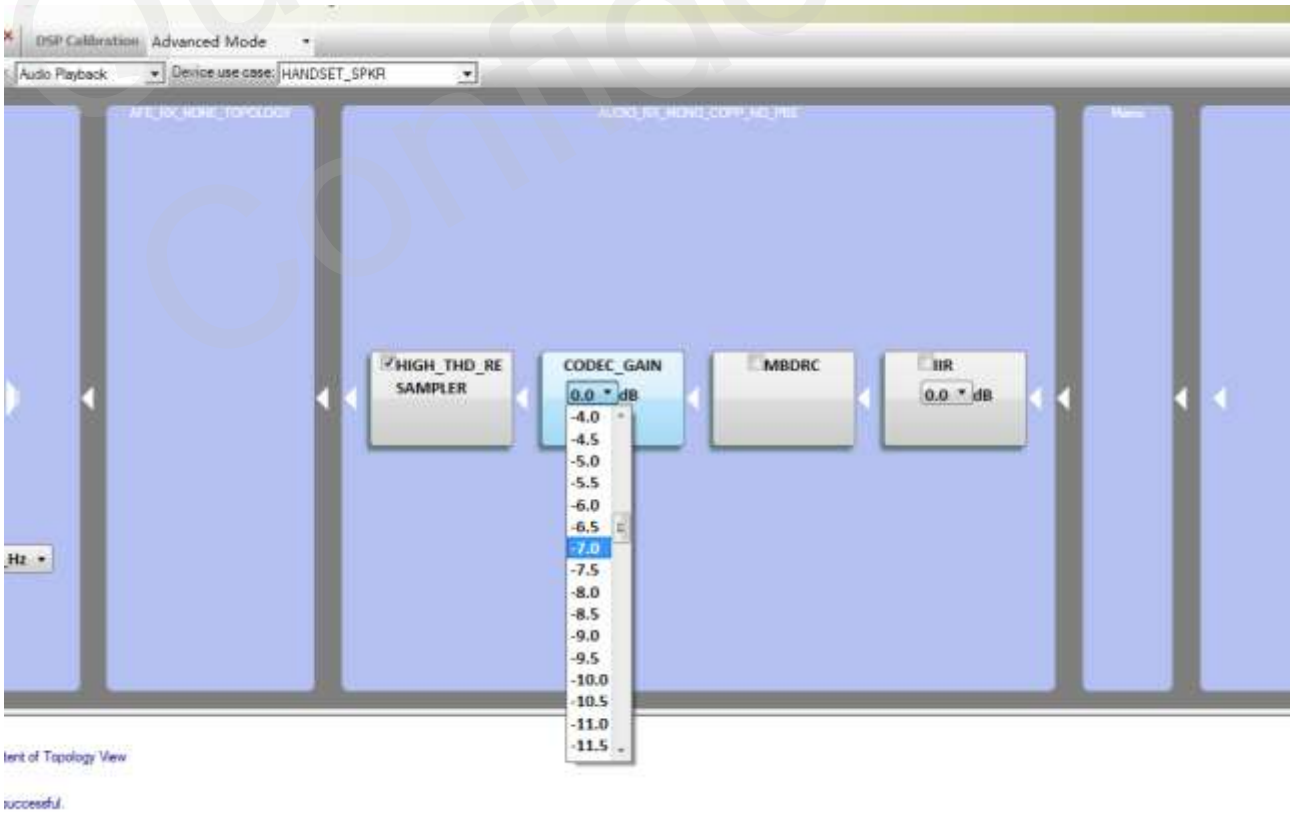

Figure 8: HANDSET SPKR

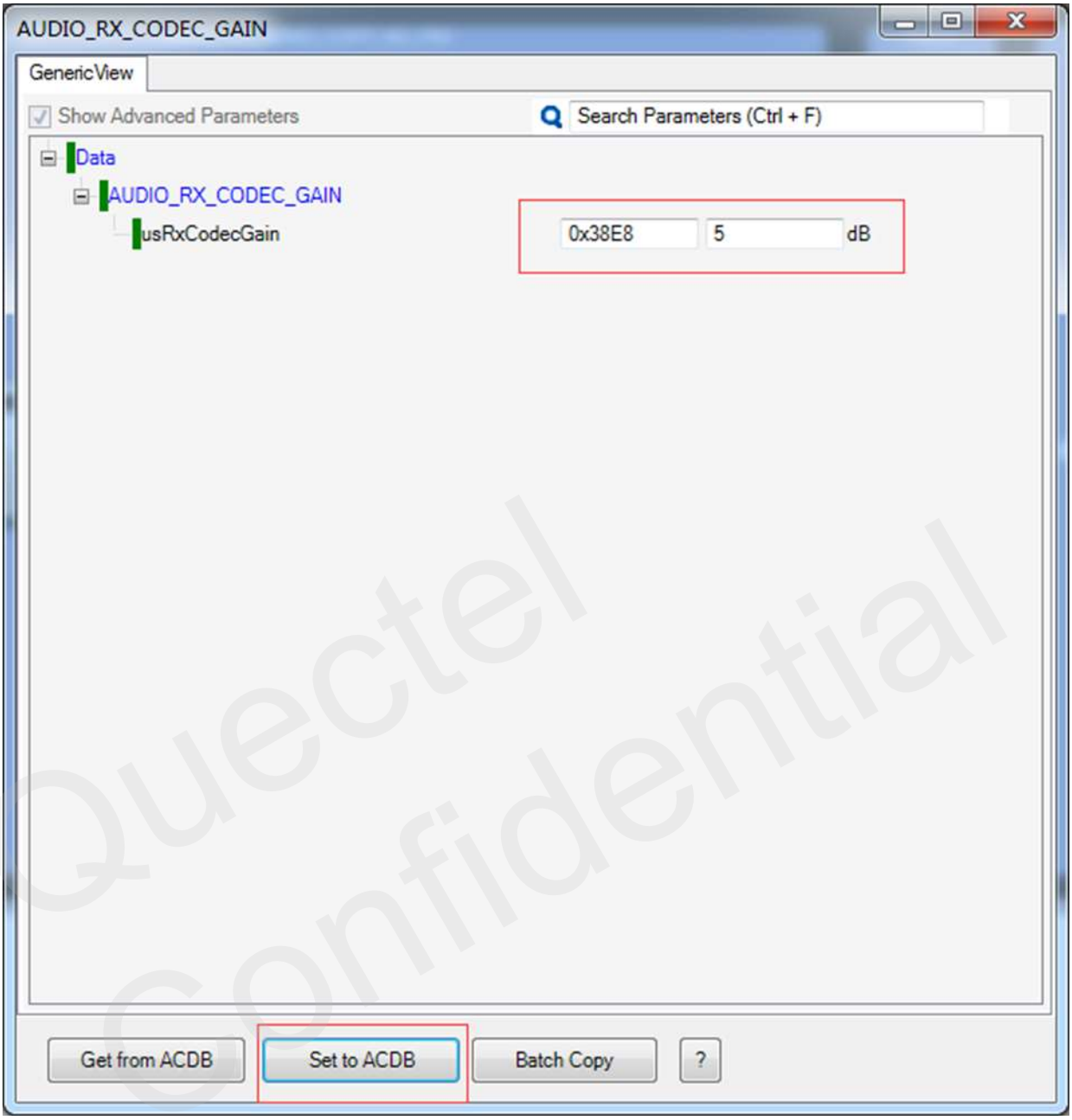

Figure 9: Setting CODEC_GAIN

2.2. Online Mode

In online mode, directly modify the ACDB parameters in the module and save them to the local. Turn on the device, open QPST, load the DM port, then open the QACT tool and connect to the device. The modification method is the same as Chapter 2.1.2. After the modification, click the button of “Save As” in the upper left corner of QACT to save the ACDB file to the local.

3 Importing ACDB File into File System

3.1. Unzipping SDK, Configuring Compilation Environment

Take the 9x07 OpenLinux platform as an example, copy SDK to linux and execute it.
Unzip sudo tar -jxvf EC20CETFDKR05A03V01M2G_OCPU_DJJ_SDK.tar.bz2, as shown in Figure 10.

```
grady@cullen-dell:~$ ls
Desktop  Documents  Downloads  EC20CETFDKR05A03V01M2G_OCPU_DJJ_SDK.tar.bz2  examples.desktop
grady@cullen-dell:~$ sudo tar -jxvf EC20CETFDKR05A03V01M2G_OCPU_DJJ_SDK.tar.bz2
```

Figure 10: Unzipping SDK

After unzipping SDK, enter ql-ol-sdk directory and execute source ql-ol-crosstool/ql-ol-crosstool-env-init to configure compilation environment, as shown in Figure 11.

```
grady@cullen-dell:~$ cd ql-ol-sdk/
grady@cullen-dell:~/ql-ol-sdk$ ls
Makefile  ql-ol-bootloader  ql-ol-crosstool  ql-ol-extsdk  ql-ol-kernel  ql-ol-rootfs
grady@cullen-dell:~/ql-ol-sdk$ source ql-ol-crosstool/ql-ol-crosstool-env-init
```

Figure 11: Configuring Compilation Environment

3.2. Copying the Debugged ACDB File to the File System

ACDB file is in the /data path of the module file system, as shown in Figure 12.

```
root@mdm9607-perf:/data# ls -l *.acdb
-rwxr-xr-x    1 root     root          1074 Jan  6 00:32 Bluetooth_cal.acdb
-rwxr-xr-x    1 root     root          4766 Jan  6 00:32 General_cal.acdb
-rwxr-xr-x    1 root     root          4438 Jan  6 00:32 Global_cal.acdb
-rwxr-xr-x    1 root     root         84298 Jan  6 00:32 Handset_cal.acdb
-rwxr-xr-x    1 root     root          1064 Jan  6 00:32 Hdmi_cal.acdb
-rwxr-xr-x    1 root     root         61654 Jan  6 00:32 Headset_cal.acdb
-rwxr-xr-x    1 root     root         72500 Jan  6 00:32 Speaker_cal.acdb
root@mdm9607-perf:/data#
```

Figure 12: ACDB File

Delete all ACDB files under ql-ol-sdk/ql-ol-rootfs/data (9x28 platform is in ql-ol-sdk/ql-ol-usrfs), copy the debugged ACDB file to this path. (workspaceFile.qwsp file does not need to be copied. This file is a project file and the module does not need to use), modify permissions, execute it.

sudo chown -R xxx:xxx *.acdb

sudo chmod 777 ./*.acdb

As shown in Figure 13

```
grady@cullen-dell:~/ql-ol-sdk/ql-ol-rootfs/data$ sudo chown -R grady:grady *.acdb
[sudo] password for grady:
grady@cullen-dell:~/ql-ol-sdk/ql-ol-rootfs/data$ sudo chmod 777 ./*.acdb
grady@cullen-dell:~/ql-ol-sdk/ql-ol-rootfs/data$ ls -l *.acdb
-rwxrwxrwx 1 grady grady  1074 Feb  1 09:37 Bluetooth_cal.acdb
-rwxrwxrwx 1 grady grady  4766 Feb  1 09:37 General_cal.acdb
-rwxrwxrwx 1 grady grady  4438 Feb  1 09:37 Global_cal.acdb
-rwxrwxrwx 1 grady grady 84298 Feb  1 09:37 Handset_cal.acdb
-rwxrwxrwx 1 grady grady  1064 Feb  1 09:37 Hdmi_cal.acdb
-rwxrwxrwx 1 grady grady 61654 Feb  1 09:37 Headset_cal.acdb
-rwxrwxrwx 1 grady grady 72500 Feb  1 09:37 Speaker_cal.acdb
grady@cullen-dell:~/ql-ol-sdk/ql-ol-rootfs/data$
```

Figure 13: Modifying Permissions

3.3. File System Compilation

Execute under the path of ql-ol-sdk：
make rootfs，compile file system, as shown in Figure 14.

```
grady@cullen-dell:~/ql-ol-sdk$ make rootfs
cd /home/grady/ql-ol-sdk ; chmod +x ./ql-ol-extsdk/tools/quectel_ubi/* ; ./ql-ol-extsdk/tools/quectel_ubi/mkfs.ubifs -r q
l-ol-rootfs -o mdm9607-perf-sysfs.ubifs -m 2048 -e 126976 -c 4292 -F ; \
        ./ql-ol-extsdk/tools/quectel_ubi/ubinize  -o mdm9607-perf-sysfs.ubi -m 2048 -p 128KiB -s 2048 ql-ol-extsdk/tools/
quectel_ubi/ubinize.cfg ; \
        mv mdm9607-perf-sysfs.ubifs mdm9607-perf-sysfs.ubi target/
ubinize: volume size was not specified in section "ubifs", assume minimum to fit image "./mdm9607-perf-sysfs.ubifs"299663
36 bytes (28.6 MiB)
grady@cullen-dell:~/ql-ol-sdk$
```

Figure 14: File System Compilation

After compilation, target folder is generated. New file system document is in the target folder, as shown in figure 15.

```
grady@cullen-dell:~/ql-ol-sdk$ ls
Makefile  ql-ol-bootloader  ql-ol-crosstool  ql-ol-extsdk  ql-ol-kernel  ql-ol-rootfs  target
grady@cullen-dell:~/ql-ol-sdk$ cd target/
grady@cullen-dell:~/ql-ol-sdk/target$ ls
mdm9607-perf-sysfs.ubi  mdm9607-perf-sysfs.ubifs
```

Figure 15: File System Document

Download the mdm9607-perf-sysfs.ubi file into the module, reboot, and the new ACDB file will be loaded by the module.