LUMANTEK®

WEIVER 2.0 Player –
HD Radio Vector Signal Generator

AM/FM 325 Libraries

HD Radio®

WEIVER 2.0 Player –
HD Radio Vector Signal Generator

Revision Number: v1.3
Distribution Date: Nov, 2016

Warranty Period
Lumantek's products comes with One(1) year limited warranty. Please contact below if you need more information. .

LUMANTEK CUSTOMER SERVICES
sales@lumantek.co.kr / TEL:02-6947-7400 / FAX:02-6947-7440

Contents

WEIVER 2.0 Player –
HD Radio Vector Signal Generator

1.1 Introduction & Features

WEIVER 2.0 Player plays back the real world RF signals which had either recorded with WEIEVR - RF capture and playback platform or other RF capture equipments. Also, it comes with a converting utility that converts the I/Q data file into a possible data format, .wpj, with WEIVER 2.0 Player. It's all for to re-generate real world RF environments that are too complex To exact simulate. It records signals directly to external SSD, and plays back directly from external SSD, so you can carry the “hassle-free heavy RF data copy“ on your own.

FEATURES

- HD Radio Vector File Signal Generation (HD Radio AM/FM, Total 176 Files)
- Supports all universal DTV and ATV broadcasting include Broadcasting Radio protocols
- Playback of real world broadcasting signals recorded with WEIEVR R & P platform to regenerate real world conditions that are too complex to simulate
- Front-access hot swappable SSD for fast reading and writing speed
- e-SATA interface for external storages
- Support for Real-time Spectrum and GPS location with Signal Tracer

WEIVER-RF capture and playback system can plays back RF signals covering the bandwidth range from 8M to 56M, or ARB in almost any environment, such as in laboratories or during on-site testing. You can see the all information about WEIVER-RF capture and playback equipment from LUMANTEK's web site (http://www.lumantek.com/)

[Weiver HD Radio Version Information]
WeiverPlayer: 2.2.0.4 or above / DSP: 4.6.1.0(HW 1.6) or above
Weiver HD Radio: 1.0.0.0 or above

1.2 User System Requirements

Your PC(System) must meet the following specification.

Computer / Processor
- Intel Pentium 4 2.33GHz
- Athlon 64 2800 or the faster processor

Operating System
- Windows XP 32 bit service pack(sp2)
- Windows Vista 32 bit service pack(sp2)
- Windows Vista 64 bit service pack(sp2)
- Windows 7 32bit
-Windows 7 64bit

Memory - 2GB RAM

Hard Disk Space
- 512 GB SSD available hard disk space

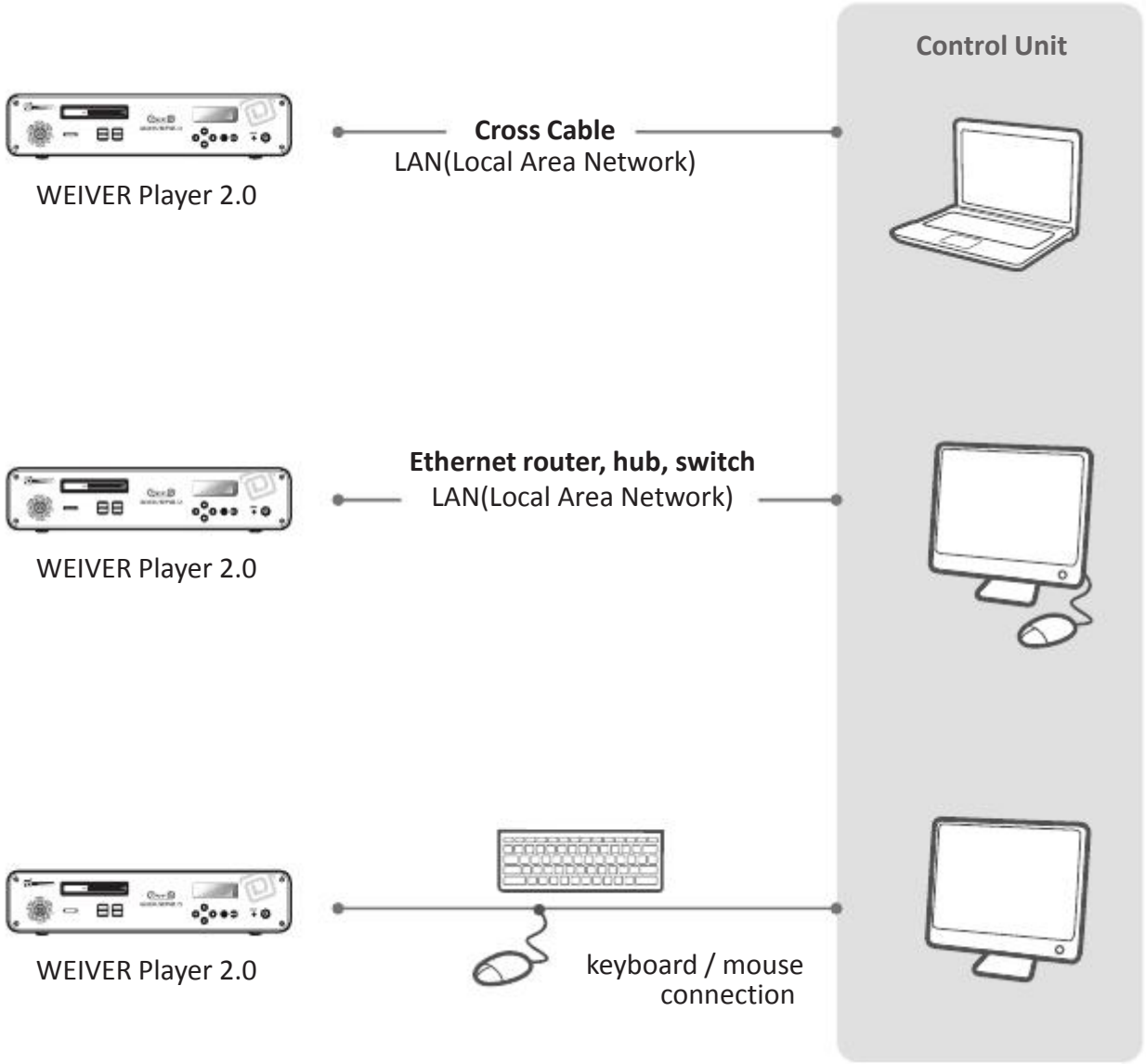
Screen
-1024x768,“32-bitTrue Color” -DirectX 9.0c or higher

Graphics Card
- DirectX 9.0c supported

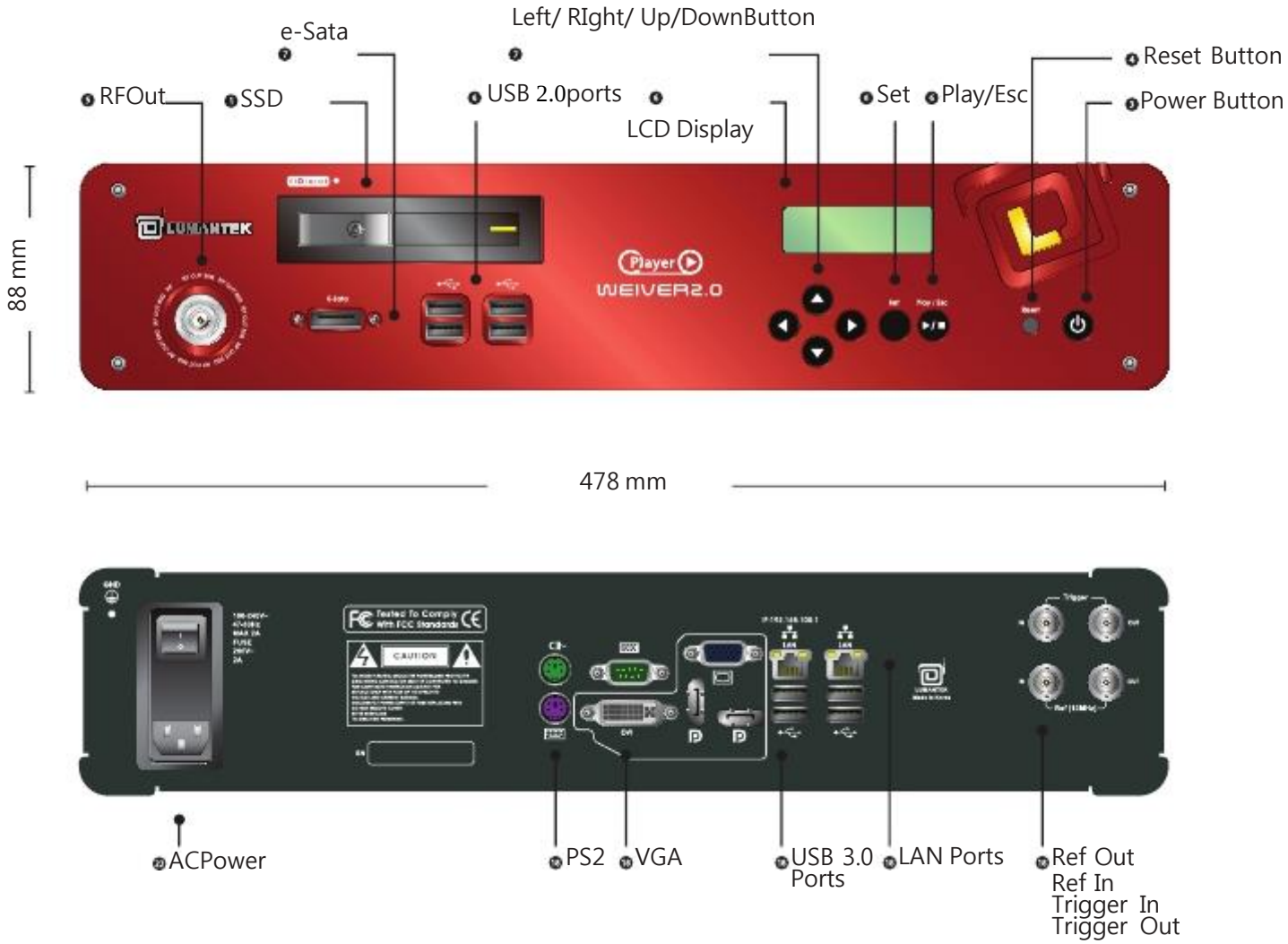
Network
- 10/100 Mbps network card, 10/100/1000Mbps network card(Recommend)
- 802.11 b/g Wireless network card

External SSD Hard disk (Optional)
- Size : 512 GB / 1TB -TRIM :Yes
- Write Speed : 220 MB / Sec(Max.) -Read Speed : 250MB / Sec(Max.)

1.3 Control Configuration(Recommended)



1.4 WEIVER 2.0 Player Components-1



1.5 WEIVER 2.0 Player Components-2

- Basic Component

WEIVER 2.0 Player Body

WEIVER 2.0 Player AC Cord

WEIVER 2.0 Player USB

WeiverEX Program
Weiver HD Radio Program
WEIVER 2.0 Player Operation Manual
WEIVER 2.0 PlayerTest Report
WEIVER 2.0 Player Restore Data

- Optional Component

WEIVER External SSD

WEIVER External SSD USB Power

WEIVER External SSD e-SATA Cable

1.6 Installation precautions

This section describes safety precautions that users shall be aware of during the system Installation and operation. Not complying such precautions may result in serious harm or personal injuries. . For that reason, please keep all details here in mind before installing or using WEIVER 2.0 Player. Please carefully read the followings for safe use of Weiver 2.0 Player

General Precautions

① Must be operated and maintained free of dust or dirt.
② The cover should be securely fastened.
③ Securely stow all the cables, external antennas, external SSD or any other tools away in a safe place after its use.
④ Avoid wearing loose, draping clothing, and dangling jewelry when using WEIVER 2.0 Player.
⑤ Do not open the WEIVER 2.0 Player case. Doing so may void the warranty and LUMANTEK takes no responsibility for the damages caused by such action. Should you be experiencing performance issues, please contact your local dealer for assistance.

Power Precautions

① Make sure if it may cause overload in wiring when you connect the power source.
② Avoid wearing necklaces or watches when connecting the system to power sources. These may cause electronic shocks to the system.
③ Avoid operating on a wet floor out in the open (e.g., raining). Make sure the power extension cable is in satisfactory condition (e.g., not worn out).
④ Disconnect the system from power source before hardware installation, rub or contact on the metal surface of the system to discharge statics from your body.
⑤ Note: Manufacturer is not responsible for damages caused by using or replacing inappropriate components or not authorized services.
⑥ Supplying power during installation may cause serious damages to the system and personal injuries.

AC Power

① This unit comes with the AC power cord, grounding connection is necessary.
② In the event of a fire, please disconnect the system from power source.

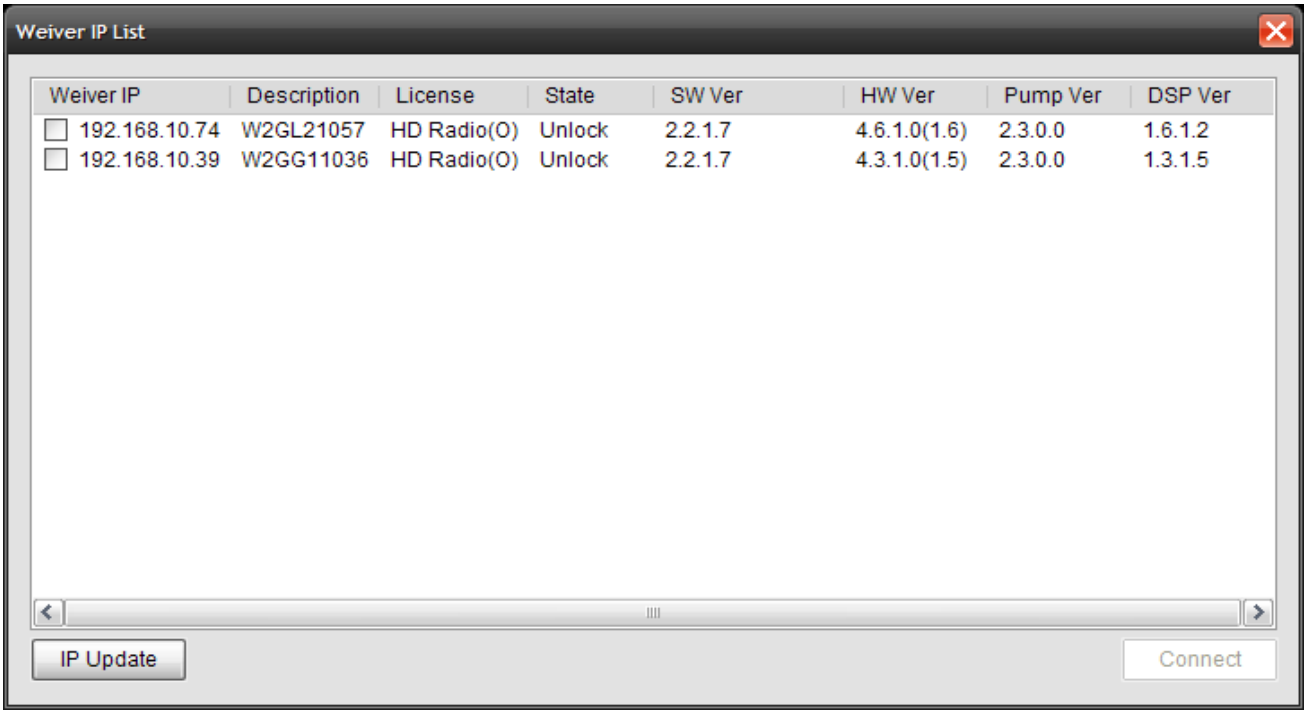
Log-off Caution

① Do NOT log off while “Remote Desktop “is running. You must shut-down the program by pressing
② the close[x] button on top right. If you log-off the player, running program will be stopped and
③ “Please Wait” sign will be appeared on front panel.
④ To recover from this state, you must log-on again after connecting the keyboard & monitor.

2.1 HD Radio Vector Signal generator – Version Info

[Weiver HD Radio RF Playback System specification]
Same as Weiver 2.0 Player specification

[Weiver HD Radio Version Information]
WeiverPlayer: 2.2.0.4 or above
DSP: 4.6.1.0(HW 1.6) or above
Weiver HD Radio: 1.0.0.0 or above



2.2 Vector File List

[Weiver HD Radio Vector Files] **Total 325 Files / sizes total 300 GB**
AM files : 91 files / FM files: 234 files

Caution: Do NOT change the file names. It won't be shown on the file list if the file name has been changed.

[AM] 91 files

IB_AM_e1_awgn.bin
IB_AM_e1a_CW_dc.bin
IB_AMr001_e1_cw_0_m10.bin
IB_AMr001_e1_cw_2_6.bin
IB_AMr001_e1_cw_m1_m2.bin
IB_AMr001_e1_cw_m7p5_10.bin
IB_AMr001_e1_cw_m30_0_30.bin
IB_AMr201_e1awfc52.bin
IB_AMr201_e1awfr1190.bin
IB_AMr208_e1awfa05.bin
IB_AMr208_e1awfa07.bin
IB_AMr208_e1awfa10.bin
IB_AMr208_e1awfb01.bin
IB_AMr208_e1awfb02.bin
IB_AMr208_e1awfb03.bin
IB_AMr208_e1awfb04.bin
IB_AMr208_e1awfb05.bin
IB_AMr208_e1awfb102.bin
IB_AMr208_e1awfb104.bin
IB_AMr208_e1awfb111.bin
IB_AMr208_e1awfb113.bin
IB_AMr208_e1awfb119.bin
IB_AMr208_e1awfb121.bin
IB_AMr208_e1awfb124.bin
IB_AMr208_e1awfb126.bin
IB_AMr208_e1awfb128.bin
IB_AMr208_e1awfb130.bin
IB_AMr208_e1awfb131.bin
IB_AMr208_e1awfb132.bin
IB_AMr208_e1awfb134.bin
IB_AMr208_e1awfb135.bin
IB_AMr208_e1awfb145.bin
IB_AMr208_e1awfb243.bin
IB_AMr208_e1awfc00.bin
IB_AMr208_e1awfc04.bin
IB_AMr208_e1awfc06.bin
IB_AMr208_e1awfc08.bin
IB_AMr208_e1awfc10.bin
IB_AMr208_e1awfc11.bin
IB_AMr208_e1awfc19.bin
IB_AMr208_e1awfc20.bin
IB_AMr208_e1awfc23.bin
IB_AMr208_e1awfc24.bin
IB_AMr208_e1awfc26.bin
IB_AMr208_e1awfc27.bin
IB_AMr208_e1awfc29.bin
IB_AMr208_e1awfc30.bin
IB_AMr208_e1awfc31.bin
IB_AMr208_e1awfc32.bin
IB_AMr208_e1awfr1221.bin
IB_AMr208_e1awfr1224.bin
IB_AMr208_e1awfr1249.bin
IB_AMr208_e1awfr1260.bin
IB_AMr208a_e1awfb00.bin
IB_AMr208a_e1awfb138.bin
IB_AMr208a_e1awfc33.bin
IB_AMr208a_e1awfc34.bin
IB_AMr208a_e1wfr1123.bin
IB_AMr208a_e1wfr1124.bin
IB_AMr208b_e1awf1123.bin
IB_AMr208b_e1wfr1124.bin
IB_AMr208c_e1awf1124.bin
IB_AMr208c_e1awfa11.bin
IB_AMr220_e1awfc28.bin
IB_AMr230_e1awfr1005.bin
IB_AMr230_e1awfr1187.bin
IB_AMr230_e1awfr1220.bin
IB_AMr230_e1awfr1261.bin
IB_AMr230a_e1awfr1001.bin
IB_AMr230a_e1awfr1002.bin
IB_AMr230b_e1awfc102.bin
IB_AMr230d_e1awfr12.bin
IB_AMr430_e1wfr1016.bin
IB_AMr430_e1wfr1017.bin
IB_AMr430_e1wfr1018.bin
IB_AMr430_e1wfr1019.bin
IB_AMr430_e1wfr1020.bin
IB_AMr430_e1wfr1021.bin
IB_AMr430_e1wfr1022.bin
IB_AMr430_e1wfr1023.bin
IB_AMr430_e1wfr1024.bin
IB_AMr430_e1wfr1025.bin
IB_AMr430_e1wfr1026.bin
IB_AMr430_e1wfr1027.bin
IB_AMr430_e1wfr1028.bin
IB_AMr440_e1wfr1116.bin
IB_AMr440_e1wfr1117.bin
IB_AMr440_e1wfr1428.bin
IB_AMr440_e1wfr1443.bin
IB_AMr440_e1wfr1447.bin
IB_AMr440_e1wfr1454.bin

[FM] 234 files

CC_TTN_FMr440a_e1wfr001.bin
CC_TTN_FMr440a_e1wfr002.bin
CC_TTN_FMr440a_e1wfr003.bin
CC_TTN_FMr440a_e1wfr004.bin
CC_TTN_FMr440a_e1wfr005.bin
CC_TTN_FMr440a_e1wfr006.bin
CC_TTN_FMr440a_e1wfr007.bin
CC_TTN_FMr440a_e1wfr008.bin
CC_TTN_FMr440a_e1wfr009.bin
CC_TTN_FMr440a_e1wfr010.bin
CC_TTN_FMr440a_e1wfr011.bin
CC_TTN_FMr440a_e1wfr012.bin
CC_TTN_FMr440a_e1wfr013.bin
CC_TTN_FMr440a_e1wfr014.bin
CC_TTN_FMr440a_e1wfr015.bin
CC_TTN_FMr440a_e1wfr016.bin
CC_TTN_FMr440a_e1wfr017.bin

[FM] 234 files

CC_TTN_FMr440a_e1wfr018.bin
CC_TTN_FMr440a_e1wfr019.bin
CC_TTN_FMr440a_e1wfr019b.bin
CC_TTN_FMr440a_e1wfr020.bin
CC_TTN_FMr440a_e1wfr020b.bin
CC_TTN_FMr440a_e1wfr021.bin
CC_TTN_FMr440a_e1wfr022.bin
CC_TTN_FMr440a_e1wfr023.bin
CC_TTN_FMr440a_e1wfr024.bin
CC_TTN_FMr440a_e1wfr025.bin
CC_TTN_FMr440a_e1wfr026.bin
CC_TTN_FMr440a_e1wfr027.bin
CC_TTN_FMr440a_e1wfr028.bin
CC_TTN_FMr440a_e1wfr029.bin
CC_TTN_FMr440a_e1wfr030.bin
CC_TTN_FMr440a_e1wfr031.bin
CC_TTN_FMr440a_e1wfr032.bin
CC_TTN_FMr440a_e1wfr033.bin
CC_TTN_FMr440a_e1wfr034.bin
CC_TTN_FMr440a_e1wfr035.bin
CC_TTN_FMr440a_e1wfr036.bin
CC_TTN_FMr440a_e1wfr037.bin
CC_TTN_FMr440a_e1wfr038.bin
CC_TTN_FMr440a_e1wfr039.bin
CC_TTN_FMr440a_e1wfr040.bin
CC_TTN_FMr440a_e1wfr041.bin
CC_TTN_FMr440a_e1wfr042.bin
CC_TTN_FMr440a_e1wfr043.bin
CC_TTN_FMr440a_e1wfr044.bin
CC_TTN_FMr440a_e1wfr045.bin
CC_TTN_FMr440a_e1wfr046.bin
CC_TTN_FMr440a_e1wfr047.bin
CC_TTN_FMr440a_e1wfr048.bin
CC_TTN_FMr440a_e1wfr049.bin
CC_TTN_FMr440a_e1wfr049b.bin
CC_TTN_FMr440a_e1wfr050.bin
CC_TTN_FMr440a_e1wfr050b.bin
CC_TTN_FMr440a_e1wfr051.bin
CC_TTN_FMr440a_e1wfr051b.bin
CC_TTN_FMr440a_e1wfr052.bin
CC_TTN_FMr440a_e1wfr053.bin
CC_TTN_FMr440a_e1wfr054.bin
CC_TTN_FMr440a_e1wfr055.bin
CC_TTN_FMr440a_e1wfr056.bin
CC_TTN_FMr440a_e1wfr057.bin
CC_TTN_FMr440a_e1wfr058.bin
CC_TTN_FMr440a_e1wfr065.bin
CC_TTN_FMr440a_e1wfr070.bin
CC_TTN_FMr440a_e1wfr071.bin
CC_TTN_FMr440a_e1wfr072.bin

CC_FMr430b_e1wfr0001.bin
CC_FMr440_e1wfr015_HighDataRate.bin
CC_FMr440d_e1wfr011.bin
CC_FMr440d_e1wfr012.bin
CC_FMr440d_e1wfr013.bin
CC_FMr440d_e1wfr014.bin
IB_FM_e1_awgn.bin
IB_FM_e1_CW_dc.bin
IB_FMr001_e1_cw_0_100.bin
IB_FMr001_e1_cw_0_m100.bin
IB_FMr001_e1_cw_0_m200.bin
IB_FMr001_e1_cw_10_20.bin
IB_FMr001_e1_cw_50_100.bin
IB_FMr001_e1_cw_m50_0.bin
IB_FMr001_e1_cw_m80_m90.bin
IB_FMr001_e1_cw_m275_0_275.bin
IB_FMr201_e1wfc52.bin
IB_FMr201_e1wfr1189.bin
IB_FMr208_e1wfr1122.bin
IB_FMr208_e1wfr1185.bin
IB_FMr208_e1wfr1246.bin
FM-I94W-US24-Loss-1222014_e1_101.1.bin
IB_FMr440_e1wfr1301.bin
IB_FMr440_e1wfr1302.bin
IB_FMr440_e1wfr1315.bin
IB_FMr440_e1wfr1332.bin
IB_FMr440_e1wfr1343.bin
IB_FMr440_e1wfr1345.bin
IB_FMr440_e1wfr1347.bin
IB_FMr440_e1wfr1360.bin
IB_FMr440_e1wfr1362.bin
IB_FMr440_e1wfr1503.bin
IB_FMr440a_e1wfr1303.bin
IB_FMr440a_e1wfr1304.bin
IB_FMr440a_e1wfr1305.bin
IB_FMr440a_e1wfr1314.bin
IB_FMr440a_e1wfr1323.bin
IB_FMr440a_e1wfr1331.bin
IB_FMr440a_e1wfr1342.bin
IB_FMr440b_e1wfr1300.bin
IB_FMr440b_e1wfr1307.bin
IB_FMr440b_e1wfr1326.bin

[FM] 234 files

IB_FMr208_e1wfr1247.bin
IB_FMr208_e1wfr1248.bin
IB_FMr208c_e1wfa05.bin
IB_FMr208c_e1wfa25.bin
IB_FMr208c_e1wfa98.bin
IB_FMr208c_e1wfa99.bin
IB_FMr208c_e1wfc00.bin
IB_FMr208c_e1wfc03.bin
IB_FMr208c_e1wfc08.bin
IB_FMr208c_e1wfc09.bin
IB_FMr208c_e1wfc10.bin
IB_FMr208c_e1wfc27.bin
IB_FMr208c_e1wfc28.bin
IB_FMr208c_e1wfc30.bin
IB_FMr208c_e1wfc46.bin
IB_FMr208c_e1wfc90.bin
IB_FMr208c_e1wfc201.bin
IB_FMr208c_e1wfc203.bin
IB_FMr208c_e1wfc204.bin
IB_FMr208c_e1wfc206.bin
IB_FMr208c_e1wfc209.bin
IB_FMr208c_e1wfc210.bin
IB_FMr208c_e1wfc211.bin
IB_FMr208c_e1wfc227.bin
IB_FMr208c_e1wfc230.bin
IB_FMr208c_e1wfc546.bin
IB_FMr208c_e1wfc547.bin
IB_FMr208c_e1wfc548.bin
IB_FMr208d_e1wfa141.bin
IB_FMr208d_e1wfc208.bin
IB_FMr208d_e1wfc538.bin
IB_FMr208d_e1wfc540.bin
IB_FMr208d_e1wfc542.bin
IB_FMr208d_e1wfc549.bin
IB_FMr208e_e1wfc12.bin
IB_FMr208e_e1wfc13.bin
IB_FMr208g_e1wfc94.bin
IB_FMr208i_e1wfa58.bin
IB_FMr208j_e1wfa11.bin
IB_FMr208j_e1wfa105.bin
IB_FMr208j_e1wfa106.bin
IB_FMr208j_e1wfa107.bin
IB_FMr208j_e1wfa108.bin
IB_FMr208j_e1wfa109.bin
IB_FMr208j_e1wfc31.bin
IB_FMr208j_e1wfc89.bin
IB_FMr208k_e1wfa104.bin
IB_FMr220_e1wfa108.bin
IB_FMr220_e1wfr1121.bin
IB_FMr220a_e1wfc100.bin
IB_FMr220a_e1wfc101.bin
IB_FMr220a_e1wfc553.bin
IB_FMr220a_e1wfc555.bin
IB_FMr220a_e1wfr1119.bin
IB_FMr220a_e1wfr1120.bin
IB_FMr220b_e1wfr1119.bin
IB_FMr220b_e1wfr1120.bin
IB_FMr220c_e1wfr1191.bin
IB_FMr230_e1wfc14.bin
IB_FMr230_e1wfr143.bin
IB_FMr230_e1wfr1061.bin
IB_FMr230a_e1wfa153.bin
IB_FMr230a_e1wfc102.bin
IB_FMr230a_e1wfr1000.bin
IB_FMr230a_e1wfr1001.bin
IB_FMr230a_e1wfr1003.bin
IB_FMr230a_e1wfr1004.bin
IB_FMr230a_e1wfr1005.bin
IB_FMr230a_e1wfr1006.bin
IB_FMr230a_e1wfr1007.bin
IB_FMr230a_e1wfr1008.bin
IB_FMr230a_e1wfr1009.bin
IB_FMr230a_e1wfr1011.bin
IB_FMr230a_e1wfr1012.bin
IB_FMr230a_e1wfr1013.bin
IB_FMr230a_e1wfr1037.bin
IB_FMr230b_e1wfr1002.bin
IB_FMr230b_e1wfr1010.bin
IB_FMr230b_e1wfr1024.bin
IB_FMr230c_e1wfd204.bin
IB_FMr230c_e1wfr1022.bin
IB_FMr230c_e1wfr1032.bin
IB_FMr230d_e1wfa78.bin
IB_FMr230d_e1wfr1025.bin
IB_FMr230f_e1wfr1023.bin
IB_FMr430_e1wfr1070.bin
IB_FMr430_e1wfr1071.bin
IB_FMr430_e1wfr1072.bin
IB_FMr430_e1wfr1073.bin
IB_FMr430_e1wfr1074.bin
IB_FMr430_e1wfr1075.bin
IB_FMr430_e1wfr1080.bin
IB_FMr430_e1wfr1082.bin
IB_FMr430_e1wfr1083.bin
IB_FMr430_e1wfr1086.bin
IB_FMr430_e1wfr1087.bin
IB_FMr430_e1wfr1088.bin
IB_FMr430_e1wfr1089.bin
IB_FMr430_e1wfr1090.bin
IB_FMr430_e1wfr1091.bin
IB_FMr430_e1wfr1092.bin
IB_FMr430_e1wfr1093.bin
IB_FMr430_e1wfr1094.bin
IB_FMr430_e1wfr1095.bin
IB_FMr430_e1wfr1096.bin
IB_FMr430_e1wfr1097.bin
IB_FMr430_e1wfr1098.bin
IB_FMr430_e1wfr1099.bin
IB_FMr430_e1wfr1100.bin
IB_FMr430_e1wfr1101.bin
IB_FMr430_e1wfr1102.bin
IB_FMr430_e1wfr1152.bin
IB_FMr430a_e1wfr1081.bin
IB_FMr430a_e1wfr1085.bin
IB_FMr440_e1wfr1111.bin
IB_FMr440_e1wfr1112.bin
IB_FMr440_e1wfr1113.bin
IB_FMr440_e1wfr1114.bin
IB_FMr440_e1wfr1115.bin
IB_FMr440_e1wfr1243.bin
IB_FMr440_e1wfr1251.bin
IB_FMr440_e1wfr1252.bin
IB_FMr440_e1wfr1253.bin
IB_FMr440_e1wfr1254.bin
IB_FMr440_e1wfr1255.bin

2.3 HD Radio Operation

There are three(3) ways to access(operate) your System

1. Standalone Operation

Connect USB monitor, Keyboard & Mouse to the H/W
Start->All Programs->©Lumantek->WeiverHDRadio->WeiverHDRadio exe.

2. Desktop PC or Network Operation

Install WeiverHDRadio program in Notebook/Desktop PC
Start->All Programs->©Lumantek->WeiverHDRadio->WeiverHDRadio exe.

3. Front Panel LCD Operation - Refer to the next page.

How to Update WEIVER 2.0 Player System

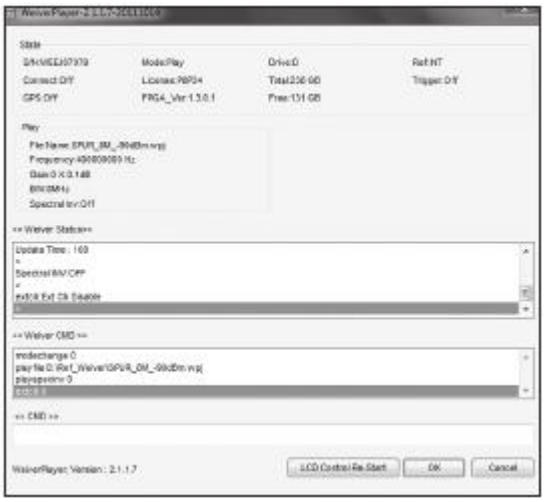
Connect the WEIVER 2.0 Player to a laptop or desktop PC with both Microsoft Windows OS and the'WeiverHDRADIO.exe' program installed. In case where system update is necessary, please have a monitor, keyboard and mouse ready.

Visit LUMANTEK online Download Center at http://www.lumantek.com/support/ to download 'WeiverHDRADIO.exe' program. Double-click on the program icon will initiate the installation.

** LCD Screen Operation

Starting WEIVER 2.0 Player – HD Radio Option

The “WeiverHDRADIO.exe” program is for connecting and controlling the Weiver 2.0 player system. You will see a “remote” sign displayed on front LCD panel when the system is being connected.



[Play]

● Set --> 1. Play File --> ● Set

--> File Select

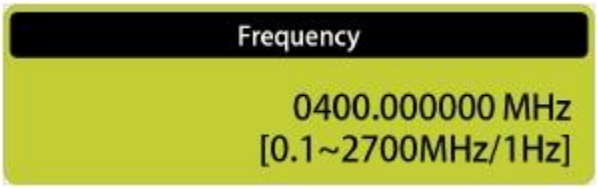
--> ● Set

--> ● Play

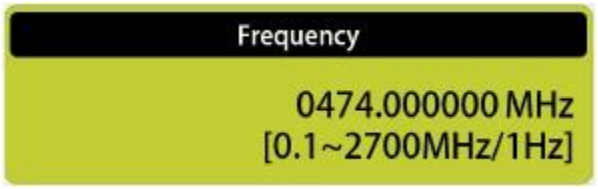
[Frequency Change]

● Set --> 2. Frequency --> ● Set

--> Change (up / down / left / right)



--> Change (up / down / left / right)



● Set --> Esc

[Gain Change]

● Set --> 3. Gain --> ● Set

--> Change (up / down / left / right)

● Set --> Esc

[Spectrum Invert]

● Set --> 4. Spectrum Inv. --> ● Set

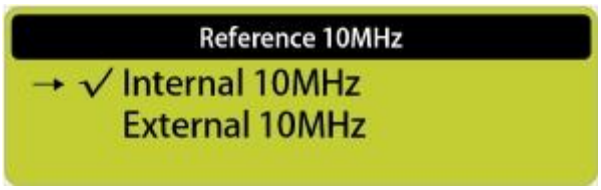
--> Change (up / down)

● Set --> Esc

[Reference 10MHz]

● Set --> 5. Ref 10MHz --> ● Set

--> Change (up / down)



● Set --> Esc

[Rescan Disk]

● Set --> – 6. Rescan Disk --> ● Set

--> Apply / Cancel

● Set --> Esc

[IP Setting]

● Set --> 7. IP Setting --> ● Set

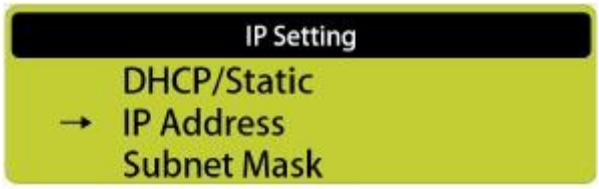
--> DHCP

Change (up)

--> Static

Change (down)

--> IP Address



--> Address Setting

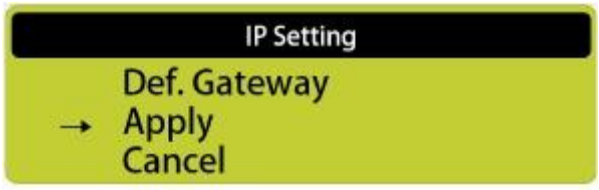
--> Subnet Mask

--> Subnet Mask Setting

--> Gateway

--> Gateway Setting

--> Apply / Cancel



[Version]

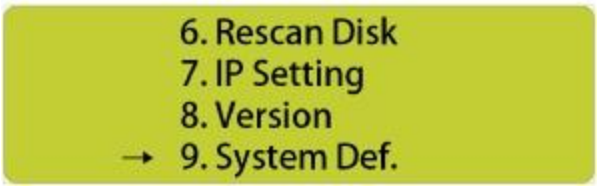
● Set --> 8.Version --> ● Set

Change (up/down)

● Set --> Esc

[System Default]

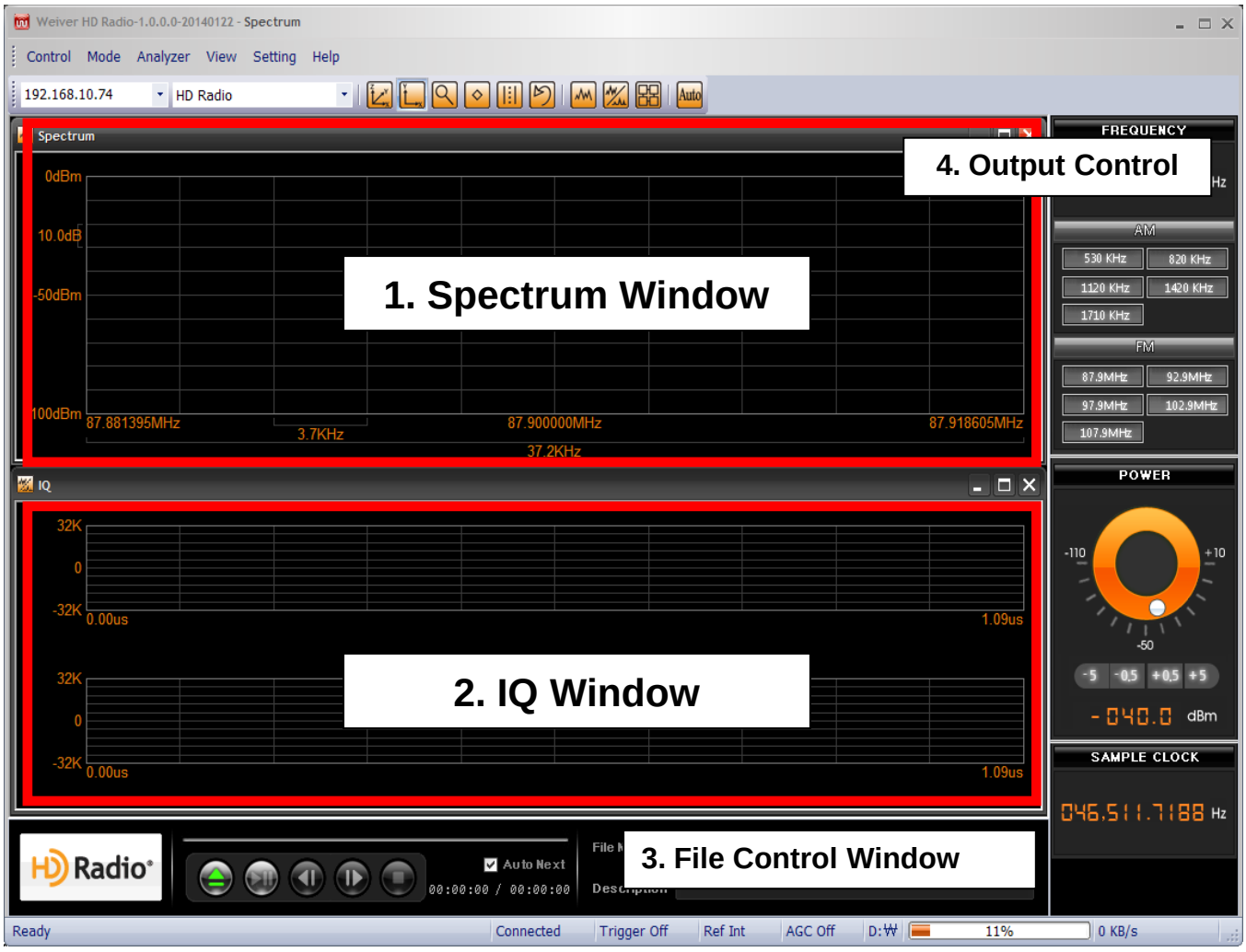
● Set --> 9.System Def. --> ● Set



--> Apply / Cancel

● Set --> Esc

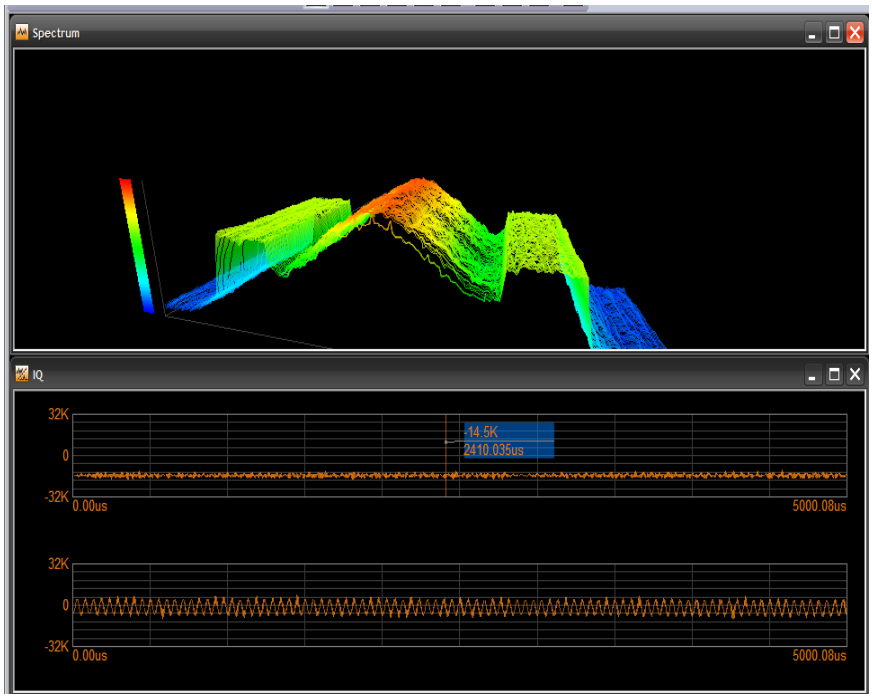
2.4 HD Radio UI



1. Spectrum Window - Showing spectrum of files currently under playback.
2. I/Q Window - Showing I(2Bytes), Q(2Byte information of files currently under playback
3. File Control Window – showing current file name and description
4. Output Control - Adjust Frequency, Power, and Sample Clock.

2.5 HD Radio screen capture

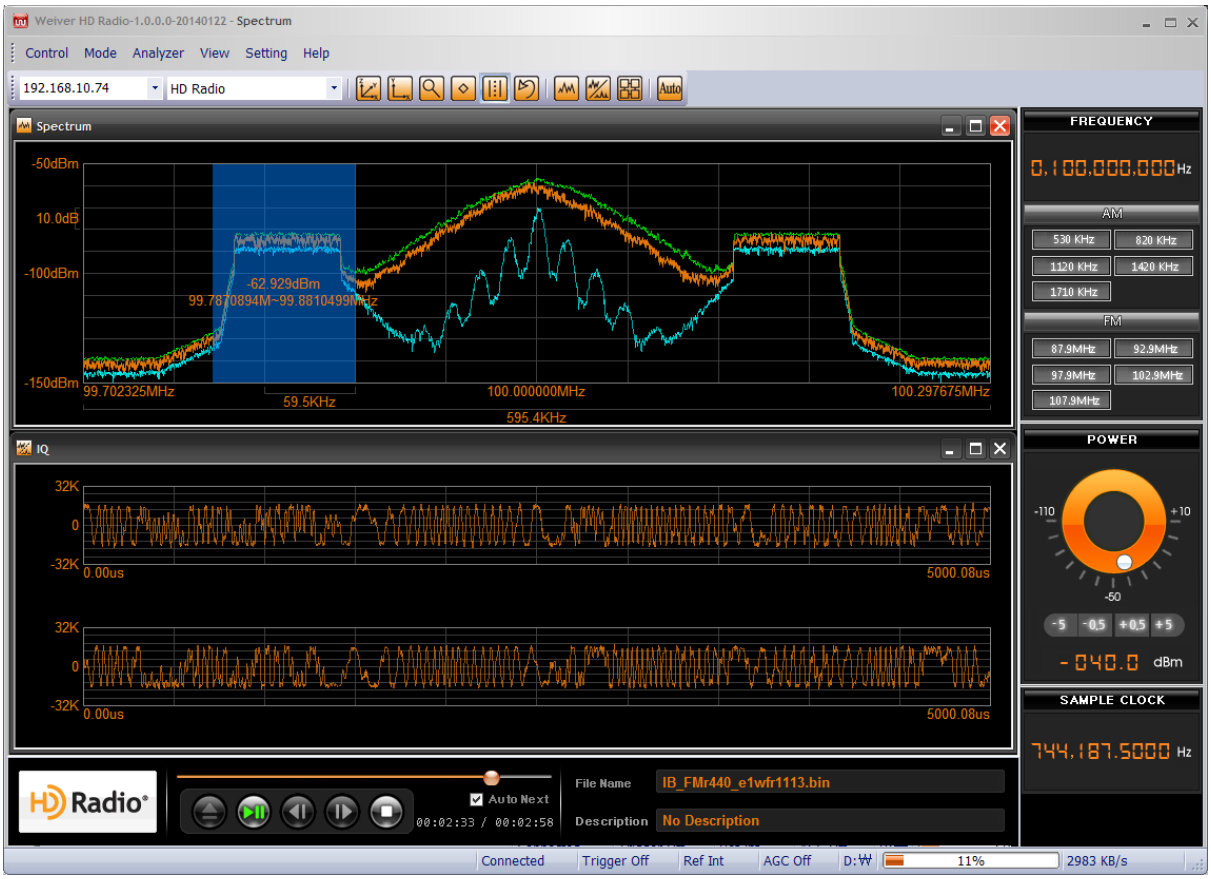
[2D Mode]



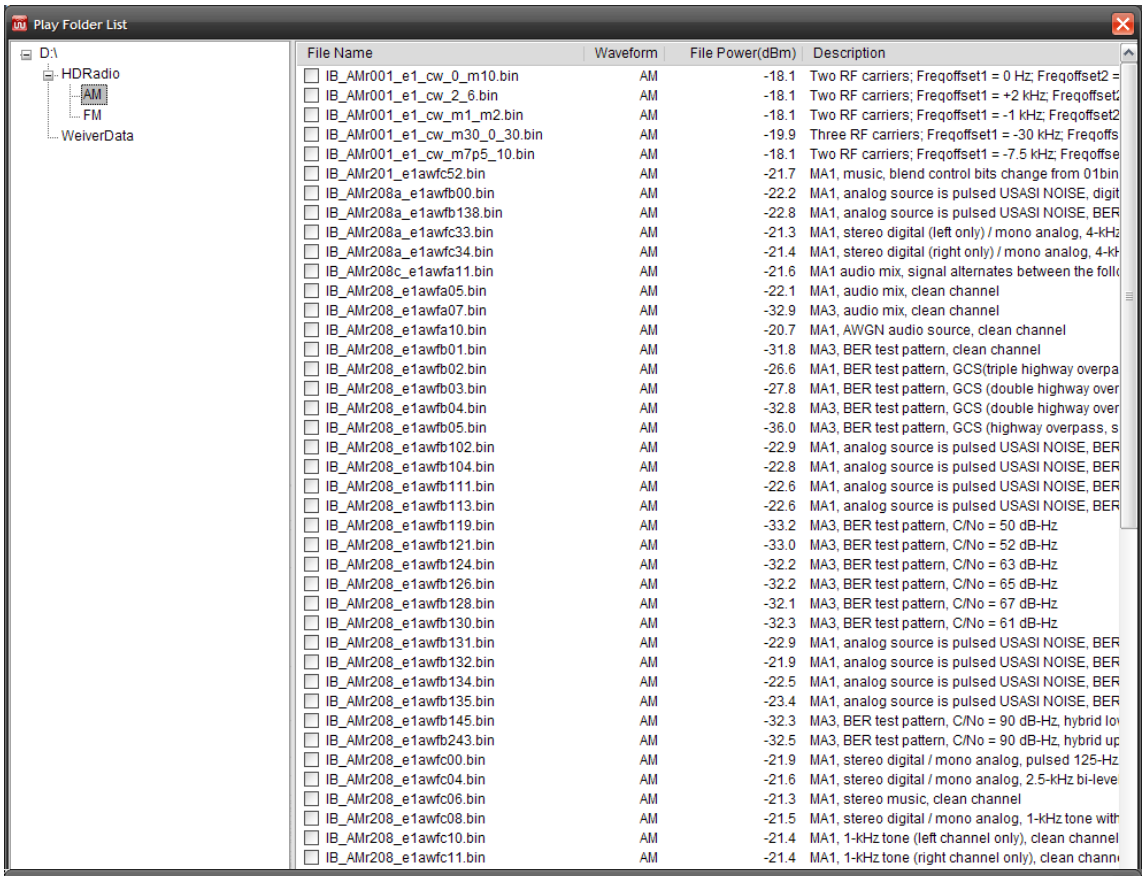
[3D Mode]

[Markers and Minimum and Maximum]

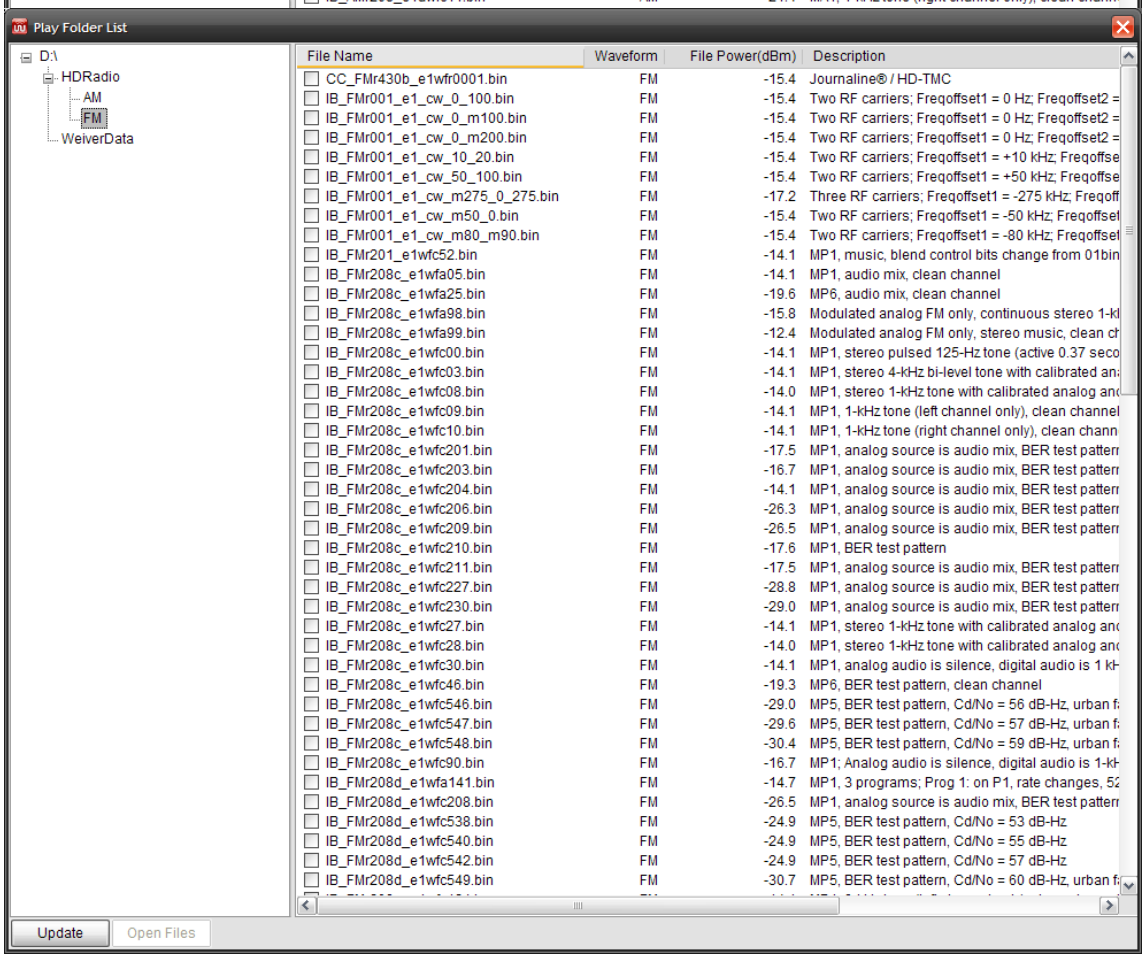
[Band Power]



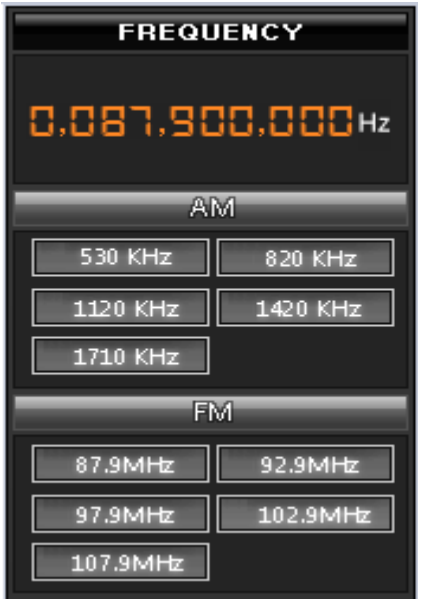
[AM]


Play Folder List
D:\
HDRadio
AM
FM
WeiverData
File Name
Waveform
File Power(dBm)
Description
IB_AMr001_e1_cw_0_m10.bin AM -18.1 Two RF carriers; Freqoffset1 = 0 Hz; Freqoffset2 =
IB_AMr001_e1_cw_2_6.bin AM -18.1 Two RF carriers; Freqoffset1 = +2 kHz; Freqoffset2
IB_AMr001_e1_cw_m1_m2.bin AM -18.1 Two RF carriers; Freqoffset1 = -1 kHz; Freqoffset2
IB_AMr001_e1_cw_m30_0_30.bin AM -19.9 Three RF carriers; Freqoffset1 = -30 kHz; Freqoffs
IB_AMr001_e1_cw_m7p5_10.bin AM -18.1 Two RF carriers; Freqoffset1 = -7.5 kHz; Freqoffse
IB_AMr201_e1awfc52.bin AM -21.7 MA1, music, blend control bits change from 01bin
IB_AMr208a_e1awfb00.bin AM -22.2 MA1, analog source is pulsed USASI NOISE, digit
IB_AMr208a_e1awfb138.bin AM -22.8 MA1, analog source is pulsed USASI NOISE, BER
IB_AMr208a_e1awfc33.bin AM -21.3 MA1, stereo digital (left only) / mono analog, 4-kH
IB_AMr208a_e1awfc34.bin AM -21.4 MA1, stereo digital (right only) / mono analog, 4-k
IB_AMr208c_e1awfa11.bin AM -21.6 MA1 audio mix, signal alternates between the foll
IB_AMr208_e1awfa05.bin AM -22.1 MA1, audio mix, clean channel
IB_AMr208_e1awfa07.bin AM -32.9 MA3, audio mix, clean channel
IB_AMr208_e1awfa10.bin AM -20.7 MA1, AWGN audio source, clean channel
IB_AMr208_e1awfb01.bin AM -31.8 MA3, BER test pattern, clean channel
IB_AMr208_e1awfb02.bin AM -26.6 MA1, BER test pattern, GCS(triple highway overpa
IB_AMr208_e1awfb03.bin AM -27.8 MA1, BER test pattern, GCS (double highway over
IB_AMr208_e1awfb04.bin AM -32.8 MA3, BER test pattern, GCS (double highway over
IB_AMr208_e1awfb05.bin AM -36.0 MA3, BER test pattern, GCS (highway overpass, s
IB_AMr208_e1awfb102.bin AM -22.9 MA1, analog source is pulsed USASI NOISE, BER
IB_AMr208_e1awfb104.bin AM -22.8 MA1, analog source is pulsed USASI NOISE, BER
IB_AMr208_e1awfb111.bin AM -22.6 MA1, analog source is pulsed USASI NOISE, BER
IB_AMr208_e1awfb113.bin AM -22.6 MA1, analog source is pulsed USASI NOISE, BER
IB_AMr208_e1awfb119.bin AM -33.2 MA3, BER test pattern, C/No = 50 dB-Hz
IB_AMr208_e1awfb121.bin AM -33.0 MA3, BER test pattern, C/No = 52 dB-Hz
IB_AMr208_e1awfb124.bin AM -32.2 MA3, BER test pattern, C/No = 63 dB-Hz
IB_AMr208_e1awfb126.bin AM -32.2 MA3, BER test pattern, C/No = 65 dB-Hz
IB_AMr208_e1awfb128.bin AM -32.1 MA3, BER test pattern, C/No = 67 dB-Hz
IB_AMr208_e1awfb130.bin AM -32.3 MA3, BER test pattern, C/No = 61 dB-Hz
IB_AMr208_e1awfb131.bin AM -22.9 MA1, analog source is pulsed USASI NOISE, BER
IB_AMr208_e1awfb132.bin AM -21.9 MA1, analog source is pulsed USASI NOISE, BER
IB_AMr208_e1awfb134.bin AM -22.5 MA1, analog source is pulsed USASI NOISE, BER
IB_AMr208_e1awfb135.bin AM -23.4 MA1, analog source is pulsed USASI NOISE, BER
IB_AMr208_e1awfb145.bin AM -32.3 MA3, BER test pattern, C/No = 90 dB-Hz, hybrid lo
IB_AMr208_e1awfb243.bin AM -32.5 MA3, BER test pattern, C/No = 90 dB-Hz, hybrid up
IB_AMr208_e1awfc00.bin AM -21.9 MA1, stereo digital / mono analog, pulsed 125-Hz
IB_AMr208_e1awfc04.bin AM -21.6 MA1, stereo digital / mono analog, 2.5-kHz bi-leve
IB_AMr208_e1awfc06.bin AM -21.3 MA1, stereo music, clean channel
IB_AMr208_e1awfc08.bin AM -21.5 MA1, stereo digital / mono analog, 1-kHz tone with
IB_AMr208_e1awfc10.bin AM -21.4 MA1, 1-kHz tone (left channel only), clean channel
IB_AMr208_e1awfc11.bin AM -21.4 MA1, 1-kHz tone (right channel only), clean chann

[FM]


Play Folder List
D:\
HDRadio
AM
FM
WeiverData
File Name
Waveform
File Power(dBm)
Description
CC_FMr430b_e1wfr0001.bin FM -15.4 Journaline® / HD-TMC
IB_FMr001_e1_cw_0_100.bin FM -15.4 Two RF carriers; Freqoffset1 = 0 Hz; Freqoffset2 =
IB_FMr001_e1_cw_0_m100.bin FM -15.4 Two RF carriers; Freqoffset1 = 0 Hz; Freqoffset2 =
IB_FMr001_e1_cw_0_m200.bin FM -15.4 Two RF carriers; Freqoffset1 = 0 Hz; Freqoffset2 =
IB_FMr001_e1_cw_10_20.bin FM -15.4 Two RF carriers; Freqoffset1 = +10 kHz; Freqoffse
IB_FMr001_e1_cw_50_100.bin FM -15.4 Two RF carriers; Freqoffset1 = +50 kHz; Freqoffse
IB_FMr001_e1_cw_m275_0_275.bin FM -17.2 Three RF carriers; Freqoffset1 = -275 kHz; Freqoff
IB_FMr001_e1_cw_m50_0.bin FM -15.4 Two RF carriers; Freqoffset1 = -50 kHz; Freqoffset
IB_FMr001_e1_cw_m80_m90.bin FM -15.4 Two RF carriers; Freqoffset1 = -80 kHz; Freqoffset
IB_FMr201_e1wfc52.bin FM -14.1 MP1, music, blend control bits change from 01bin
IB_FMr208c_e1wfa05.bin FM -14.1 MP1, audio mix, clean channel
IB_FMr208c_e1wfa25.bin FM -19.6 MP6, audio mix, clean channel
IB_FMr208c_e1wfa98.bin FM -15.8 Modulated analog FM only, continuous stereo 1-k
IB_FMr208c_e1wfa99.bin FM -12.4 Modulated analog FM only, stereo music, clean ch
IB_FMr208c_e1wfc00.bin FM -14.1 MP1, stereo pulsed 125-Hz tone (active 0.37 seco
IB_FMr208c_e1wfc03.bin FM -14.1 MP1, stereo 4-kHz bi-level tone with calibrated an
IB_FMr208c_e1wfc08.bin FM -14.0 MP1, stereo 1-kHz tone with calibrated analog an
IB_FMr208c_e1wfc09.bin FM -14.1 MP1, 1-kHz tone (left channel only), clean channel
IB_FMr208c_e1wfc10.bin FM -14.1 MP1, 1-kHz tone (right channel only), clean chann
IB_FMr208c_e1wfc201.bin FM -17.5 MP1, analog source is audio mix, BER test patter
IB_FMr208c_e1wfc203.bin FM -16.7 MP1, analog source is audio mix, BER test patter
IB_FMr208c_e1wfc204.bin FM -14.1 MP1, analog source is audio mix, BER test patter
IB_FMr208c_e1wfc206.bin FM -26.3 MP1, analog source is audio mix, BER test patter
IB_FMr208c_e1wfc209.bin FM -26.5 MP1, analog source is audio mix, BER test patter
IB_FMr208c_e1wfc210.bin FM -17.6 MP1, BER test pattern
IB_FMr208c_e1wfc211.bin FM -17.5 MP1, analog source is audio mix, BER test patter
IB_FMr208c_e1wfc227.bin FM -28.8 MP1, analog source is audio mix, BER test patter
IB_FMr208c_e1wfc230.bin FM -29.0 MP1, analog source is audio mix, BER test patter
IB_FMr208c_e1wfc27.bin FM -14.1 MP1, stereo 1-kHz tone with calibrated analog an
IB_FMr208c_e1wfc28.bin FM -14.0 MP1, stereo 1-kHz tone with calibrated analog an
IB_FMr208c_e1wfc30.bin FM -14.1 MP1, analog audio is silence, digital audio is 1 k
IB_FMr208c_e1wfc46.bin FM -19.3 MP6, BER test pattern, clean channel
IB_FMr208c_e1wfc546.bin FM -29.0 MP5, BER test pattern, Cd/No = 56 dB-Hz, urban f
IB_FMr208c_e1wfc547.bin FM -29.6 MP5, BER test pattern, Cd/No = 57 dB-Hz, urban f
IB_FMr208c_e1wfc548.bin FM -30.4 MP5, BER test pattern, Cd/No = 59 dB-Hz, urban f
IB_FMr208c_e1wfc90.bin FM -16.7 MP1; Analog audio is silence, digital audio is 1-k
IB_FMr208d_e1wfa141.bin FM -14.7 MP1, 3 programs; Prog 1: on P1, rate changes, 5
IB_FMr208d_e1wfc208.bin FM -26.5 MP1, analog source is audio mix, BER test patter
IB_FMr208d_e1wfc538.bin FM -24.9 MP5, BER test pattern, Cd/No = 53 dB-Hz
IB_FMr208d_e1wfc540.bin FM -24.9 MP5, BER test pattern, Cd/No = 55 dB-Hz
IB_FMr208d_e1wfc542.bin FM -24.9 MP5, BER test pattern, Cd/No = 57 dB-Hz
IB_FMr208d_e1wfc549.bin FM -30.7 MP5, BER test pattern, Cd/No = 60 dB-Hz, urban f
Update
Open Files

2.6 HD Radio Setting



- [Weiver HD Radio Frequency Setting]

Adjustable Range : 100KHz ~ 200MHz

How to Change
- Click on the number section, adjust the frequency with mouse wheel or keyboard arrow buttons
- Click on pre-selected AM, FM Frequency icon

- [Weiver HD Radio Power Setting]

Adjustable Range : -110dBm~+10dBm

It depend on the power level of each AM/FM Vector files which have different Max/Min level. Max(+10dBm)/Min (-110dBm) may not be applicable to some of the files

How to Change
- Click on the number section, adjust the power level with mouse wheel or keyboard arrow buttons
- Click on the White button on UI and scroll
- Click on pre-selected ranges, -5, -0.5, +0.5, +5

- [Weiver HD Radio Sample Clock Setting]

AM/FM Default Sample Clock
AM: 46511.7188 Hz
FM: 744187.5 Hz

Adjustable Range
AM: 46558.2305 Hz~46558.2305 Hz
FM: 743443.3125 Hz~744931.6875 Hz

How to Change
Click on the number section, adjust the with mouse wheel or keyboard arrow buttons

WEIVER 2.0 / WEIVER Player API for HD Radio Vector Signal Generator

- Summary

There are Three (3) ways to control both Weiver & WeiverPlayer
1. with keypad control for 'WeiverPlayer'
2. with 'WeiverEx' Software
3. with 'API'

#1 and #2 above are the general control method,

#3 is to control the system with command via UDP packet.
API control is supported for 'Weiver 1.0', 'Weiver Player 1.0', 'Weiver 2.0' and 'Weiver Player 2.0'
Supported in 'Playback' mode ONLY. (Not supported for 'Recording')
HD Radio™ is supported with 'Weiver 2.0' and 'Weiver Player 2.0' system ONLY.

3.1 Principle of Weiver/ WeiverPlayer API

'WeiverEx' software connects with 'Weiver/ WeiverPlayer' software via IP, control the system with UDP packets. Since the main purpose of utilizing the API is to control the 'Weiver/ WeiverPlayer' system without the 'WeiverEx' program, this documentation will open sample API programs with its packet data information.

You may download reference sources on Lumantek website at www.lumantek.com

Following is the API packet structure for 'Weiver/WeiverPlayer'. (Reference: WvPlayerRemoteAPI.h)

```
#define     WEIVER_PLAYER_ID                    0x057b50b7
#define     WEIVER_PLAYER_API_RX_PORT           50504

#define     REMOTE_STRING_SUCCESS _T("Received")
#define     REMOTE_STRING_INVALIDE_PARAM _T("Invalid Param")

typedef enum _WvPlayerRemoteCMD_e
{
            WV_PLAYER_REMOTE_CMD_SET_STRING,
            WV_PLAYER_REMOTE_CMD_MAX
}WvPlayerRemoteCMD_t;

typedef struct _WvPlayerRemotePacket_t
{
            unsigned int u4_ID;// WEIVER_PLAYER_ID
            unsigned int u4_Reserved;//WV_PLAYER_REMOTE_CMD_SET_STRING
            unsigned int u4_DataSize;
            unsigned char u1a_Data[1000];
}WvPlayerRemotePacket_t, *WvPlayerRemotePacket_tp;
```

UDP port: 50504 is left open to support API with 'Weiver/WeiverPlayer'.
The 'Weiver/Weiverplayer' executes certain commands when the 'WvPlayerRemotePacket' structure data is received.

The opened 'Source' is the sample window programs. (May applicable to the other types of the OS)

Before moving into the command list description, the sample program 'WeiverPlayerRemoteAPI.exe' uses following five (5) assumptions.

Assumption 1: IP address for 'Weiver/WeiverPlayer' is 192.168.100.1.
Assumption 2: Port left open for API support 'Weiver/WeiverPlayer' is 50504 (fixed)
Assumption 3: Port left open for UDP reception is 50000. (Vary)
Assumption 4: Each characters have 2byte in its size.
Assumption 5: The 'Time Out' will occur if UDP gets no response after its transmission.

3.2 Data Packet Structure

'WeiverEx' software connects with 'Weiver/ WeiverPlayer' software via IP, control the system with UDP packets. Since the main purpose of utilizing the API is to control the 'Weiver/ WeiverPlayer' system without the 'WeiverEx' program, this documentation will open sample API programs with its packet data information.

connect 50000

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	26	L"connect 50000"
Size(Byte)	4	4	4	13x2

disconnect

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	20	L"disconnect"
Size(Byte)	4	4	4	10x2

rescandisk

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	20	L"rescandisk"
Size(Byte)	4	4	4	10x2

openfile test.wpj

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	34	L"openfile test.wpj"
Size(Byte)	4	4	4	17x2

get openfile

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	24	L"get openfile"
Size(Byte)	4	4	4	12x2

play start

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	20	L"play start"
Size(Byte)	4	4	4	10x2

play stop

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	18	L"play stop"
Size(Byte)	4	4	4	9x2

play pause

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	20	L“play pause”
Size(Byte)	4	4	4	10x2

play resume

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	22	L” play resume”
Size(Byte)	4	4	4	11x2

get play_status

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	30	L“get play_status”
Size(Byte)	4	4	4	15x2

set freq 400000000

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	36	L“set freq 400000000”
Size(Byte)	4	4	4	18x2

set freq 400000000 hz

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	42	L“set freq 400000000 hz”
Size(Byte)	4	4	4	21x2

set freq 400000 khz

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	38	L“set freq 400000 khz”
Size(Byte)	4	4	4	19x2

set freq 400 mhz

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	32	L“set freq 400 mhz”
Size(Byte)	4	4	4	16x2

get freq

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	16	L“get freq”
Size(Byte)	4	4	4	8x2

get gain_min

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	24	L“get gain_min”
Size(Byte)	4	4	4	12x2

get gain_max

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	24	L“get gain_max”
Size(Byte)	4	4	4	12x2

get gain

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	16	L“get gain”
Size(Byte)	4	4	4	8x2

get rf_out_level 1

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	36	L“get rf_out_level 1”
Size(Byte)	4	4	4	18x2

set gain 100

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	24	L“set gain 100”
Size(Byte)	4	4	4	12x2

set gain -100

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	26	L“set gain -100”
Size(Byte)	4	4	4	13x2

get play_time_total

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	38	L“get play_time_total”
Size(Byte)	4	4	4	19x2

get play_time_current

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	42	L“get play_time_current”
Size(Byte)	4	4	4	21x2

set play_time_current 5

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	46	L“set play_time_current 5”
Size(Byte)	4	4	4	23x2

get play_section

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	32	L“get play_section”
Size(Byte)	4	4	4	16x2

set play_section 10 20

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	44	L“set play_section 10 20”
Size(Byte)	4	4	4	22x2

get spec_inv

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	24	L“get spec_inv”
Size(Byte)	4	4	4	12x2

set spec_inv off

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	32	L“set spec_inv off”
Size(Byte)	4	4	4	16x2

set spec_inv on

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	30	L“set spec_inv on”
Size(Byte)	4	4	4	15x2

get sample_rate

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	30	L“get sample_rate”
Size(Byte)	4	4	4	15x2

set sample_rate 10000000

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	48	L“set sample_rate 10000000”
Size(Byte)	4	4	4	24x2

set sample_rate 10000000 hz

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	54	L“set sample_rate 10000000 hz”
Size(Byte)	4	4	4	27x2

set sample_rate 10 mhz

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	44	L“set sample_rate 10 mhz”
Size(Byte)	4	4	4	22x2

set sample_rate 10000 khz

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	54	L“set sample_rate 10000 khz”
Size(Byte)	4	4	4	27x2

set trigger_out_level low

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	50	L“set trigger_out_level low”
Size(Byte)	4	4	4	25x2

set trigger_out_level high

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	52	L“set trigger_out_level low”
Size(Byte)	4	4	4	26x2

get trigger_out_level

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	42	L“get trigger_out_level”
Size(Byte)	4	4	4	21x2

set extclk off

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	28	L“set extclk off”
Size(Byte)	4	4	4	14x2

set extclk on

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	26	L“set extclk on”
Size(Byte)	4	4	4	13x2

get extclk

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	20	L“get extclk”
Size(Byte)	4	4	4	10x2

get trigger_in

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	28	L“get trigger_in”
Size(Byte)	4	4	4	14x2

set trigger_in off

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	36	L“set trigger_in off”
Size(Byte)	4	4	4	18x2

set trigger_in on

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	34	L“set trigger_in on”
Size(Byte)	4	4	4	17x2

set current_directory d:\

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	50	L“set current_directory d:\”
Size(Byte)	4	4	4	25x2

set current_directory d:\WeiverData

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	70	L“set current_directory d:\WeiverData”
Size(Byte)	4	4	4	35x2

get current_directory

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	42	L“get current_directory”
Size(Byte)	4	4	4	21x2

get file_count

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	28	L“get file_count”
Size(Byte)	4	4	4	14x2

get filename 0

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	28	L“get filename 0”
Size(Byte)	4	4	4	14x2

get folder_count

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	32	L“get folder_count”
Size(Byte)	4	4	4	16x2

get foldername 0

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	32	L“get foldername 0”
Size(Byte)	4	4	4	16x2

set power -200

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	28	L“set power -200”
Size(Byte)	4	4	4	14x2

get power

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	18	L“get power”
Size(Byte)	4	4	4	9x2

get power_min

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	26	L“get power_min”
Size(Byte)	4	4	4	13x2

get power_max

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	26	L“get power_max”
Size(Byte)	4	4	4	13x2

get freq_min

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	24	L“get freq_min”
Size(Byte)	4	4	4	12x2

get freq_max

Type	ID	Reserved	Data Size	Data
Data	0x057b50b7	0	24	L“get freq_max”
Size(Byte)	4	4	4	12x2

3.3 Command

Command	connect <parameter>
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	To connect with the device <Parameter> temporary port number. Ex) To send the response from Weiver system, 'connect 50000' command sends UDP packets to the 50000 port. If the System is being controlled by either 'WeiverEx' or 'HD Radio™ S/W', the 'connect' command will fail. When 'connect' command fails, it transmits 'connect' command after sending 'disconnect' command first. The temporarily opened port number information must be saved.

Command	Disconnect
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Command disconnecting from the device. RF transmission will be terminated automatically.

Command	rescandisk
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Rescanning the disk drive connected to the device.

Command	openfile <parameter>
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Open (select) the file for the playback. <Parameter> is a sample (temporary) file name. Move to the folder where the playback file is located before executing 'openfile' command. Please refer to 'set current_directory' for folder location commands 'openfile' fails if the system is under 'play' status. To stop 'play', please refer the commands list for 'play'. For Weiver file, it automatically configures (sets) center frequency and power level at the time of recording. Please refer to the 'set freq' and 'set gain' commands list to adjust center frequency and power level of the Weiver file. For HD Radio™ FM file, it automatically configures (sets) frequency at 87.9MHz, power level at its distinct level. For HD Radio™ AM file, it automatically configures (sets) frequency 830KHz, power level at its distinct level. Please refer to the 'set freq' and 'set power' commands list for HD Radio™ file center frequency and power level adjustment.

Command	get openfile
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Get information of file names that is currently being opened.

Commmand	play start play stop play pause play resume
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	play start : Start RF Transmission. play stop : Stop RF Transmission. play pause : Pause RF Transmission. play resume : Resume RF Transmission. Use 'play' commands after entering 'openfile' commands.

Command	get play_status
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Get status information of 'play'.

Command	set freq <parameter> set freq <parameter> hz set freq <parameter> khz set freq <parameter> mhz
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Set the frequency of the device. <parameter> is temporary(sample) frequency. No specific adjustment is necessary since 'openfile' commands automatically configures its sample and center frequency. Use 'set freq' commands to configure frequency for 'HD Radio™' file, the frequency automatically sets at 87.9MHz for FM and 830KHz for AM respectively.
Command	get freq
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Get the frequency information of the device.

Command	set gain <parameter>
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : X
Description	Commands for adjusting the RF transmission power of the device. <parameter> is a temporary(sample) gain value. <parameter> is 100 when +10.0 dB gain is applied. <parameter> is -123 when -12.3 dB gain is applied. HD Radio™ files NOT supported. Please refer to the 'set power' command list for HD Radio™ files.

Command	get gain get gain_min get gain_max
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : X
Description	get gain : get gain information of the RF transmission. get gain_min : get the minimum gain information of the RF transmission. get gain_max : get the maximum gain information of the RF transmission. It is -12.3 dB when gain information is -123. HD Radio™ files NOT supported Please refer to the 'set power' command for HD Radio™ files.

Command	get rf_out_level <parameter>
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : X
Description	Get information of RF output power. <parameter> is a temporary(sample) time value (in seconds). The time value must be less than its total time. It is -12.3 dB when gain information is -123. Please refer to 'get play_time_total' commands for total time of the file. HD Radio™ files NOT supported Please refer to the 'get power' command list for HD Radio™ files.

Command	set play_time_current <parameter>
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Command configures (sets) the starting point of the 'play' file by seconds. <parameter> is a temporary(sample) time value (in seconds).

Command	get play_time_total get play_time_current
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	get play_time_total : get information of the file size that is currently being played by its seconds. get play_time_current : get information of the file location that is currently being played by its seconds.

Command	set play_section <parameter1> <parameter2>
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : X
Description	Command configure (sets) the 'Loop' playback function. <parameter1> is temporary (sample) loop starting point time (in seconds) value. <parameter2> is temporary (sample) loop ending point time (in seconds) value.

Command	get play_section
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : X
Description	Get time value information of loop segments.

Command	set spec_inv off set spec_inv on
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
description	set spec_inv off : transmission without inverse. (left/right) set spec_inv on : transmission with inverse. (left/right)

Command	get spec_inv
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	get spec_inv : get information of spectrum inverse value.

Command	set sample_rate <parameter> set sample_rate <parameter> hz set sample_rate <parameter> khz set sample_rate <parameter> mhz
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : X
Description	Command for sample frequency adjustment. <parameter> is a temporary(sample) frequency. 'openfile' command automatically configures (sets) center frequency and power level at the time of recording. NO specific adjustment necessary.

Command	get sample_rate
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : X
Description	Get information of the sample frequency.

Command	set trigger_out_level low set trigger_out_level high get trigger_out_level
Support	Weiver 1.0 : X Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Command for output control of 'Trigger Out' port. set trigger_out_level low : Low Level(0V) output. set trigger_out_level high : High Level(3.3V) output. get trigger_out_level : Get information on status of 'Trigger Out' port.

Command	set extclk off set extclk on get extclk
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Command for configuration of external/internal 10MHz synchronization. set extclk off : Use internal 10MHz. set extclk on : Use external 10MHz. get extclk : Get information on status of 10MHz sync status.

Command	set trigger_in off set trigger_in on get trigger_in
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Operation control command for number of devices by output level of 'Trigger In Port'. set trigger_in off : Operates regardless of 'Trigger In Port' input level. set trigger_in on : Operates only when 'Trigger In Port' Level is High(3.3v) get trigger_in : Get information on status of operation based on sync status.

Command	set current_directory <parameter>
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Command for designate the folder where the device file is located. <parameter> is absolute path name of the folder. Please change the folder before using 'openfile' command.

Command	get current_directory
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Get information on current device folder.

Command	get file_count
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Get information on number of Weiver and HD Radio™ files in current device.

Command	get filename <parameter>
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Get information on file names of Weiver and HD Radio™ of appropriate index among retrieved file list of current device folder. <parameter> Designate Index of the file list. The value of Index must be less than value of 'get file_count' command.

Command	get folder_count
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Change the number of sub-folders in device folder.

Command	get foldername <parameter>
Support	Weiver 1.0 : O Weiver Player 1.0 : O Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	Get information on name of sub-folders in current device folder. <parameter> Designate Index of the folder. The value of Index must be less than value of 'get foder_count' command.

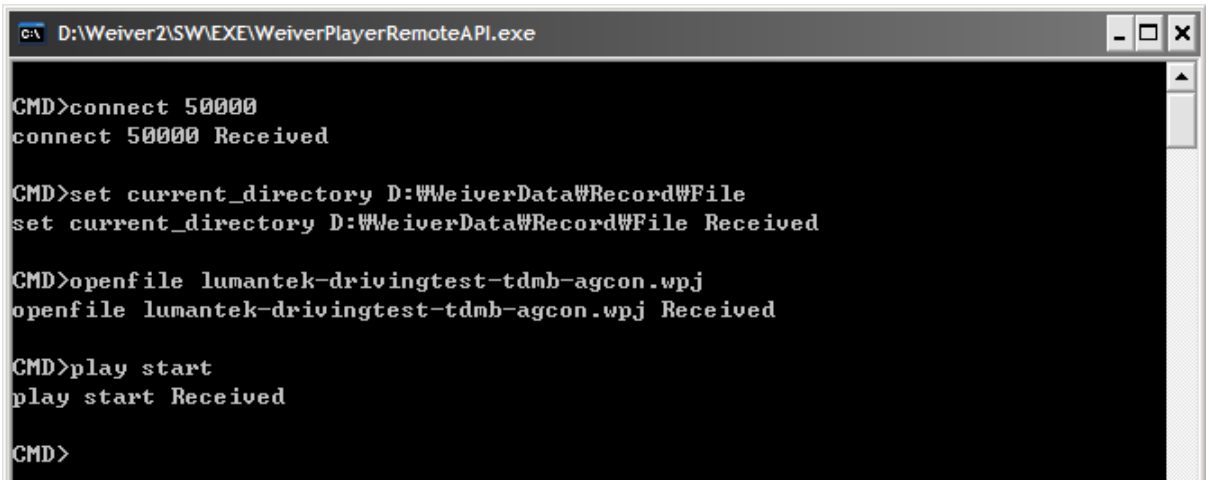
Command	set power <parameter>
Support	Weiver 1.0 : X Weiver Player 1.0 : X Weiver 2.0 : X Weiver Player 2.0 : X HD Radio™ : O
Description	Command for HD Radio™ RF transmission power control. <parameter> is a temporary(sample) value. <parameter> is -205 when power -20.5 dBm is applied. <parameter> is -432 when gain -43.2 dBm is applied.

Command	get power get power_min get power_max
Support	Weiver 1.0 : X Weiver Player 1.0 : X Weiver 2.0 : X Weiver Player 2.0 : X HD Radio™ : O
Description	get power : get information on HD Radio™ RF transmission power. get power_min : get information on the min. HD Radio™ RF transmission power. get power_max : get information on the max. HD Radio™ RF transmission power. Transmission power is -20.5 dBm when return value is -205. Transmission power is -43.2 dBm when return value is -423.

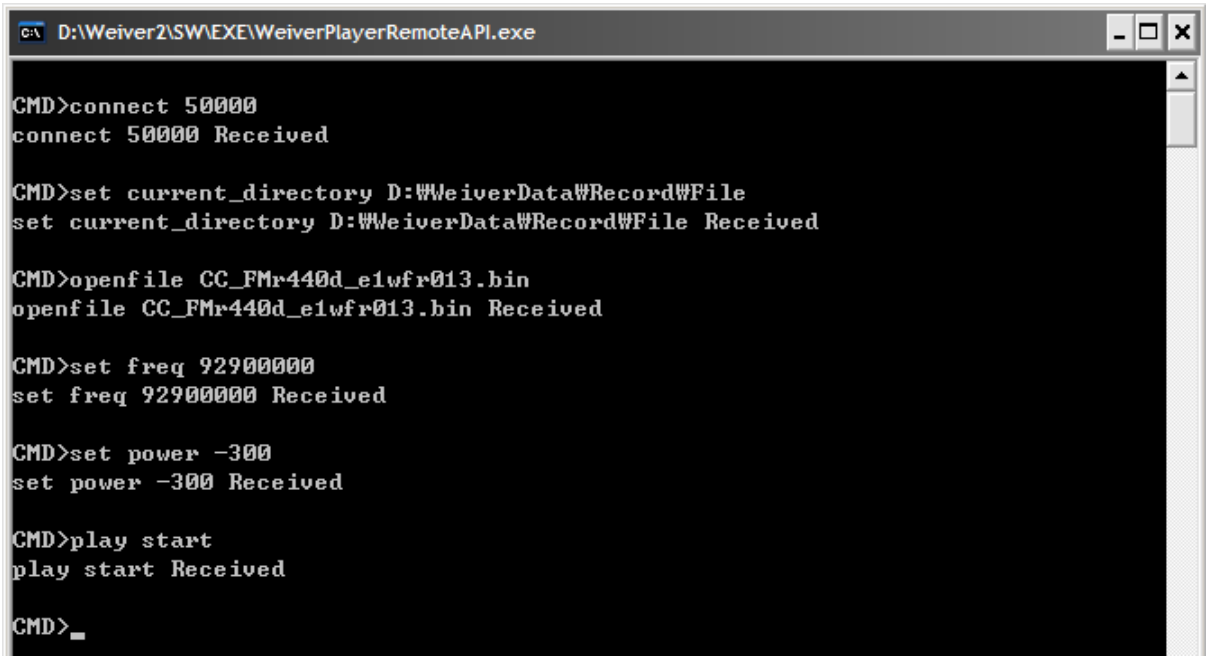
Command	get freq_min get freq_max
Support	Weiver 1.0 : X Weiver Player 1.0 : X Weiver 2.0 : O Weiver Player 2.0 : O HD Radio™ : O
Description	get freq_min : get the information on minimum frequency value of the device. get freq_max : get the information on maximum frequency value of the device.

3.4 Example

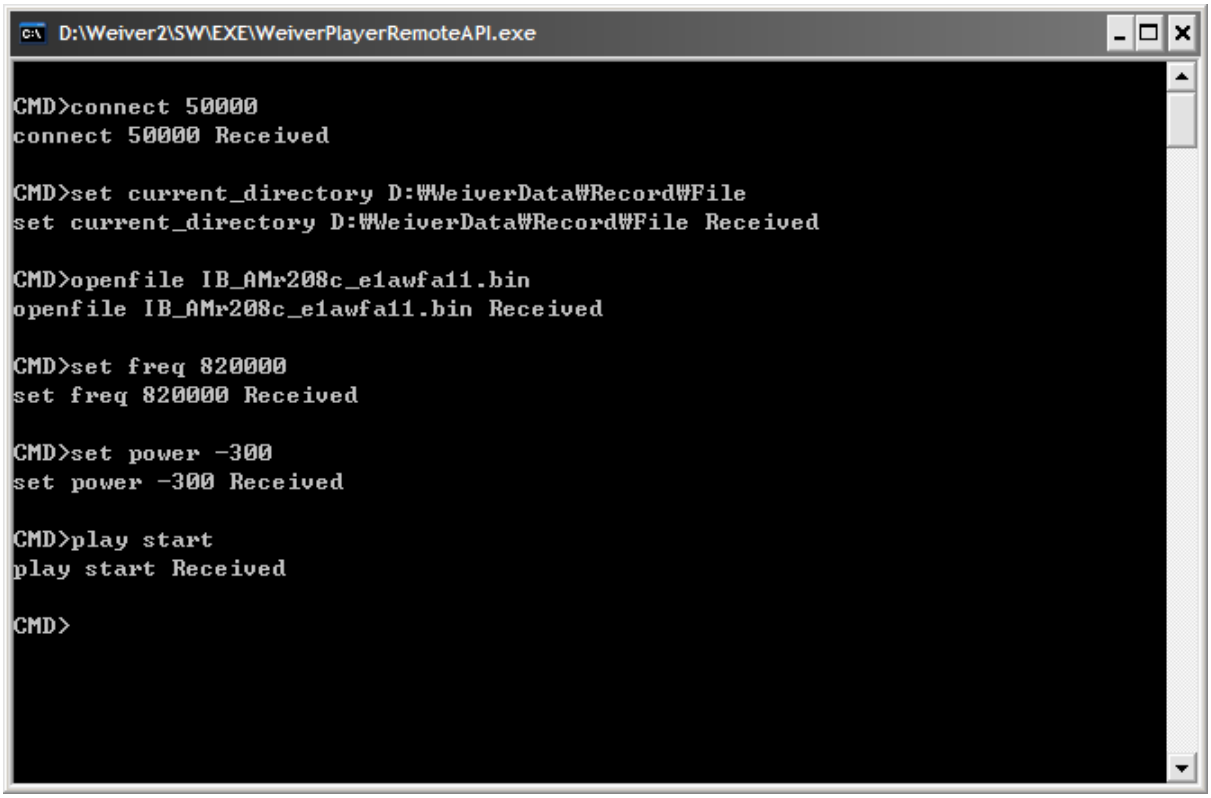
[Example: Play Weiver file]
connect 50000
set current_directory D:\WeiverData\Record\File
openfile lumantek-drivingtest-tdmb-agcon.wpj
play start



[Example: Play HD Radio™ FM file]
connect 50000
set current_directory D:\WeiverData\Record\File
openfile CC_FMr440d_e1wfr013.bin
set freq 92900000
set power -300
play start



[Example : Play HD Radio™ AM files]
connect 50000
set current_directory D:\WeiverData\Record\File
openfile IB_AMr208c_e1awfa11.bin
set freq 820000
set power -300
play start



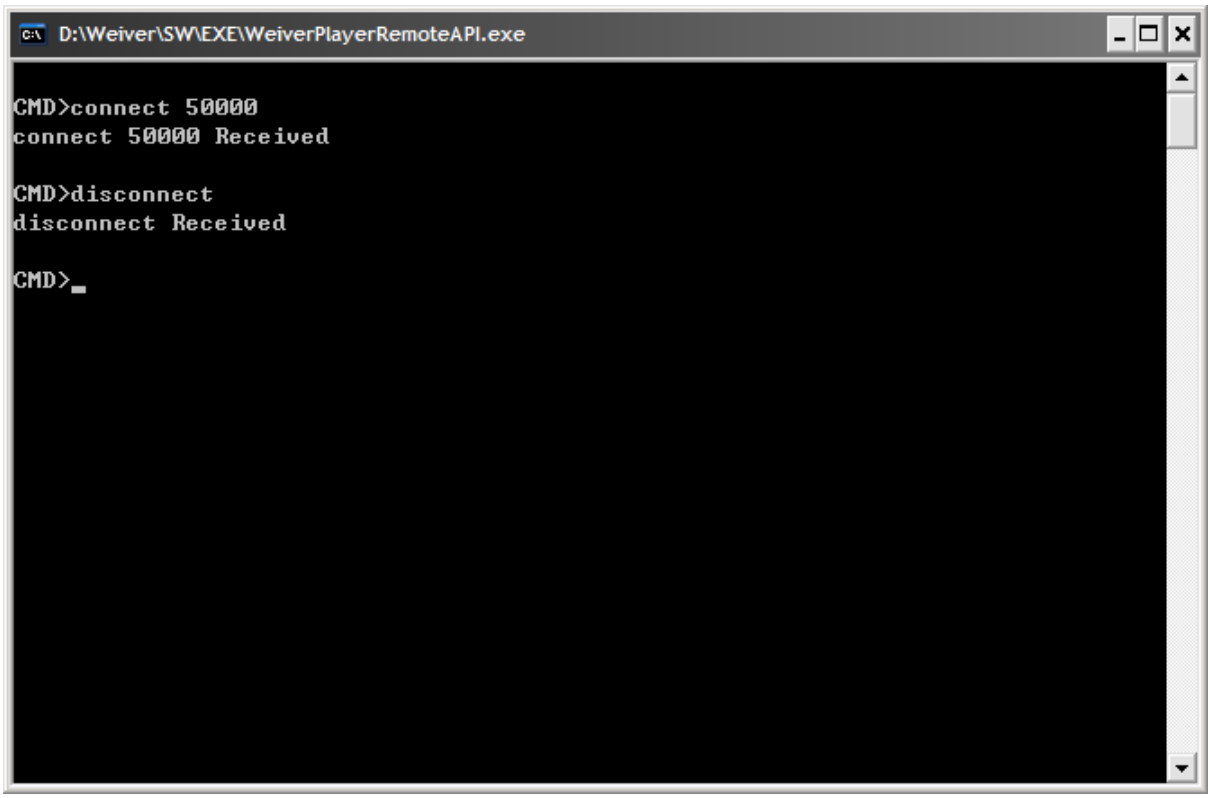
[Command examples]
'connect' and 'disconnect' command

- How to use
'connect port number' : Port number is the number of port where the response will be received after sending the packets.
'disconnect' : Terminate the connection

- Description
UDP port number must be notified when use 'connect' command to receive a proper response. Designate UDP port number as same as the u2_RecvPortFromWeiverPlayer variable value for program examples. UDP port number is adjustable.
Receiving UDP port number of the device is fixed at 50504.

Disconnect command will stop device operation.

- Example
1. connect 50000,
2. disconnect

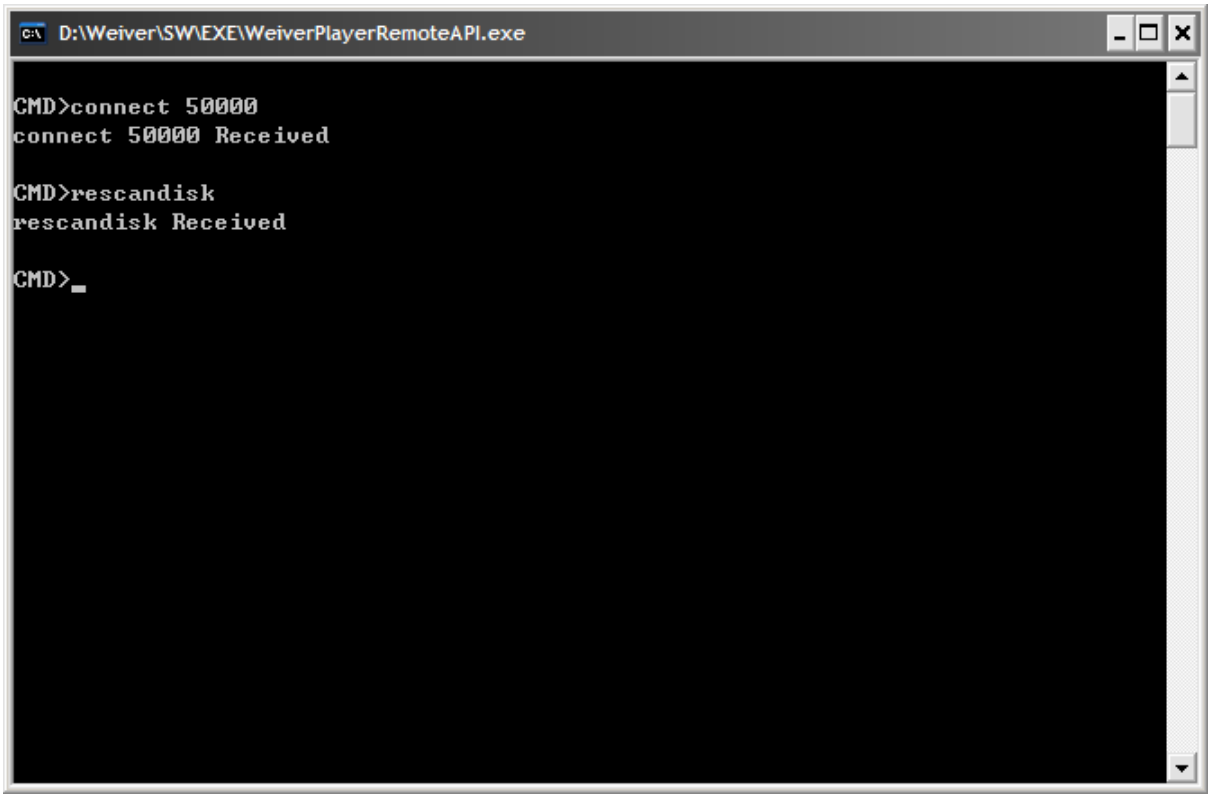


'rescandisk' command

How to use)
'rescandisk' command will refresh device disk drive.
If e-SATA disk drive is not found, 'rescandisk' command may able device to find the disk drive.

Example)
Use when disk drive is not refreshing automatically.

Examples)
1. rescandisk



'set current_directory', 'get current_directory' command
'get file_count', 'get filename', 'get folder_count', 'get foldername' command
'openfile', 'get openfile' command

How to use)
set current_directory : directory is the disk drive of the device.
get file_count : Bring number of files with .wpj file extension.
get filename index: The maximum number of Index is [number of files – 1].
get folder_count : set current_directory: Bring number of folders from directory.
get foldername index : The maximum number of Index is [number of folders – 1].
openfile filename : file name with .wpj extension.

Description)
Designate a xxx.wpj for play files.
Stop first before using 'openfile' command.

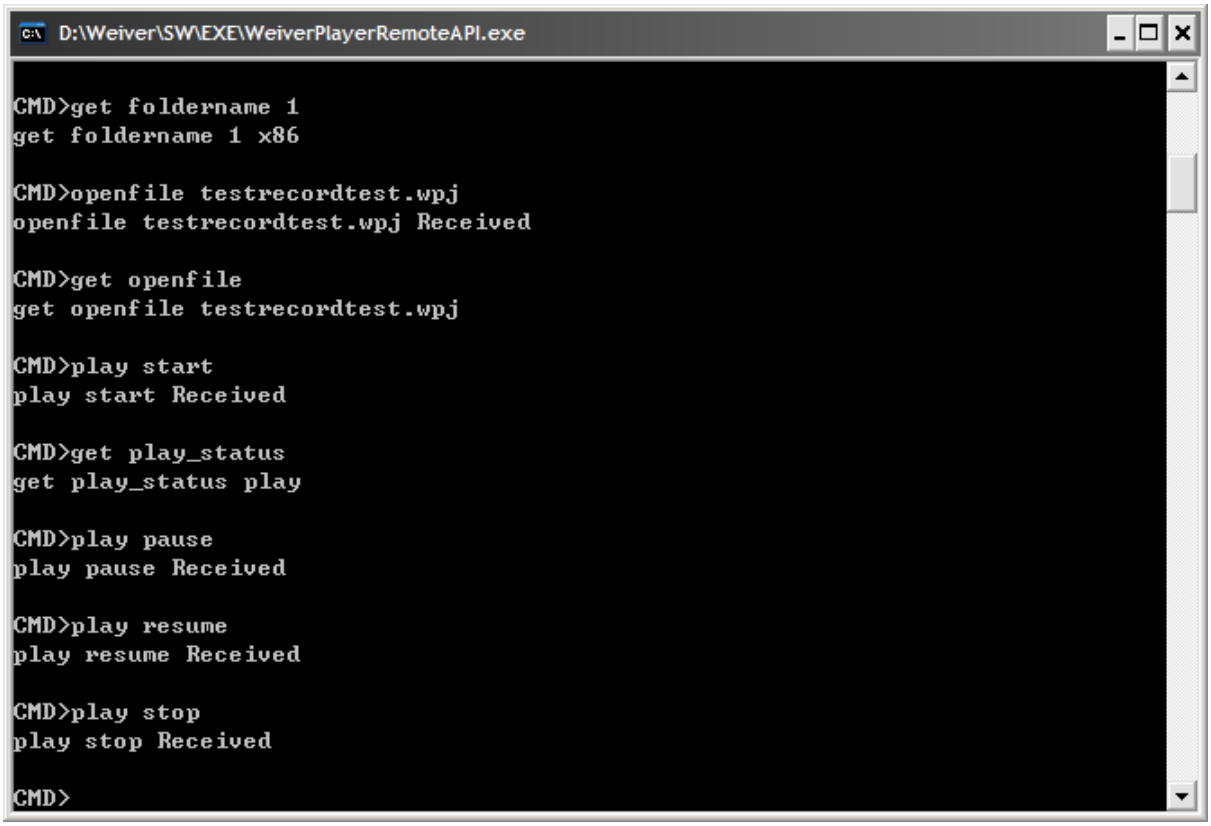
Example)
1. set current_directory f:\
2. get current_directory
3. get file_count
4. get filename 0
5. get filename 1
6. get folder_count
7. get foldername 0
8. get foldername 1
9. openfile testrecordtest.wpj
10. get openfile

‘play start’, ‘play stop’, ‘play pause’, ‘play resume’, ‘get play_status’ Command

How to use)
‘play start’, ‘play stop’, ‘play pause’, ‘play resume’, ‘get play_status’

Description)
Use ‘play start’, ‘play stop’, ‘play pause’ and ‘play resume’ command after selecting the file with ‘openfile’ command.
‘play start’ starts the file Play.
‘play stop’ stops the file Play.
‘play pause’ pauses the file Play.
‘play resume’ resumes the file Play.
‘get play_status’ retrieves the current status.

Example)
1. play start
2. get play_status
2. play pause
3. play resume
4. play stop

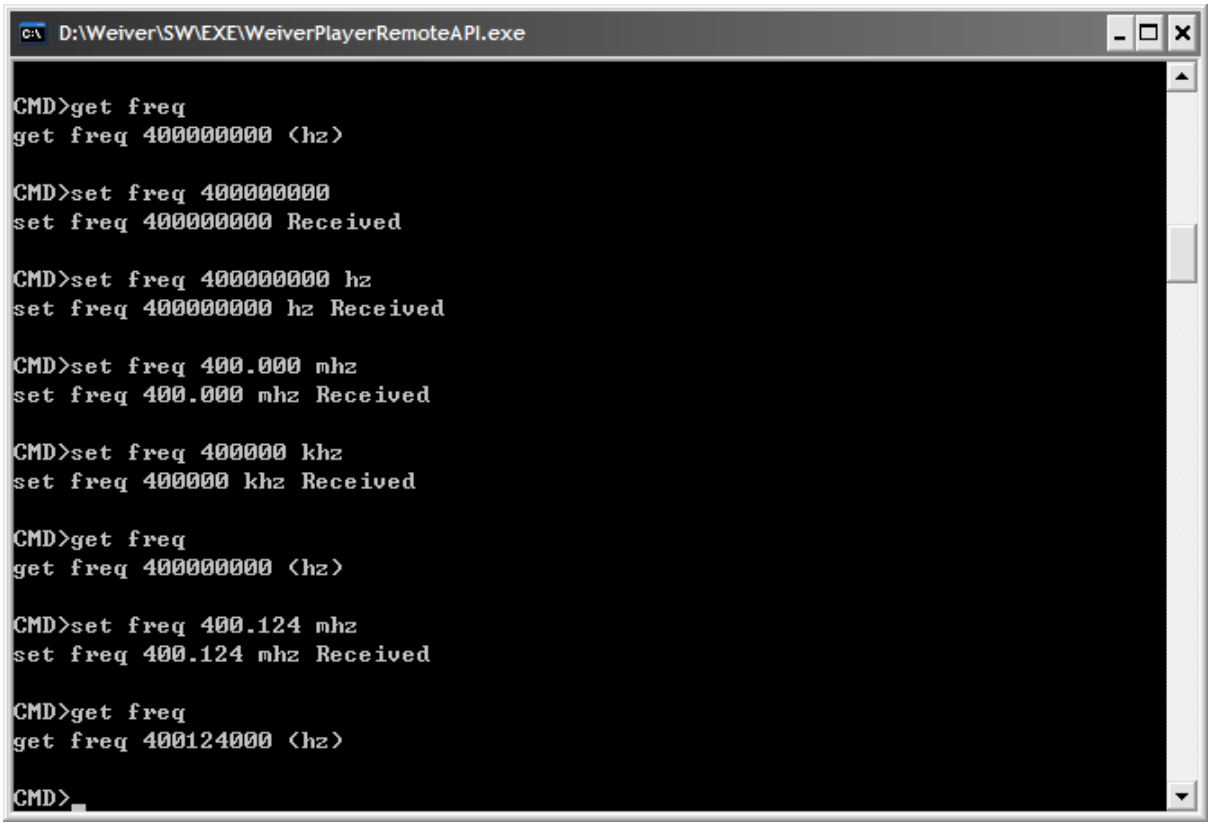


set freq', 'get freq' Command

how to use)
'set freq' frequency
Weiver 1.0 : frequency range is from 50000000(50MHz) to 1000000000(1GHz).
Weiver 2.0 : Please refer to 'get freq_min', get 'freq_max' for frequency commands.

Description)
Select the file with 'openfile' commend and adjust frequency with 'set freq' command. The units are in Hz, KHz, and MHz.

Example)
1. set freq 400000000 (400MHz configuration)
2. get freq
3. set freq 400 MHz (400MHz configuration)
4. set freq 400000 KHz (400MHz configuration)
5. set freq 400000000 Hz (400MHz configuration)

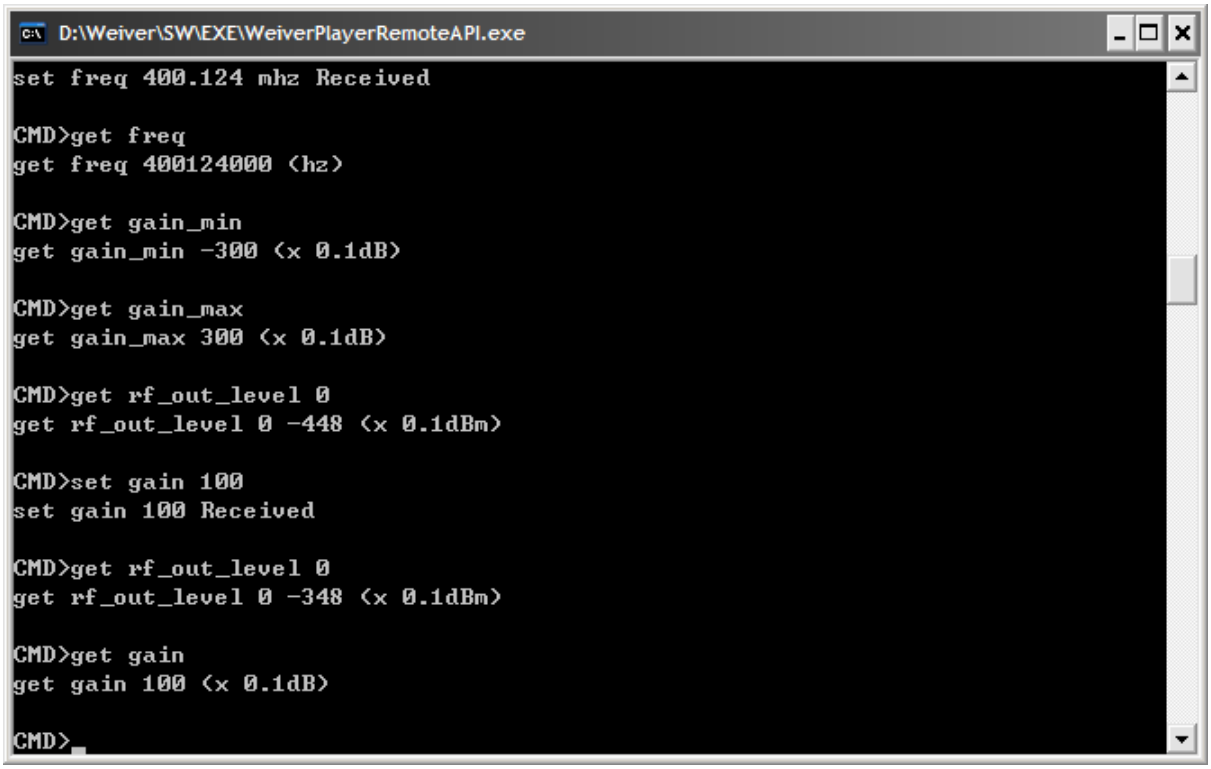


'get gain_min', 'get gain_max', 'get rf_out_level 0', 'get gain', 'set gain' command.

How to use)
To apply more gain during Play after using 'openfile' command.
Generally, the range of Gain is from -30dB to +30dB. However, when the file has been selected, the range of the Gain can be checked with 'get gain_min' and 'get gain_max' command.
'get rf_out_level' time: the range of time is the total play time of the file.

Description)
Units in 0.1 dB.
For example, type in 'set gain 100' to apply +10dB Gain.

Example)
1. get gain_min
2. get gain_max
3. get rf_out_level 0
4. set gain 100
5. get rf_out_level 0
6. get gain

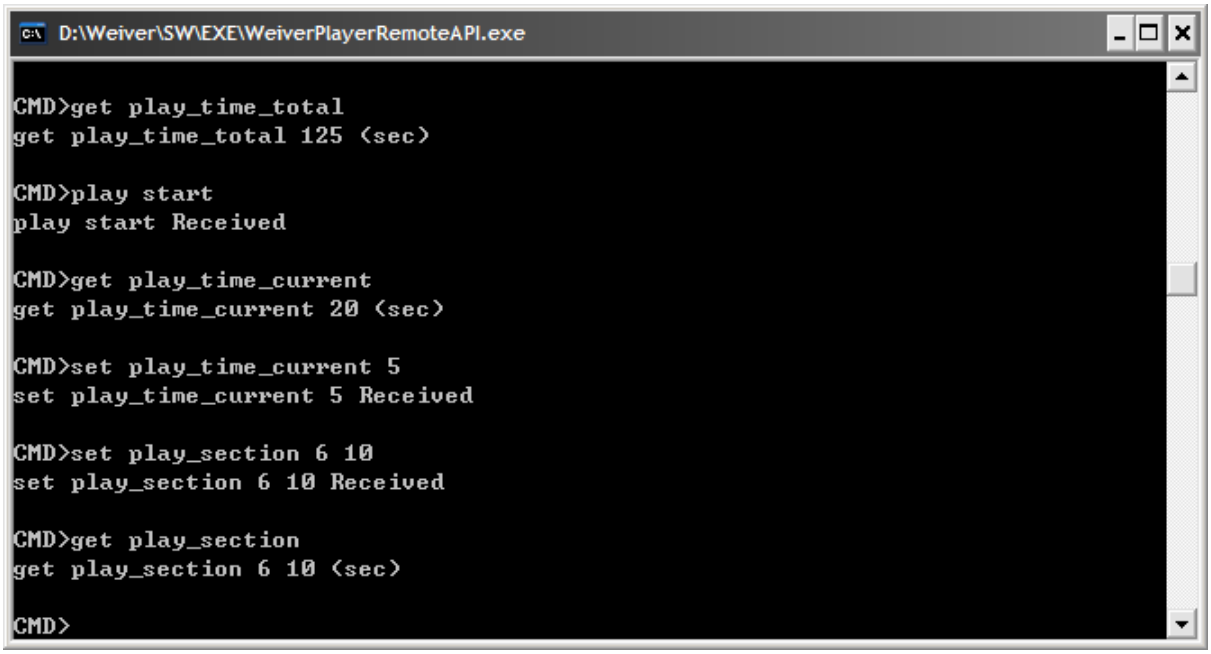


'get play_time_total', 'get play_time_current', 'set play_time_current', 'get play_section' and 'set play_section' command

How to use)
'play time' related commands: allows designate the play starting points and obtain 'time' related information of the file after such file has been selected by using 'openfile' command.
'get play_time_total' command: displays the total play time of the selected file in seconds.
'set play_time_current' command: allows to configure starting and end point of the file for a continuous loop play. Time at starting point must be less than the ending point, and the time at the ending point must be less than the total play time of the file.

Description)
Units in seconds.
Possible playback time of the play file is 630 seconds.
'pp 5' moves the file play point to the 5 sec. location
'playrepeat' 6 10: Continuous loop play between 6 sec and 10 sec file location.

Example)
1. get play_time_total
2. get play_time_current
3. set play_time_current 5
4. set play_section 6 10
5. get play_section

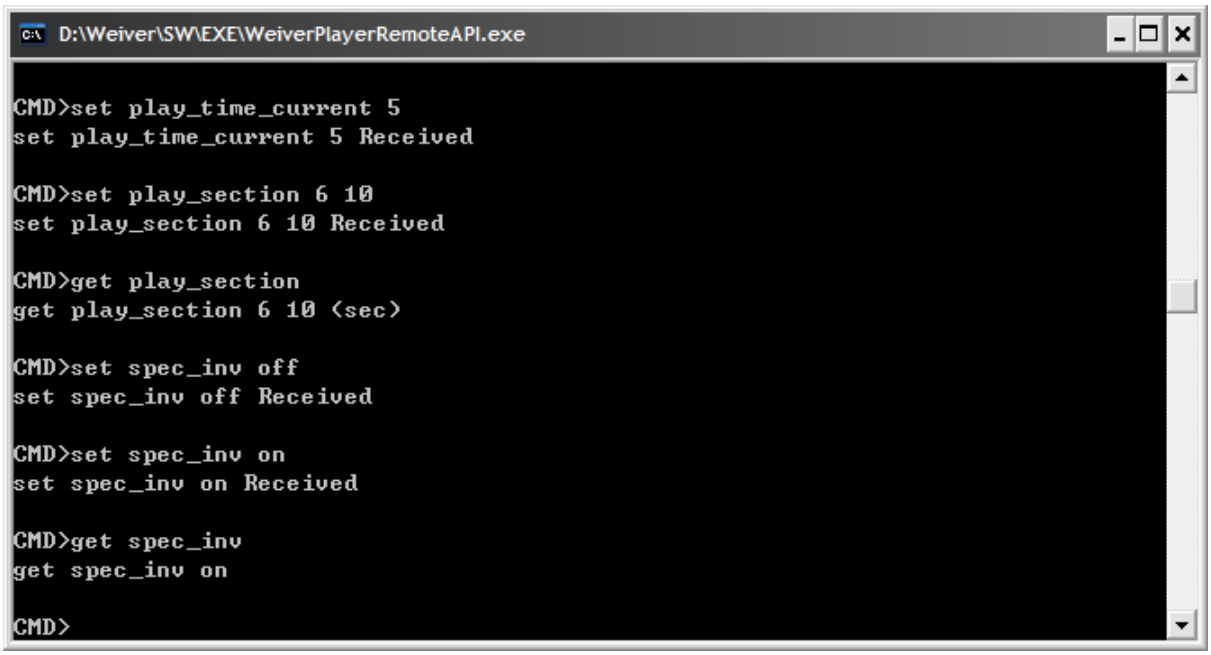


‘set spec_inv off/on’ and ‘get spec_inv’ command

How to use)
‘set spec_inv off’: Transmitting the original RF spectrum of the play file.
‘set spec_inv on’ :Transmitting the inverted RF spectrum of the play file

Description)
Use ‘set spec_inv’ command to transmit inverted spectrum

Example)
1. set spec_inv off
2. set spec_inv on



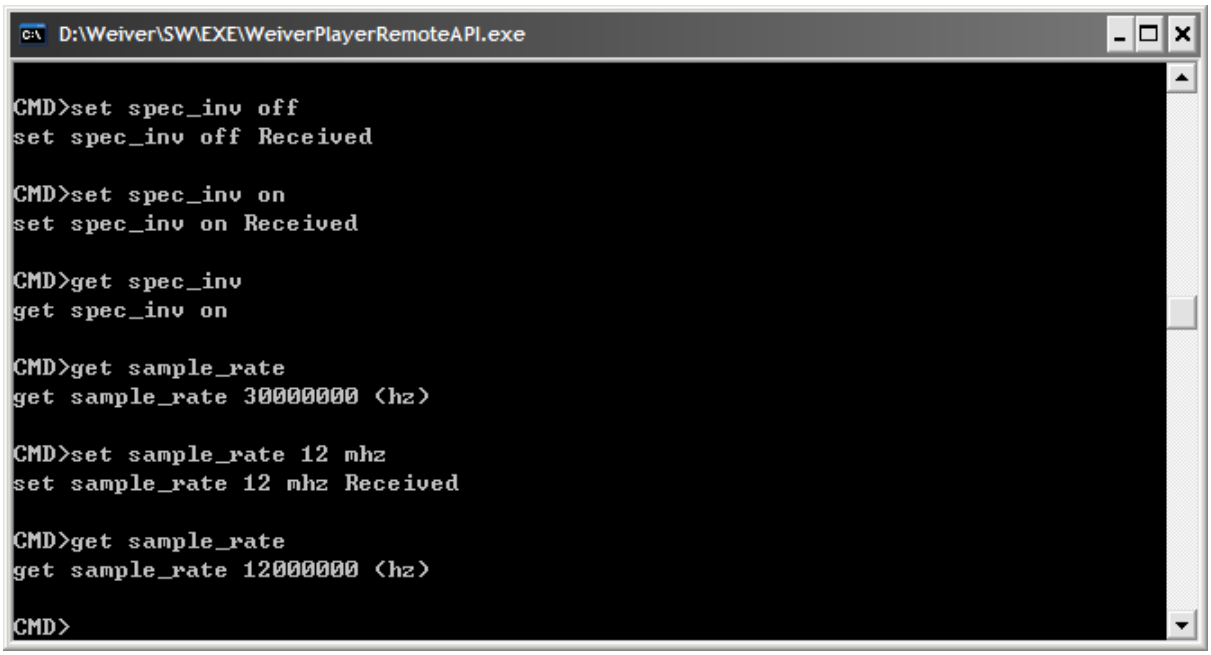
'set sample_rate' and 'get sample_rate' Command

How to use)
'set sample_rate' sample frequency
Weiver 1.0 : Sample frequency range 5000000(5MHz) ~ 30000000(30MHz)
Weiver 2.0: Sample frequency range 1250000(1.25MHz) ~ 70000000(70MHz)

Sample frequency means the I/Q Rate. I/Q rate for BW 8MHz is 10MHz when BW 24MHz is 30MHz.

Description)
Sample frequency will be selected automatically when file has been selected by 'openfile' command
Use 'set sample_rate' to manually adjust the rate.
Units in Hz.

Example)
1. set sample_rate 12000000, set sample_rate 12 mhz, set sample_rate 12000 khz
2. get sample_rate

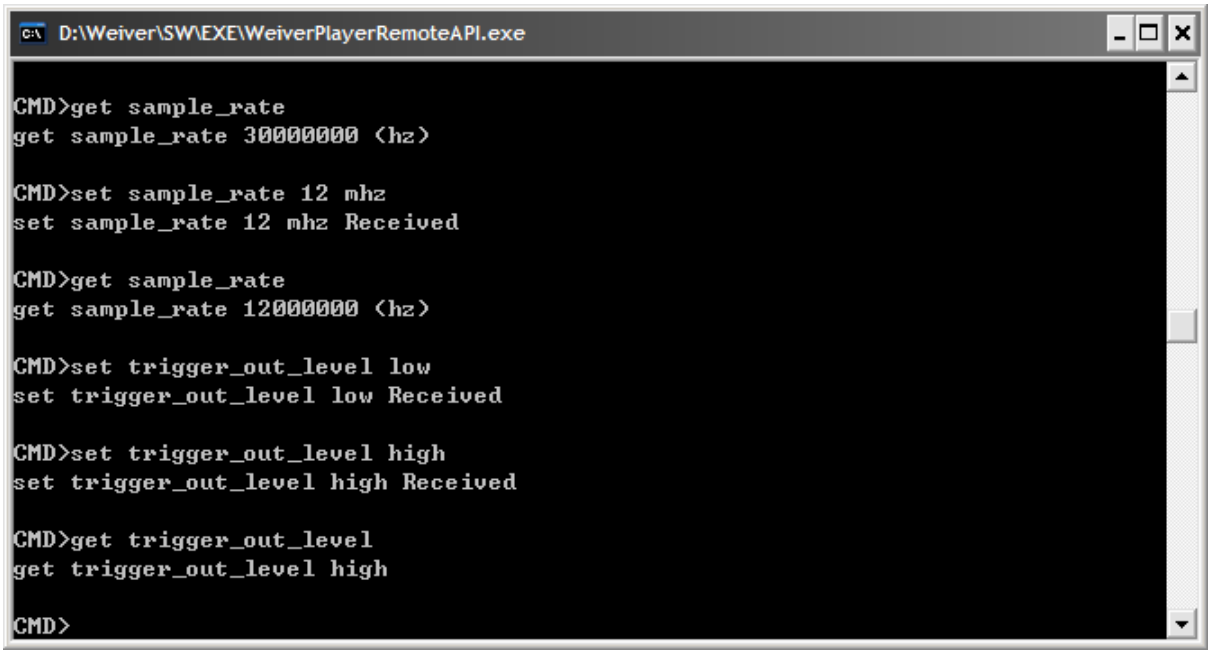


‘set trigger_out_level’ and ‘get trigger_out_level’ Command.

How to use)
WeiverPlayer 1.0, Weiver 2.0 and WeiverPlayer 2.0 have Trigger out port.
‘set trigger_out_level low’: Transmitting LOW level(0V) via Trigger Out port.
‘set trigger_out_level high’: Transmitting HIGH level(3.3V) via Trigger Out port.

Description)
Configure output level of ‘Trigger Out’ port.

Example)
1. set trigger_out_level low
2. set trigger_out_level high



‘set extclk’ and ‘get extclk’ Command

How to use)
It can synchronized external 10MHz with device operating clock
‘set extclk off’: Use internal 10MHz clock.
‘set extclk on’: Sync with external 10MHz clock input

Description)
Generally use ‘set extclk off’.

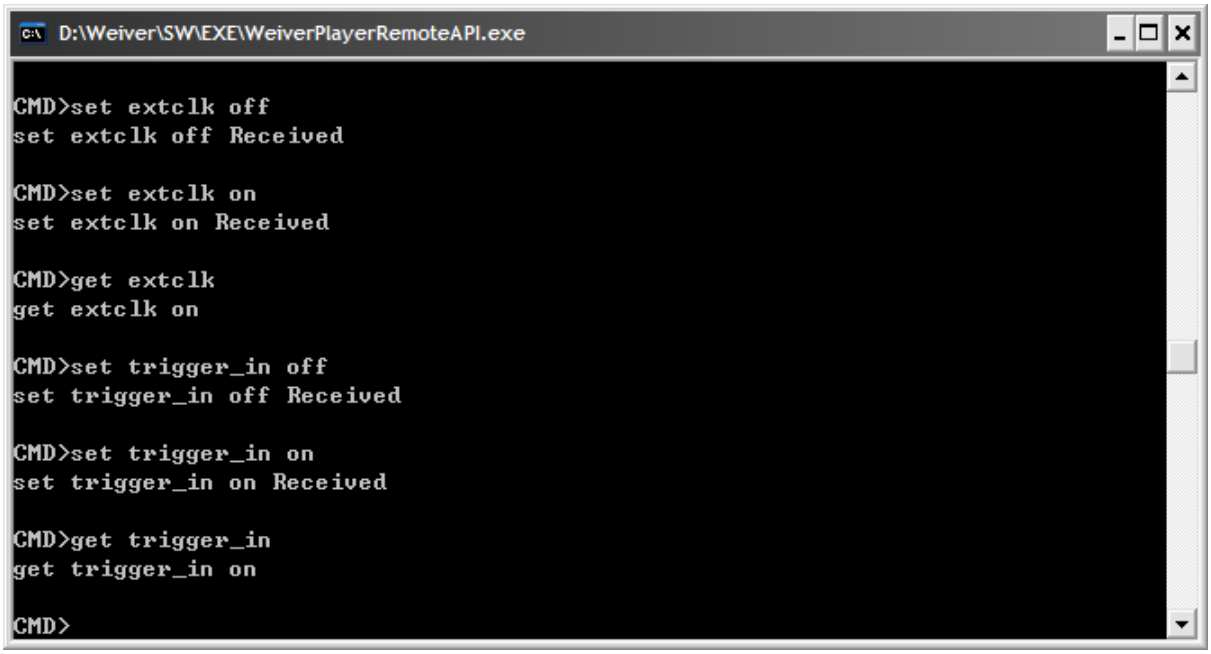
Example)
1. set extclk off
2. set extclk on
3. get extclk

‘set trigger_in’ and ‘get trigger_in’ command

How to use)
‘set trigger_in off’: RF transmission regardless of its input level of ‘trigger In’ port.
‘set trigger_in on’: RF transmitting, only if the input level is HIGH(3.3V)

Description)
When ‘set trigger_in’ command was used for play start, the file play is based on the input level of ‘trigger In’ port. It enables the synchronized operation of multiple Weivers.

Example)
1. set trigger_in off
2. set trigger_in on
3. get trigger_in



[WeiverPlayerRemoteAPI Source]

```
CWinApp theApp;
using namespace std;
const unsigned short u2_WeiverPlayerPort = WEIVER_PLAYER_API_RX_PORT;
const unsigned short u2_RecvPortFromWeiverPlayer = 50000;
const char s1a_WeiverPlayerIP[] = "192.168.100.1";
//const char s1a_WeiverPlayerIP[] = "127.0.0.1";
//const char s1a_WeiverPlayerIP[] = "192.168.10.15";

int _tmain(int argc, TCHAR* argv[], TCHAR* envp[])
{
		int nRetCode = 0;

		if (!AfxWinInit(::GetModuleHandle(NULL), NULL, ::GetCommandLine(), 0))
		{
			_tprintf(_T("AfxWinInit failed.\n"));
			nRetCode = 1;
		}
		else
		{
			wchar_t wCmdLine[1000];
			wchar_t wSendCmd[1000];

			WSADATA wsaData;
			WSAStartup(MAKEWORD(2,2), &wsaData);

			while(nRetCode == 0)
			{
				memset(wCmdLine, 0, 1000);
				memset(wSendCmd, 0, 1000);

				wprintf(_T("\nCMD>"));

				_getws_s(wCmdLine, 1000);

				if(wcsncmp(wCmdLine, _T("exit"), wcslen(_T("exit")))
== 0)
				{
					nRetCode = 1;
				}
```

```
                    else
                    {
                    SendToStringCMD((char*)wCmdLine, 2*wcslen(wCmdLine));
                    }
                    wprintf(_T("\n"));
                    }
                    WSACleanup();
        }

        return nRetCode;
}

void SendToStringCMD(char *s1p_Data, unsigned int u4_Size)
{
        SOCKET h_Socket;
        SOCKADDR_IN Addr;

        int SentBytes;

        WvPlayerRemotePacket_t t_Packet;
        WvPlayerRemotePacket_tp tp_Packet;

        char *pSendData = (char *)&t_Packet;


        char   message[2000];
        memset(message, 0, 2000);


        int clntAddrSize = sizeof(Addr);

        h_Socket = socket(PF_INET, SOCK_DGRAM, IPPROTO_UDP);

        memset(&Addr, 0, sizeof(Addr));
        Addr.sin_family = AF_INET;
        Addr.sin_port = htons(u2_RecvPortFromWeiverPlayer);
        Addr.sin_addr.s_addr = htonl(INADDR_ANY);


        struct timeval timeout;
        timeout.tv_sec = 2;//2 seconds
        timeout.tv_usec = 0;
        int optlen = sizeof(timeout);
```

```
setsockopt(h_Socket, SOL_SOCKET, SO_RCVTIMEO, (const char *)&timeout, optlen);
 if(bind(h_Socket, (SOCKADDR*)&Addr, sizeof(Addr)) == SOCKET_ERROR)
 {
                    wprintf(_T("bind() Error"));
                    closesocket(h_Socket);
                    return;
 }


 t_Packet.u4_ID = WEIVER_PLAYER_ID;
 t_Packet.u4_Reserved = WV_PLAYER_REMOTE_CMD_SET_STRING;
 t_Packet.u4_DataSize = u4_Size;

 memset(t_Packet.u1a_Data, 0, 1000);
 memcpy(t_Packet.u1a_Data, s1p_Data, u4_Size);

 memset(&Addr, 0, sizeof(Addr));
 Addr.sin_family = AF_INET;
 Addr.sin_port = htons(u2_WeiverPlayerPort);
 Addr.sin_addr.s_addr = inet_addr(s1a_WeiverPlayerIP);

 SentBytes = sendto(h_Socket
                                                ,pSendData
                                                ,4/*sizeof(t_Packet.u4_ID)*/
                     + 4/*sizeof(t_Packet.u4_Reserved)*/
                     + 4/*sizeof(t_Packet.u4_DataSize)*/ + u4_Size
                     ,0, (SOCKADDR*)&Addr, sizeof(SOCKADDR_IN));

 if(SentBytes == SOCKET_ERROR)
 {
                    wprintf(_T("\nSend Fail"));
 }
```

```
		else
		{
			memset(&Addr, 0, sizeof(Addr));

			int length = recvfrom(h_Socket
			    , message, 2000, 0, (SOCKADDR*)&Addr, &clntAddrSize);

			if(length == -1)
			{
				wprintf(_T("Time Out"));
			}
			else
			{
				tp_Packet = (WvPlayerRemotePacket_tp)(message);

				wprintf(_T("%s"), tp_Packet->u1a_Data);
			}
		}

		closesocket(h_Socket);
}
```