

User Guide – AT Command Specification

Aurawave AW100 Series

v1.3.0, 13 Oct 2026

Revision History

Version	Date	Notes
1.0	7 Aug 2025	Initial release.
1.0-r2	5 Sep 2025	Add examples. Fix error codes.
1.0-r3	24 Oct 2025	Fix incorrect command in 3.2 step 8.
1.0-r4	11 Feb 2026	Update AT+PBSNKDEMO. Add Sink use case. Clarify line endings on the serial interface.
1.1-r1	23 Feb 2026	Update AT+PBSRCCONF, AT+DFU. Add AT+PBGRPCONF, AT+PBSTRMCONF, AT+USBID.
1.1-r2	04 Mar 2026	Clarifications; complete serial terminal settings; Broadcast Sink Demo instructions; I2S settings for the 40-pin header analog input; AT+CONFDEF resets the board.
1.1-r3	11 Mar 2026	Small clarifications. Broadcast Sink Demo only works on boards with analog outputs.
1.2-r1	31 Mar 2026	Update AT+PBSRCCONF (Broadcast ID parameter). Update AT+PBGRPCONF (Subgroup Audio Source parameter).
1.2-r2	26 May 2026	Remove AT+PBSNKDEMO. Add AT+PBSNKSCAN, AT+PBSNKEXPLORE, AT+PBSNKSTART, AT+PBSNKSTOP. Add small clarifications.
1.2-r3	22 Jun 2026	Update AT+AUDDEVLIST. Clarify that the I2S_MAIN audio device refers to the analog jacks. Uniformize single and double quote characters.
1.3	13 Oct 2026	USB Audio Class 2; macOS host support. Add AT+PBSNKCONF, AT+PBSNKSTATE, AT+USBAUD. Sink automatic retry/reconnect. Added 24kHz quality presets. Default device name ends in the chip ID and names the USB device and new broadcast sources. AT+PBSNKSTART: two formats, by name (a prefix by default, or exact only) or by Broadcast ID.

Contents

Revision History	2
1. Introduction	5
1.1. Scope of the Document	5
1.2. Conventions and Abbreviations	5
2. AT Command Syntax	7
2.1. Set Operation <CMD>=[...]	7
2.2. Read Operation <CMD>?	7
2.3. Test Operation <CMD>=?	8
2.4. Execute Operation <CMD>[=...]	8
2.5. Responses	8
2.6. Entering Successive AT Commands	9
2.7. Echo	9
2.8. Supported Character Set	9
2.9. Supported Line Endings	10
3. LED Indications	11
4. General Commands	12
4.1. Attention command (AT)	12
4.2. Echo Mode disable (ATE0)	12
4.3. Echo Mode enable (ATE1)	12
4.4. Version (AT+CGMR)	13
4.5. List available commands (AT+CLAC)	13
4.6. Reboot (AT+REBOOT)	14
5. Bluetooth Commands	15
5.1. Device name (AT+BTDEVNAME)	15
6. LE Audio Commands	17
6.1. Public Broadcast Source create (AT+PBSRCCREATE)	17
6.2. Public Broadcast Source delete (AT+PBSRCDEL)	17
6.3. Public Broadcast Source configure (AT+PBSRCCONF)	18
6.4. Public Broadcast subgroup configure (AT+PBGRPCONF)	20
6.5. Public Broadcast stream configure (AT+PBSTRMCONF)	22
6.6. Public Broadcast Source start (AT+PBSRCSTART)	23
6.7. Public Broadcast Source state (AT+PBSRCSTATE)	24
6.8. Public Broadcast Source stop (AT+PBSRCSTOP)	25
6.9. Public Broadcast Sink scan (AT+PBSNKSCAN)	25
6.10. Public Broadcast Sink explore (AT+PBSNKEXPLORE)	26
6.11. Public Broadcast Sink configure (AT+PBSNKCONF)	27
6.12. Public Broadcast Sink start (AT+PBSNKSTART)	29
6.13. Public Broadcast Sink stop (AT+PBSNKSTOP)	31

6.14. Public Broadcast Sink state (AT+PBSNKSTATE)	31
7. Audio System Commands	33
7.1. Audio devices list (AT+AUDDEVLIST)	33
7.2. Audio device set (AT+AUDDEVSET)	33
8. System Configuration Commands	35
8.1. Configuration save (AT+CONFSAVE)	35
8.2. Configuration defaults (AT+CONFDEF)	35
9. USB Commands	37
9.1. USB VID/PID (AT+USBID)	37
9.2. USB audio stream status (AT+USBAUD)	37
10. Firmware Update Commands	39
10.1. Enter DFU Mode (AT+DFU)	39
11. Worked Examples	40
11.1. The Factory Default State	40
11.2. Example 1: A Named Broadcast from the Analog Input, Kept Across Power Cycles	40
11.3. Example 2: An AW100 Receiving Another AW100	41
11.4. Example 3: A Stereo Pair of AW100 Sinks	42
11.5. Example 4: Receiving a Phone or Third-Party Broadcast	43
11.6. Example 5: A Two-Language Broadcast and Sinks Choosing by Language	43
12. Supported Result Codes	46
12.1. Summary of CME ERROR Codes	46
12.2. Summary of Unsolicited Result Codes	47
Additional Information - Cloud2GND	48
Additional Information - Ezurio	49

1. Introduction

This document applies to the following Aurawave firmware:

- v1.3.0, tag 1.3.0-63f2ff1 (reported by AT+CGMR)

1.1. Scope of the Document

This document describes the AT command set of the Cloud2GND Aurawave AW100 series modules. The commands configure an AW100 that works stand-alone. Such an AW100 is set up once, its configuration is saved with AT+CONFSAVE, and it starts by itself at every power-up. The commands also control an AW100 from a host while it runs. The document is generated for each firmware release. Use the document whose version matches the firmware.

1.2. Conventions and Abbreviations

The AW100 and the Host

In this document **the AW100** is the Cloud2GND Aurawave AW100 series module.

The **host** is whatever sends AT commands to the AW100 and reads its responses. It can be a terminal program on a PC, an app on a mobile phone, or a microcontroller (MCU) connected over UART or USB. On USB the AW100 presents a serial port (USB CDC) next to its USB audio device. The serial port is a COM port on Windows, or /dev/cu.usbmodem... on macOS.

AT Commands

An AT command is a line of text the host sends to the AW100. The AW100 answers each command with any information it returns and a final result: OK, ERROR or +CME ERROR: <cause value>. Every command starts with AT. The name after the prefix usually starts with +. The format follows ITU-T V.250 and is described in [AT Command Syntax](#).

The command names are built from these parts:

Part	Meaning
PB	Public Broadcast (the Auracast™ broadcast the AW100 sends or receives)
SRC, SNK	Broadcast Source, Broadcast Sink
GRP, STRM	Subgroup, stream
CONF	Configuration
AUD, DEV	Audio, device
BT, USB	Bluetooth, USB

Abbreviations

Abbreviation	Meaning
ASCII	American Standard Code for Information Interchange
BIG	Broadcast Isochronous Group: the set of BISes one Broadcast Source transmits
BIS	Broadcast Isochronous Stream: one audio stream within a BIG
CDC	USB Communications Device Class (the AW100's serial port over USB)
CR, LF	Carriage return (\r, 0x0D), line feed (\n, 0x0A)
DFU	Device firmware update (the mode entered with AT+DFU)
I2S	Inter-IC Sound: the digital audio interface to the codec or an external audio device
LC3	Low Complexity Communication Codec, the LE Audio codec
LE	Bluetooth Low Energy
LE Audio	The Bluetooth audio architecture over LE, of which Auracast broadcasts are part
LED	Light-emitting diode (the status indicator)
MCU	Microcontroller unit
PD	Presentation delay: the time a Broadcast Sink waits before it plays the audio, announced by the source
QoS	Quality of service (the retransmission and latency settings of a broadcast)
UART	Universal asynchronous receiver-transmitter (a wired serial port)
USB	Universal Serial Bus
VID, PID	USB vendor ID and product ID

The +CME ERROR codes and their names (EINVAL, EBUSY, ...) are listed in [Summary of CME ERROR Codes](#).

2. AT Command Syntax

Each command line starts with AT and ends with a line ending: CR, LF or CRLF.

Words in <angle brackets> are placeholders for parameters. Words in [square brackets] are optional. **The brackets are not typed in the command line.**

String parameters are enclosed in double quotes, "like this". A string cannot contain a double quote. "" is the empty string. Numbers are typed without quotes.

The prefix and the command name may be typed in upper or lower case: at+cgmr is AT+CGMR. Quoted parameters keep the case they are sent in. A tag such as "NAME" must be given in upper case. Responses are always in upper case.

The command name is followed by characters that select the operation. The operations are described in the following subsections.

2.1. Set Operation <CMD>=[...]

A Set Operation sets a value or carries out an action with parameters. Example:

```
AT+CMD=1
```

Where:

- **AT** is the command line prefix
- **+** is the prefix for commands
- **CMD** is the body of a basic command
- **=** specifies the Set Operation
- **1** is a parameter (multiple parameters are separated by commas)

2.2. Read Operation <CMD>?

A Read Operation reads the current value. The entry of each command states whether it has a read form. The configuration commands read a value through a set form with fewer parameters, for example AT+PBRCCNF=0, "NAME". Example of a read form:

```
AT+CMD?
```

Where:

- **AT** is the command line prefix
- **+** is the prefix for commands
- **CMD** is the body of a basic command

- ? specifies the Read Operation

2.3. Test Operation <CMD>=?

A Test Operation confirms that the command exists. Every command answers OK to its test form. Commands with parameters first print a line that shows them. Example:

```
AT+CMD=?
```

Where:

- **AT** is the command line prefix
- **+** is the prefix for commands
- **CMD** is the body of a basic command
- **=?** specifies the Test Operation

2.4. Execute Operation <CMD>[=...]

An Execute Operation needs no operator character unless it takes a parameter. Example:

```
AT+CMD
```

Where:

- **AT** is the command line prefix
- **+** is the prefix for commands
- **CMD** is the body of a basic command

2.5. Responses

The AW100 answers every command line with a final result. It is one of the following:

```
OK<CR><LF>
ERROR<CR><LF>
+CME ERROR: <cause value><CR><LF>
```

OK means the command was carried out.

ERROR means the line could not be used. This is the case for:

- a line that does not start with AT, or an unknown command

- stray spaces, or a character outside printable ASCII
- the wrong number of parameters, an empty parameter, or a parameter for a command that takes none
- a parameter that is neither a number nor a quoted string
- a read form (AT+X?) on a command that has none
- a line that is too long

+CME ERROR: <cause value> means the command was understood but could not be carried out. The **Errors** row of each command explains its cause values. [Summary of CME ERROR Codes](#) lists them.

A command that requests a state answers OK whenever the AW100 ends up in that state. It does not matter whether anything had to change. Selecting the active audio device again, starting a running source, stopping a stopped source and creating the source that already exists all answer OK.

A command that writes a configuration value answers OK only when the value was accepted. A source that has been started and a sink that has found its source refuse every configuration write, even for a value that is already set. An error always means that the request was not carried out.

Three cause values can come from any command that takes parameters:

- 22: a number was given in quotes, or a decimal number contains letters.
- 12: a value is longer than its field.
- 34: a number is above 2147483647.

Some commands also send information response lines before the final result. The **Response** row of the command gives them. An information response line usually starts with + and the command name:

```
+CMD: [...]<CR><LF>
```

2.6. Entering Successive AT Commands

Send one command line at a time. Wait for its final result (OK, ERROR or +CME ERROR) before sending the next. A command such as AT+PBSNKSCAN answers only when it has finished.

2.7. Echo

The AW100 echoes the characters it receives. ATE0 turns the echo off and ATE1 turns it on again. Echo is on after every boot.

2.8. Supported Character Set

Only printable ASCII characters can be used in AT commands.

2.9. Supported Line Endings

CR, LF and CRLF are accepted as line endings. A command line of 192 characters or more is answered with ERROR.

3. LED Indications

The AW100 has one status LED with a green and a blue element. It shows green, blue, both together (cyan) or off, steady or blinking. The LED shows the state of the Broadcast Source or the Broadcast Sink, whichever changed last. An AW100 that acts as a sink shows the sink states. An AW100 that acts as a source shows the source states. The same states are read over AT with AT+PBSNKSTATE and AT+PBSRCSTATE.

LED	Meaning	State read over AT
Cyan, three flashes	The AW100 has powered up or rebooted and the firmware is running. The flashes end after about 1.5 seconds, or earlier if a source or sink starts. The LED then shows the state below. It goes off when no source or sink is started	-
Off	Nothing is started: no broadcast on the air and no sink started. Also after a stop	IDLE (source or sink), source STOPPED
Blue, medium blink	The broadcast is being brought up after AT+PBSRCSTART	source STARTING
Blue, steady	The broadcast is on the air	source STREAMING
Green, slow blink	The sink is searching for its source. It retries until the source is found or the sink is stopped	sink SEARCHING
Green, fast blink	The sink has been started and is synchronising to its source	sink STARTING
Green, steady	The sink is receiving audio	sink STREAMING
Green, medium blink	The source's audio streams ended while the sink stays synchronised to its announcement. The source paused or stopped. Audio resumes by itself when the streams return	sink STOPPED
Green and blue alternating, slow	The source was found but none of its streams fits the sink's location and language (AT+PBSNKCONF). Nothing is played until the configuration or the broadcast changes	sink NOMATCH
Cyan, steady	The button is pressed. Releasing it reboots the AW100	-
Cyan, fast blink	The button has been held for 5 seconds. Releasing it erases the saved configuration and reboots the AW100	-
Green and blue alternating, fast	The firmware could not recover from an internal fault. Power-cycle the AW100	-

Slow means about one flash per second, medium about two flashes per second, fast about four.

An AW100 normally recovers from an internal fault by rebooting. The power-up flashes then appear again. After three faults in quick succession the AW100 discards the saved configuration and starts with the factory defaults.

The bootloader's own indications during a firmware update are outside this document.

4. General Commands

4.1. Attention command (AT)

Execute Operation	AT
Description	Confirms that the AT interface is active. Answers OK and nothing else.
Parameters	None
Response	OK
Example	<pre>AT OK</pre>
Test Operation	AT=?
Description	Confirms the command exists. No parameter information is reported.
Response	OK
Example	<pre>AT=? OK</pre>

4.2. Echo Mode disable (ATE0)

Execute Operation	ATE0
Description	Disables echo. Characters sent to the AW100 are not echoed back. Echo stays off until ATE1 or a reboot. Note: The ATE0 line itself is still echoed. Echo stops from the next command.
Parameters	None
Response	OK
Example	<pre>ATE0 OK</pre>
Test Operation	ATE0=?
Description	Confirms the command exists. No parameter information is reported.
Response	OK
Example	<pre>ATE0=? OK</pre>

4.3. Echo Mode enable (ATE1)

Execute Operation	ATE1
Description	Enables echo. Characters sent to the AW100 are echoed back. Echo is on at boot.

Parameters	None
Response	OK
Example	<pre>ATE1 OK</pre>
Test Operation	ATE1=?
Description	Confirms the command exists. No parameter information is reported.
Response	OK
Example	<pre>ATE1=? OK</pre>

4.4. Version (AT+CGMR)

Execute Operation	AT+CGMR
Description	Reports the application version. Reports the SoftDevice Controller version as well when the Bluetooth controller is Nordic's. Firmware built for the external-antenna variant appends (EXT-ANT) to the application line. Note: In the standard variant the application line ends with a space after the closing quote.
Parameters	<version> The release tag, for example "1.3.0-f562cb9". A build that is not on a release tag reports the tag, the number of commits since it and a 12-character commit ID.
Response	<pre>+CGMR: Application: "<version>" [(EXT-ANT)] [Controller: SoftDevice: Version <version as string> (<version as hex>), Revision <revision>] OK</pre>
Example	<pre>AT+CGMR +CGMR: Application: "1.3.0-f562cb9" Controller: SoftDevice: Version 6.2 (0x10), Revision 8224 OK</pre>
Test Operation	AT+CGMR=?
Description	Confirms the command exists. No parameter information is reported.
Response	OK
Example	<pre>AT+CGMR=? OK</pre>

4.5. List available commands (AT+CLAC)

Execute Operation	AT+CLAC
Description	Lists every AT command the firmware accepts, one command per line, then OK. The list may contain commands reserved for production use that this document does not describe. The order is fixed by the firmware build and is not alphabetical.
Parameters	None

Response	<Command> ... OK
Example	<pre> AT+CLAC AT AT+AUDDEVLIST AT+AUDDEVSET ... ATE1 OK </pre>
Test Operation	AT+CLAC=?
Description	Confirms the command exists. No parameter information is reported.
Response	OK
Example	<pre> AT+CLAC=? OK </pre>

4.6. Reboot (AT+REBOOT)

Execute Operation	AT+REBOOT
Description	Resets the AW100. Nothing is sent in reply. Anything not saved with AT+CONFSAVE is lost. The USB connection drops and comes back when the AW100 has restarted. Note: Reconnect to the COM port if your terminal does not reconnect automatically.
Parameters	None
Response	None (System Reboot)
Example	<pre> AT+REBOOT </pre>
Test Operation	AT+REBOOT=?
Description	Confirms the command exists. No parameter information is reported.
Response	OK
Example	<pre> AT+REBOOT=? OK </pre>

5. Bluetooth Commands

5.1. Device name (AT+BTDEVNAME)

Set Operation	AT+BTDEVNAME=<name>
Description	Set the device name. It is the Bluetooth device name and the USB product string. The host shows the USB product string for the AW100's audio and serial devices. A Broadcast Source created with AT+PBSRCCREATE is given this name. The name is stored at once and is kept across reboots. USB shows the new name after the next boot. Setting the name that is already set answers OK and changes nothing. Note: If the name could not be stored the AW100 still answers OK. Read the name after a reboot to check it. Note: A Broadcast Source that already exists keeps its own name. The factory auto-start source is created with the factory default name at the first boot and at the boot after AT+CONFDEF. To give it the device name, delete it and create it again (AT+PBSRCDEL, AT+PBSRCCREATE), or set its name with AT+PBSRCCONF=<PBSourceID>,"NAME",<ParamValue>. Then save with AT+CONFSAVE. Note: The factory default is "Aurawave-" followed by the last four hex digits of the chip ID, for example "Aurawave-4F6D". These are also the last four digits of the USB serial number. AT+CONFDEF restores the factory default. Note: A name set to plain "Aurawave" becomes the factory default again at the next boot. Note: AT+PBSNKSTART matches the start of the source name. A sink started with AT+PBSNKSTART="Aurawave" joins any source whose name starts with Aurawave.
Parameters	<name> Quoted string of 0 to 31 printable ASCII characters. A name shorter than 4 characters is kept as the Bluetooth device name but is too short for a broadcast name. USB and new Broadcast Sources then show the factory default instead. This includes the empty name.
Response	OK
Example	<pre>AT+BTDEVNAME="Kitchen" OK</pre>
Read Operation	AT+BTDEVNAME?
Description	Read the device name.
Response	+BTDEVNAME: <name> OK
Example	<pre>AT+BTDEVNAME? +BTDEVNAME: "Aurawave-4F6D" OK</pre>
Test Operation	AT+BTDEVNAME=?
Description	Confirms the command exists and shows its parameters.
Response	+BTDEVNAME=<name> OK
Example	<pre>AT+BTDEVNAME=? +BTDEVNAME=<name> OK</pre>

Errors	• +CME ERROR: 22 (EINVAL); The name is a bare number instead of a quoted string. Other unquoted text is answered with ERROR. • +CME ERROR: 12 (ENOMEM); The name is longer than 31 characters.
--------	--

6. LE Audio Commands

6.1. Public Broadcast Source create (AT+PBSRCCREATE)

Note: The source present after AT+CONFDEF, and on an AW100 with no saved configuration, is the factory default source. It differs from a new source in two points. Its subgroup has no language, and it is started.

Execute Operation	AT+PBSRCCREATE
Description	Creates the Public Broadcast Source instance. The response carries its index. When the source already exists, the command answers OK with its index and keeps the existing configuration. A new source is not encrypted and uses preset 3. Its Broadcast ID is fixed for each AW100. It has one subgroup with language "eng" and audio source 0. The subgroup has two streams, with Audio Location 1 (Front Left) and 2 (Front Right). The Broadcast Name of a new source is the device name of the AW100 (AT+BTDEVNAME). The default device name is "Aurawave-" followed by the last four hexadecimal digits of the chip ID. A device name shorter than 4 characters is not used, and the source takes the default name instead. AT+PBSRCCONF=0, "NAME", ... sets another name before the source is started.
Parameters	None
Response	+PBSRCCREATE: <PBSourceID> OK
Example	<pre>AT+PBSRCCREATE +PBSRCCREATE: 0 OK</pre>
Test Operation	AT+PBSRCCREATE=?
Description	Confirms the command exists. No parameter information is reported.
Response	OK
Example	<pre>AT+PBSRCCREATE=? OK</pre>
Errors	+CME ERROR: 12 (ENOMEM): No free source is left. This firmware has one source and answers OK with index 0 when it exists.

6.2. Public Broadcast Source delete (AT+PBSRCDEL)

Execute Operation	AT+PBSRCDEL=<PBSourceID>
Description	Deletes the Public Broadcast Source instance. A broadcast that is starting or streaming is stopped first, and its advertising stops with it. The command answers OK also when no source exists at that index.
Parameters	<PBSourceID> Index of the Public Broadcast Source. This firmware has one source, index 0.
Response	OK
Example	<pre>AT+PBSRCDEL=0 OK</pre>

Test Operation	AT+PBSRCDEL=?
Description	Confirms the command exists and shows its parameters.
Response	OK
Example	<pre>AT+PBSRCDEL=? +PBSRCDEL: <(<0>)> OK</pre>
Errors	+CME ERROR: 37 (ECHNRG): The source index is not 0. This firmware has one source. +CME ERROR: 116 (ETIMEDOUT): The broadcast did not stop within about half a second. Read AT+PBSRCSTATE, then send the command again.

6.3. Public Broadcast Source configure (AT+PBSRCCONF)

Execute "Read" Operation	AT+PBSRCCONF=<PBSourceID>,<ParamTag>												
Description	Reads one parameter of the Public Broadcast Source.												
Parameters	<PBSourceID> Index of the Public Broadcast Source. This firmware has one source, index 0. <ParamTag> Tag of the parameter to read, in upper case as listed. <table> <tr> <th>ParamTag String</th><th>Description</th></tr> <tr> <td>"NAME"</td><td>Broadcast Name</td></tr> <tr> <td>"PRESET"</td><td>LC3 codec and QoS preset, 0 to 5</td></tr> <tr> <td>"CODE"</td><td>Broadcast Code. "" when the broadcast is not encrypted</td></tr> <tr> <td>"SUBGROUPS"</td><td>Number of subgroups</td></tr> <tr> <td>"ID"</td><td>Broadcast ID, six upper-case hexadecimal digits, not quoted</td></tr> </table>	ParamTag String	Description	"NAME"	Broadcast Name	"PRESET"	LC3 codec and QoS preset, 0 to 5	"CODE"	Broadcast Code. "" when the broadcast is not encrypted	"SUBGROUPS"	Number of subgroups	"ID"	Broadcast ID, six upper-case hexadecimal digits, not quoted
ParamTag String	Description												
"NAME"	Broadcast Name												
"PRESET"	LC3 codec and QoS preset, 0 to 5												
"CODE"	Broadcast Code. "" when the broadcast is not encrypted												
"SUBGROUPS"	Number of subgroups												
"ID"	Broadcast ID, six upper-case hexadecimal digits, not quoted												
Response	+PBSRCCONF: <PBSourceID>,<ParamTag>,<ParamValue> OK												
Example	<pre>AT+PBSRCCONF=0,"NAME" +PBSRCCONF: 0,"NAME","Aurawave-4F6D" OK</pre>												
Execute "Set" Operation	AT+PBSRCCONF=<PBSourceID>,<ParamTag>,<ParamValue>												

Description	Sets one parameter of the Public Broadcast Source. A source that has been started accepts no configuration write, even after it is stopped. Note: Every "SUBGROUPS" write deletes the subgroups and creates new ones, also when the count does not change. The new subgroups have no streams, no language, no name, and audio source 0. Set "STREAMS", "LOC" and the other subgroup parameters again before the start. Note: Presets 0 and 4 send each frame 3 times. Presets 1, 2, 3 and 5 send each frame 5 times. Preset 2 (48 kHz Low Latency) interleaves the two streams. All presets announce a presentation delay of 40 ms. 16 kHz and 24 kHz are Standard Quality. Every Auracast receiver decodes them, and 24 kHz is what phones and hearing aids use. 48 kHz is High Quality.																		
Parameters	<PBSourceID> Index of the Public Broadcast Source. This firmware has one source, index 0. <ParamTag> Tag of the parameter to set, in upper case as listed. <ParamValue> New value. Its type depends on <ParamTag>. Strings are in double quotes, numbers are not. <table><tr><th>ParamTag String</th><th>Description</th><th>Acceptable Input</th></tr><tr><td>"NAME"</td><td>Broadcast Name</td><td>ASCII string of 4 to 32 characters</td></tr><tr><td>"PRESET"</td><td>LC3 codec and QoS preset. A new source, and the source after AT+CONFDEF, uses 3.</td><td>0 – 16 kHz Low Latency (16_2_1) 1 – 16 kHz High Reliability (16_2_2) 2 – 48 kHz Low Latency (48_4_1, interleaved) 3 – 48 kHz High Reliability (48_4_2) 4 – 24 kHz Low Latency (24_2_1) 5 – 24 kHz High Reliability (24_2_2)</td></tr><tr><td>"CODE"</td><td>Broadcast Code. A source with a code is encrypted.</td><td>ASCII string of 4 to 16 characters. "" removes the code and turns encryption off.</td></tr><tr><td>"SUBGROUPS"</td><td>Number of subgroups</td><td>1 or 2</td></tr><tr><td>"ID"</td><td>Broadcast ID, 24 bits</td><td>Up to 6 hexadecimal digits, not quoted, no prefix (1A2B3C)</td></tr></table>	ParamTag String	Description	Acceptable Input	"NAME"	Broadcast Name	ASCII string of 4 to 32 characters	"PRESET"	LC3 codec and QoS preset. A new source, and the source after AT+CONFDEF, uses 3.	0 – 16 kHz Low Latency (16_2_1) 1 – 16 kHz High Reliability (16_2_2) 2 – 48 kHz Low Latency (48_4_1, interleaved) 3 – 48 kHz High Reliability (48_4_2) 4 – 24 kHz Low Latency (24_2_1) 5 – 24 kHz High Reliability (24_2_2)	"CODE"	Broadcast Code. A source with a code is encrypted.	ASCII string of 4 to 16 characters. "" removes the code and turns encryption off.	"SUBGROUPS"	Number of subgroups	1 or 2	"ID"	Broadcast ID, 24 bits	Up to 6 hexadecimal digits, not quoted, no prefix (1A2B3C)
ParamTag String	Description	Acceptable Input																	
"NAME"	Broadcast Name	ASCII string of 4 to 32 characters																	
"PRESET"	LC3 codec and QoS preset. A new source, and the source after AT+CONFDEF, uses 3.	0 – 16 kHz Low Latency (16_2_1) 1 – 16 kHz High Reliability (16_2_2) 2 – 48 kHz Low Latency (48_4_1, interleaved) 3 – 48 kHz High Reliability (48_4_2) 4 – 24 kHz Low Latency (24_2_1) 5 – 24 kHz High Reliability (24_2_2)																	
"CODE"	Broadcast Code. A source with a code is encrypted.	ASCII string of 4 to 16 characters. "" removes the code and turns encryption off.																	
"SUBGROUPS"	Number of subgroups	1 or 2																	
"ID"	Broadcast ID, 24 bits	Up to 6 hexadecimal digits, not quoted, no prefix (1A2B3C)																	
Response	OK																		
Example	AT+PBSRCCONF=0,"NAME","TV 1" OK AT+PBSRCCONF=0,"ID",1A2B3C OK																		
Test Operation	AT+PBSRCCONF=?																		
Description	Confirms the command exists and shows its parameters.																		
Response	+PBSRCCONF: <(0)>,<ParamTag>,<ParamValue>																		
Example	AT+PBSRCCONF=? +PBSRCCONF: <(0)>,<ParamTag>,<ParamValue> OK																		

Errors	+CME ERROR: 37 (EHRNG): The source index is not 0. This firmware has one source. +CME ERROR: 22 (EINVAL): No source exists at that index. Create one with AT+PBSRCCREATE first. The code is also given for an unknown or empty tag. It is given for a value that is not accepted: a name shorter than 4 characters, a preset above 5, a code of 1 to 3 characters, an ID above FFFFFFFF, or a subgroup count of 0. +CME ERROR: 12 (ENOMEM): The value is too long or too large. A name has at most 32 characters, a code at most 16, and a tag at most 9. The source has at most 2 subgroups. +CME ERROR: 16 (EBUSY): The source has been started. No configuration write is accepted after that, not even for a value that is already set. Delete the source and create it again.
--------	---

6.4. Public Broadcast subgroup configure (AT+PBGRPCONF)

Execute "Read" Operation	AT+PBGRPCONF=<PBSourceID>,<PBSubGroupID>,<ParamTag>										
Description	Reads one parameter of a subgroup of the Public Broadcast Source.										
Parameters	<PBSourceID> Index of the Public Broadcast Source. This firmware has one source, index 0. <PBSubGroupID> Index of the subgroup, starting at 0. <ParamTag> Tag of the parameter to read, in upper case as listed. <table border="1"> <thead> <tr> <th>ParamTag String</th><th>Description</th></tr> </thead> <tbody> <tr> <td>"LANG"</td><td>Language of the subgroup. +CME ERROR: 61 when not set. The source present after AT+CONFDEF has no language. A source created with AT+PBSRCCREATE has "eng".</td></tr> <tr> <td>"NAME"</td><td>Name of the subgroup (Program_Info metadata). +CME ERROR: 61 when not set</td></tr> <tr> <td>"STREAMS"</td><td>Number of streams in the subgroup</td></tr> <tr> <td>"AUD_SRC"</td><td>Audio source of the subgroup, 0 to 3</td></tr> </tbody> </table>	ParamTag String	Description	"LANG"	Language of the subgroup. +CME ERROR: 61 when not set. The source present after AT+CONFDEF has no language. A source created with AT+PBSRCCREATE has "eng".	"NAME"	Name of the subgroup (Program_Info metadata). +CME ERROR: 61 when not set	"STREAMS"	Number of streams in the subgroup	"AUD_SRC"	Audio source of the subgroup, 0 to 3
ParamTag String	Description										
"LANG"	Language of the subgroup. +CME ERROR: 61 when not set. The source present after AT+CONFDEF has no language. A source created with AT+PBSRCCREATE has "eng".										
"NAME"	Name of the subgroup (Program_Info metadata). +CME ERROR: 61 when not set										
"STREAMS"	Number of streams in the subgroup										
"AUD_SRC"	Audio source of the subgroup, 0 to 3										
Response	+PBGRPCONF: <PBSourceID>,<PBSubGroupID>,<ParamTag>,<ParamValue> OK										
Example	<pre>AT+PBGRPCONF=0,0,"LANG" +PBGRPCONF: 0,0,"LANG","eng" OK</pre>										
Execute "Set" Operation	AT+PBGRPCONF=<PBSourceID>,<PBSubGroupID>,<ParamTag>,<ParamValue>										
Description	Sets one parameter of a subgroup of the Public Broadcast Source. A source that has been started accepts no configuration write, even after it is stopped.										

Parameters	<PBSourceID> Index of the Public Broadcast Source. This firmware has one source, index 0. <PBSubGroupID> Index of the subgroup, starting at 0. <ParamTag> Tag of the parameter to set, in upper case as listed. <ParamValue> New value. Its type depends on <ParamTag>. Strings are in double quotes, numbers are not. <table><tr><th>ParamTag String</th><th>Description</th><th>Acceptable Input</th></tr><tr><td>"LANG"</td><td>Language of the subgroup</td><td>Three lower-case letters (ISO 639-3), for example "eng". "" clears the language.</td></tr><tr><td>"NAME"</td><td>Name of the subgroup. It is sent as the Program_Info metadata of the subgroup. A receiver can use it to tell subgroups of the same language apart.</td><td>String of up to 32 characters. "" clears the name.</td></tr><tr><td>"STREAMS"</td><td>Number of streams in the subgroup</td><td>1 or 2. The source has at most 2 streams in total across all subgroups. Every write creates the streams of the subgroup again with Audio Location 0. Set "LOC" afterwards.</td></tr><tr><td>"AUD_SRC"</td><td>Audio source of the subgroup. AT+PBSRCSTART answers +CME ERROR: 22 when the value does not fit the number of streams.</td><td>0 – Auto (default). One stream carries the left channel, or the right channel when its Audio Location is a right-side location. Two streams carry stereo. 1 – Left channel (one stream) 2 – Right channel (one stream) 3 – Stereo (two streams)</td></tr></table>	ParamTag String	Description	Acceptable Input	"LANG"	Language of the subgroup	Three lower-case letters (ISO 639-3), for example "eng". "" clears the language.	"NAME"	Name of the subgroup. It is sent as the Program_Info metadata of the subgroup. A receiver can use it to tell subgroups of the same language apart.	String of up to 32 characters. "" clears the name.	"STREAMS"	Number of streams in the subgroup	1 or 2. The source has at most 2 streams in total across all subgroups. Every write creates the streams of the subgroup again with Audio Location 0. Set "LOC" afterwards.	"AUD_SRC"	Audio source of the subgroup. AT+PBSRCSTART answers +CME ERROR: 22 when the value does not fit the number of streams.	0 – Auto (default). One stream carries the left channel, or the right channel when its Audio Location is a right-side location. Two streams carry stereo. 1 – Left channel (one stream) 2 – Right channel (one stream) 3 – Stereo (two streams)
ParamTag String	Description	Acceptable Input														
"LANG"	Language of the subgroup	Three lower-case letters (ISO 639-3), for example "eng". "" clears the language.														
"NAME"	Name of the subgroup. It is sent as the Program_Info metadata of the subgroup. A receiver can use it to tell subgroups of the same language apart.	String of up to 32 characters. "" clears the name.														
"STREAMS"	Number of streams in the subgroup	1 or 2. The source has at most 2 streams in total across all subgroups. Every write creates the streams of the subgroup again with Audio Location 0. Set "LOC" afterwards.														
"AUD_SRC"	Audio source of the subgroup. AT+PBSRCSTART answers +CME ERROR: 22 when the value does not fit the number of streams.	0 – Auto (default). One stream carries the left channel, or the right channel when its Audio Location is a right-side location. Two streams carry stereo. 1 – Left channel (one stream) 2 – Right channel (one stream) 3 – Stereo (two streams)														
Response	OK															
Example	<pre>AT+PBGRPCONF=0,0,"LANG","eng" OK</pre>															
Test Operation	AT+PBGRPCONF=?															
Description	Confirms the command exists and shows its parameters. Note: The second field reads <(0)> although subgroups 0 and 1 exist.															
Response	+PBGRPCONF: <(0)>,<(0)>,<ParamTag>,<ParamValue>															
Example	<pre>AT+PBGRPCONF=? +PBGRPCONF: <(0)>,<(0)>,<ParamTag>,<ParamValue> OK</pre>															

Errors	• +CME ERROR: 37 (ECHRNG): The source index is not 0, or the subgroup index is beyond the subgroup count. An invalid "STREAMS" or "AUD_SRC" value is reported first, as +CME ERROR: 22. • +CME ERROR: 22 (EINVAL): No source exists at that index. Create one with AT+PBSRCCREATE first. The code is also given for an unknown or empty tag, a stream count of 0, a language that is not three lower-case letters, and an audio source that is not 0 to 3. • +CME ERROR: 12 (ENOMEM): The value is too long or too large. A name has at most 32 characters, a language 3, and a tag 7. The source has at most 2 streams in total across all subgroups. • +CME ERROR: 61 (ENODATA): The language or name has not been set on that subgroup. • +CME ERROR: 16 (EBUSY): The source has been started. No configuration write is accepted after that, not even for a value that is already set. Delete the source and create it again.
--------	---

6.5. Public Broadcast stream configure (AT+PBSTRMCONF)

Execute "Read" Operation	AT+PBSTRMCONF=<PBSourceID>,<PBSubGroupID>,<PBStreamID>,<ParamTag>				
Description	Reads one parameter of a stream of the Public Broadcast Source.				
Parameters	<PBSourceID> Index of the Public Broadcast Source. This firmware has one source, index 0. <PBSubGroupID> Index of the subgroup, starting at 0. <PBStreamID> Index of the stream within the subgroup, starting at 0. <ParamTag> Tag of the parameter to read, in upper case as listed. <table> <tr> <th>ParamTag String</th><th>Description</th></tr> <tr> <td>"LOC"</td><td>Audio Location of the stream, as a decimal number</td></tr> </table>	ParamTag String	Description	"LOC"	Audio Location of the stream, as a decimal number
ParamTag String	Description				
"LOC"	Audio Location of the stream, as a decimal number				
Response	+PBSTRMCONF: <PBSourceID>,<PBSubGroupID>,<PBStreamID>,<ParamTag>,<ParamValue> OK				
Example	<pre>AT+PBSTRMCONF=0,0,0,"LOC" +PBSTRMCONF: 0,0,0,"LOC",1 OK</pre>				
Execute "Set" Operation	AT+PBSTRMCONF=<PBSourceID>,<PBSubGroupID>,<PBStreamID>,<ParamTag>,<ParamValue>				
Description	Sets one parameter of a stream of the Public Broadcast Source. A source that has been started accepts no configuration write, even after it is stopped.				

Parameters	<PBSourceID> Index of the Public Broadcast Source. This firmware has one source, index 0. <PBSubGroupID> Index of the subgroup, starting at 0. <PBStreamID> Index of the stream within the subgroup, starting at 0. <ParamTag> Tag of the parameter to set, in upper case as listed. <ParamValue> New value, a decimal number. <table><thead><tr><th>ParamTag String</th><th>Description</th><th>Acceptable Input</th></tr></thead><tbody><tr><td>"LOC"</td><td>Audio Location of the stream</td><td>0 for mono, or one Audio Location bit from Bluetooth Assigned Numbers 6.12.1, as a decimal number: 1 Front Left, 2 Front Right, 4 Front Center. Two streams of one subgroup must have different locations, unless both are 0.</td></tr></tbody></table>	ParamTag String	Description	Acceptable Input	"LOC"	Audio Location of the stream	0 for mono, or one Audio Location bit from Bluetooth Assigned Numbers 6.12.1, as a decimal number: 1 Front Left, 2 Front Right, 4 Front Center. Two streams of one subgroup must have different locations, unless both are 0.
ParamTag String	Description	Acceptable Input					
"LOC"	Audio Location of the stream	0 for mono, or one Audio Location bit from Bluetooth Assigned Numbers 6.12.1, as a decimal number: 1 Front Left, 2 Front Right, 4 Front Center. Two streams of one subgroup must have different locations, unless both are 0.					
Response	OK						
Example	<pre>AT+PBSTRMCONF=0,0,0,"LOC",1 OK</pre>						
Test Operation	AT+PBSTRMCONF=?						
Description	Confirms the command exists and shows its parameters.						
Response	+PBSTRMCONF: <(0-0)>,<(0-1)>,<(0-1)>,<ParamTag>,<ParamValue>						
Example	<pre>AT+PBSTRMCONF=? +PBSTRMCONF: <(0-0)>,<(0-1)>,<(0-1)>,<ParamTag>,<ParamValue> OK</pre>						
Errors	• +CME ERROR: 37 (ECHRNG): The source index is not 0, or the subgroup index or stream index is beyond its count. • +CME ERROR: 22 (EINVAL): No source exists at that index. Create one with AT+PBSRCCREATE first. The code is also given for a tag other than "LOC", and for a location that is neither 0 nor a single Audio Location bit. • +CME ERROR: 12 (ENOMEM): The tag is longer than 3 characters. • +CME ERROR: 16 (EBUSY): The source has been started. No configuration write is accepted after that, not even for a value that is already set. Delete the source and create it again.						

6.6. Public Broadcast Source start (AT+PBSRCSTART)

Execute Operation	AT+PBSRCSTART=<PBSourceID>
Description	Starts the LE Audio broadcast of the Public Broadcast Source. The command answers OK also when the source is already STARTING or STREAMING. A STOPPED source is restarted with its configuration and Broadcast ID unchanged. The command does not restart a source whose stop has not completed. This is the case after AT+PBSRCSTOP answered +CME ERROR: 116. The source still reports STREAMING, and the command answers OK without effect. Read AT+PBSRCSTATE until it reports STOPPED, then send AT+PBSRCSTART again.
Parameters	<PBSourceID> Index of the Public Broadcast Source. This firmware has one source, index 0.
Response	OK

Example	<pre>AT+PBSRCSTART=0 OK</pre>
Test Operation	AT+PBSRCSTART=?
Description	Confirms the command exists and shows its parameters.
Response	<pre>+PBSRCSTART: <(0)> OK</pre>
Example	<pre>AT+PBSRCSTART=? +PBSRCSTART: <(0)> OK</pre>
Errors	• +CME ERROR: 37 (ECHRNG): The source index is not 0. This firmware has one source. • +CME ERROR: 22 (EINVAL): No source exists at that index. Create one with AT+PBSRCCREATE first. The code is also given for an incomplete configuration: a subgroup without streams, an audio source that does not fit the number of streams, or two streams of one subgroup with the same Audio Location other than 0. • +CME ERROR: 12 (ENOMEM): No radio resources are left for the broadcast. Delete the source and try again.

6.7. Public Broadcast Source state (AT+PBSRCSTATE)

Execute Operation	AT+PBSRCSTATE=<PBSourceID>
Description	Reads the streaming state of the Public Broadcast Source. The response carries the state as a number and as a name. LED Indications describes the LED. • 0 IDLE: The source is created and not started. LED off • 1 STARTING: AT+PBSRCSTART is accepted and the broadcast is starting. LED blue, blinking • 2 STREAMING: The broadcast is on the air. This state is also reported for the short time a stop takes to complete. LED blue, steady • 3 STOPPED: The source was stopped with AT+PBSRCSTOP and is off the air. AT+PBSRCSTART restarts it and AT+PBSRCDEL deletes it. It cannot be configured again. LED off
Parameters	<PBSourceID> Index of the Public Broadcast Source. This firmware has one source, index 0.
Response	<pre>+PBSRCSTATE: <PBSourceID>,<StateCode>,"<StateName>" OK</pre>
Example	<pre>AT+PBSRCSTATE=0 +PBSRCSTATE: 0,2,"STREAMING" OK</pre>
Test Operation	AT+PBSRCSTATE=?
Description	Confirms the command exists and shows its parameters.
Response	<pre>+PBSRCSTATE: <(0)> OK</pre>
Example	<pre>AT+PBSRCSTATE=? +PBSRCSTATE: <(0)> OK</pre>

Errors	• +CME ERROR: 37 (ECHRNG): The source index is not 0. This firmware has one source. • +CME ERROR: 22 (EINVAL): No source exists at that index. Create one with AT+PBSRCCREATE first.
--------	---

6.8. Public Broadcast Source stop (AT+PBSRCSTOP)

Execute Operation	AT+PBSRCSTOP=<PBSourceID>
Description	Stops the BIG of the Public Broadcast Source. Its extended and periodic advertising stops too. Scanners no longer find the broadcast, and a sink synchronised to it searches for it again. The command answers OK also when the source was never started or is already STOPPED. A source that is STARTING is stopped as soon as its streams are up. AT+PBSRCSTART restarts a STOPPED source.
Parameters	<PBSourceID> Index of the Public Broadcast Source. This firmware has one source, index 0.
Response	OK
Example	<pre>AT+PBSRCSTOP=0 OK</pre>
Test Operation	AT+PBSRCSTOP=?
Description	Confirms the command exists and shows its parameters.
Response	+PBSRCSTOP: <(0)> OK
Example	<pre>AT+PBSRCSTOP=? +PBSRCSTOP: <(0)> OK</pre>
Errors	• +CME ERROR: 37 (ECHRNG): The source index is not 0. This firmware has one source. • +CME ERROR: 22 (EINVAL): No source exists at that index. Create one with AT+PBSRCCREATE first. • +CME ERROR: 116 (ETIMEDOUT): The broadcast did not stop within about half a second. Read AT+PBSRCSTATE, then send the command again.

6.9. Public Broadcast Sink scan (AT+PBSNKSCAN)

Execute Operation	AT+PBSNKSCAN=<Timeout>
Description	Scan for Broadcast Sources in range for <Timeout> seconds. Every announcement received is printed as it arrives. A source normally appears several times. OK is answered when the time is up. Note: <SourceName> is the name of the source. <SourceID> is its Broadcast ID, in hexadecimal. <LEAddress> is its Bluetooth LE address. <LEAddressType> is 0 for a public address and 1 for a random address. <AdvSetID> is its advertising set ID, 0 to 15. Give these values unchanged to AT+PBSNKEXPLORE or AT+PBSNKSTART. Note: <SourceEncrypted> is the encryption bit of the Public Broadcast Announcement of the source. It is 1 when a Broadcast Code is required. A source that sends no Public Broadcast Announcement is listed with 0. Its name is empty unless it advertises a Broadcast Name.
Parameters	<Timeout> Scan time in seconds, 1 to 4294967.

Response	+PBSNKSCAN: <SourceName>,<SourceID>,<SourceEncrypted>,<LEAddress>,<LEAddressType>,<AdvSetID> OK
Example	<pre> AT+PBSNKSCAN=5 +PBSNKSCAN: "Aurawave-4F6D",9E26A2,0,"CE:18:12:13:57:37",1,0 +PBSNKSCAN: "Aurawave-4F6D",9E26A2,0,"CE:18:12:13:57:37",1,0 OK </pre>
Test Operation	AT+PBSNKSCAN=?
Description	Confirms the command exists and shows its parameters.
Response	+PBSNKSCAN: <Timeout> OK
Example	<pre> AT+PBSNKSCAN=? +PBSNKSCAN: <Timeout> OK </pre>
Errors	+CME ERROR: 22 (EINVAL): The duration is 0 or above 4294967. Scan for 1 second or more. +CME ERROR: 16 (EBUSY): The sink is searching for its source and its own scan is running at this moment. Stop the sink, or try again.

6.10. Public Broadcast Sink explore (AT+PBSNKEXPLORE)

Execute Operation	AT+PBSNKEXPLORE=<SourceID>[,<LEAddress>,<LEAddressType>,<AdvSetID>]
Description	Read the parameters of a Broadcast Source: its presentation delay, subgroups, codec, language, subgroup name and streams. The command takes 1 or 4 parameters. With the Broadcast ID alone the AW100 scans for the source for up to 10 seconds. The scan ends as soon as the source is heard. With the address, address type and advertising set ID from AT+PBSNKSCAN the AW100 does not scan. The lines are printed from the first periodic advertising report received, within 5 seconds. A source whose report does not describe its streams answers OK with no lines. Note: <PresentationDelay> is in microseconds. <CodecID>, <CompanyID> and <VendorCodecID> are hexadecimal. 06,0000,0000 is LC3. <Location> is the Audio Location of the stream in hexadecimal. Convert it to decimal for AT+PBSNKCONF. Note: The "LANG" and "NAME" lines are printed only when the subgroup advertises them. The "LOC" line is printed only when the stream itself advertises an Audio Location. A mono stream does not. A stream whose Audio Location is given for the whole subgroup does not either. <SubGroup> and <Stream> count from 0 in the order the source lists them.
Parameters	<SourceID> Broadcast ID, unquoted, hexadecimal. <LEAddress> Bluetooth LE address of the source, in quotes, in the form XX:XX:XX:XX:XX:XX. <LEAddressType> Address type of the source, 0 to 3, as AT+PBSNKSCAN reports it. <AdvSetID> Advertising set ID of the source, 0 to 15.

Response	+PBSNKEXPLORE: "PD",<PresentationDelay> +PBSNKEXPLORE: "SUBGROUPS",<Count> +PBSNKEXPLORE: <SubGroup>,"CODEC",<CodecID>,<CompanyID>,<VendorCodecID> +PBSNKEXPLORE: <SubGroup>,"LANG",<Language> +PBSNKEXPLORE: <SubGroup>,"NAME",<Name> +PBSNKEXPLORE: <SubGroup>,"STREAMS",<Count> +PBSNKEXPLORE: <SubGroup>,<Stream>,"LOC",<Location> OK
Example	<pre> AT+PBSNKEXPLORE=9E26A2,"CE:18:12:13:57:37",1,0 +PBSNKEXPLORE: "PD",40000 +PBSNKEXPLORE: "SUBGROUPS",1 +PBSNKEXPLORE: 0,"CODEC",06,0000,0000 +PBSNKEXPLORE: 0,"LANG","eng" +PBSNKEXPLORE: 0,"NAME","Stage Left" +PBSNKEXPLORE: 0,"STREAMS",2 +PBSNKEXPLORE: 0,0,"LOC",1 +PBSNKEXPLORE: 0,1,"LOC",2 OK </pre>
Test Operation	AT+PBSNKEXPLORE=?
Description	Confirms the command exists and shows its parameters.
Response	+PBSNKEXPLORE: <SourceID>[,<LEAddress>,<LEAddressType>,<AdvSetID>] OK
Example	<pre> AT+PBSNKEXPLORE=? +PBSNKEXPLORE: <SourceID>[,<LEAddress>,<LEAddressType>,<AdvSetID>] OK </pre>
Errors	+CME ERROR: 22 (EINVAL): The address was given without its type and advertising set ID, or a parameter is not in the accepted form. Give 1 or 4 parameters. +CME ERROR: 19 (ENODEV): No source with this Broadcast ID was heard during the 10 second scan. Check the ID and that the source is broadcasting, or give its address. +CME ERROR: 16 (EBUSY): The sink has found its source, or one of its search attempts is running at this moment. Stop the sink first. +CME ERROR: 116 (ETIMEDOUT): Nothing was received from the periodic advertising of the source within 5 seconds. The source stopped, or the address, type or advertising set ID is wrong. +CME ERROR: 12 (ENOMEM): The address is longer than 17 characters, or the previous synchronisation is still being released. Wait a moment and try again. +CME ERROR: 120 (EALREADY): The source stopped advertising while it was being read. Try again.

6.11. Public Broadcast Sink configure (AT+PBSNKKCONF)

Execute "Read" Operation	AT+PBSNKKCONF=<ParamTag>
Description	Read the value of a Public Broadcast Sink parameter.

Parameters	<ParamTag> Tag of the parameter to read. <table><tr><th>ParamTag String</th><th>Description</th></tr><tr><td>"LOC"</td><td>Audio Location of the stream the sink synchronises to, in decimal</td></tr><tr><td>"LANG"</td><td>Language of the subgroup the sink synchronises to. Empty when no language is set</td></tr></table>	ParamTag String	Description	"LOC"	Audio Location of the stream the sink synchronises to, in decimal	"LANG"	Language of the subgroup the sink synchronises to. Empty when no language is set			
ParamTag String	Description									
"LOC"	Audio Location of the stream the sink synchronises to, in decimal									
"LANG"	Language of the subgroup the sink synchronises to. Empty when no language is set									
Response	+PBSNKCONF: <ParamTag>,<ParamValue> OK									
Example	AT+PBSNKCONF="LOC" +PBSNKCONF: "LOC", 1 OK AT+PBSNKCONF="LANG" +PBSNKCONF: "LANG", "" OK									
Execute "Set" Operation	AT+PBSNKCONF=<ParamTag>,<ParamValue>									
Description	Set a Public Broadcast Sink parameter. AT+CONFSAVE saves the parameters. A sink that is not started uses the new value at the next AT+PBSNKSTART. A sink that is still SEARCHING uses it at its next attempt. Once the sink has found its source (state STARTING, STREAMING, STOPPED or NOMATCH) every write is refused, even for a value that is already set. The sink synchronises to exactly one stream and plays it on both outputs. It chooses among the streams it can decode (see AT+PBSNKSTART). The candidates are the streams whose Audio Location includes "LOC". If there are none, the candidates are the mono streams, which advertise no Audio Location. With "LOC" set to 0 every decodable stream is a candidate. Among the candidates a 48 kHz stream is preferred over a lower rate. Otherwise the first stream offered is used. When no stream satisfies the location and the language, AT+PBSNKSTATE reports NOMATCH and nothing is played.									
Parameters	<ParamTag> Tag of the parameter to set. <ParamValue> New value of the parameter. Its type depends on <ParamTag>. <table><tr><th>ParamTag String</th><th>Description</th><th>Acceptable Input</th></tr><tr><td>"LOC"</td><td>Audio Location of the stream to synchronise to, in decimal. Any single Audio Location bit is accepted, for example 4 (Front Center) or 16 (Back Left). A combination such as 3 is refused</td><td>0 – Any 1 – Front left (Default) 2 – Front right</td></tr><tr><td>"LANG"</td><td>Language of the subgroup to synchronise to. Subgroups that advertise a different language are skipped. Subgroups that advertise no language are used as if no language were set</td><td>ISO 639-3 code, three lowercase letters, in quotes "" – Any language (Default)</td></tr></table>	ParamTag String	Description	Acceptable Input	"LOC"	Audio Location of the stream to synchronise to, in decimal. Any single Audio Location bit is accepted, for example 4 (Front Center) or 16 (Back Left). A combination such as 3 is refused	0 – Any 1 – Front left (Default) 2 – Front right	"LANG"	Language of the subgroup to synchronise to. Subgroups that advertise a different language are skipped. Subgroups that advertise no language are used as if no language were set	ISO 639-3 code, three lowercase letters, in quotes "" – Any language (Default)
ParamTag String	Description	Acceptable Input								
"LOC"	Audio Location of the stream to synchronise to, in decimal. Any single Audio Location bit is accepted, for example 4 (Front Center) or 16 (Back Left). A combination such as 3 is refused	0 – Any 1 – Front left (Default) 2 – Front right								
"LANG"	Language of the subgroup to synchronise to. Subgroups that advertise a different language are skipped. Subgroups that advertise no language are used as if no language were set	ISO 639-3 code, three lowercase letters, in quotes "" – Any language (Default)								
Response	OK									

Example	AT+PBSNKCONF="LOC",2 OK
	AT+PBSNKCONF="LANG","fra" OK
Test Operation	AT+PBSNKCONF=?
Description	Confirms the command exists and shows its parameters.
Response	+PBSNKCONF: <ParamTag>,<ParamValue> OK
Example	AT+PBSNKCONF=? +PBSNKCONF: <ParamTag>,<ParamValue> OK
Errors	• +CME ERROR: 22 (EINVAL): The tag is not "LOC" or "LANG", the location is not 0 or a single Audio Location bit, or the language is not three lowercase letters. Tags are upper case and in quotes. • +CME ERROR: 12 (ENOMEM): The tag is longer than 4 characters, or the language is longer than 3. • +CME ERROR: 13 (EACCES): The sink has found its source. Stop it with AT+PBSNKSTOP before changing its configuration.

6.12. Public Broadcast Sink start (AT+PBSNKSTART)

Note: The command has two formats. The first parameter tells them apart: a quoted string is a source name, an unquoted hexadecimal number is a Broadcast ID. A source is chosen by its name or by its ID, never by both. Parameters cannot be left empty. To give a code with a name, give <ExactOnly> too.

Note: OK means the sink is started. The command makes a first attempt to find the source and answers OK when that attempt ends. This takes about 1 second when the source is not heard. When the source is heard the answer normally comes within a second, and in the worst case after about 7 seconds. After OK the state is STARTING when the first attempt found the source, otherwise SEARCHING. The state then becomes STREAMING or NOMATCH (AT+PBSNKSTATE).

Note: A sink that is SEARCHING repeats a 1 second scan and a 1 second pause until the source is found or AT+PBSNKSTOP is sent. It accepts a new AT+PBSNKSTART, which replaces its target. The sink has found its source when the state is STARTING, STREAMING, STOPPED or NOMATCH.

Note: The sink decodes LC3 streams with 10 ms frames at 16, 24 or 48 kHz. These are the Standard Quality configurations 16_2 and 24_2 and the High Quality configurations 48_2, 48_4 and 48_6. Streams with 7.5 ms frames are skipped. A source with no stream the sink can decode is reported as NOMATCH.

Note: The AW100's own Broadcast Source is deleted when the sink starts. This also happens at boot when a saved started sink resumes. AT+CONFSAVE saves the target. Example: see Example 2: An AW100 Receiving Another AW100.

Start by name	AT+PBSNKSTART="<SourceName>"[,<ExactOnly>[,<Code>]]
Description	Start the sink on the Broadcast Source with this name. By default the name is a prefix. The sink joins the first source it finds whose name starts with <SourceName>. For example "Aurawave" joins any source with a factory name, such as "Aurawave-4F6D". With <ExactOnly> set to 1 the sink joins only a source whose whole name is <SourceName>.

Parameters	<SourceName> Name of the Broadcast Source, in quotes, 4 to 128 characters, case-sensitive. <ExactOnly> How the name matches. <Code> Broadcast Code, in quotes, 4 to 16 characters. Required for an encrypted source. Give "" or leave it out for a source that is not encrypted. <table border="1"> <thead> <tr> <th>ParamTag String</th><th>Description</th></tr> </thead> <tbody> <tr> <td>0</td><td>The name is a prefix: any source whose name starts with <SourceName> (Default)</td></tr> <tr> <td>1</td><td>Only a source whose whole name is <SourceName></td></tr> </tbody> </table>	ParamTag String	Description	0	The name is a prefix: any source whose name starts with <SourceName> (Default)	1	Only a source whose whole name is <SourceName>
ParamTag String	Description						
0	The name is a prefix: any source whose name starts with <SourceName> (Default)						
1	Only a source whose whole name is <SourceName>						
Response	OK						
Example	<pre>AT+PBSNKSTART="Aurawave" OK</pre> <pre>AT+PBSNKSTART="Aurawave-4F6D",1 OK</pre> <pre>AT+PBSNKSTART="Aurawave",0,"secret12" OK</pre>						
Start by ID	AT+PBSNKSTART=<SourceID>[,<Code>]						
Description	Start the sink on the Broadcast Source with this Broadcast ID, as AT+PBSNKSCAN reports it. An ID always matches exactly.						
Parameters	<SourceID> Broadcast ID, unquoted, hexadecimal, 000000 to FFFFFFFF. Upper or lower case. Leading zeros are optional. <Code> Broadcast Code, in quotes, 4 to 16 characters. Required for an encrypted source. Give "" or leave it out for a source that is not encrypted.						
Response	OK						
Example	<pre>AT+PBSNKSTART=9E26A2 OK</pre> <pre>AT+PBSNKSTART=9E26A2,"secret12" OK</pre>						
Test Operation	AT+PBSNKSTART=?						
Description	Confirms the command exists and shows its parameters.						
Response	<pre>+PBSNKSTART: "<SourceName>"[,<ExactOnly>[,<Code>]] +PBSNKSTART: <SourceID>[,<Code>] OK</pre>						
Example	<pre>AT+PBSNKSTART=? +PBSNKSTART: "<SourceName>"[,<ExactOnly>[,<Code>]] +PBSNKSTART: <SourceID>[,<Code>] OK</pre>						

Errors	• +CME ERROR: 22 (EINVAL): The name or the code is shorter than 4 characters, the code is not in quotes, <ExactOnly> is not 0 or 1, the Broadcast ID is above FFFFFFFF, or more than a code follows the ID. • +CME ERROR: 12 (ENOMEM): The name is longer than 128 characters, the code is longer than 16, or the Broadcast ID has more than 10 digits. • +CME ERROR: 13 (EACCES): The sink has found its source. Stop it with AT+PBSNKSTOP before giving it another target. • +CME ERROR: 120 (EALREADY): The sink found its previous target while the command was waiting. It keeps that source. Stop it with AT+PBSNKSTOP and send the command again.
--------	---

6.13. Public Broadcast Sink stop (AT+PBSNKSTOP)

Execute Operation	AT+PBSNKSTOP
Description	Stop the sink. The sink releases its source and ends the search, also the search that follows a lost source. The state becomes IDLE. An AT+CONFSAVE sent afterwards saves the sink as stopped. OK is answered whether or not the sink was started. The answer normally comes at once. While a synchronisation attempt is in progress it can take up to about 6 seconds. Error codes, if any, come from the Bluetooth host and are not listed here.
Parameters	None
Response	OK
Example	<pre>AT+PBSNKSTOP OK</pre>
Test Operation	AT+PBSNKSTOP=?
Description	Confirms the command exists. No parameter information is reported.
Response	OK
Example	<pre>AT+PBSNKSTOP=? OK</pre>

6.14. Public Broadcast Sink state (AT+PBSNKSTATE)

Execute Operation	AT+PBSNKSTATE
--------------------------	---------------

Description	Read the state of the Public Broadcast Sink and the Broadcast Source it found. <Index> is always 0. The LED shows the state (see LED Indications). The state codes 0 to 3 are also used by AT+PBSRCSTATE. The codes 4 and 5 exist only for the sink. • 0 IDLE: The sink is not started, or it was stopped with AT+PBSNKSTOP. LED off • 1 STARTING: The first attempt of AT+PBSNKSTART found the source and the sink is synchronising to it. A source found by a later attempt goes from SEARCHING directly to STREAMING or NOMATCH. LED green, blinking • 2 STREAMING: Audio is being received. LED green, steady • 3 STOPPED: The streams of the source ended while its periodic advertising is still received. Audio resumes without a command when the streams return. This state is also seen for a few seconds when the source goes out of range, before SEARCHING. LED green, blinking • 4 NOMATCH: The source was found but none of its streams can be used. No stream satisfies the location and language of the sink (AT+PBSNKCONF), or the sink cannot decode any stream. Nothing is played. LED green and blue, alternating • 5 SEARCHING: The sink is scanning for its source. This follows an AT+PBSNKSTART that did not find the source, or the loss of the source. LED green, blinking The source fields identify the source the sink found. They are valid in STARTING, STREAMING, STOPPED and NOMATCH. They are kept after AT+PBSNKSTOP. While the state is SEARCHING they are placeholders: FFFFFFFF,"FF:FF:FF:FF:FF:FF",0,255. Before the first start they are 000000,"00:00:00:00:00:00",0,0.
Parameters	None
Response	+PBSNKSTATE: <Index>,<State>,"<StateStr>",<SourceID>",<LEAddress>",<LEAddressType>,<AdvSetID> OK
Example	<pre>AT+PBSNKSTATE +PBSNKSTATE: 0,2,"STREAMING",9E26A2,"CE:18:12:13:57:37",1,0 OK</pre>
Test Operation	AT+PBSNKSTATE=?
Description	Confirms the command exists and shows its parameters.
Response	+PBSNKSTATE: <Index>,<State>,<StateStr>,<SourceID>,<LEAddress>,<LEAddressType>,<AdvSetID> OK
Example	<pre>AT+PBSNKSTATE=? +PBSNKSTATE: <Index>,<State>,<StateStr>,<SourceID>,<LEAddress>,<LEAddressType>,<AdvSetID> OK</pre>

7. Audio System Commands

7.1. Audio devices list (AT+AUDDEVLIST)

Execute Operation	AT+AUDDEVLIST
Description	List the audio devices present on this AW100, with the index AT+AUDDEVSET uses. Note: The I2S_AUX device is the I2S port on the 40-pin header: stereo, 48 kHz, 32-bit samples, with the AW100 as clock master. Note: The I2S_MAIN device is the on-board codec with the analog jacks.
Parameters	None
Response	+AUDDEVLIST: <ListLength> 0,<Device0> 1,<Device1> ... OK
Example	<pre>AT+AUDDEVLIST +AUDDEVLIST: 3 0,USB 1,I2S_MAIN 2,I2S_AUX OK</pre>
Test Operation	AT+AUDDEVLIST=?
Description	Confirms the command exists. No parameter information is reported.
Response	OK
Example	<pre>AT+AUDDEVLIST=? OK</pre>

7.2. Audio device set (AT+AUDDEVSET)

Execute Operation	AT+AUDDEVSET=<AudioDeviceId>
Description	Select the active audio device. One setting serves both roles. It is the audio input of Public Broadcast Source instances and the audio output of the Public Broadcast Sink. A running audio path is stopped and restarted on the new device. Selecting the device that is already active answers OK and changes nothing. The selection is kept across reboots only after AT+CONFSAVE. AT+AUDDEVLIST reports the devices and their indexes.
Parameters	<AudioDeviceId> Index of the device in the list reported by AT+AUDDEVLIST. The list depends on the hardware detected at boot. The same device may have a different index on different AW100 variants. Read the list instead of assuming an index.
Response	OK
Example	<pre>AT+AUDDEVSET=0 OK</pre>
Read Operation	AT+AUDDEVSET?

Description	Read the index of the active audio device.
Response	+AUDDEVSET: <AudioDeviceId> OK
Example	<pre>AT+AUDDEVSET? +AUDDEVSET: 0 OK</pre>
Test Operation	AT+AUDDEVSET=?
Description	Confirms the command exists and shows its parameters. Note: <N> is one less than the number of devices reported by AT+AUDDEVLIST. With a single device the response is +AUDDEVSET: <(0)>.
Response	+AUDDEVSET: <(0-<N>)> OK
Example	<pre>AT+AUDDEVSET=? +AUDDEVSET: <(0-2)> OK</pre>
Errors	• +CME ERROR: 37 (ECHRNG): The index is beyond the list given by AT+AUDDEVLIST. • +CME ERROR: 22 (EINVAL): The index is not a decimal number.

8. System Configuration Commands

8.1. Configuration save (AT+CONFSAVE)

Execute Operation	AT+CONFSAVE
Description	Save the current system configuration. It includes: • Public Broadcast Source instances and their configuration. That no instance exists is saved too. • Broadcast state. A Broadcast Source that is streaming when you save starts streaming again at boot. • Active audio device • Public Broadcast Sink configuration: the source name and match mode, the Broadcast ID and the Broadcast Code, the stream location and language (AT+PBSNKCONF), and whether the sink is started. A saved started sink searches for its source again after a reboot. The device name and the USB Vendor and Product IDs are not part of this set. AT+BTDEVNAME and AT+USBID store them at once. After three faults in quick succession the AW100 discards the saved configuration at the next boot. It then starts with the factory defaults, as after AT+CONFDEF.
Parameters	None
Response	OK
Example	<pre>AT+CONFSAVE OK</pre>
Test Operation	AT+CONFSAVE=?
Description	Confirms the command exists. The command has no parameters.
Response	+CONFSAVE OK
Example	<pre>AT+CONFSAVE=? +CONFSAVE OK</pre>
Errors	• +CME ERROR: 28 (ENOSPC): The settings storage is full. Restore the defaults with AT+CONFDEF and save again. • +CME ERROR: 12 (ENOMEM): The settings storage has no room for another entry. Restore the defaults with AT+CONFDEF and save again. • +CME ERROR: 13 (EACCES): The settings storage is not available. Reboot and try again. • +CME ERROR: 5 (EIO): The settings could not be written. Try again. A repeat is a hardware fault.

8.2. Configuration defaults (AT+CONFDEF)

Execute Operation	AT+CONFDEF
-------------------	------------

Description	Erases the saved configuration and reboots. Everything AT+CONFSAVE saves, the device name and the USB Vendor and Product IDs return to the factory defaults. The defaults are stored again at that boot. AT+CONFSAVE is not required. Note: OK is sent first. The AW100 then resets. Reconnect to the COM port if your terminal does not reconnect automatically.
Parameters	None
Response	OK
Example	<pre>AT+CONFDEF OK</pre>
Test Operation	AT+CONFDEF=?
Description	Confirms the command exists. The command has no parameters.
Response	+CONFDEF OK
Example	<pre>AT+CONFDEF=? +CONFDEF OK</pre>
Errors	• +CME ERROR: 5 (EIO): The saved configuration could not be erased. Try again. A repeat is a hardware fault.

9. USB Commands

9.1. USB VID/PID (AT+USBID)

Note: The factory default is 1915,530A. AT+CONFDEF restores it.

Set Operation	AT+USBID=<VendorID>,<ProductID>
Description	Set the USB Vendor and Product IDs. The command stores the values at once. AT+CONFSAVE is not needed and does not cover them. The new IDs are used from the next boot. The current USB connection is not changed.
Parameters	<VendorID> USB Vendor ID, 1 to 4 hexadecimal digits (0000 to FFFF), upper or lower case, no 0x prefix. <ProductID> USB Product ID, 1 to 4 hexadecimal digits (0000 to FFFF), upper or lower case, no 0x prefix.
Response	OK
Example	<pre>AT+USBID=1915,530A OK</pre>
Read Operation	AT+USBID?
Description	Read the USB Vendor and Product IDs in use by the current USB connection. A pair set with AT+USBID is reported from the next boot.
Response	+USBID: <VID>,<PID> OK
Example	<pre>AT+USBID? +USBID: 1915,530A OK</pre>
Test Operation	AT+USBID=?
Description	Confirms the command exists and shows its parameters.
Response	+USBID=<Vendor-ID>,<Product-ID> OK
Example	<pre>AT+USBID=? +USBID=<Vendor-ID>,<Product-ID> OK</pre>
Errors	• +CME ERROR: 22 (EINVAL): An ID is larger than FFFF or is quoted. A 0x prefix or other non-hex text is answered with ERROR. • +CME ERROR: 28 (ENOSPC): The settings storage is full. Restore the defaults with AT+CONFDEF and try again. • +CME ERROR: 12 (ENOMEM): The settings storage has no room for another entry. Restore the defaults with AT+CONFDEF and try again. • +CME ERROR: 13 (EACCES): The settings storage is not available. Reboot and try again. • +CME ERROR: 5 (EIO): The new IDs could not be written. Try again. A repeat is a hardware fault.

9.2. USB audio stream status (AT+USBAUD)

Note: Available since firmware v1.3.

Note: Only the read form AT+USBAUD? reports anything. AT+USBAUD answers OK alone.

Read Operation	AT+USBAUD?
Description	Reports the state of the USB audio streams and the USB audio counters. The host opens and closes the streams. The counters are cumulative since boot. Check them first when audio over USB clicks or drops. Both streams carry 48 samples in every USB frame. The AW100 absorbs the offset between the host clock and its own audio clock (<RatePPM>) by dropping or repeating a 1 ms block of audio, about every 1000/<RatePPM> seconds: every 50 seconds at 20 ppm. On a healthy link the counters can therefore grow slowly. A fast growth points to a problem on the USB link or in the host.
Response	+USBAUD: <HpStream>,<MicStream>,<Rate>,<RatePPM>,<Frames>,<RxOverruns>,<RxDropped>,<TxUnderruns> OK
Example	<pre>AT+USBAUD? +USBAUD: 1,0,786485,67,245086,0,0,0 OK</pre>
Test Operation	AT+USBAUD=?
Description	Confirms the command exists and shows the response fields.
Response	+USBAUD: <hp-stream>,<mic-stream>,<feedback-q14>,<feedback-ppm>,<frames-measured>,<rx-overruns>,<rx-dropped>,<tx-underruns> OK
Example	<pre>AT+USBAUD=? +USBAUD: <hp-stream>,<mic-stream>,<feedback-q14>,<feedback-ppm>,<frames-measured>,<rx-overruns>,<rx-dropped>,<tx-underruns> OK</pre>

10. Firmware Update Commands

10.1. Enter DFU Mode (AT+DFU)

Execute Operation	AT+DFU
Description	Enters firmware update (DFU) mode. The AW100 reboots into the bootloader and appears to the host as a serial (CDC ACM) device for the image upload. Nothing is sent in reply. If the AW100 cannot enter DFU mode it answers ERROR and stays in the application. The AW100 Firmware Update Guide describes the update procedure. Note: AW100s shipped with firmware v1.0 do not support this command. Their bootloader must be updated first.
Parameters	None
Response	None (Reboot into DFU mode)
Example	AT+DFU
Test Operation	AT+DFU=?
Description	Confirms the command exists. No parameter information is reported.
Response	OK
Example	AT+DFU=? OK

11. Worked Examples

Each example sets up one or more AW100s for a task. The commands are given in the order to send them. The rules for each command are in its entry in the previous chapters. Each example names the entries it uses.

Every example has the same parts: the goal and the equipment, the starting state, the commands with the responses the AW100 sends, a check of the result, and whether the configuration has to be saved. The numbered callouts explain each step.

The host is a terminal application on the USB CDC serial port. The baud rate does not matter on USB. The audio device indexes are those of the example. Read them with AT+AUDDEVLIST on the AW100 at hand. The Broadcast IDs, addresses and names are examples too.

11.1. The Factory Default State

An AW100 is in the factory default state when it leaves the factory, after AT+CONFDEF, and after a factory reset with the button (hold it for 5 seconds until the LED blinks cyan fast, then release it, see [LED Indications](#)). In that state:

- One Public Broadcast Source exists and is started. The AW100 broadcasts whenever it is powered. The LED is steady blue.
- The broadcast is named after the AW100: "Aurawave-" and the last four hex digits of the chip ID, for example "Aurawave-4F6D". This is also the device name (AT+BTDEVNAME) and the name the USB host shows for the AW100.
- The broadcast has one subgroup with two streams, front left and front right. It has no language. It uses 48 kHz High Reliability ("PRESET" 3) and is not encrypted.
- The audio device is USB. The AW100 is a USB sound card and broadcasts what the host plays to it.
- The Public Broadcast Sink is not started. Its location is 1 (front left) and it has no language (AT+PBSNKCONF).

The simplest use of an AW100 needs no command. Plug a factory default AW100 into a USB audio host and play audio. Receive the broadcast with any Auracast receiver and its assistant. The broadcast has the name the host shows for the AW100. The examples start from this state unless they say otherwise.

11.2. Example 1: A Named Broadcast from the Analog Input, Kept Across Power Cycles

Goal: broadcast the audio on the analog line input under a name of your choice. The broadcast restarts by itself at every power-up.

Equipment: one AW100, an analog audio source on its line input, an Auracast receiver and assistant to listen with.

Starting state: factory default.

```

AT+PBSRCDEL=0 ①
OK
AT+AUDDEVLIST ②
+AUDDEVLIST: 3
0,USB
1,I2S_MAIN
2,I2S_AUX
OK
AT+AUDDEVSET=1
OK
AT+PBSRCCREATE ③
+PBSRCCREATE: 0
OK
AT+PBSRCCONF=0,"NAME","Television-1" ④
OK
AT+PBSRCCONF=0,"CODE","secret12" ⑤
OK
AT+PBSRCSTART=0 ⑥
OK
AT+PBSRCSTATE=0 ⑦
+PBSRCSTATE: 0,2,"STREAMING"
OK
AT+CONFSAVE ⑧
OK

```

- ① Delete the factory source. A source that has been started cannot be reconfigured. It has to be deleted and created again (AT+PBSRCDEL). The command answers OK also when no source exists.
- ② Select the analog input. The I2S_MAIN audio device uses the analog jacks. Read the list first. The index depends on the AW100 variant (AT+AUDDEVLIST, AT+AUDDEVSET).
- ③ Create the new source. It has two streams, front left and front right, language eng, 48 kHz High Reliability, and the name of the AW100 (AT+PBSRCCREATE). Unlike the factory source, it advertises a language.
- ④ Name the broadcast, 4 to 32 characters. This is the name receivers list. Other settings also go here, before the start: a Standard Quality preset for hearing aids ("PRESET" 4 or 5), or a subgroup language (AT+PBSRCCONF, AT+PBGRPCONF).
- ⑤ Optional. A Broadcast Code of 4 to 16 characters makes the broadcast encrypted. Receivers then need the code. An AW100 sink takes it in AT+PBSNKSTART. Leave this step out for an open broadcast.
- ⑥ Start the broadcast (AT+PBSRCSTART).
- ⑦ **Check:** STREAMING and a steady blue LED mean the broadcast is on the air (AT+PBSRCSTATE). Receivers now list "Television-1".
- ⑧ **Persist:** save the source, its started state and the audio device. At the next power-up the AW100 broadcasts "Television-1" from the analog input again (AT+CONFSAVE). Without this step the next boot returns to the last saved configuration.

11.3. Example 2: An AW100 Receiving Another AW100

Goal: use a second AW100 as the receiver of the first one's broadcast. The received audio goes to the analog output. **Equipment:** two AW100s, a USB audio host for the source, headphones or an amplifier on the sink's analog output, a terminal on the sink.

Starting state: both AW100s factory default. Plug the source into the USB audio host and play audio. Its LED is steady blue. On the sink:

```

AT+PBSNKSTART="Aurawave" ①
OK
AT+PBSNKSTATE ②
+PBSNKSTATE: 0,2,"STREAMING",9E26A2,"CE:18:12:13:57:37",1,0
OK
AT+AUDDEVSET=1 ③
OK
AT+CONFSAVE ④
OK

```

- ① Start the sink with the source's name. The name is a prefix. "Aurawave" joins the first AW100 found with a factory name. With several factory-named AW100s on the air, give the whole name and 1: AT+PBSNKSTART="Aurawave-4F6D",1. Starting the sink deletes this AW100's own source. An AW100 is a source or a sink, not both (AT+PBSNKSTART).
- ② **Check:** the state is STARTING (LED green, fast blink) while the sink synchronises, then STREAMING (LED steady green). When the source is not on the air yet, the state is SEARCHING (LED green, slow blink) and the sink keeps searching by itself (AT+PBSNKSTATE).
- ③ Route the received audio to the analog output, the I2S_MAIN audio device. The index is the one AT+AUDDEVLIST reports. With USB selected, the sink delivers the audio to its USB host as a microphone (AT+AUDDEVSET).
- ④ **Persist** (optional): a saved started sink searches for its source again at every power-up and plays as soon as it is found (AT+CONFSAVE).



An AW100 sink plays one stream of the broadcast and sends it to both of its outputs. With the factory sink location, front left, it plays the source's left channel. For stereo with two AW100 sinks see [Example 3: A Stereo Pair of AW100 Sinks](#). To make the AW100 a source again, send AT+PBSNKSTOP, AT+PBSRCCREATE and AT+PBSRCSTART=0, then AT+CONFSAVE if the sink had been saved. AT+CONFDEF returns to the factory state instead.

11.4. Example 3: A Stereo Pair of AW100 Sinks

Goal: two AW100 sinks play the left and the right channel of one broadcast, each on its own output, for example into the two channels of an amplifier or two active speakers. **Equipment:** three AW100s (one source, two sinks), or any stereo Auracast source and two AW100s, and a terminal on each sink.

Starting state: the source broadcasts as in example 2, here "Aurawave-4F6D". Sink A is set up exactly as in example 2. Its factory location, front left, is the left channel. On sink B:

```

AT+PBSNKCONF="LOC",2 ①
OK
AT+PBSNKSTART="Aurawave-4F6D",1 ②
OK
AT+PBSNKSTATE
+PBSNKSTATE: 0,2,"STREAMING",9E26A2,"CE:18:12:13:57:37",1,0
OK
AT+AUDDEVSET=1
OK
AT+CONFSAVE ③
OK

```

- ① Set the sink's location to front right before starting it. The configuration of a sink that has found its source cannot be changed (AT+PBSNKCONF).

- ② Start with the whole name and 1 (exact match). Both sinks then join the same source whatever else is on the air. Both sinks report the same Broadcast ID in AT+PBSNKSTATE.
- ③ **Persist:** the location is saved with the sink. The pair comes back as left and right at every power-up (AT+CONFSAVE).

Check: both LEDs steady green. Sink A plays the left channel and sink B the right one. A sink whose location the broadcast does not offer reports NOMATCH (LED green and blue alternating) and plays nothing.

11.5. Example 4: Receiving a Phone or Third-Party Broadcast

Goal: receive an Auracast broadcast from a phone or another vendor's transmitter on an AW100 sink. The broadcast may be encrypted.

Equipment: one AW100 with a terminal, and an Auracast source. Here the source is a phone that broadcasts with a Broadcast Code and shows the code on its screen.

Starting state: factory default. The phone's broadcast is on the air.

```
AT+PBSNKSCAN=5 ①
+PBSNKSCAN: "Galaxy S24",3A7F10,1,"5C:2E:59:1B:7D:E4",1,0
+PBSNKSCAN: "Galaxy S24",3A7F10,1,"5C:2E:59:1B:7D:E4",1,0
OK
AT+PBSNKEXPLORE=3A7F10,"5C:2E:59:1B:7D:E4",1,0 ②
+PBSNKEXPLORE: "PD",40000
+PBSNKEXPLORE: "SUBGROUPS",1
+PBSNKEXPLORE: 0,"CODEC",06,0000,0000
+PBSNKEXPLORE: 0,"STREAMS",1
OK
AT+PBSNKSTART=3A7F10,"1234" ③
OK
AT+PBSNKSTATE ④
+PBSNKSTATE: 0,2,"STREAMING",3A7F10,"5C:2E:59:1B:7D:E4",1,0
OK
```

- ① Scan for 5 seconds. Each line is one announcement received: the source's name, its Broadcast ID, whether it is encrypted (1 means a code is needed), its address, address type and advertising set ID. A source is normally listed more than once (AT+PBSNKSCAN).
- ② Optional. Read the broadcast's structure with the ID, address, address type and advertising set ID from the scan. This one has one subgroup with one stream and no "LOC" line. It is a mono stream, which a sink with any location takes (AT+PBSNKEXPLORE).
- ③ Start the sink with the Broadcast ID (unquoted, as the scan shows it) and the code the phone displays. An ID always matches exactly. A name would do as well. The AW100 decodes the Standard Quality (16 and 24 kHz) and the 10 ms High Quality (48 kHz) configurations that phones and transmitters use (AT+PBSNKSTART).
- ④ **Check:** STREAMING and a steady green LED. A stereo broadcast gives the left channel with the factory sink location. AT+PBSNKCONF="LOC",0 before the start takes the first stream offered, whatever its location (AT+PBSNKSTATE).

Persist: AT+CONFSAVE keeps the ID and the code. The sink rejoins that broadcast at every power-up.

11.6. Example 5: A Two-Language Broadcast and Sinks Choosing by Language

Goal: one broadcast carries two languages. The left input channel is English and the right one is French. Each is a mono stream in its

own subgroup. A sink picks the language. **Equipment:** one AW100 as the source with a USB or analog stereo feed that carries one language per channel, one or more AW100s as sinks (or any Auracast receiver whose assistant lists subgroups), and a terminal on each.

Starting state: factory default. On the source:

```
AT+PBSRCDEL=0
OK
AT+PBSRCCREATE
+PBSRCCREATE: 0
OK
AT+PBSRCCONF=0,"NAME","Conference-1"
OK
AT+PBSRCCONF=0,"SUBGROUPS",2 ①
OK
AT+PBGRPCONF=0,0,"STREAMS",1 ②
OK
AT+PBSTRMCONF=0,0,0,"LOC",1
OK
AT+PBGRPCONF=0,0,"AUD_SRC",1
OK
AT+PBGRPCONF=0,0,"LANG","eng"
OK
AT+PBGRPCONF=0,0,"NAME","Floor" ③
OK
AT+PBGRPCONF=0,1,"STREAMS",1 ④
OK
AT+PBSTRMCONF=0,1,0,"LOC",2
OK
AT+PBGRPCONF=0,1,"AUD_SRC",2
OK
AT+PBGRPCONF=0,1,"LANG","fra"
OK
AT+PBGRPCONF=0,1,"NAME","Interpreter"
OK
AT+PBSRCSTART=0
OK
AT+CONFSAVE
OK
```

- ① Two subgroups. Every "SUBGROUPS" write creates the subgroups again, empty. The subgroup settings follow it (AT+PBSRCCONF).
- ② Subgroup 0: one stream, located front left, fed by the left input channel ("AUD_SRC" 1), language eng. The two streams of a broadcast may be spread over the subgroups in any way. There are two streams in total (AT+PBGRPCONF, AT+PBSTRMCONF).
- ③ Optional. A subgroup name is sent as the subgroup's Program_Info. An assistant can show "Floor" and "Interpreter" next to the languages.
- ④ Subgroup 1: one stream, front right, fed by the right input channel ("AUD_SRC" 2), language fra.

On a sink that should play French:

```
AT+PBSNKCONF="LOC",2 ①
OK
AT+PBSNKCONF="LANG","fra" ②
OK
AT+PBSNKSTART="Conference-1",1 ③
OK
AT+PBSNKSTATE
+PBSNKSTATE: 0,2,"STREAMING",B41C02,"CE:18:12:13:57:37",1,0
OK
AT+CONFSAVE
OK
```

- ① The French stream is located front right. Set the sink's location to 2 before the start, or to 0 to take any stream of the subgroup. With the factory location, front left, the sink would report NOMATCH (AT+PBSNKCONF).
- ② Set the language before the start. The sink then skips subgroups that advertise a different language. A source that advertises no language at all plays as if no language were set.
- ③ Start with the broadcast's name. The sink takes the stream of the fra subgroup and plays it on both outputs. A second sink with "LANG","eng", or with no language set, and the factory location plays the English channel (AT+PBSNKSTART).

Check: STREAMING and a steady green LED. A language that is not on the air gives NOMATCH (LED green and blue alternating, slow). Nothing is played until the sink's language or the broadcast changes. AT+PBSNKCONF="LANG","" returns the sink to any subgroup.

Persist: AT+CONFSAVE on the source keeps the two-language broadcast. On each sink it keeps the language, the location and the target.

12. Supported Result Codes

12.1. Summary of CME ERROR Codes

A result code of +CME ERROR: <cause value> indicates an error. The following table lists the cause values the AW100 returns to AT commands. The **Errors** row of each command gives the cause for that command.

Value	Error Code	Context	AT Commands
5	EIO	A write to the settings storage failed	AT+CONFDEF AT+CONFSAVE AT+USBID
12	ENOMEM	No room. A value is longer than its field, or there is no free slot, subgroup, stream or storage entry	AT+BTDEVNAME AT+CONFSAVE AT+PBGRPCONF AT+PBSNKCONF AT+PBSNKEXPLORE AT+PBSNKSTART AT+PBSRCCONF AT+PBSRCCREATE AT+PBSRCSTART AT+PBSTRMCONF AT+USBID
13	EACCES	Refused in the current state. The sink has found its source, or the settings storage is not available	AT+CONFSAVE AT+PBSNKCONF AT+PBSNKSTART AT+USBID
16	EBUSY	Something else is in progress or forbids the change. A scan or a search is running, or the source has been started and cannot be reconfigured	AT+PBGRPCONF AT+PBSNKEXPLORE AT+PBSNKSCAN AT+PBSRCCONF AT+PBSTRMCONF
19	ENODEV	Nothing found. No broadcast with that ID was heard	AT+PBSNKEXPLORE

Value	Error Code	Context	AT Commands
22	EINVAL	A value is not acceptable, or the source it refers to does not exist	AT+AUDDEVSET AT+BTDEVNAME AT+PBGRPCONF AT+PBSNKCONF AT+PBSNKEXPLORE AT+PBSNKSCAN AT+PBSNKSTART AT+PBSRCCONF AT+PBSRCSTART AT+PBSRCSTATE AT+PBSRCSTOP AT+PBSTRMCONF AT+USBID
28	ENOSPC	The settings storage is full	AT+CONFSAVE AT+USBID
37	ECHRNA	An index is beyond the list it refers to	AT+AUDDEVSET AT+PBGRPCONF AT+PBSRCCONF AT+PBSRCDEL AT+PBSRCSTART AT+PBSRCSTATE AT+PBSRCSTOP AT+PBSTRMCONF
61	ENODATA	The value asked for has not been set	AT+PBGRPCONF
116	ETIMEDOUT	The operation did not complete in time	AT+PBSNKEXPLORE AT+PBSRCDEL AT+PBSRCSTOP
120	EALREADY	The sink already has its source, or the source went away while it was being read	AT+PBSNKEXPLORE AT+PBSNKSTART

Cause values 12, 22 and 34 can also come from any command that takes parameters. See [Responses](#).

12.2. Summary of Unsolicited Result Codes

The AW100 writes one unsolicited line, +READY, when it has finished booting and accepts commands. An empty line precedes it. The saved configuration has been applied by then, including a source or sink that starts at boot. There are no other unsolicited result codes.

Additional Information - Cloud2GND

Headquarters

Cloud2GND, LLC
4500 Bowling Blvd #100
Louisville, KY 40207 USA

Website

<https://www.cloud2gnd.com/>

Technical Support

<https://cloud2gnd.com/contact/>

Note: Information contained in this document is subject to change.

Additional Information - Ezurio

Ezurio supplies the BL5340PA radio module of the AW100 and is the sales and distribution channel of the AW100. Please contact your local Ezurio sales representative for ordering and availability:

Headquarters	Ezurio 50 S. Main St. Suite 1100 Akron, OH 44308 USA
Website	https://www.ezurio.com/
Sales Contact	https://www.ezurio.com/contact

Ezurio's products are subject to standard [Terms & Conditions](#).

Copyright © 2026 Cloud2GND, LLC. All rights reserved. Any information furnished by Cloud2GND and its agents is believed to be accurate but cannot be guaranteed. All specifications are subject to change without notice. Responsibility for the use and application of Cloud2GND materials or products rests with the end user since Cloud2GND and its agents cannot be aware of all potential uses. Cloud2GND makes no warranties as to non-infringement nor as to the fitness, merchantability, or sustainability of any Cloud2GND materials or products for any specific or general uses. Cloud2GND or any of its affiliates or agents shall not be liable for incidental or consequential damages of any kind. Nothing herein provides a license under any Cloud2GND or any third-party intellectual property right. Aurawave is a trademark of Cloud2GND, LLC. The Bluetooth® word mark and logos and Auracast™ are trademarks owned by Bluetooth SIG, Inc. Ezurio and its associated logos are trademarks owned by Ezurio and/or its affiliates. Other trademarks are the property of their respective owners.