

# Aaware AEV13MZ Far-Field Voice and Sound Capture Quickstart Guide

---

## Powering Up

Connect micro-USB cable to the JTAG-UART USB port (J2) as shown in figure 1, and the other side to a PC/laptop (see figure 2). If you do not need access to the terminal (i.e. WiFi is setup), a USB battery can also be used as the power source, and alternately plugged into the MiniZed AUX-PWR USB port (J6).



Figure 1: Power via UART-JTAG USB

## Far-Field Voice Assistant Operation

Upon power up and boot, the platform will go into voice-assistant mode and start listening for voice commands based on the Wakeword "Alexa" (or optionally both "Alexa" and "Ok Google"). An LED will blink in the direction of the detected WakeWord, and noise/interference suppressed audio will be captured from that direction. See the User Guide for more details.

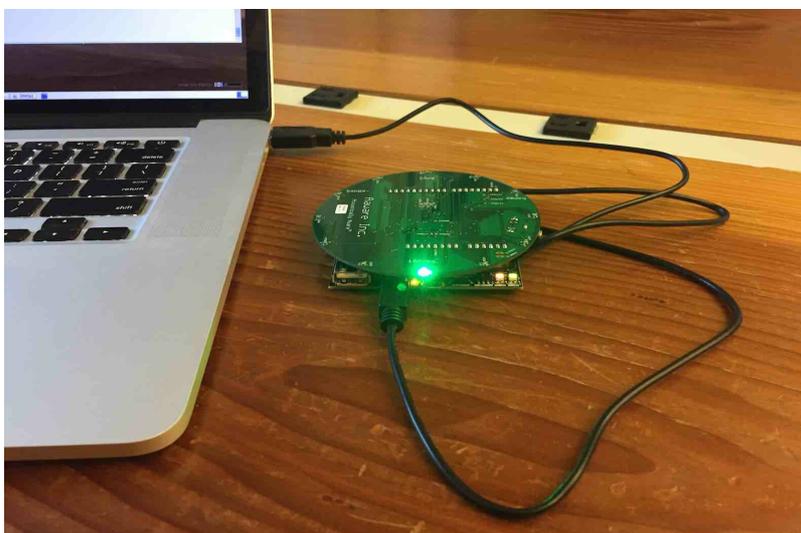


Figure 2: Aaware Wakeword Directional LED Blink

---

# Basic Linux Features and Administration

## Terminal Connection

If you have connected the USB port to a laptop/USB, then you can use terminal software to log into the Aaware embedded Linux environment. The terminal serial settings must be 115200,8N1 (115200 baud rate, 8-bit, no parity bits, 1 stop bit). Once connected you will be logged in as root and will see a linux prompt:

```
port is      : /dev/ttyUSB1
...
Terminal ready

root@aawmz1:~# uname -a
Linux aawmz1 4.4.0-aawos+ #1 SMP PREEMPT Fri Nov 3 21:28:18 PDT 2017 armv7l armv7l armv7l GNU/
root@aawmz1:~#
...
```

Note: you will automatically be logged in as root user.

## WiFi Setup

The Aaware Linux platform is setup with a network WiFi interface as wlan0, as shown by the ifconfig command below.

```
root@aawmz1:~# ifconfig
lo      Link encap:Local Loopback
        inet addr:127.0.0.1  Mask:255.0.0.0
        inet6 addr: ::1/128 Scope:Host
        UP LOOPBACK RUNNING  MTU:65536  Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
        RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

wlan0   Link encap:Ethernet  HWaddr e0:e5:cf:00:fb:93
        inet addr:192.168.0.102  Bcast:192.168.0.255  Mask:255.255.255.0
        inet6 addr: fe80::e2e5:cfff:fe00:fb93/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:200 errors:0 dropped:0 overruns:0 frame:0
        TX packets:27 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:41146 (41.1 KB)  TX bytes:3842 (3.8 KB)

root@aawmz1:~#
```

To set this up for your own basic WiFi network, you can follow the steps below. For networks with more

complex security or more details on wpa\_supplicant, some good starting points are:

- [https://wiki.archlinux.org/index.php/WPA\\_supplicant](https://wiki.archlinux.org/index.php/WPA_supplicant)
- [http://manpages.ubuntu.com/manpages/trusty/man5/wpa\\_supplicant.conf.5.html](http://manpages.ubuntu.com/manpages/trusty/man5/wpa_supplicant.conf.5.html)
- <https://wiki.debian.org/WiFi/HowToUse>

Basic WiFi setup steps:

1. Add your network to /etc/wpa\_supplicant.conf using a text editor of choice (vi/vim is available on the platform). An example entry is shown below for a standard password protected network, just change the placeholder parameters like myssid to your real network values.

```
root@aawmz1:~# vim /etc/wpa_supplicant.conf
##### wpa_supplicant configuration file #####

ctrl_interface=/var/run/wpa_supplicant
ctrl_interface_group=netdev
update_config=1

network={
    auth_alg=OPEN
    key_mgmt=WPA-PSK
    psk="mypassword"
    ssid="myssid"
    proto=RSN
    mode=0
}
...
```

1. Optionally you can protect your access point password using wpa\_passphrase, and use the result psk in wpa\_supplicant.conf instead of having the real password:

```
root@aawmz1:~$ wpa_passphrase myssid mypasswd
network={
    ssid="myssid"
    #psk="mypasswd"
    psk=e39d75850c60012e6164b936d1fcfc3b5c96119026210727ba79081a9f4f8e85
}
```

1. Rescan for networks using the iw wlan0 scan command. This should automatically connect to your interface if it is in range.

```
root@aawmz1:~# iw wlan0 scan | grep SSID
SSID: NETGEAR05
SSID: familyabchome
SSID: TurnKey Vacation Rentals
...
```

## Stop/Start/Restart WiFi Interface

The standard ifdown, ifup, ifconfig commands will work for stopping and starting the WiFi interface.

```
root@aawmz1:~# ifdown wlan0
root@aawmz1:~# ifconfig
wlan0      Link encap:Ethernet  HWaddr e0:e5:cf:00:fb:93
           UP BROADCAST MULTICAST  MTU:1500  Metric:1
           RX packets:509 errors:0 dropped:0 overruns:0 frame:0
           TX packets:54 errors:0 dropped:0 overruns:0 carrier:0
           collisions:0 txqueuelen:1000
           RX bytes:105161 (105.1 KB)  TX bytes:7684 (7.6 KB)
...
root@aawmz1:~# ifup wlan0
```

## The Administrative User: aawadm

For nearly all use cases, one should operate as a non-root user. For the Aaware platform, the sound capture administrator user aawadm is the recommended account for using the platform. The default passwd is admaaw17.

## Remote Login with ssh

Once WiFi is up, use secure shell to login as aawadm:

```
[laptop@vmel6 ~]$ ssh aawadm@192.168.0.102
aawadm@192.168.0.102's password:
Welcome to Ubuntu 14.04 (GNU/Linux 4.4.0-aawos+ armv7l)
 * Documentation:  http://www.ubuntu.com
Last login: Thu Aug 10 00:53:02 2017 from 192.168.0.106

aawadm@aawmz1:~$
```

## Installing Linux Applications

The AEV13MZ provides a standard Ubuntu 14.04 environment with applications available via the Ubuntu repositories. Simply use apt-get to install your favorite applications or development tools. Many of the most popular development environments like Python2 and 3 are already installed.

```
aawadm@aawmz1:~$ sudo apt-get install python
[sudo] password for aawadm:
Reading package lists... Done
Building dependency tree
Reading state information... Done
python is already the newest version.
python set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 4 not upgraded.
```

---

# Voice Application Development and Support

The AEV13MZ platform is a fully integrated far-field voice/sound capture solution ready for voice application development and is designed for exploration of various configurations.

Standard ALSA (Advanced Linux Sound Architecture) interfaces are provided to deliver "clean" audio following a wakeword (i.e. after "Alexa" or "OK Google" is spoken). This provides a familiar programming interface that supports a variety of voice application frameworks like Google Assistant and Amazon Voice Services. Please see the User Guide for details on how to use the API to bring up various voice assistant frameworks, play audio, and develop your own voice applications.

The default microphone array configuration uses all 13 microphones included on the Aaware board, but is easily re-configured to use a 7-mic subset. Other configurations are possible. See the User Guide on how to switch configurations. Other array configurations and algorithm customization is possible. Please contact your Aaware representative for details on algorithm customization.

## User Guide Included on AEV13MZ

The User Guide is included on the AEV13MZ in directory /home/aawadm/aev13mz-userguide-vxxx.pdf. Updates are available by contacting Aaware support.

## Community Support

General support for the MiniZed is available at: <http://zedboard.org/forums/zed-english-forum>

and there is an Aaware Voice Kit thread available here: <http://zedboard.org/forums/aaware-voice-kit>

Contact Aaware Inc. at: [support@aaware.com](mailto:support@aaware.com)

ver: aev13mz-v1.0