Sonic Pad
Firmware Burning Tutorial

（Ubuntu）

Version	V1.1
Date	2023/1/10
version record	V1.0 initial version V1.1 Chapter 4 Add Notes Before Burning
editor	

Shenzhen Creality 3D Technology Co., Ltd

Catalog:

1. Tools：

1. LiveSuitV306_For_Linux64.zip for 64-bit systems, and LiveSuitV306_For_Linux32.zip for 32-bit systems

2. （burn firmware）t800-sonic_lcd_uart0_1.0.6.40.112.img

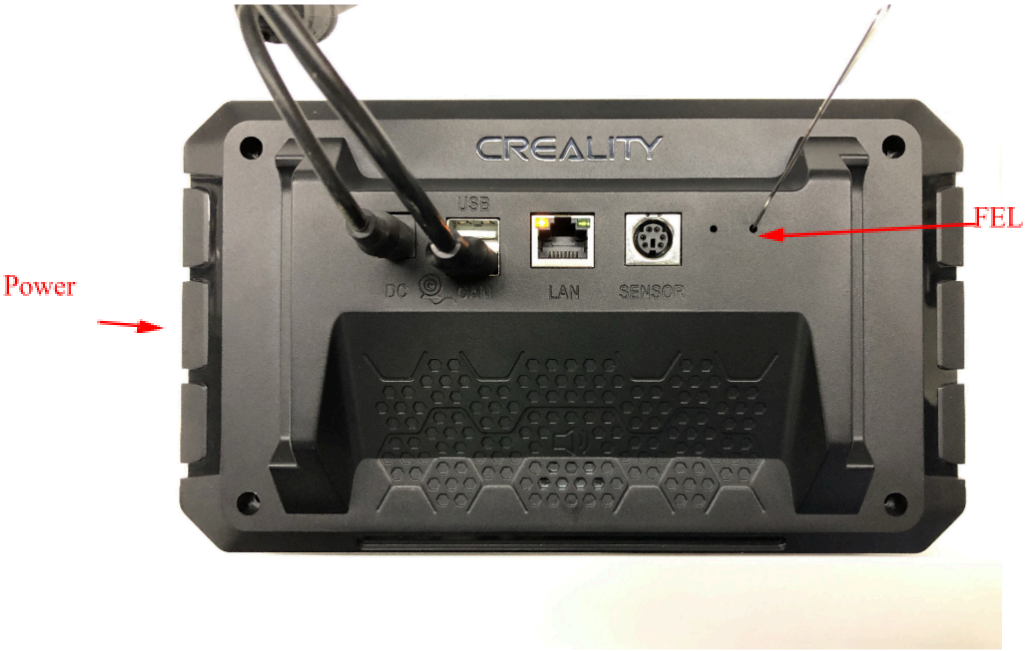
3. Male to male USB cable(TYPE-A TO TYPE-A)

2. Enter the burning mode

1. When the Sonic pad is turned off, insert the USB cable into the CAM port

2. Press and hold the FEL button on the right with a paper clip

3. Press the Power button to power on the Sonic pad, then the Sonic pad will enter the burning mode



3. Driver Installation

It is recommended to use Ubuntu 18 or Ubuntu 20. Other versions of Ubuntu may encounter installation problems, please solve the problem by yourself.

Installation Guide:

1. The binary executable file is LiveSuit.run. If the file does not have execution permission, please use the command: chmod +x LiveSuit.run in the terminal to add

executable permission to the file.

2. Open the terminal, enter the LiveSuit_For_Linux64 directory, and enter "./LiveSuit.run' ' to run the installation program. (If the dkms module is missing, please use the command: "sudo apt-get install dkms" to install dkms)

3. The program is installed in the Bin directory of the current user's Home directory. After the program is installed, enter the program directory and enter the command "./LiveSuit" to run LiveSuit. The Ubuntu release version can add udev rules as follows to read and write hardware devices without root privileges

   a) Open a terminal and enter "sudo vim /etc/udev/rules.d/10-local.rules"
   b) Enter the content of the following udev rules, and be sure to replace test with the user group where the current user belongs.
      ```
      SUBSYSTEM!="usb_device", ACTION!="add", GOTO="objdev_rules_end"
      #USBasp
      ATTRS{idVendor}=="1f3a", ATTRS{idProduct}=="efe8", GROUP="test",
      MODE="0666"
      LABEL="objdev_rules_end"
      ```

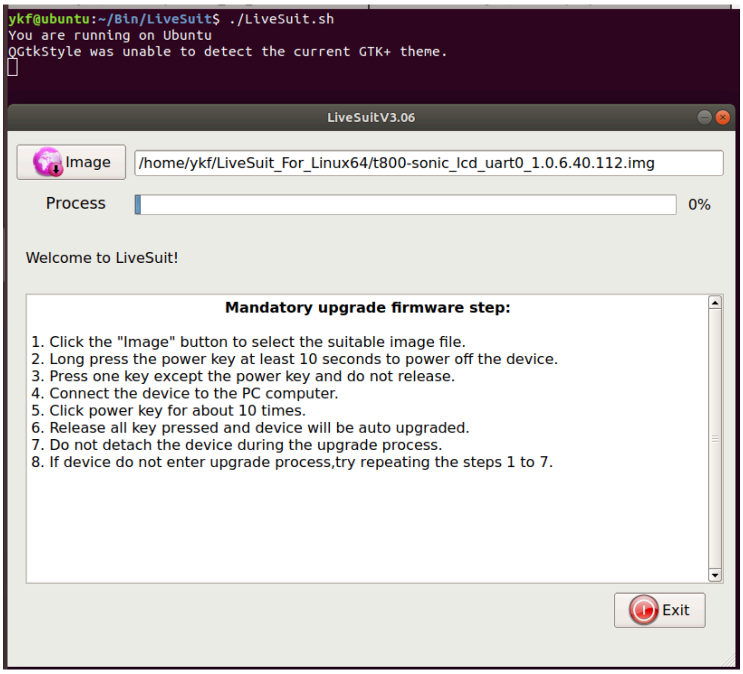
   c) Save and restart the machine or restart the udev service to run LiveSuit.sh as a normal user. The command to restart the udev service is: sudo service udev restart.

4. During the process of flashing the firmware, please do not close the program or disconnect the device to avoid hardware damage.

4. Sonic pad firmware burning

Notes before burning the firmware: Run the LiveSuit software first, connect the Sonic pad to the computer with a male-to-male USB cable, and then let the Sonic pad enter the burning mode. The specific steps are as follows:

1、Enter the LiveSuit directory and execute the ./LiveSuit.sh command to start the LiveSuit program and select the firmware to be burned



```
ykf@ubuntu:~/Bin/LiveSuit$ ./LiveSuit.sh
You are running on Ubuntu
QGtkStyle was unable to detect the current GTK+ theme.
```

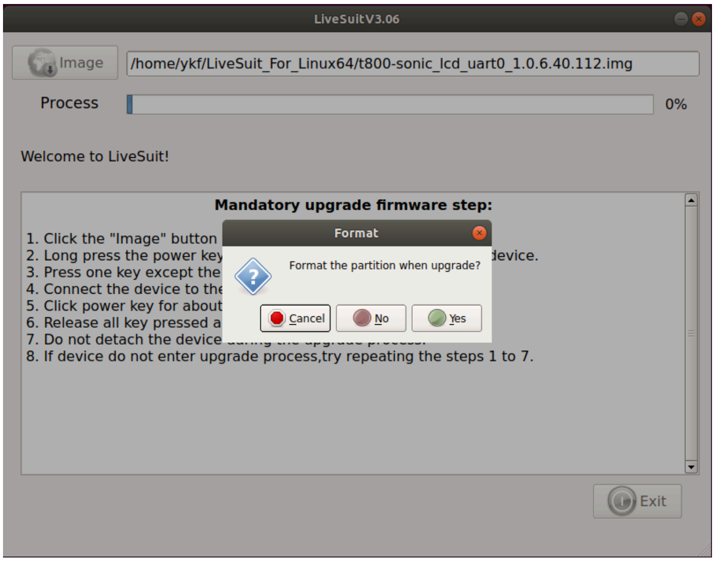
/home/ykf/LiveSuit_For_Linux64/t800-sonic_lcd_uart0_1.0.6.40.112.img

Welcome to LiveSuit!

Mandatory upgrade firmware step:

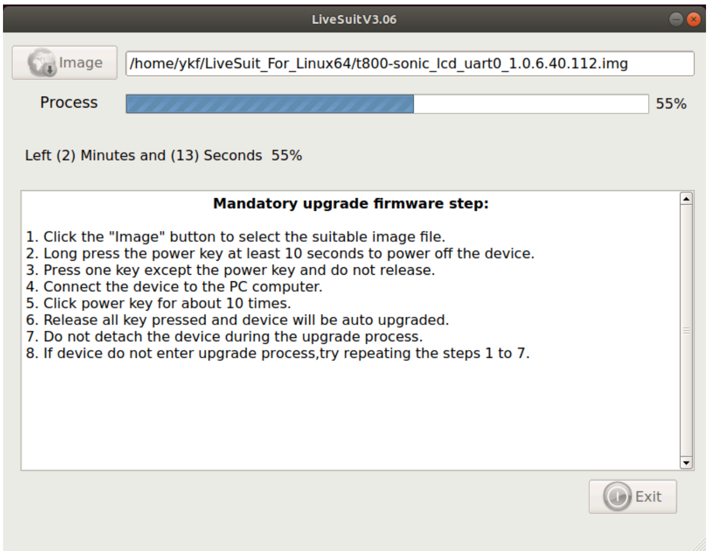
1. Click the "Image" button to select the suitable image file.
2. Long press the power key at least 10 seconds to power off the device.
3. Press one key except the power key and do not release.
4. Connect the device to the PC computer.
5. Click power key for about 10 times.
6. Release all key pressed and device will be auto upgraded.
7. Do not detach the device during the upgrade process.
8. If device do not enter upgrade process,try repeating the steps 1 to 7.

2、 Connect the Sonic pad to the PC and enter the burning mode (please refer to Chapter 2). The following page appears, click the "Yes" button



Format the partition when upgrade?

burning...



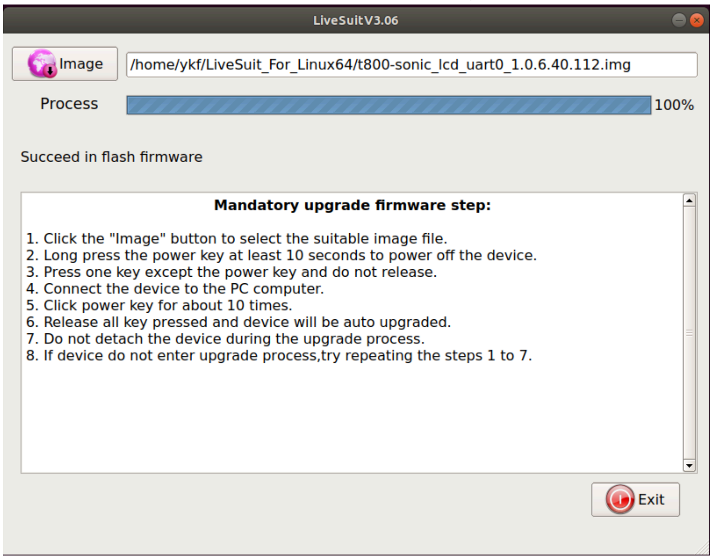
/home/ykf/LiveSuit_For_Linux64/t800-sonic_lcd_uart0_1.0.6.40.112.img

Left (2) Minutes and (13) Seconds 55%

Mandatory upgrade firmware step:

1. Click the "Image" button to select the suitable image file.
2. Long press the power key at least 10 seconds to power off the device.
3. Press one key except the power key and do not release.
4. Connect the device to the PC computer.
5. Click power key for about 10 times.
6. Release all key pressed and device will be auto upgraded.
7. Do not detach the device during the upgrade process.
8. If device do not enter upgrade process,try repeating the steps 1 to 7.

Burning is successful, the device restarts automatically



/home/ykf/LiveSuit_For_Linux64/t800-sonic_lcd_uart0_1.0.6.40.112.img

Succeed in flash firmware

Mandatory upgrade firmware step:

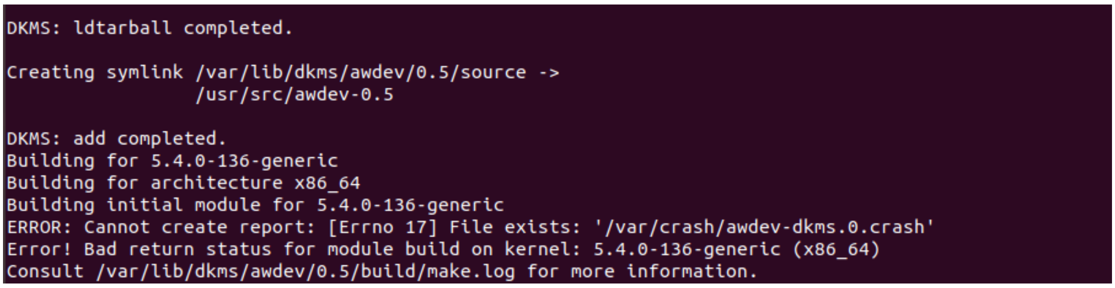
1. Click the "Image" button to select the suitable image file.
2. Long press the power key at least 10 seconds to power off the device.
3. Press one key except the power key and do not release.
4. Connect the device to the PC computer.
5. Click power key for about 10 times.
6. Release all key pressed and device will be auto upgraded.
7. Do not detach the device during the upgrade process.
8. If device do not enter upgrade process,try repeating the steps 1 to 7.

5. Precautions

1、 If the kernel version has been updated, the driver is not loaded automatically, please install the driver manually, the driver installation requires root privileges

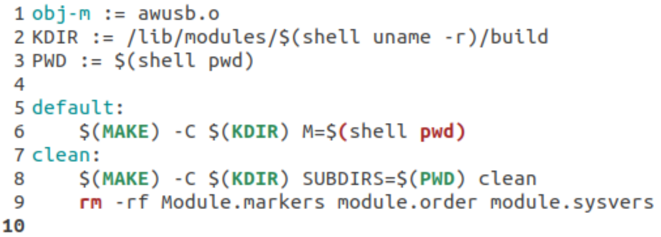
2、 sudo dpkg -l to check the software installation status, confirm that dkms, flex, and bison are installed, if the xxx library is missing, please execute sudo apt-get update; sudo apt-get install xxx

3、 If awdev installation fails, please modify the Makefile and awusb.c files as follows



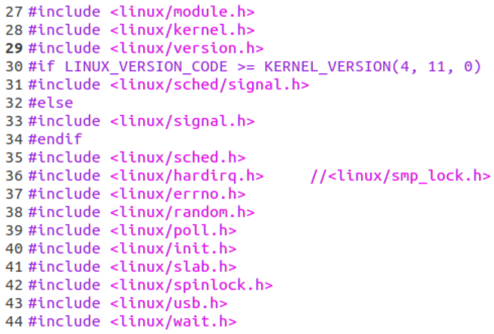
```
DKMS: ldtarball completed.

Creating symlink /var/lib/dkms/awdev/0.5/source ->
                 /usr/src/awdev-0.5

DKMS: add completed.
Building for 5.4.0-136-generic
Building for architecture x86_64
Building initial module for 5.4.0-136-generic
ERROR: Cannot create report: [Errno 17] File exists: '/var/crash/awdev-dkms.0.crash'
Error! Bad return status for module build on kernel: 5.4.0-136-generic (x86_64)
Consult /var/lib/dkms/awdev/0.5/build/make.log for more information.
```

3.1 Modify the Makefile file, sudo vim /usr/src/awdev-0.5/Makefile



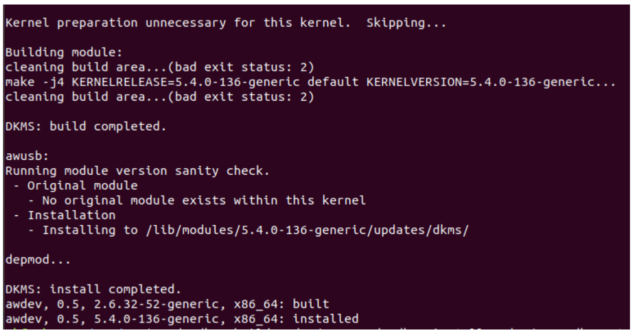
```
obj-m := awusb.o
KDIR := /lib/modules/$(shell uname -r)/build
PWD := $(shell pwd)

default:
	$(MAKE) -C $(KDIR) M=$(shell pwd)
clean:
	$(MAKE) -C $(KDIR) SUBDIRS=$(PWD) clean
	rm -rf Module.markers module.order module.sysvers

```

3.2 Prompt awusb/awusb.c:376:8: error: implicit declaration of function 'signal_pending' [-Werror=implicit-function-declaration]. Open the awusb.c file, add 29 to 34 lines of code, modify the content as follows:



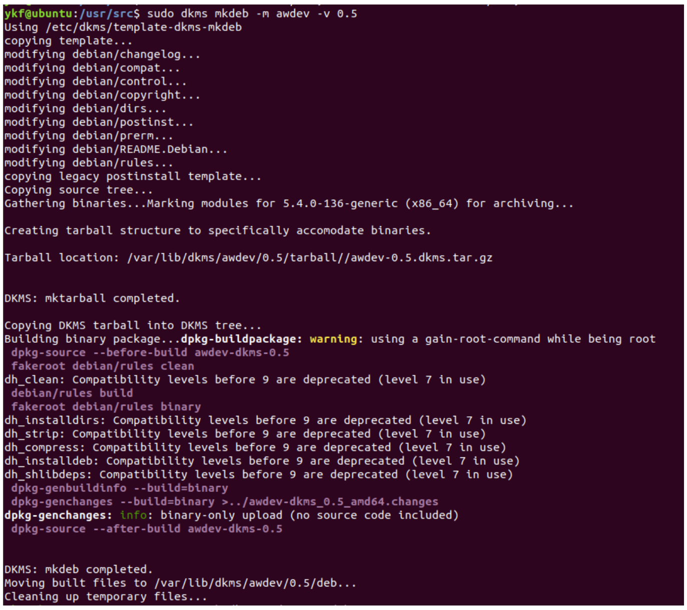
```
27 #include <linux/module.h>
28 #include <linux/kernel.h>
29 #include <linux/version.h>
30 #if LINUX_VERSION_CODE >= KERNEL_VERSION(4, 11, 0)
31 #include <linux/sched/signal.h>
32 #else
33 #include <linux/signal.h>
34 #endif
35 #include <linux/sched.h>
36 #include <linux/hardirq.h>     //<linux/smp_lock.h>
37 #include <linux/errno.h>
38 #include <linux/random.h>
39 #include <linux/poll.h>
40 #include <linux/init.h>
41 #include <linux/slab.h>
42 #include <linux/spinlock.h>
43 #include <linux/usb.h>
44 #include <linux/wait.h>
```

3.3 Modify the Makefile and execute sudo dkms build awdev/0.5; sudo dkms install awdev/0.5; dkms status command



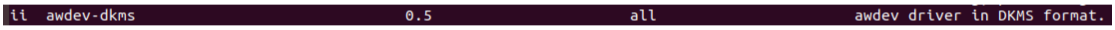
```
Kernel preparation unnecessary for this kernel.  Skipping...

Building module:
cleaning build area...(bad exit status: 2)
make -j4 KERNELRELEASE=5.4.0-136-generic default KERNELVERSION=5.4.0-136-generic...
cleaning build area...(bad exit status: 2)

DKMS: build completed.

awusb:
Running module version sanity check.
 - Original module
   - No original module exists within this kernel
 - Installation
   - Installing to /lib/modules/5.4.0-136-generic/updates/dkms/

depmod...

DKMS: install completed.
awdev, 0.5, 2.6.32-52-generic, x86_64: built
awdev, 0.5, 5.4.0-136-generic, x86_64: installed
```

3.4 Execute sudo dkms mkdeb -m awdev -v 0.5. After successful compilation, the deb package is stored in the /var/lib/dkms/awdev/0.5/deb directory, as shown in the figure.



```
ykf@ubuntu:/usr/src$ sudo dkms mkdeb -m awdev -v 0.5
Using /etc/dkms/template-dkms-mkdeb
copying template...
modifying debian/changelog...
modifying debian/compat...
modifying debian/control...
modifying debian/copyright...
modifying debian/dirs...
modifying debian/postinst...
modifying debian/prerm...
modifying debian/README.Debian...
modifying debian/rules...
copying legacy postinstall template...
Copying source tree...
Gathering binaries...Marking modules for 5.4.0-136-generic (x86_64) for archiving...

Creating tarball structure to specifically accomodate binaries.

Tarball location: /var/lib/dkms/awdev/0.5/tarball//awdev-0.5.dkms.tar.gz


DKMS: mktarball completed.

Copying DKMS tarball into DKMS tree...
Building binary package...dpkg-buildpackage: warning: using a gain-root-command while being root
 dpkg-source --before-build awdev-dkms-0.5
 fakeroot debian/rules clean
dh_clean: Compatibility levels before 9 are deprecated (level 7 in use)
 debian/rules build
 fakeroot debian/rules binary
dh_installdirs: Compatibility levels before 9 are deprecated (level 7 in use)
dh_strip: Compatibility levels before 9 are deprecated (level 7 in use)
dh_compress: Compatibility levels before 9 are deprecated (level 7 in use)
dh_installdeb: Compatibility levels before 9 are deprecated (level 7 in use)
dh_shlibdeps: Compatibility levels before 9 are deprecated (level 7 in use)
 dpkg-genbuildinfo --build=binary
 dpkg-genchanges --build=binary >../awdev-dkms_0.5_amd64.changes
dpkg-genchanges: info: binary-only upload (no source code included)
 dpkg-source --after-build awdev-dkms-0.5


DKMS: mkdeb completed.
Moving built files to /var/lib/dkms/awdev/0.5/deb...
Cleaning up temporary files...
```

The above successfully compiled the driver deb package, and entered the sudo dpkg -i awdev-dkms_0.5_amd64.deb command to install it. After the installation is successful, execute sudo dpkg -l, you can see the awdev driver module in the module directory, as shown in the figure below



```
ii  awdev-dkms                                          0.5                          all                          awdev driver in DKMS format.
```