

SB35 Development Platform for Android 11

User Guide

Aug. 2021

Version 1.0

Document Revisions

Date	Revision Number	Document Changes
08/2021	1.0	Initial version

Table of Contents

1	Overview	4
2	Preparation.....	4
2.1	Setting up your computer.....	4
3	Building the Android platform for SB35.....	4
3.1	Downloading the SB35 source.....	4
3.2	Building Android images.....	4
4	Downloading SB35 image with Flash Tool.....	6
5	Booting SB35.....	7
6	Serial console terminal	8
7	Clone project.....	9
8	Reference	9

1 Overview

This tutorial guides new developers how to build Android 11 with the MTK i350 based development platform – SB35 board. It provides manuals for:

- Setting up a Linux® OS build machine.
- Building SB35 Android system images.
- Downloading the images to SB35 development board.

For more information about building the Android platform, see <https://source.android.com/setup>

2 Preparation

The minimum recommended system requirements are as follows:

- 16 GB RAM
- 300 GB hard disk

2.1 Setting up your computer

To build the Android source files, you need a 64-bit version of Ubuntu (14.04 is recommended).

After installing the computer running Linux OS, check whether all the necessary packages are installed for an Android build. See "Setting up a Linux build environment" on the Android website source.android.com/setup/build/initializing

```
$ sudo apt-get install git-core gnupg flex bison gperf build-essential  
zip curl zlib1g-dev gcc-multilib g++-multilib libc6-dev-i386  
lib32ncurses5-dev x11proto-core-dev libx11-dev lib32z-dev libgl1-mesa-dev  
libxml2-utils xsltproc unzip
```

In addition to the packages requested on the Android website, the following packages are also needed:

```
$ sudo apt-get install libssl-dev libswitch-perl
```

NOTE

The Android 11 in AOSP comes with a prebuilt version of OpenJDK, so no additional JDK installation is required.

3 Building the Android platform for SB35

3.1 Downloading the SB35 source

Please contact your Innocomm contact window to download the SB35 source code.

3.2 Building Android images

Use the command **lunch** to set up the build configuration and **make** to start the build process are executed.

The build configuration command **lunch** can be issued with an argument <Build name>-<Build type> string, such as **lunch full_sb35-userdebug**, or can be issued without the argument presenting a menu of selection.

The build type is used to specify what debug options are provided in the final image. The following table lists the build types.

Build type	Description
user	Limited access; suited for production
userdebug	Like user but with root access and debug capability; preferred for debugging
eng	Development configuration with additional debugging tools

Android build steps are as follows:

1. Change to the top level build directory. Assuming `${SB35}` is your root directory of SB35 source.

```
$ cd ${SB35}
```

2. Set up the environment for building. This only configures the current terminal.

```
$ source build/envsetup.sh
```

3. Execute the Android **lunch** command.

```
$ lunch full_sb35-userdebug
```

4. Execute the **make** command to generate the image.

```
$ make 2>&1 | tee build-log.txt
```

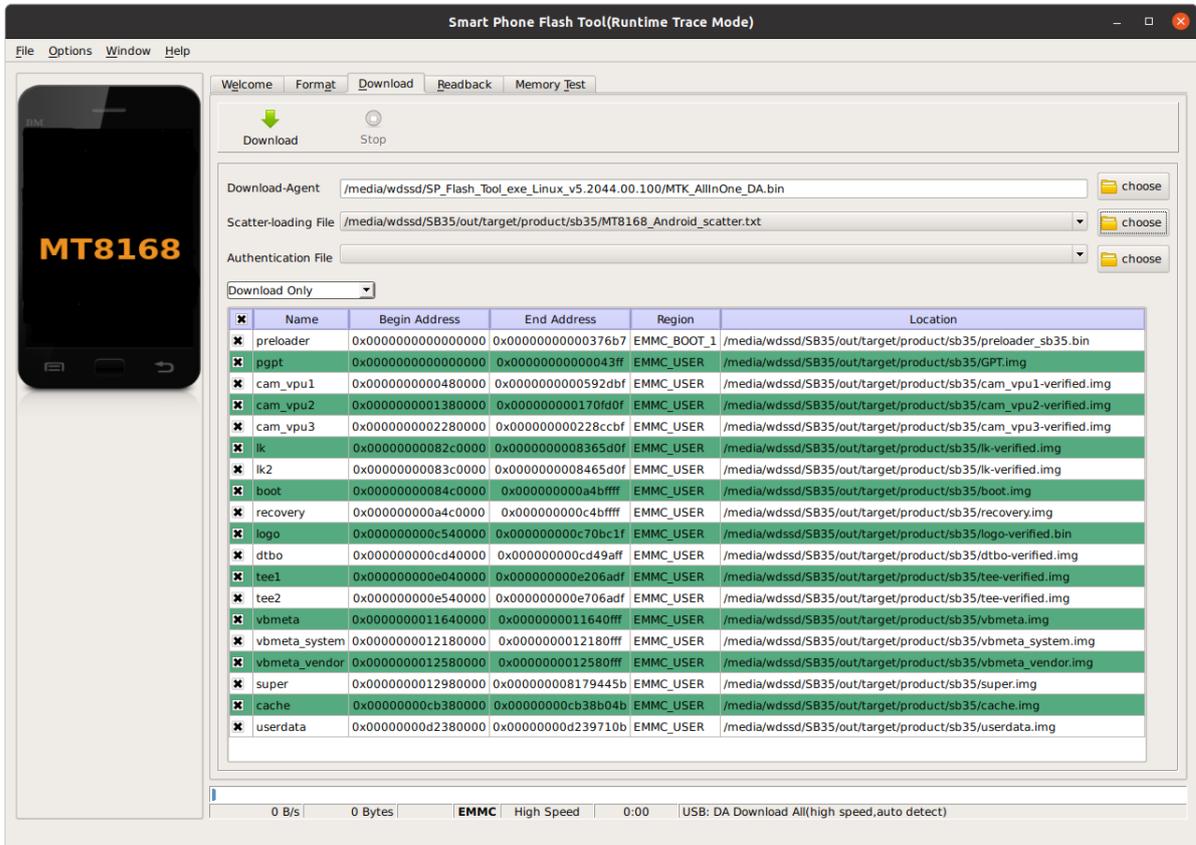
When the **make** command is complete, the `build-log.txt` file contains the execution output. Check for any errors.

For building images for SB35, the following outputs are generated by default in `${SB35}/out/target/product/sb35`.

- `${SB35}/out/target/product/sb35/preloader_sb35.bin`
- `${SB35}/out/target/product/sb35/GPT.img`
- `${SB35}/out/target/product/sb35/cam_vpu1-verified.img`
- `${SB35}/out/target/product/sb35/cam_vpu2-verified.img`
- `${SB35}/out/target/product/sb35/cam_vpu3-verified.img`
- `${SB35}/out/target/product/sb35/lk-verified.img`
- `${SB35}/out/target/product/sb35/boot.img`
- `${SB35}/out/target/product/sb35/recovery.img`
- `${SB35}/out/target/product/sb35/logo-verified.bin`
- `${SB35}/out/target/product/sb35/dtbo-verified.img`
- `${SB35}/out/target/product/sb35/tee-verified.img`
- `${SB35}/out/target/product/sb35/vbmeta.img`
- `${SB35}/out/target/product/sb35/vbmeta_system.img`
- `${SB35}/out/target/product/sb35/vbmeta_vendor.img`
- `${SB35}/out/target/product/sb35/super.img`
- `${SB35}/out/target/product/sb35/cache.img`
- `${SB35}/out/target/product/sb35/userdata.img`

4 Downloading SB35 image with Flash Tool

Please refer to SB35-Flash_Tool_Download_Guide.pdf for details.

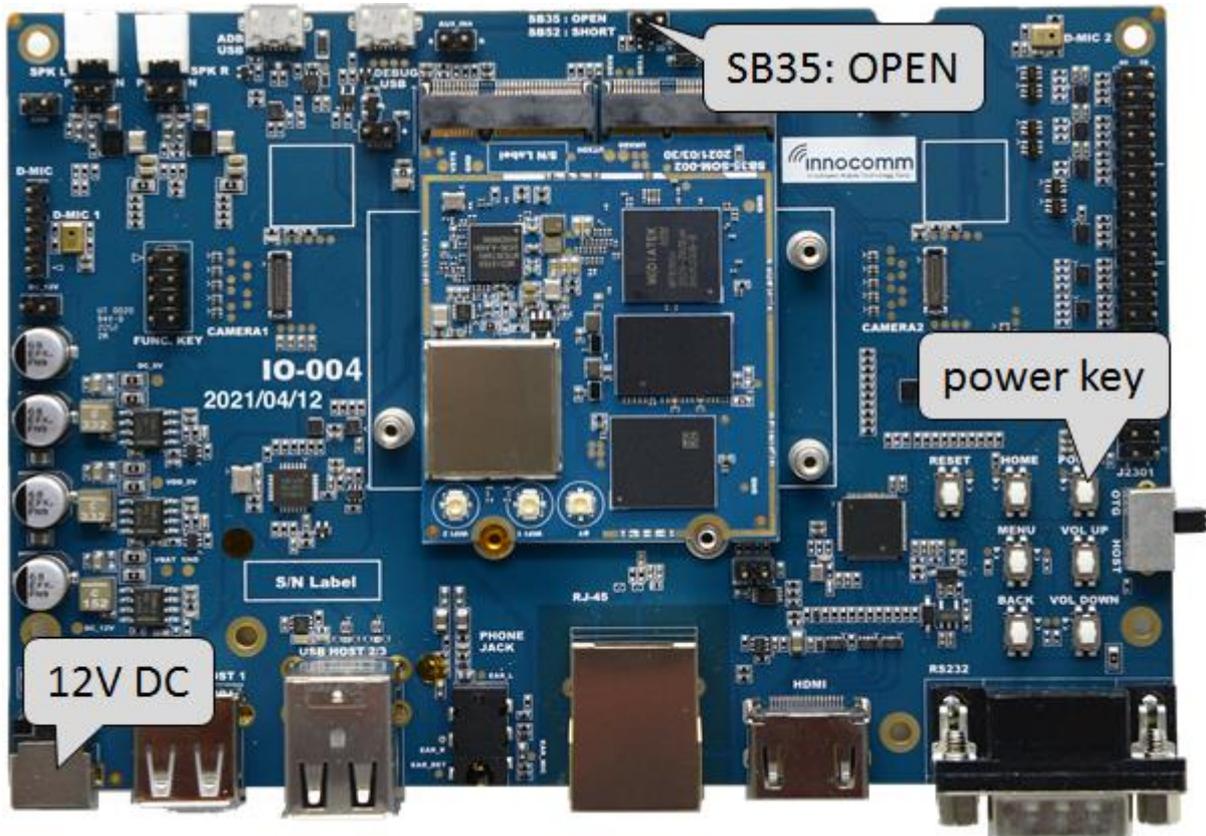


The screenshot shows the 'Smart Phone Flash Tool (Runtime Trace Mode)' interface. The 'Download' tab is active, and the 'Download-Agent' is set to '/media/wdssd/SP_Flash_Tool_exe_Linux_v5.2044.00.100/MTK_AllInOne_DA.bin'. The 'Scatter-loading File' is set to '/media/wdssd/SB35/out/target/product/sb35/MT8168_Android_scatter.txt'. The 'Authentication File' is empty. The 'Download Only' dropdown is set to 'All'. A table lists the files to be downloaded, including preloader, pgpt, cam_vpu1, cam_vpu2, cam_vpu3, lk, lk2, boot, recovery, logo, dtbo, tee1, tee2, vbmeta, vbmeta_system, vbmeta_vendor, super, cache, and userdata. The status bar at the bottom shows '0 B/s', '0 Bytes', 'EMMC High Speed', '0:00', and 'USB: DA Download All (high speed, auto detect)'.

Name	Begin Address	End Address	Region	Location
preloader	0x0000000000000000	0x0000000000376b7	EMMC_BOOT_1	/media/wdssd/SB35/out/target/product/sb35/preloader_sb35.bin
pgpt	0x0000000000000000	0x00000000000043ff	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/GPT.img
cam_vpu1	0x0000000000480000	0x0000000000592dbf	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/cam_vpu1-verified.img
cam_vpu2	0x0000000001380000	0x000000000170fd0f	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/cam_vpu2-verified.img
cam_vpu3	0x0000000002280000	0x000000000228ccbf	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/cam_vpu3-verified.img
lk	0x00000000082c0000	0x0000000008365d0f	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/lk-verified.img
lk2	0x00000000083c0000	0x0000000008465d0f	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/lk-verified.img
boot	0x00000000084c0000	0x000000000a4bffff	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/boot.img
recovery	0x000000000a4c0000	0x000000000c4bffff	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/recovery.img
logo	0x000000000c540000	0x000000000c70bc1f	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/logo-verified.bin
dtbo	0x000000000cd40000	0x000000000cd49aff	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/dtbo-verified.img
tee1	0x000000000e040000	0x000000000e206adf	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/tee-verified.img
tee2	0x000000000e540000	0x000000000e706adf	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/tee-verified.img
vbmeta	0x0000000011640000	0x0000000011640fff	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/vbmeta.img
vbmeta_system	0x0000000012180000	0x0000000012180fff	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/vbmeta_system.img
vbmeta_vendor	0x0000000012580000	0x0000000012580fff	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/vbmeta_vendor.img
super	0x0000000012980000	0x00000000179445b	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/super.img
cache	0x00000000cb380000	0x00000000cb38b04b	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/cache.img
userdata	0x00000000d2380000	0x00000000d239710b	EMMC_USER	/media/wdssd/SB35/out/target/product/sb35/userdata.img

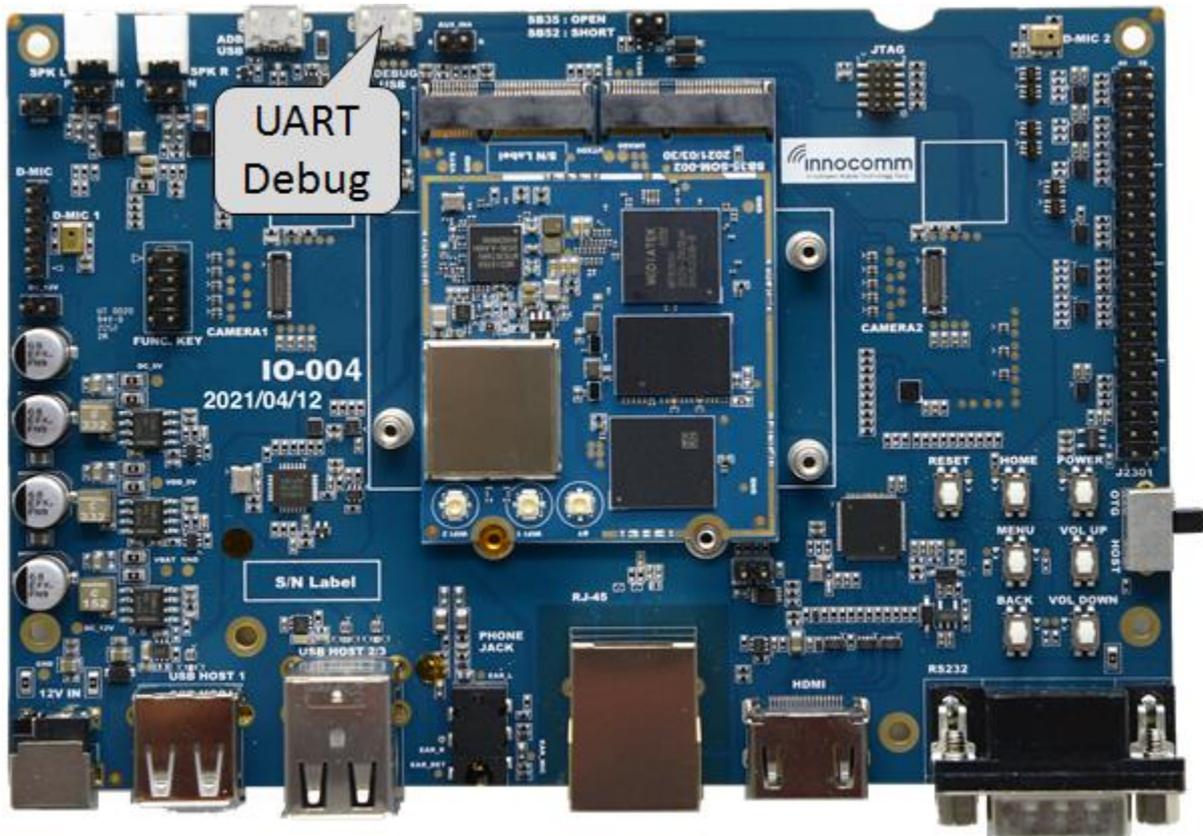
5 Booting SB35

SB35 development board is powered by 12V DC from DC Jack. Set SB35 jumper to OPEN. Long press power key to boot.



6 Serial console terminal

The serial console is a helpful tool for debugging your board and reviewing system log information. The console is the default output location for kernel log messages (i.e. dmesg), and it also provides access to a full shell prompt that you can use to access commands such as logcat.



Recommended tools for serial communication terminal:

- **Putty** for Windows.
- **Minicom** for Ubuntu. (\$ sudo apt-get install minicom)

Configure the serial port as follows:

- Baud rate: 921600
- Data bits: 8
- Parity: None
- Stop bits: 1
- Hardware Flow Control : No
- Software Flow Control : No

7 Clone project

You may want to create a new project for your hardware design. The script “project_clone.pl” can quickly clone a new project based on an existing project.

```
$ cd ${SB35}/vendor/mediatek/proprietary/scripts/project_clone
$ cat readme.txt
```

Usage:

```
[Common]
perl project_clone.pl -p {codebase_full_path} -o
{base_company/base_project} -n {new_company/new_project}
[Arguments]
-p : source codebase full path in local device.(usually is release
package uncompress path)
-o : base_company/base_project which the new project created from,
support cross company.
-n : new_company/new_project need to be created.
```

For examples:

Create new_project from "/home/lin/SB35/device/inoco/sb35".

```
$ perl project_clone.pl -p "/home/lin/SB35" -o "inoco/sb35" -n
"new_company/new_project"
```

8 Reference

- <https://source.android.com>
- SB35-Flash_Tool_Download_Guide.pdf