MT7981 Single Image SOP

2022/4/25

Document Revision History

Revision	Date	Author (Optional)	Description
0.1	2022-3-6	Micheal Su	Initial draft
1.0	2022-4-25	Jones Huang	Official release

Outline

- ☐ Generate SPIM-NAND Single Image
- ☐ Generate eMMC Single Image

Generate SPIM-NAND Single Image

Prepare File for SPIM-NAND Single Image

- **bl2.img**
  - **Please refer to MT7981_Build_SOP_xxx.pdf application note.**
- **fip.bin**
  - **Please refer to MT7981_Build_SOP_xxx.pdf application note.**
- **kernal_image**
  - **Please refer to MT7981_Build_SOP_xxx.pdf application note.**
- **mk_image.sh**
  - **In ATF folder, i.e. atf/tools/dev/single_img_wrapper/mk_image.sh**

*Note: You can also find the mk_image.sh in atf on MTK DCC center

How to Generate SPIM-NAND Single Image

- **Put all those files under the same folder,**
  - **bl2.img**
  - **fip.bin**
  - **kernal_image, e.g. openwrt-mediatek-mt7981-mt7981-spim-nand-rfb-squashfs-factory.bin**
  - **mk_image.sh**
- **Run mk_image.sh**
  - **CMD:~/#> ./mk_image.sh -p <CHIP Name> -d <Flash Type> -b <bl2.img> -f <fip.bin> -k <Kernel image>**
  - **For example:**
    CMD:~/#> ./mk_image.sh -p mt7981abd -d spim-nand -b bl2.img -f fip.bin -k openwrt-mediatek-mt7981-mt7981-spim-nand-rfb-squashfs-factory.bin

  Note: If you are using mt7981a/7981b/7981d then use **mt7981abd** as <CHIP Name>
- **The single image "mt7981-spim-nand-XXXX-single-image.bin" generated in the same folder.**

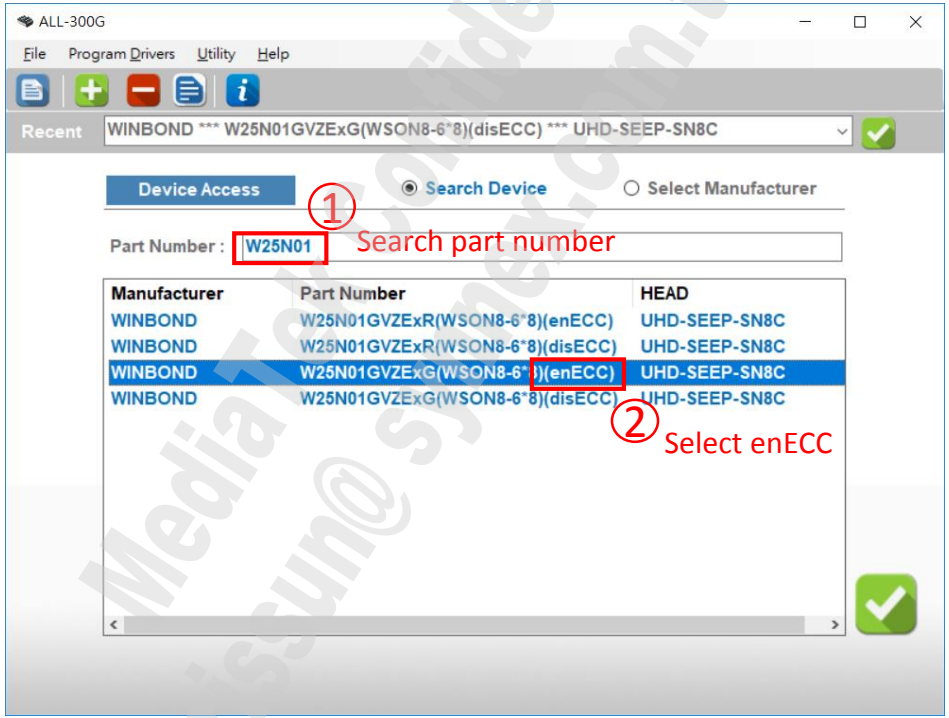
Customize partition config

- You can customize your own partition config in ./partitions
- Take spim-nand-default.yml for example:

```
1   # NAND flash layout:
2   #       0x0~0x100000 : BL2,          1024K
3   #  0x100000~0x180000 : Uboot env,    512K
4   #  0x180000~0x380000 : RF,          2048K
5   #  0x380000~0x1080000: FIP,           13M
6   #  0x1080000~        : firmware
7
8   spim-nand:
9     bl2_start: 0x0
10    rf_start: 0x180000
11    fip_start: 0x380000
12    kernel_start: 0x1080000
```

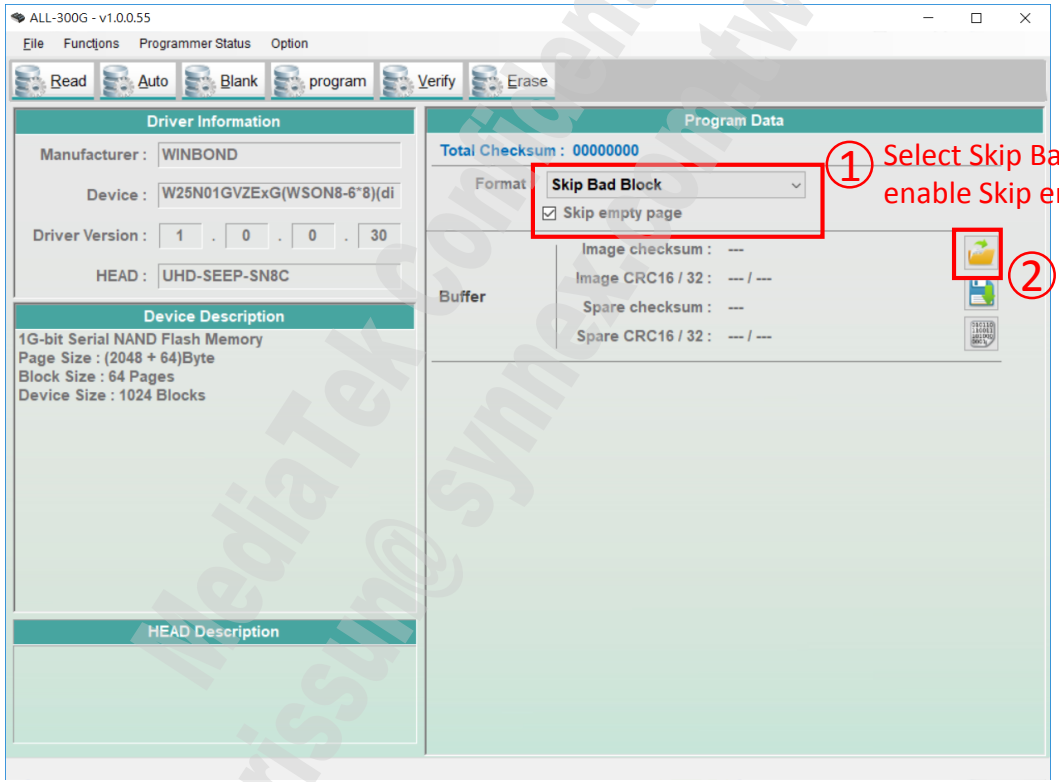
How to Program SPIM-NAND Single Image by Programmer
The example of ALL-300G programmer

- **Select SPI-NAND P/N (Winbond W25N01GVZEIG part as example)**



How to Program SPIM-NAND Single Image by Programmer
The example of ALL-300G programmer

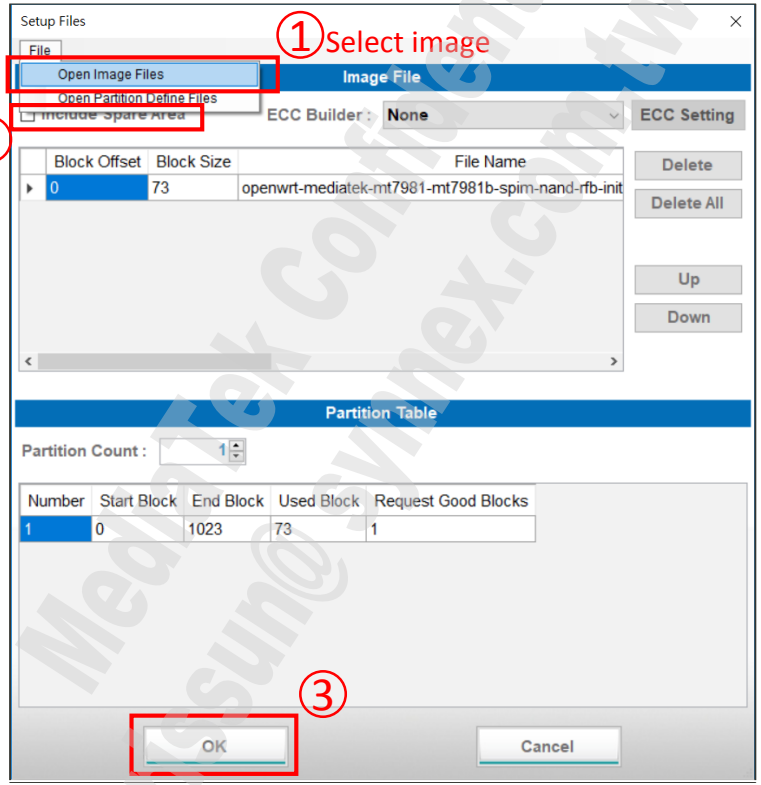
- **Load image**



How to Program SPIM-NAND Single Image by Programmer
The example of ALL-300G programmer

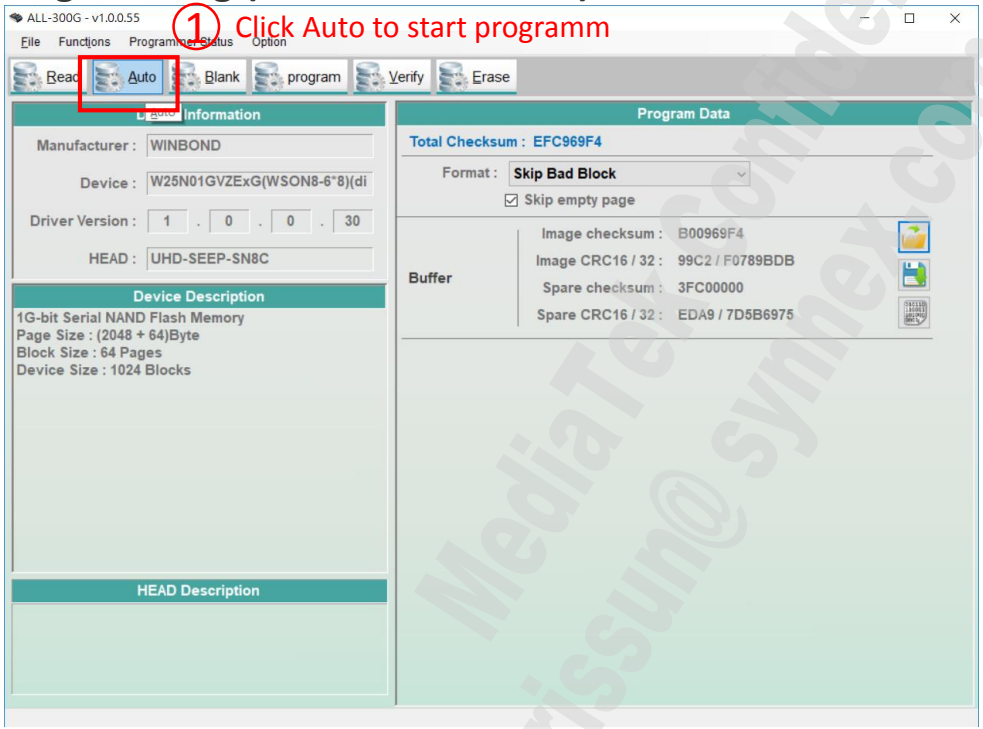
- **Load image**

Uncheck Include Spare Ares



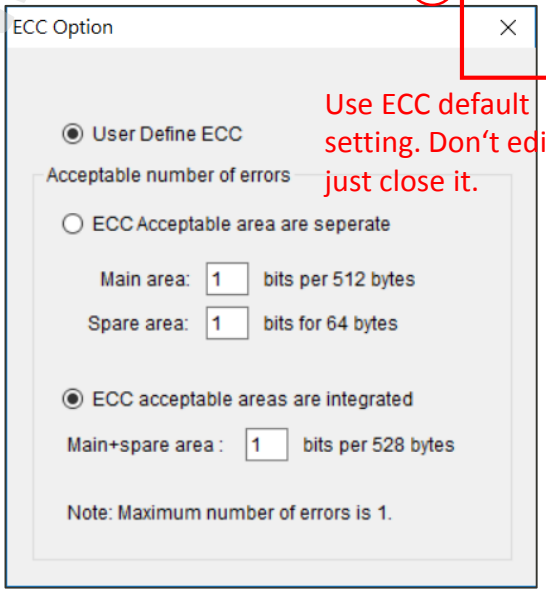
How to Program SPIM-NAND Single Image by Programmer
The example of ALL-300G programmer

- **Programming (check contact first)**



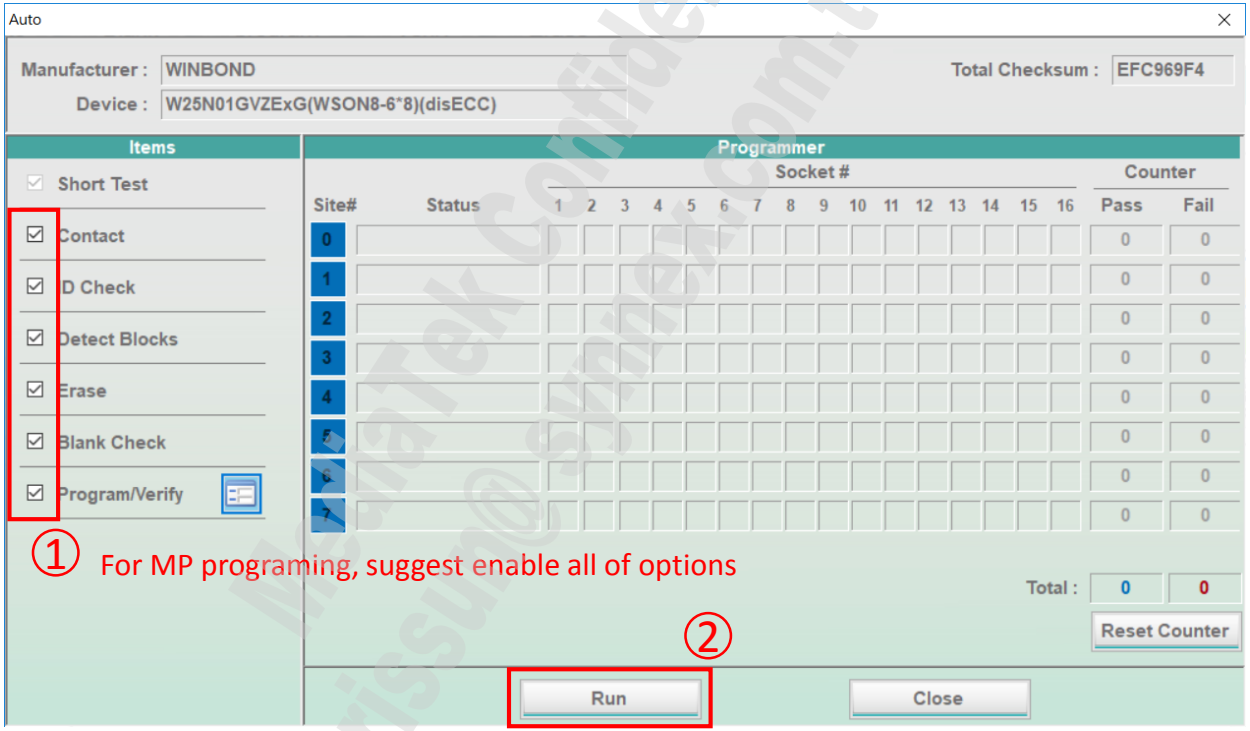
① Click Auto to start programm

② Use ECC default setting. Don't edit it, just close it.



How to Program SPIM-NAND Single Image by Programmer
The example of ALL-300G programmer

- **Programming (formal programming)**



Generate eMMC Single Image

eMMC Physical Partitions

- According to eMMC standard 5.1 section 6.2.1, eMMC devices have the following physical partitions
- In MTK's platforms, we use boot area partition 1 & user data area (UDA) only.
- BL2 is placed at boot partition 1, and the rest is at UDA.

6.2.1 General

The default area of the memory device consists of a User Data Area to store data, two possible boot area partitions for booting (see 6.3.2) and the Replay Protected Memory Block Area Partition (see 6.6.22) to manage data in an authenticated and replay protected manner. The memory configuration initially consists (before any partitioning operation) of the User Data Area and RPMB Area Partitions and Boot Area Partitions (whose dimensions and technology features are defined by the memory manufacturer).

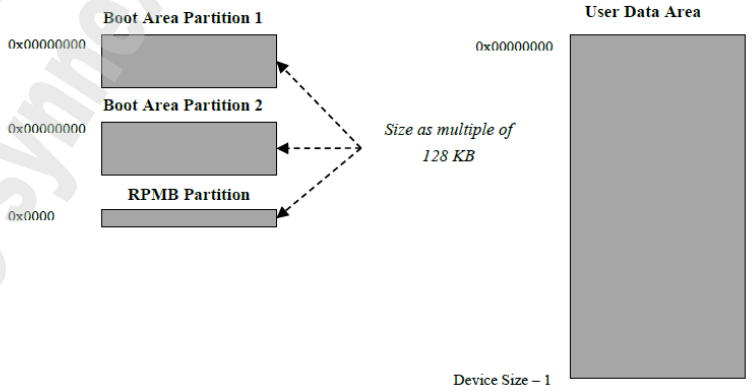
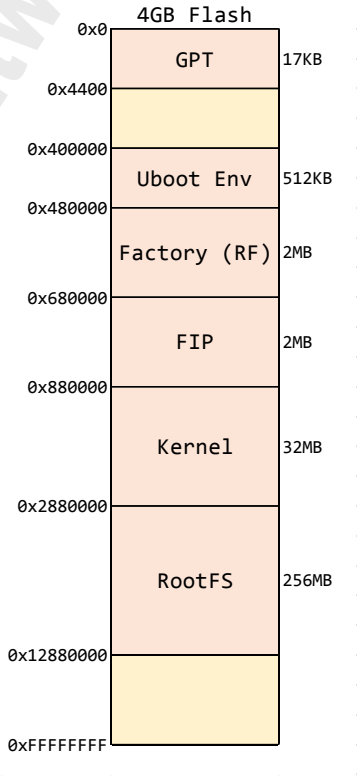


Figure 14 — *e*•MMC memory organization at time zero

MT7981 eMMC Partition Layout

- **Boot partition 1: BL2**
- **UDA:**
  The flash partition layout define at
  atf/tools/dev/gpt_editor/example/mt7981-emmc.json



MT7981 eMMC Partition Layout

- For Programmer need to prepare bl2.img for boot partition 1 and mt7981-eMMC-single-image.bin for UDA.
- mt7981-eMMC-single-image.bin is includes,
  - GPT (GPT_EMMC)
  - FIP (fip.bin)
  - firmware (kernel image, openwrt-mediatek-mt7981-xxxxxx.bin)

Prepare File for eMMC Single Image

- **bl2.img**
  - Please refer to MT7981_Build_SOP_xxx.pdf application note.
- **fip.bin**
  - Please refer to MT7981_Build_SOP_xxx.pdf application note.
- **kernal_image**
  - Please refer to MT7981_Build_SOP_xxx.pdf application note.
- **GPT_EMMC**
  - cd atf/tools/dev/gpt_editor
  - python mtk_gpt.py --i example/mt7981-emmc.json --o GPT_EMMC
- **mk_image.sh**
  - In ATF folder, i.e. atf/tools/dev/single_img_wrapper/mk_image.sh

*Note
If you want to customize partition layout, please modify the setting in 2 files:
- atf/tools/dev/gpt_editor/example/**mt7981-emmc.json**
- atf/tools/dev/single_img_wrapper/partitions/**emmc-default.yml**

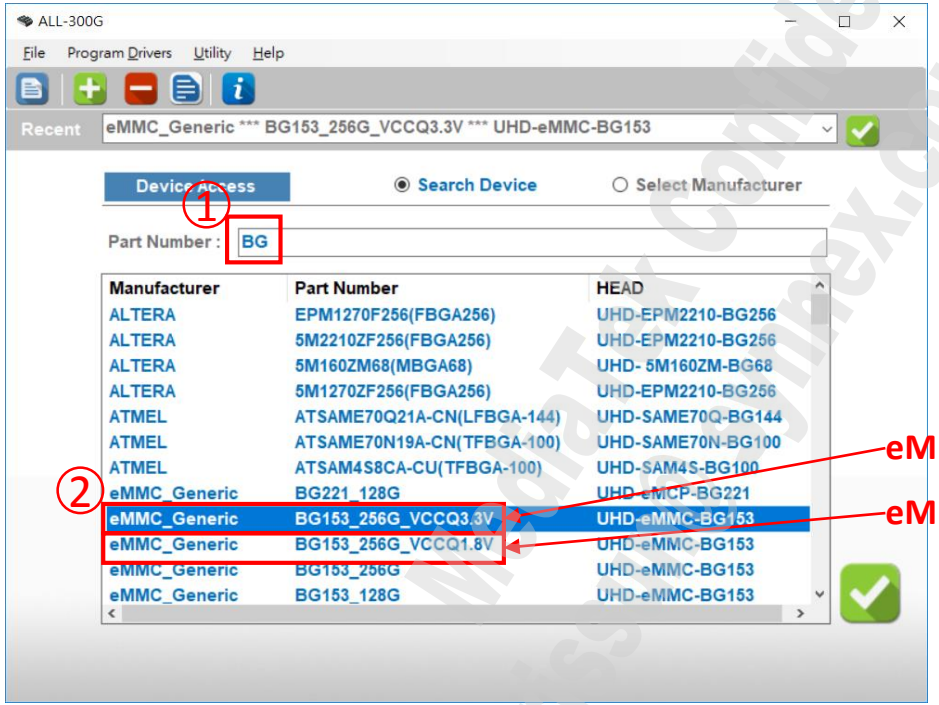
How to Generate eMMC Single Image

- **Put all those files under the same folder,**
  - **GPT_EMMC**
  - **fip.bin**
  - **kernal_image, e.g. OF_openwrt-mediatek-mt7981-mt7981-emmc-rfb-squashfs-sysupgrade-xxxx.bin**
  - **mk_image.sh**
- **Run mk_image.sh**
  - **CMD:~/#> ./mk_image.sh -p <CHIP Name> -d <Flash Type> -g <GPT table> -f <fip.bin> -k <Kernel image>**
  - **For example:
CMD:~/#> ./mk_image.sh -p mt7981abd -d emmc -g GPT_EMMC -f fip.bin -k OF_openwrt-mediatek-mt7981-mt7981-emmc-rfb-squashfs-sysupgrade-xxxx.bin**

  Note: If you are using mt7981a/7981b/7981d then use **mt7981abd** as <CHIP Name>
- **The single image "mt7981-eMMC-single-image.bin" generated in the same folder.**

How to Program Single Image by Programmer
The example of ALL-300G programmer

- **Select eMMC flash type,**

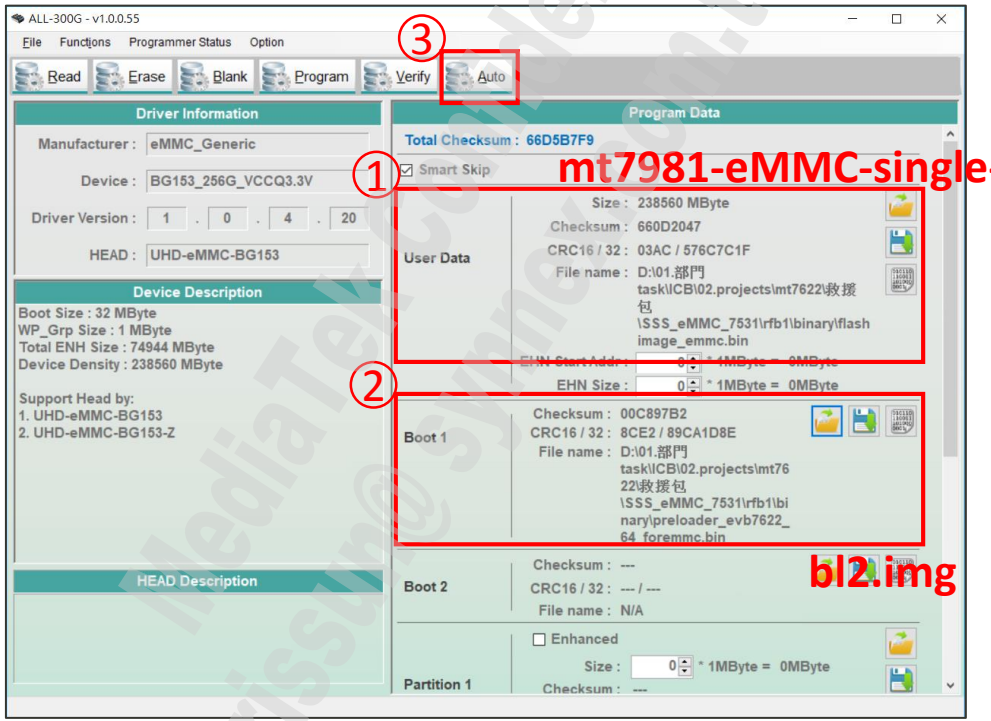


eMMC 4.5 → use BG153_256G_VCCQ3.3V

eMMC 5.1 → use BG153_256G_VCCQ1.8V

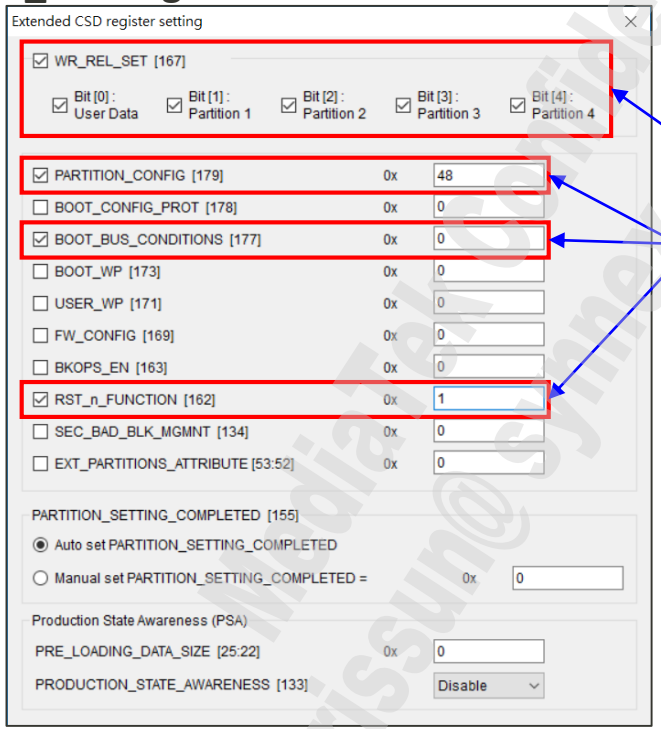
How to Program Single Image by Programmer
The example of ALL-300G programmer

- **Load image,**



How to Program Single Image by Programmer
The example of ALL-300G programmer

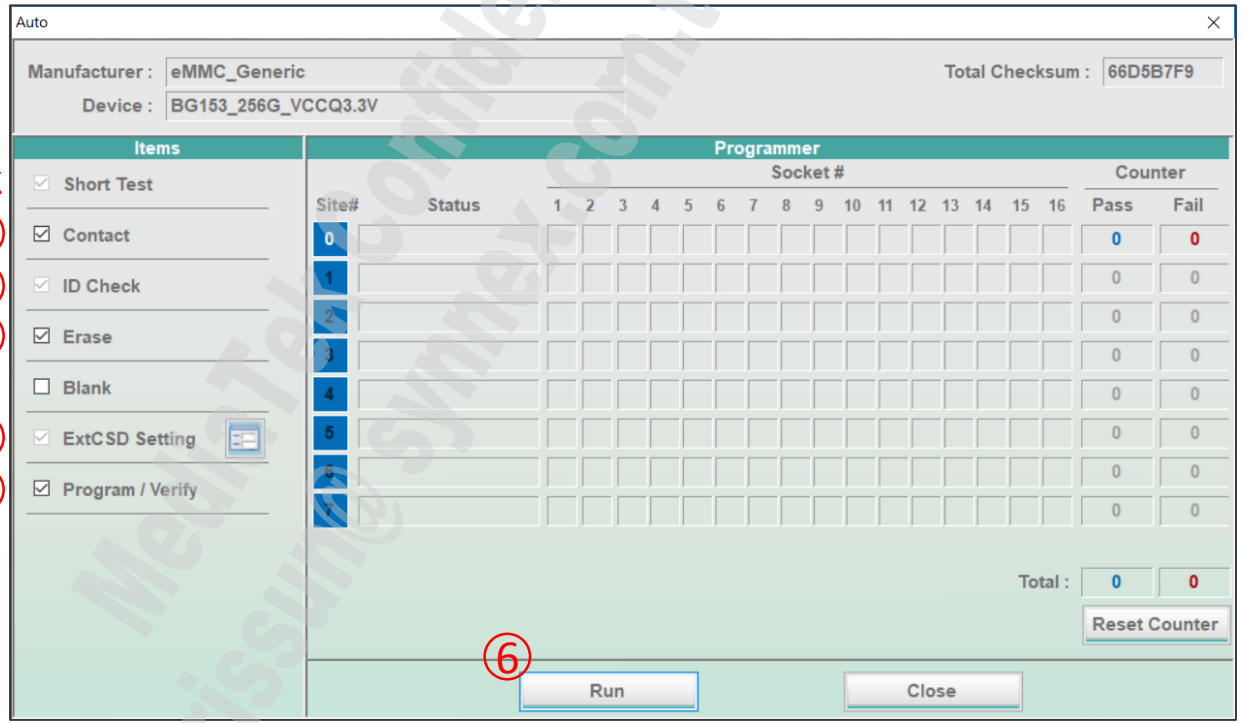
- **Configure EXT_CSD register**



- ext_csd[162] = 0x01
- ext_csd[167] = 0x1F
- ext_csd[177] = 0x00
- ext_csd[179] = 0x48

How to Program Single Image by Programmer
The example of ALL-300G programmer

- **Start program,**



MediaTek Proprietary and Confidential