

Forum:[Hack Forums](#)

Thread: [SM-G991B - Android 14 - FRP + OEM Locked](#)

DigitalZero - May 25, 2025, 4:50 am

Hi,

I've got an SM-G991B on Android 14 FW using the Exynos 2100 Chipset. Phone is FRP + OEM Locked.

I'm wondering how can I remove it? Any free tools/software?

DramaKing - May 25, 2025, 5:55 pm

(May 25, 2025 04:50 AM)DigitalZero Wrote: [\[CITE\]](#) Hi,

I've got an SM-G991B on Android 14 FW using the Exynos 2100 Chipset. Phone is FRP + OEM Locked.

I'm wondering how can I remove it? Any free tools/software?
Return it to the owner?

Infinity. - May 26, 2025, 11:30 pm

(May 25, 2025 05:55 PM)DramaKing Wrote: [\[CITE\]](#) Return it to the owner?
Sure, that's why he posted it here

frp locked: gacc protection is active.
oem locked: bootloader is locked, you can't flash custom binaries.

No fully free tools currently support Android 14 frp bypass on SM-G991B reliably without:
-Authorized Samsung Service access
-Downgrading firmware (but its blocked)

Old tools like:
samfirm, frp hijacker, samfw, or easy-samsung frp tool but these tools use to work on android 9-12?

It seems that you literally can't do anything op, unless you take a risk and do what I suggest you to.
(note, it will only get your frp lock removed)

-this is a hardware level recovery, follow it at your own risk

1. disassemble your phone
2. access the test point (TP) on the motherboard
3. connect a UART or JTAG box (e.g., Z3X, Octoplus, Medusa, Hydra Dongle, UFI Box)
4. short the test point while connecting USB to Force the phone into EDL/Qualcomm Download Mode (Exynos doesn't use EDL, so it may go into "Exynos BootROM mode" or "Upload mode")

Above steps don't actually unlock your frp it's just a gateway for you to do:

1. Bypass frp by flashing a special bootloader or patched firmware
2. Write back persist, efs, or misc partition to disable frp lock

how you do above is entirely upto you. Cheers

DigitalZero - May 30, 2025, 6:46 am

(May 26, 2025 11:30 PM)Infinity. Wrote: [\[CITE\]](#) Sure, that's why he posted it here

frp locked: gacc protection is active.

oem locked: bootloader is locked, you can't flash custom binaries.

No fully free tools currently support Android 14 frp bypass on SM-G991B reliably without:

- Authorized Samsung Service access
- Downgrading firmware (but its blocked)

Old tools like:

samfirm, frp hijacker, samfw, or easy-samsung frp tool but these tools use to work on android 9-12?

It seems that you literally can't do anything op, unless you take a risk and do what I suggest you to.
(note, it will only get your frp lock removed)

-this is a hardware level recovery, follow it at your own risk

1. disassemble your phone
2. access the test point (TP) on the motherboard
3. connect a UART or JTAG box (e.g., Z3X, Octoplus, Medusa, Hydra Dongle, UFI Box)
4. short the test point while connecting USB to Force the phone into EDL/Qualcomm Download Mode (Exynos doesn't use EDL, so it may go into "Exynos BootROM mode" or "Upload mode")

Above steps don't actually unlock your frp it's just a gateway for you to do:

1. Bypass frp by flashing a special bootloader or patched firmware
2. Write back persist, efs, or misc partition to disable frp lock

how you do above is entirely upto you. Cheers

I'm pretty fu*led, but I have actually looked into this as well. I've gone into Bootloader Mode and tried flashing a lower firmware but I get SW Rev Error which means i can not flash a binary firmware lower than the one that's on my phone... so i was thinking what about downgrading or spoofing the firmware binary on a lower stock firmware to look like it's actually a higher firmware to the phone.. but no one has released a working tool for this that i can find..

I have also tried booting into EDL mode but Exynos doesn't use EDL Mode.

I have also on purpose bricked the normal bootmode in hope that the phone will boot into a deeper level BootROM mode but that didn't work either. I have already erased the nand and reflashed a stock firmware with the same binary but that didn't erase the FRP or OEM Lock. I also can't flash any special bootloaders or patched firmware since the Bootloader only accepts official firmware with a binary higher than the one the phone is currently using.

Don't know what else i can try...

Infinity. - May 31, 2025, 9:14 pm

(May 30, 2025 06:46 AM)DigitalZero Wrote: [\[CITE\]](#) I'm pretty fu*led, but I have actually looked into this as well. I've gone into Bootloader Mode and tried flashing a lower firmware but I get SW Rev Error which means i can not flash a binary firmware lower than the one that's on my phone... so i was thinking what about downgrading or spoofing the firmware binary on a lower stock firmware to look like it's actually a higher firmware to the phone.. but no one has released a working tool for this

that i can find..

I have also tried booting into EDL mode but Exynos doesn't use EDL Mode.

I have also on purpose bricked the normal bootmode in hope that the phone will boot into a deeper level BootROM mode but that didn't work either. I have already erased the nand and reflashed a stock firmware with the same binary but that didn't erase the FRP or OEM Lock. I also can't flash any special bootloaders or patched firmware since the Bootloader only accepts official firmware with a binary higher than the one the phone is currently using.

Don't know what else i can try...
you're in deep Samsung hell. why?

Hell #1

You'd need to spoof the revision number inside the firmware, which:
-Requires unpacking + patching TAR/IMG files (boot.img, sboot.img, etc.)
-Is signed and verified by Samsung's bootloader (via AVB and secure boot)

Therefore, completely useless unless BL is already unlocked, which isn't the case

But hey, Spoofing firmware version doesn't work because bootloaders only accept firmware signed by Samsung's private key.

Hell #2

nand erase + same binary firmware flash doesn't remove frp:
Because frp is stored in persistent partitions (e.g., frp, misc, efs)
Knox(one of the components of samsung's security stack) also stores fuse information and bootloader lock state
Bootloader will always reject custom sboot or non-Samsung-signed kernel/bootloaders unless BL is unlocked (which it isn't)

WHAT MIGHT WORK: Let me elaborate on my first post (NOTE: This method is costlier than the cost of new Samsung S21?)

Hardware required:

Hot Air Rework Station - For safe disassembly and shielding removal.
Multimeter - To identify ground and trace test point pins.
USB to UART (3.3V) Adapter - Needed to interface with the device's BootROM/Serial debug.
Common chipsets: CP2102, FT232RL, CH340G.
Octopus Pro / Z3X Pro / Medusa / UFI Box - pro boxes to write to EMMC/NAND directly and handle Samsung partitions.
Samsung Test Point Diagram - shows exact short points on the motherboard.
Samsung USB Drivers + Samsung Tool Pro (software) - used for driver interfacing and port detection.

1. Disassemble the Device
Remove the back glass carefully (heat gun + suction cup).
Unscrew the shields and logic board.

Locate test point (use a diagram or probe with multimeter).

2: Connect UART

Now that you have your mobile's board exposed, Locate test point pads near USB-C area or under the EMI shield

And get the following hardware ready.

USB to UART Adapter (CP2102, FTDI FT232RL, CH340G 3.3V logic)

Soldering station or tweezers

Test Point Diagram for SM-G991B (Helps locate TX/RX/GND pins)

Samsung USB Drivers installed on PC and Multimeter for tracing/confirming GND lines

-- steps --Connect TX, RX, GND to phone UART points.Wire up your USB-UART adapter:

TX from adapter-> RX on phone

RX from adapter -> TX on phone

GND -> GND on phone

DO NOT connect 3.3V or 5V VCC unless explicitly required (not needed for Samsung).

Attach USB-to-UART adapter to PC.Plug the UART adapter into your PC.

Open Device Manager->confirm COM port appears under "Ports (COM & LPT)".

Open PuTTY, TeraTerm, or any UART terminal and set the following settingBaud rate: 115200 or 921600

Data bits: 8

Stop bits: 1

Parity bits: None

Flow control: None

Select your UART COM port and click "open"

You may see logs or a prompt like SBL> or Samsung Bootrom>.

3. Trigger Upload Mode or Crash Bootloader.

What's happening?

By shorting a test point or sending a UART signal at power-up, the phone will enter a BootROM handler called Upload Mode (on Exynos).

You bypass the bootloader stage, allowing tools like Octoplus, UFI, or Z3X to access partitions directly

Disconnect the battery.

Use tweezers to short the Test Point to GND.

While holding the short:Connect USB cable to PC.

If successful:

On Windows, you'll hear a device connect sound.

In Device Manager, it may appear as Exynos USB Device, Qualcomm HS-USB, or a diagnostic COM port.

Onwards:

Load up one of the below tools:

Z3X Samsung Tool Pro- Unlock, FRP reset, read EFS over UART.

Octoplus Box - FRP Remove, Read/Write EMMC, Boot Repair

UFI Box - Advanced EMMC dump + write

Hydra Dongle/Chimera - Used with UART for advanced unlock and partition writing

Run the following operations:Read FRP partition

Wipe FRP or reset flag
Optionally restore EFS from a clean dump
Example in UFI BOX
Open UFI eMMC ToolBox
Choose proper COM port or direct EMMC access
Click Identify eMMC/NAND
Navigate to Special Tasks or User Partition
Locate and select:
frp -> right-click -> Erase or Format
Optional: efs -> Backup or write clean dump
Click Write or Execute

You may see log lines like:

Code Resetting FRP...
Writing partition...
Rebooting to normal mode...

NOTE: if you make 1 mistake, you fry your board.

NOTE: if your IMEI/baseband is corrupted you need to write Clean EFS dump using "Write EMMC Image" or "fastboot flash"