

How to edit EBR1 and EBR2 to resize your /data partition in MTK6589 device. by agismaniax @xda-developers.com

Tools:

HxD hex editor

Windows 7 Calculator (programmer and standard mode)

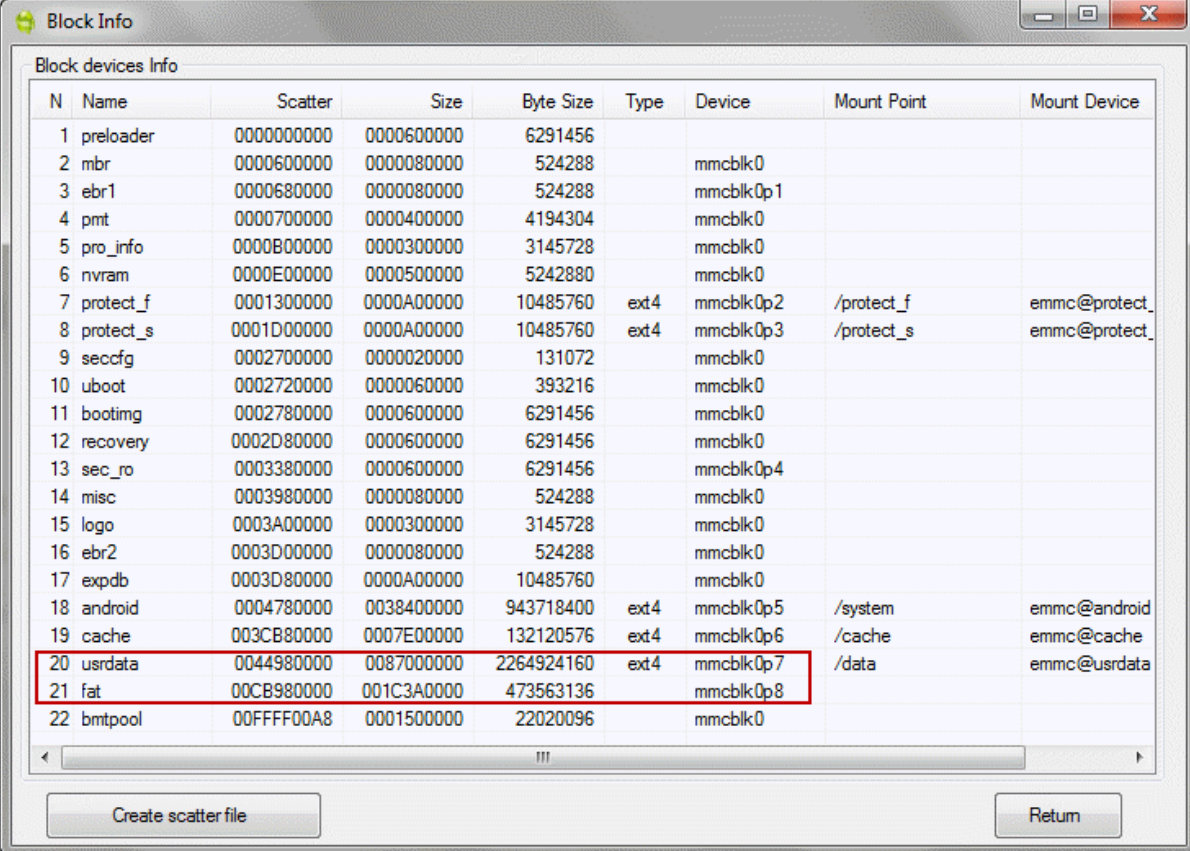
Notepad++

MTK Droid Tools (MDT)

SP Flash Tool

Byte Converter (<http://www.alterlinks.com/byte-converter/byte-converter.php>)

Check your device scatter file (MDT Block Info):



N	Name	Scatter	Size	Byte Size	Type	Device	Mount Point	Mount Device
1	preloader	000000000	000060000	6291456				
2	mbr	000060000	000008000	524288		mmcblk0		
3	ebr1	000068000	000008000	524288		mmcblk0p1		
4	pmt	000070000	000040000	4194304		mmcblk0		
5	pro_info	0000B0000	000030000	3145728		mmcblk0		
6	nvrnm	0000E0000	000050000	5242880		mmcblk0		
7	protect_f	000130000	0000A0000	10485760	ext4	mmcblk0p2	/protect_f	emmc@protect_
8	protect_s	0001D0000	0000A0000	10485760	ext4	mmcblk0p3	/protect_s	emmc@protect_
9	seccfg	000270000	000002000	131072		mmcblk0		
10	uboot	000272000	000006000	393216		mmcblk0		
11	bootimg	000278000	000060000	6291456		mmcblk0		
12	recovery	0002D8000	000060000	6291456		mmcblk0		
13	sec_ro	000338000	000060000	6291456		mmcblk0p4		
14	misc	000398000	000008000	524288		mmcblk0		
15	logo	0003A0000	000030000	3145728		mmcblk0		
16	ebr2	0003D0000	000008000	524288		mmcblk0		
17	expdb	0003D8000	0000A0000	10485760		mmcblk0		
18	android	000478000	003840000	943718400	ext4	mmcblk0p5	/system	emmc@android
19	cache	003CB8000	0007E0000	132120576	ext4	mmcblk0p6	/cache	emmc@cache
20	usrdata	004498000	008700000	2264924160	ext4	mmcblk0p7	/data	emmc@usrdata
21	fat	00CB98000	001C3A000	473563136		mmcblk0p8		
22	bmtpool	00FFF00A8	000150000	22020096		mmcblk0		

(a) USRDATA 0x44980000 (hex)

(b) FAT 0xCB980000 (hex)

Check and calculate existing size of mount point /data:

$Y = b - a = 0xCB980000 - 0x44980000$

$= 0x87000000$ (hex) = 2,264,924,160 bytes (hex to bytes conversion)

Open existing EBR1:

```

00000130 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000140 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000150 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000160 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000170 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000180 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000190 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001C0 00 00 83 00 00 00 00 08 02 00 00 20 1C 00 00 00 ..f.....
000001D0 00 00 83 00 00 00 00 28 1E 00 00 F0 03 00 00 00 ..f...{...8...
000001E0 00 00 83 00 00 00 00 18 22 00 00 80 43 00 00 00 ..f.....".eC...
000001F0 00 00 05 00 00 00 00 B4 01 00 FF FF FF FF 55 AA .....'.yyyyU*

```

A = 0x00221800 = 0x221800 (00 18 22 00)
B = 0x00438000 = 0x438000 (00 80 43 00)
C = 0x0001B400 = 0x1B400 (00 B4 01 00)
Max Value = 0xFFFFFFFF (FF FF FF FF)

Open existing EBR2:

```

00000130 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000140 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000150 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000160 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000170 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000180 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000190 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001C0 00 00 83 00 00 00 00 E4 63 00 FF 67 9A FF 00 00 ..f....äc.ÿgÿ..
000001D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 AA .....U*

```

D = 0x0063E400 = 0x63E400 (00 E4 63 00)
E = 0xFF9A67FF (FF 67 9A FF)

Calculate D and E to make sure:

$D = (A + B) - C = (0x221800 + 0x438000) - 0x1B400 = 0x63E400$
 $E = \text{Max Value} - (A + B) = 0xFFFFFFFF - (0x221800 + 0x438000) = 0xFF9A67FF$

Calculate target size (I use my device as example):

Target size (X) = 2,4GB = 2576980377.6 bytes
1 sector in NAND flash is 512 bytes.

$X = 2576980377.6 \text{ bytes} / 512 \text{ sector} = 5033164.8 \Rightarrow 5033180$ (rounded up)
Convert 5033180 (dec) to hex.
X = 4CCCCDC (hex) = 0x4CCCCDC (DC CC 4C 00)

Change EBR1 from this:

01E0 00 00 83 00 00 00 00 18 22 00 00 80 43 00 00 00

Into this:

01E0 00 00 83 00 00 00 00 18 22 00 DC CC 4C 00 00 00

```

00000140 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000150 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000160 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000170 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000180 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000190 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001C0 00 00 83 00 00 00 00 08 02 00 00 20 1C 00 00 00 ..f.....
000001D0 00 00 83 00 00 00 00 28 1E 00 00 F0 03 00 00 00 ..f....(...8....
000001E0 00 00 83 00 00 00 00 18 22 00 DC CC 4C 00 00 00 ..f.....".ÜIL...
000001F0 00 00 05 00 00 00 00 B4 01 00 FF FF FF FF 55 AA .....'.yyyyU*

```

Calculate new D and E:

$D = A + X + C = 0x221800 + 0x4CCDC + 0x1B400 = 0x7098DC$ (DC 98 70 00)
 $E = 0xFFFFFFFF - (A + X) = 0xFFFFFFFF - (0x221800 + 0x4CCDC) =$
 $0xFF911B23$ (23 1B 91 FF)

Change EBR2 from this:

```
01C0 00 00 83 00 00 00 00 E4 63 00 FF 67 9A FF 00 00
```

Into this:

```
01C0 00 00 83 00 00 00 DC 98 70 00 23 1B 91 FF 00 00
```

```

00000140 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000150 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000160 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000170 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000180 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000190 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001C0 00 00 83 00 00 00 DC 98 70 00 23 1B 91 FF 00 00 ..f...Ü~p.#.`y..
000001D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001F0 00 00 00 00 00 00 00 00 00 00 00 00 00 55 AA .....U*

```

Flash only new EBR1 and EBR2 using SP Flash Tools with your device scatter file above, or create CWM flashable zip to install new EBR.

Source: <http://www.mobile01.com/topicdetail.php?f=586&t=3468729&p=1>