

[English](#) • [Deutsch](#) • [Español](#) • [Français](#) • [Italiano](#) • [???](#) • [Polski](#) • [Português](#) • [??????](#) • [Svenska](#) • [???\(????\)?](#) • [???\(??\)?](#)

Contents

- [1 NVRAM](#)
- [2 Boardflags](#)
- [3 Overclocking](#)
- [4 Hardware Modifications](#)

NVRAM

NVRAM (non-volatile RAM) also called flash space is the place where the permanent settings are stored. This includes

- ◇ DD-WRT settings that you normally change using [Web Interface](#)
- ◇ settings for user [Startup Scripts](#)

You can run the following commands using [Command Line](#):

- Show the whole NVRAM content:

```
nvrnram show
```

- Show NVRAM content containing the pattern `<search_pattern>` (useful for quickly finding things in NVRAM):

```
nvrnram show | grep <search_pattern>
```

Replace `<replace_pattern>` with what you are actually looking for. This can simply be a word or a [regular expression](#)

- Show value of a certain variable:

```
nvrnram get <variable_name>
```

- Change variable value *in RAM only*:

```
nvrnram set <variable_name>="<value>"
```

(Quotes not needed for numeric/boolean values, needed for texts like `192.168.1.1` or `443 8080` or `sometext`).

Now you can play with the new settings or test user scripts but this new value will be lost after reboot unless you do `nvrnram commit`.

- Delete a variable (and its value). Useful when you made a typo in `<variable_name>` above:

```
nvrnram unset <variable_name>
```

Hardware

- Save all changed variables to NVRAM:

```
nvrnm commit
```

Reboot the router for new settings in NVRAM to go into effect.

- To erase the nvrnm block when running DD-WRT:

```
erase nvrnm
```

- To erase nvrnm in the Broadcom CFE (bootloader):

```
nvrnm erase
```

- These are often confused!

Boardflags

```
/* boardflags */
```

1. define BFL_BTCEXIST 0x0001 /* This board implements Bluetooth coexistence */
2. define BFL_PACTRL 0x0002 /* This board has gpio 9 controlling the PA */
3. define BFL_AIRLINEMODE 0x0004 /* This board implements gpio13 radio disable indication */
4. define BFL_ENETROBO 0x0010 /* This board has robo switch or core */
5. define BFL_CCKHIPWR 0x0040 /* Can do high-power CCK transmission */
6. define BFL_ENETADM 0x0080 /* This board has ADMtek switch */
7. define BFL_ENETVLAN 0x0100 /* This board has vlan capability */
8. define BFL_AFTERBURNER 0x0200 /* This board supports Afterburner mode */
9. define BFL_NOPCI 0x0400 /* This board leaves PCI floating */
10. define BFL_FEM 0x0800 /* This board supports the Front End Module */
11. define BFL_EXTLNA 0x1000 /* This board has an external LNA */
12. define BFL_HGPA 0x2000 /* This board has a high gain PA */
13. define BFL_BTCMOD 0x4000 /* This board' BTCEXIST is in the alternate gpios */

Overclocking

BCM3303 v0.7 clock frequencies: (todo...)

MIPS/CPU	Backplane	Comment
----------	-----------	---------

BCM3303 v0.8 clock frequencies:

MIPS/CPU	Backplane	Comment
183	91	

Hardware

188	94
197	98
200	100
206	103
212	106
216	108
217	109
225	113
238	119
240	120
250	125

Hardware Modifications

Hardware Modifications (mods) are not necessarily DD-WRT specific or compatible.

- Serial Port
- Additional external SMA-R connector
- Dual Serial Port
- SD-Card
- Temperature sensor for WRT54G 2.x and WRT54GS with DS1820
- DIR-320 64 Mb RAM Upgrade