

RIoTboard Boot Switches

The useful combinations of boot switches on the RIoTboard are as follows:

| Switch | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 |
|-----------------|----|----|----|----|----|----|----|----|
| Serial Download | 0 | 1 | x | x | x | x | x | x |
| SD (J6, bottom) | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 |
| uSD (J7, top) | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| eMMC | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 |

D4 basically enables some extra eMMC related things for port 4, otherwise it's a simple case of D7 & D8 selecting which port to boot from.

D6 could be used to force the port into single bit mode, but this doesn't seem useful unless you have a troublesome card as it'll slow down access to the card considerably.

Something worth noting is that with D1=1, D2=0 for Internal Boot mode, there's a fuse setting in the SoC, BT_FUSE_SEL, which if inadvertently set will cause the SoC to ignore the other switches and boot according to the eFUSE settings. As it's possible to blow the fuses with the MFG tools, it's possible, if unlikely, that you can mess up your board.

Lastly, the SoC has a feature called SD/MMC Manufacture Mode. Basically this only happens when the configured boot device fails and it causes the SoC to first look at USDHC-1 & USDHC-2 for an SD card with a valid boot image, and if none is found it'll finally fall back to Serial Download mode. So USDHC-1 isn't implemented on the board, USDHC-2 is the full size SD slot at J6 on the bottom of the board. So it's possible to have the switches configured for eMMC boot, but the eMMC either blank or corrupted and find that the board will boot from a card in J6.

The real detail of how this all works can be found in Chapter 8 of the i.MX 6Solo/6DualLite Applications Processor Reference Manual.

Download IMX6SLDRM.pdf from Freescale [here](#). The doc is just short of 5800 pages long so it should keep you in bedtime reading for a while ;-)

Detailed Boot Switch Table

| | Switch | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 |
|-----------------|---|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|----------------|-----------------|
| | Boot Option | BOOT MODE[1] | BOOT MODE[0] | BOOT CFG1[6] | BOOT CFG1[5] | BOOT CFG2[6] | BOOT CFG2[5] | BOOT CFG[4] | BOOT CFG2[3] |
| Boot from Fuses | | 0 | 0 | x | x | x | x | x | x |
| Serial Download | | 0 | 1 | x | x | x | x | x | x |
| Reserved | | 1 | 1 | ? | ? | ? | ? | ? | ? |
| Internal Boot | | 1 | 0 | | | | | | |
| SD | | | | 1 | 0 | | | | |
| | don't care due to BOOT_CFG2[7]=0 | | | | | x | | | |
| | 1 bit | | | | | | 0 | | |
| | 4 bit | | | | | | 1 | | |
| eMMC | | | | 1 | 1 | | | | |
| | 1 bit | | | | | 0 | 0 | | |
| | 4 bit | | | | | 0 | 1 | | |
| | 8 bit, not implemented on pcb | | | | | 1 | 0 | | |
| USDHC-1 | not implemented | | | | | | | 0 | 0 |
| USDHC-2 | J6 full size SD slot on bottom of board | | | | | | | 0 | 1 |
| USDHC-3 | J7 uSD slot on top of board | | | | | | | 1 | 0 |
| USDHC-4 | eMMC | | | | | | | 1 | 1 |

Now some combinations don't make a lot of sense, for example you can configure the board to boot from interfaces that have not been implemented on the PCB, and switch D5 is essentially useless due to that. Other combinations that could be useful are ruled out due to all of the remaining BOOT_CFG1-4 values being defaulted to zero as can be seen on the left hand side of p18 of the schematics.

There are other complexities, while the SoC has a total of four SD/MMC/eSD/eMMC/SDXC capable ports, they're not all created equal. Some are more suited to eMMC devices. With that knowledge it should be simple to conclude that configuring eMMC access on a port that's not suited to that use probably isn't worthwhile.