An overview of the linux SPI NOR subsystem

Tudor Ambarus, Pratyush Yadav, Michael Walle

Linux SPI NOR team

- Tudor Ambarus, co-maintainer
  - Embedded linux development: bootloaders, kernel development, board bring-up.
  - Living in Bucharest, Romania
  - contractor
- Pratyush Yadav, co-maintainer
  - Living in Dresden, Germany
- Michael Walle, reviewer
  - Living in Saarbruecken, Germany
  - Working at Kontron mostly on upstreaming the ARM boards

What is this talk about?

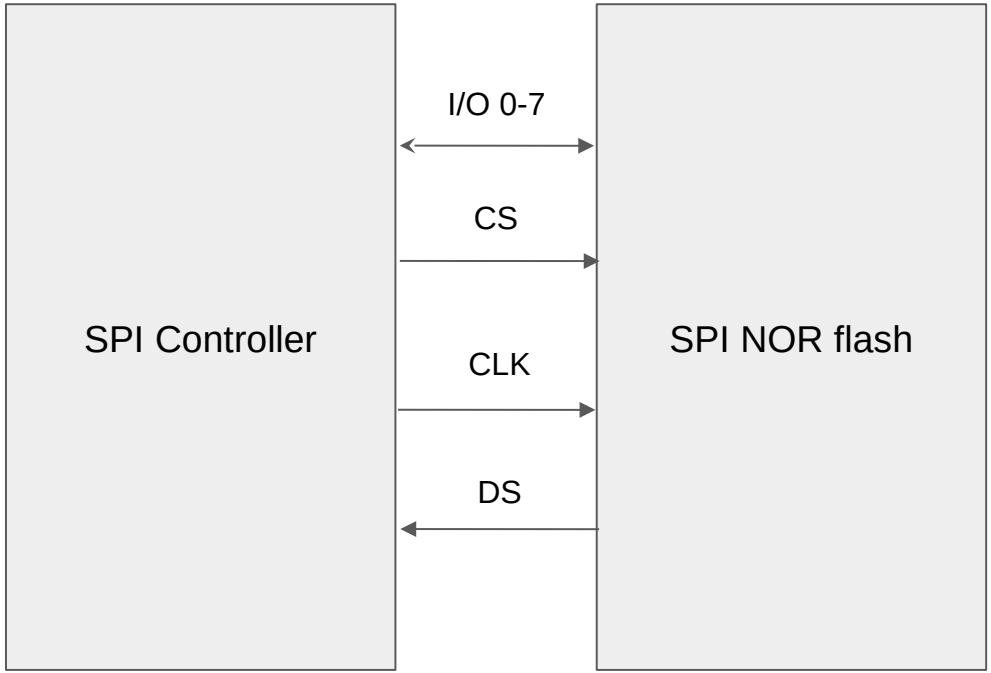
- Overview of the linux kernel SPI NOR subsystem (what happened in the last few years)
- SPI NOR parameters initialization
- How to update or propose new flash additions
- Challenges and future development

An overview of the linux SPI NOR subsystem

Overall architecture

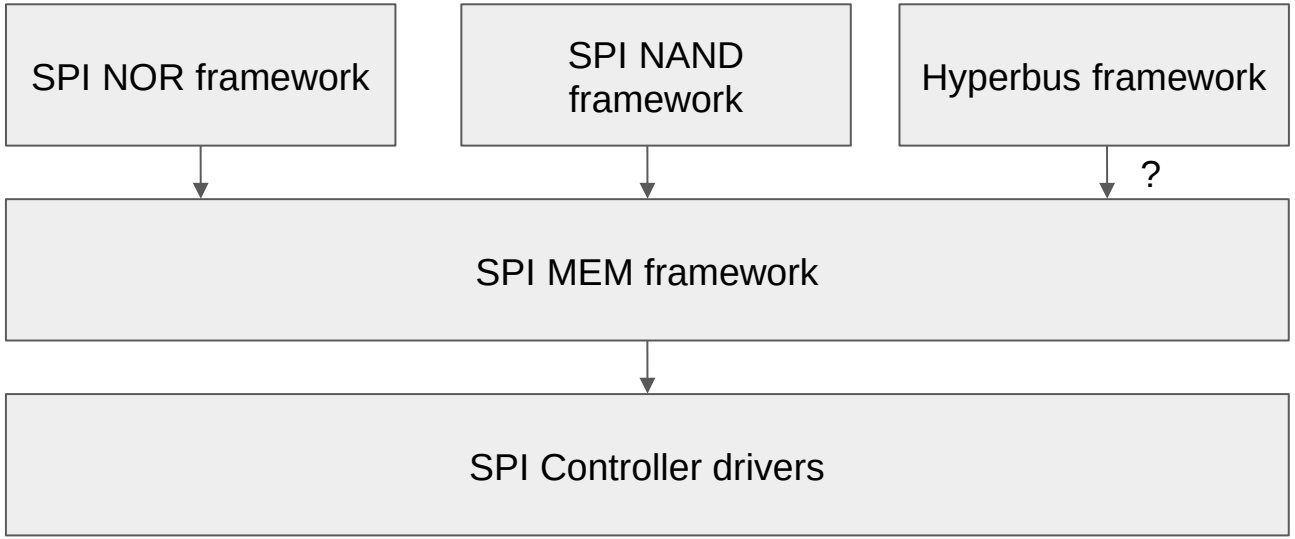
Linux SPI NOR: Overall architecture

SPI NOR introduction



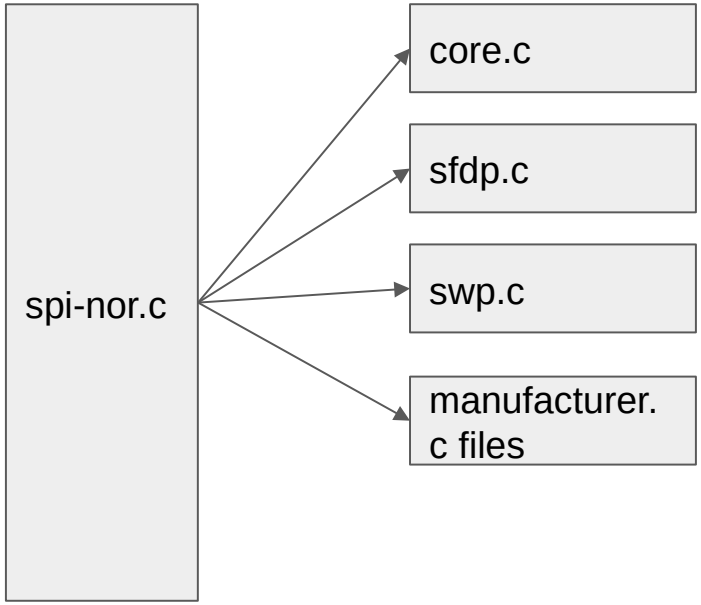
Linux SPI NOR: Overall architecture

SPI NOR introduction



Linux SPI NOR: Overall architecture

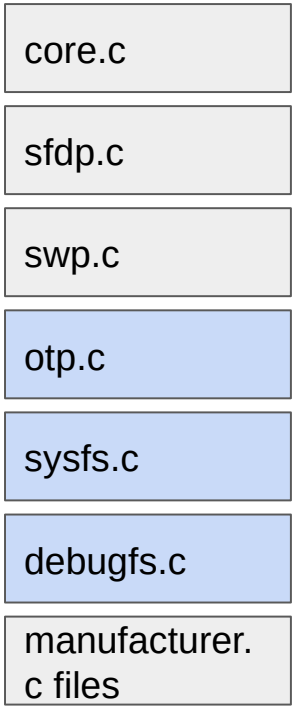
Splitting SPI NOR



Linux SPI NOR: Overall architecture

Adding new functionalities

linux/mtd/spi-nor/



- SWP - Software Write Protection
- Memory array splitted in “protection blocks”
- Top/bottom, complement array protection
- Will be extended with individual block/sector array protection.

winbond

7.1.14 **Status Register Memory Protection (WPS = 0, CMP = 0)**

STATUS REGISTER(1)					W25Q32JW (32M-BIT) MEMORY PROTECTION(3)			
SEC	TB	BP2	BP1	BP0	PROTECTED BLOCK(S)	PROTECTED ADDRESSES	PROTECTED DENSITY	PROTECTED PORTION(2)
X	X	0	0	0	NONE	NONE	NONE	NONE
0	0	0	0	1	63	3F0000h – 3FFFFFh	64KB	Upper 1/64
0	0	0	1	0	62 and 63	3E0000h – 3FFFFFh	128KB	Upper 1/32
0	0	0	1	1	60 thru 63	3C0000h – 3FFFFFh	256KB	Upper 1/16
0	0	1	0	0	56 thru 63	380000h – 3FFFFFh	512KB	Upper 1/8
0	0	1	0	1	48 thru 63	300000h – 3FFFFFh	1MB	Upper 1/4
0	0	1	1	0	32 thru 63	200000h – 3FFFFFh	2MB	Upper 1/2

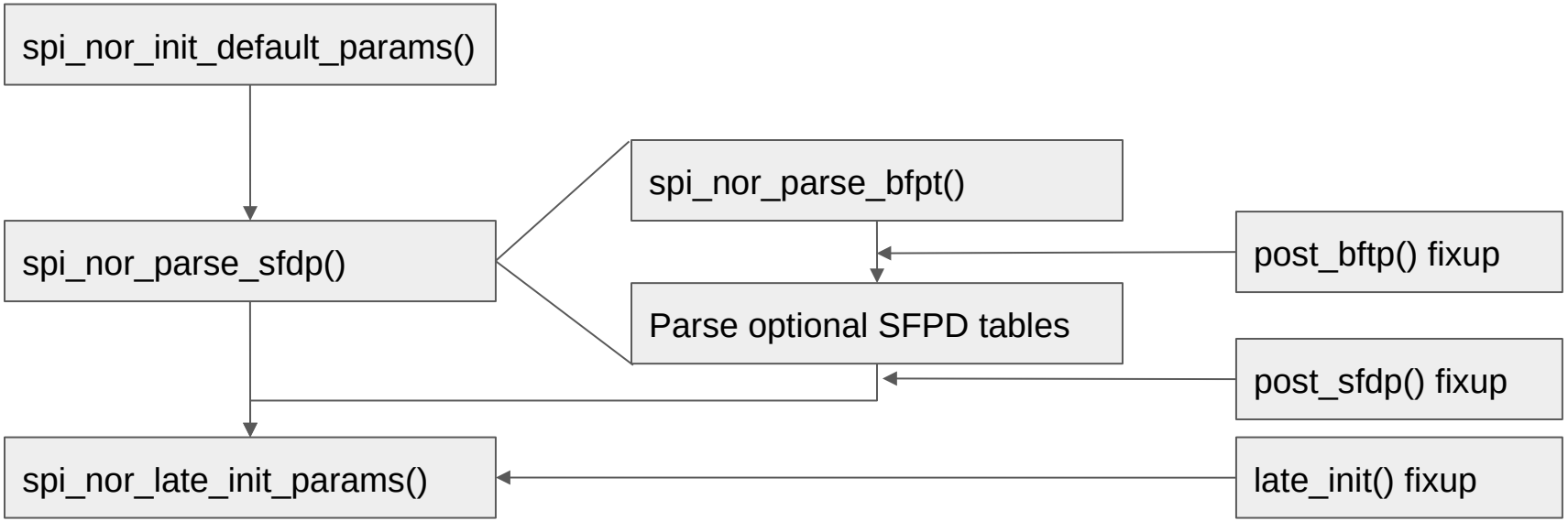
An overview of the linux SPI NOR subsystem

Flash parameters initialization

Linux SPI NOR: Flash parameters initialization

Parameters update sequence

linux/mtd/spi-nor/core.c



An overview of the linux SPI NOR subsystem

Requirements for flash updates or new flash proposals

Example of a flash update:

```
        { "gd25q256", INFO(0xc84019, 0, 64 * 1024, 512)
+               PARSE_SFDP
                FLAGS(SPI_NOR_HAS_LOCK | SPI_NOR_HAS_TB | SPI_NOR_TB_SR_BIT6)
-               NO_SFDP_FLAGS(SECT_4K | SPI_NOR_DUAL_READ | SPI_NOR_QUAD_READ)
                FIXUP_FLAGS(SPI_NOR_4B_OPCODES)
                .fixups = &gd25q256_fixups },
```

Linux SPI NOR: Requirements for flash updates/additions

Minimum testing requirement

- Dump the sysfs jedec_id, manufacturer, partname and sfdp tables
- Check the correctness of erases, writes and reads with mtd_utils

An overview of the linux SPI NOR subsystem

Challenges and future development

- Use SFDP tables to differentiate the flashes, where possible
- Where run-time differentiation is not possible, use DT to differentiate between them (used as a last resort)

- Some manufacturers swap the bytes on a 16-bit boundary when flashes are configured in Octal DTR mode.
- Bad design decision -> endianness problems. It can affect the boot sequence if the entire boot sequence is not handled in the same mode
- Some controllers can swap the bytes back on the fly

- Different kind of memories with their peculiarities
- Some FRAMs, MRAMs slip in the SPI NOR driver
- Make a generic driver with hooks to handle all

- Needed for high clock speeds in 8D-8D-8D mode
- Read predetermined pattern data from the flash and runs a sequence of test reads to find the optimal delays
- Nvm cells which point to a fixed partition containing the data to be read. Partitions defined ba command line?

- Detecting the flash mode when the flash boots in anything other than 1-1-1 mode (or the flash does not support 1-1-1 mode at all)

- License for the vendor SFDP tables
- To introduce the SFDP decoder into mtd-utils

- Flash parameters and settings solely discovered by parsing without static parameters declaration.

Questions? Suggestions?

Tudor Ambarus, Pratyush Yadav, Michael Walle