Nand Flash Support Guide

Guidelines to check Nand flash compatibility with Zynq

Introduction:

This document can be divided into two parts:

1. Checklist to see if the selected flash is compatible with Zynq BootROM and PS SMC or not.
2. Guidelines for editing u-boot to support on-board programming for compatible flashes that are not listed under (Xilinx_Answer_50991).

Checklist for checking flash compatibility:

BootROM reads the ONFI complaint parameter in 8-bit mode to determine the device width. Any width other than 8-bit and 16-bit is not supported with BootROM.

BootROM uses the following NAND timing values in the smc.SET_CYCLES register:
• t_rr = 2, t_ar = 2, t_clr = 1, t_wp = 3, t_rea = 2, t_wc = 5, t_rc = 5

For more details, refer to “Boot Time Optimizations” under “6.3.5 NAND Boot”

Unlike QSPI, there are not many restrictions on Nand flash compatibility with BootROM. It expects the BootROM header to be present within the first 128MB address range.

The major compatibility restrictions arises because of the SMC controller (NAND flash controller). Please refer to the 11.1.1 Features section in (UG585) for the features of the SMC controller.

Here is the checklist for flash compatibility with Zynq:

- ✓ The page size has to be between 512 & 2048.
- ✓ Make sure that the flash has either on-Die ECC or 1-bit ECC. Anything other than this is not compatible with Zynq.
- ✓ The I/O width should only be 8-bits or 16-bits.
- ✓ SMC controller supports flashes with one CS bit.
- ✓ Check if the flash is ONFI specification 1.0 compatible.
- ✓ Other Hardware considerations are:
    - I/O Supply Voltage & Electrical Specifications – These should be within the operational limits of MIO pins-specified in DS187 for 7z010 &7z020 devices and DS191 for 7z030 & 7z045 devices.
    - Timing Specifications - Compare setup/hold timing requirements of Zynq datasheets with that of the flash datasheet. You should also perform IBIS timing simulation while designing the board.

Guidelines for editing u-boot to support on-board programming for compatible flashes

U-boot can access the NAND flash and perform read, erase, and write operations only if the flash is listed in the nand.h file.

The below function nand_scan_ident, identifies the nand flash.

```
28  struct mtd_info;
29  struct nand_flash_dev;
30  /* Scan and identify a NAND device */
31  extern int nand_scan (struct mtd_info *mtd, int max_chips);
32  /* Separate phases of nand_scan(), allowing board driver to intervene
33   * and override command or ECC setup according to flash type */
34  extern int nand_scan_ident(struct mtd_info *mtd, int max_chips,
35                             const struct nand_flash_dev *table);
36  extern int nand_scan_tail(struct mtd_info *mtd);
37
```

This function is called in the zynq_nand.c file and if the flash is not recognized, it returns an error:

```
    /* first scan to find the device and get the page size */
    if (nand_scan_ident(mtd, 1, NULL)) {
        printf("%s: nand_scan_ident failed\n", __func__);
        goto fail;
    }
```

To add a new flash, all you need to do is add the flash manufacture ID under the manufacture ID section in the nand.h file:

```
566 /*
567  * NAND Flash Manufacturer ID Codes
568  */
569 #define NAND_MFR_TOSHIBA    0x98
570 #define NAND_MFR_SAMSUNG    0xec
571 #define NAND_MFR_FUJITSU    0x04
572 #define NAND_MFR_NATIONAL   0x8f
573 #define NAND_MFR_RENESAS    0x07
574 #define NAND_MFR_STMICRO    0x20
575 #define NAND_MFR_HYNIX      0xad
576 #define NAND_MFR_MICRON     0x2c
577 #define NAND_MFR_AMD        0x01
578 #define NAND_MFR_MACRONIX   0xc2
579 #define NAND_MFR_EON        0x92
580
581 /**
582  * struct nand_flash_dev - NAND Flash Device ID Structure
583  * @name:   Identify the device type
584  * @id:     device ID code
585  * @pagesize:   Pagesize in bytes. Either 256 or 512 or 0
586  *      If the pagesize is 0, then the real pagesize
587  *      and the eraseize are determined from the
588  *      extended id bytes in the chip
589  * @erasesize:  Size of an erase block in the flash device.
590  * @chipsize:   Total chipsize in Mega Bytes
591  * @options:    Bitfield to store chip relevant options
592  */
593 struct nand_flash_dev {
594     char *name;
595     int id;
596     unsigned long pagesize;
597     unsigned long chipsize;
598     unsigned long erasesize;
599     unsigned long options;
600 };
601
```

NOTE: Xilinx recommends that you avoid flashes not included in the manufacture ID list in nand.h

Examples:

Let us consider the following flashes and see if they are compatible with Zynq or not:

Note: Conclusion was highlighted in green for compatible flashes and in red for incompatible.

1. S34ML04G2:
   Checklist:

✓ The page size has to be between 512 & 2048 bits.
Note: The page size of this flash is either 1024 or 2048 based on I/O width (1024 bits for x16 and 2048 bits for x8). So this criterion is met.

✓ Make sure that the flash has either **on-Die ECC or 1-bit ECC**. Anything other than this is not compatible with Zynq.
Note: The following flash has 4-bit ECC. The SMC controller is not compatible with flashes that have 4-bit ECC, so, this flash cannot be used with Zynq.

Performance

- **Page Read / Program**
  - Random access: 25 µs (Max) **(S34ML01G2)**
  - Random access: 30 µs (Max) **(S34ML02G2, S34ML04G2)**
  - Sequential access: 25 ns (Min)
  - Program time / Multiplane Program time: 300 µs (Typ)
- **Block Erase (S34ML01G2)**
  - Block Erase time: 3 ms (Typ)
- **Block Erase / Multiplane Erase (S34ML02G2, S34ML04G2)**
  - Block Erase time: 3.5 ms (Typ)
- **Reliability**
  - 100,000 Program / Erase cycles (Typ) (with 4-bit ECC per 528 bytes (x8) or 264 words (x16))
  - 10 Year Data retention (Typ)
  - Blocks zero and one are valid and will be valid for at least 1000 program-erase cycles with ECC
- **Package Options**
  - Lead Free and Low Halogen
  - 48-Pin TSOP 12 x 20 x 1.2 mm
  - 63-Ball BGA 9 x 11 x 1 mm
  - 67-Ball BGA 8 x 6.5 x 1 mm **(S34ML01G2)**

✓ The I/O width should only be 8-bits or 16-bits.
- This flash supports both x8 and x16 I/O widths.

✓ SMC controller supports flashes with one CS bit.
- This flash has only one-bit chip enable (CS).

✓ Check if the flash is ONFI specification 1.0 compatible.
- S34ML04G2 flash supports ONFI specification 1.0.

Conclusion: S34ML04G2 meets all of the points in the checklist except for the ECC requirement. So, this flash can be termed as incompatible with Zynq.

2. S34ML08G1

   Checklist:

✓ The page size has to be between 512 & 2048 bits.
- The page size of this flash is 2048 bits (within the range supported by BootROM).

✓ Make sure that the flash has either on-Die ECC or 1-bit ECC. Anything other than this is not compatible with Zynq.
- As per the datasheet, S34ML08G1 supports 1-bit ECC.

✓ The I/O width should be either 8-bits or 16-bits.

- This flash can support x8 I/O width which meets the requirement of SMC.

✓ SMC controller supports flashes with one CS bit.
- This flash family, has only one port for chip-select.

✓ Check if the flash is ONFI specification 1.0 compatible.
- S34ML08G1 is ONFI spec. 1.0 compatible.

Conclusion: **S34ML08G1** is compatible with Zynq based on the checklist, and from the datasheet of this flash, the manufacture ID is 0x01h (which is already present in the nand.h file).

You should able to program this flash through Zynq PS and boot from it.

3. S34ML08G101TFI200

Checklist:

✓ The page size has to be between 512 & 2048 bits.
- The page size of this flash is 2048 bits (within the range supported by BootROM).

✓ Make sure that the flash has either on-Die ECC or 1-bit ECC. Anything other this is not compatible with Zynq.
- As per the datasheet, S34ML08G101TFI200 supports 1-bit ECC.

✓ The I/O width should be either 8-bits or 16-bits.
- This flash can support x8 I/O width which meets the requirement of SMC.

✓ The SMC controller supports flashes with one CS bit.
- This flash family has two-bit chip-select. As a result, this is not compatible with PS SMC.

✓ Check if the flash is ONFI specification 1.0 compatible.
- S34ML08G101TFI200 is ONFI spec. 1.0 compatible.

Conclusion- **SM34ML08G101TFI200** has two-bit CS. As a result, it is not compatible with Zynq.

4. MT29F2G08AABWP

Checklist:

✓ The page size must be between 512 & 2048 bits.
- The page size of this flash is either 1024 or 2048 based on I/O width (1024 bits for x16 and 2048 bits for x8). So, this criterion is met.

✓ Make sure that the flash has either on-Die ECC or 1-bit ECC. Anything than other this is not compatible with Zynq.
- As per the datasheet, MT29F2G08AABWP supports 1-bit ECC.

✓ The I/O width should be either 8-bits or 16-bits.
- This flash can support both x8 and x16 I/O widths.

✓ The SMC controller supports flashes with one CS bit.
- This flash family, has one-bit chip-select. As a result, this is compatible with PS SMC.

✓ Check if the flash is ONFI specification 1.0 compatible.

- MT29F2G08AABWP is ONFI spec. 1.0 compatible.

Conclusion: **MT29F2G08AABWP** is compatible with Zynq as it meets all of the points in the checklist.
Because Micron flash's manufacture ID is already added to nand.h file, you do not need to make any changes to this file.