Quad SPI Flash: Benefits and Uses in General Purpose Microcontrollers

NXP Semiconductors
BL Microcontrollers
San Jose CA
October 2010
**LPC4300: SPI Flash Interface**

**CORE**
- ARM CORTEX-M4
  - Up to 150MHz
- MPU
- NVIC
- WIC
- Floating Point Unit
- Debug Trace

**SYSTEM**
- GPDMA
- Audio PLL
- Brownout Detector
- USB PLL
- Power-On Reset
- CPU PLL
- Watchdog Timer
- IRC

**MEMORY**
- Flash
  - Up to 1 MB Dual Bank
- SRAM
  - Up to 264 KB
- ROM
  - ROM/OTP

**SUBSYSTEM**
- ARM CORTEX-M0
  - Up to 150MHz
- IPC
- NVIC

**BUS SYSTEM**
- 2 x HS USB 2.0
- SPI Flash Interface
- 4 x UART
- 4 x 32 bit Timers
- Motor Ctrl PWM
- 2 x 8 Ch 10-Bit ADC

**INTERFACES**
- Ethernet MAC
- External Mem Ctrl
- SD/MMC
- Quad Enc Interface
- 3 x SSP/SPI
- 2 x I²C
- 2 x I²S
- RTC
- Alarm Timer

**TIMERS**
- CAN 2.0B
- External Mem Ctrl
- SD/MMC
- Quad Enc Interface

**CONFIGURABLE INTERFACES**
- Serial GPIO
- State Config Timer
- OTP Key Storage
- AES Decryption

**SECURITY**
- 2 x 8 Ch 10-Bit ADC
- 10-Bit DAC

**ANALOG**
- 10-Bit DAC
Unique NXP feature that maps low-cost serial flash memories into the internal memory system.
What is Quad SPI?

- A couple of years ago, PCs started using Quad-SPI Flash for loading BIOS. The high PC volumes forced prices down to low levels.
- Advantages: High speeds, small packages/few pins, low cost.
- Disadvantages: Not supported by standard MCUs – UNTIL NOW!

SPI Flash Interface uses either 4 or 6 lines

- Standard SPI flash uses CLK, CS, MISO and MOSI.
- Quad SPI flash uses CLK, CS IO0, IO1, IO2 and IO3.
External Flash Performance Comparison

Throughput rate for different flash types

<table>
<thead>
<tr>
<th>Interface</th>
<th>Mode</th>
<th>Access Time (nS)</th>
<th>Effective Throughput(MB/S)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parallel</td>
<td>8 Bits</td>
<td>90</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>16 Bits</td>
<td>90</td>
<td>22</td>
</tr>
<tr>
<td>Serial</td>
<td>Single</td>
<td>12.5</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Dual</td>
<td>12.5</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Quad</td>
<td>12.5</td>
<td>40</td>
</tr>
</tbody>
</table>
Traditional View

- CPU
- Control Reg
- Single read buffer
- Or big buffer
- Serial Flash
- CPU RAM
Serial Flash

SPIFI Interface

Full serial flash memory is visible to the CPU. CPU can read any location randomly.
Most of the 157 serial flashes noted above are SPI-only:

- **Read Status (opcode 05), input data 02, SPI mode**

Most Basic SPI Read command:

- **Read Command (opcode 0x03), SPI mode, first byte read = 0x38**
- Next faster mode is for slave to send read data in dual format

Fast Read Dual Out command (opcode 0x3B), first byte read = 0x38

- Next faster mode: master sends the address & mode in dual format
Note that Read command requires the serial flash to provide data in the clock period after it samples the last bit of the address.

This is a difficult requirement, and constrains the serial clock rate at which the Read command can be used.

- Most devices limit Read to 20–50 MHz.

Fast Read command adds dummy byte between address and data.

- Allows time for device to get its data pipeline filled and ready.
- Can be used up to max serial clock frequency of device (50–133 MHz).

Fast Read command (opcode 0x0B), SPI mode, first byte read = 0x31
- Quad mode adds signals IO3:2; here slave sends data in quad format

- Here master sends the address, mode, 2 dummy bytes in quad format.

Fast Read Quad Out Command (opcode 0x6B), first byte read = 0x87

Fast Read Quad Out Command (opcode 0xEB), first byte read = 0x38
If the mode byte in the Read Dual/Quad I/O command is 0xA5, most serial flashes will not expect the next command to have an opcode.

Next mode byte(s) can be 0xA5, or 0xFF to end No Opcode mode.

Driver automatically uses no-opcode mode for devices that can do it.

“No opcode mode” is our term, flash vendors use “continuous read mode”, “XIP mode”
ROM-based Drivers

- **Boot from Quad SPI Flash using SPIFI**
  - Faster than single lane serial flash
  - Boot source selected by pin or NV location

- **Initialization API**
  - Checks what kind of device
  - Writes to control registers for optimal read performance

- **Write API**
  - Block writes
  - Erase
  - Write protection.
SPIFI – Supported Devices

- All NXP’s devices with the unique SPI Flash Interface (SPIFI) support all major suppliers of QSPI flash
Three requirements for serial flash include most existing devices:
- Read JEDEC ID command
- Page programming command (byte programming not supported)
- At least one erase command that is uniform across the whole device

Most (all?) of the 17 unsupported devices lack page programming:
- Elite F25L004, F25L008, F25L016
- Eon 25B64

157 serial flash devices from 11 vendors meet these requirements.

95 have been successfully tested with the SPIFI driver API.

Most of the untested 62 are old, small, more or less obsolete.
Thank You