



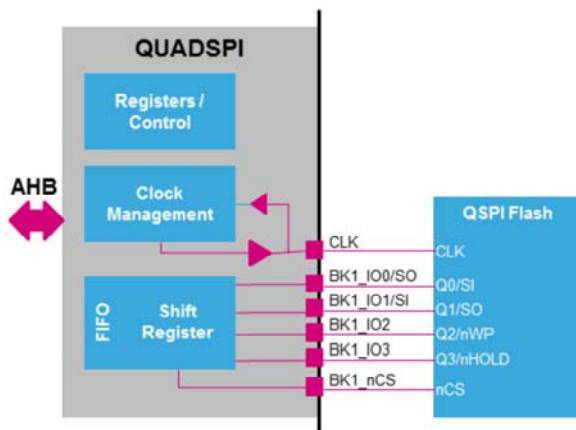
STM32L4 - QUADSPI

Quad-SPI memory interface

Revision 2.1



Hello, and welcome to this presentation of the STM32 Quad-SPI memory interface. It covers the main features of this interface, which is widely used for connecting external memories to the microcontroller.



- The Quad-SPI memory interface provides a communication interface with external serial Flash memories
 - Fully configurable
 - Supports Execute in Place (XiP)
 - Memory mapped

Application benefits

- Supports all SPI Flash memories
- Only a few pins needed
- Simple to integrate additional memory in existing project

The Quad-SPI memory interface integrated inside STM32 products provides a communication interface, allowing the microcontroller to communicate with external SPI and Quad-SPI memories. This interface is fully configurable, allowing easy connection of any existing serial memories available today on the market. Applications can benefit from the easy connection of external serial memories, requiring only few pins. Thanks to the memory mapping feature, external memories can be simply accommodated in the existing project whenever more memory space is needed.

Key features 3

- Three operating modes
 - Indirect
 - Status-polling
 - Memory-mapped
- Optimized operations up to 48 MHz
 - Dual-flash mode (8 bits accessing two flash memories in parallel) (*)
 - Single data rate (SDR) and Dual data rate (DDR) mode support



(*) except on STM32L47x/48x devices

The Quad-SPI memory interface integrated inside STM32 products offers three operating modes and is optimized for communication with external memories with support for dual flash mode, allowing to access 8 bits in single reading cycle. It supports both single- and dual-data rate operation.

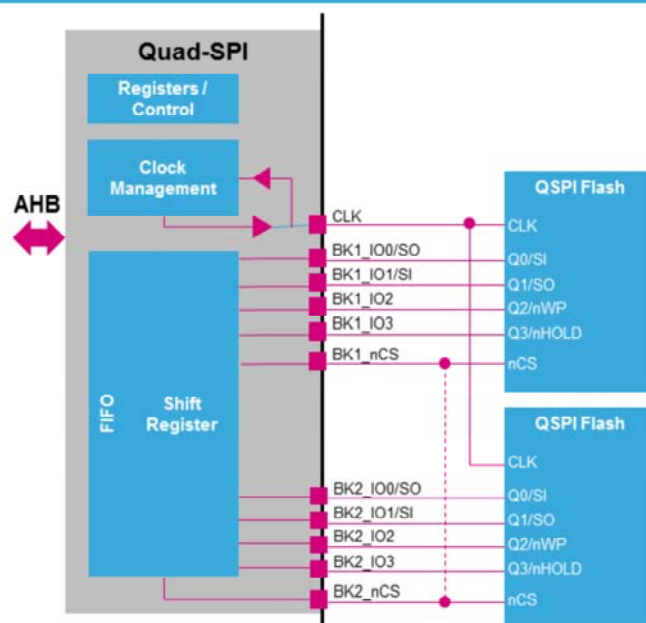
Single or Dual Flash mode

4

- Flexible selection between number of pins and performance

- QuadSPI supports single and dual QuadSPI external FLASH operation (*)
- 4 bit or 8 bit data bus

(*) except on STM32L47x/48x devices



The Quad-SPI memory interface supports the connection of one or two external memories. This means that data can be transferred over a 4- or 8-bit data bus in between the memory and the microcontroller. It gives the user flexibility to choose between the number of pins required for connection (6 for a single and 10 for a double connection) and the performance of the data transfer (4 bits for a single or 8 bits for a double connection).

• Flexible operating modes to reduce CPU load

- Indirect mode
 - All the operations are performed through registers (classical SPI)
- Status-polling mode
 - Automatic periodical read of Flash memory status registers and interrupt generation on match
- Memory-mapped mode
 - External Flash memory seen as being internal for read operations

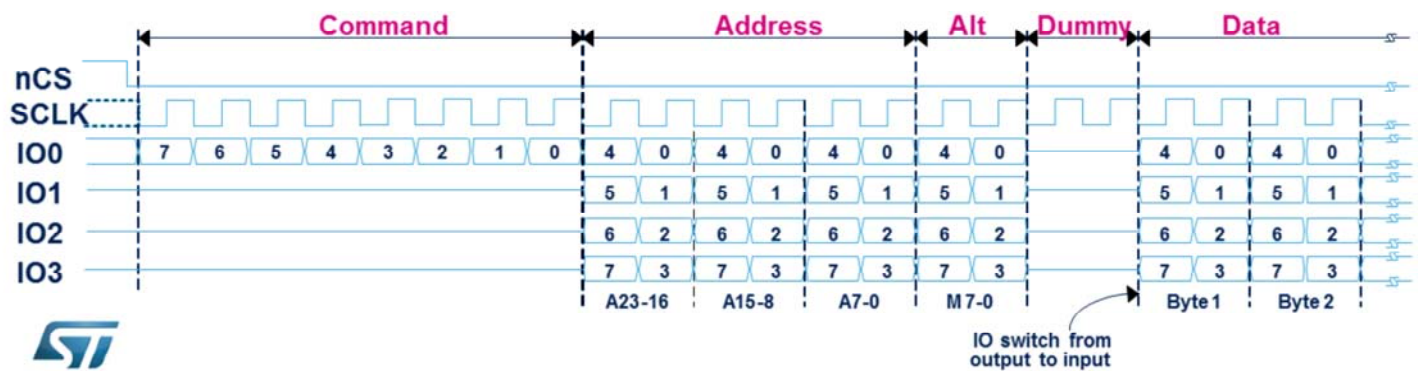


The Quad-SPI memory interface operates in three different modes:

1. Indirect mode, where it behaves as classical SPI interface and all operations are performed through registers,
2. Status-polling mode, where the Flash status registers are read periodically with interrupt generation,
3. Memory-mapped mode, where the external memory is seen as an internal memory for read operations.

Compatible with any single/dual/quad SPI serial Flash memory

- QSPI commands contains up to 5 configurable phases
 - Each phase can be enabled or disabled
 - Configurable length for each phase
 - Configurable number of lines for each phase



The Quad-SPI memory interface offers high flexibility in frame format configuration. This flexibility allows addressing any serial Flash memory. Users can enable or disable each of the five phases and configure the length of each phase as well as the number of lines used for each phase.

Classical SPI interface

- Same usage as a classical communication IP
 - The data is transferred writing or reading data register
 - Number of bytes specified in the data length register
- Management of data FIFO with
 - Interrupts flag (Transfer Complete flag)
 - DMA support
- Launching a command
 - When writing the instruction if only instruction is needed
 - When writing the address if only instruction & address are needed
 - When writing the data when only data phase is needed



The Quad-SPI memory interface used in indirect operating mode behaves like a classical SPI interface. Transferred data goes through the data register with FIFO. Data exchanges are driven by software or by DMA, using related interrupt flags in the Quad-SPI status registers.

Each command is launched by writing the instruction, address or data, depending on the instruction context.

Reduced software overhead

- Specific mode for polling a Status register
 - Programmable register length: 8/16/24/32-bit
 - Repeated read operation at a defined rate
- Mask the response and generate an interrupt in case of match
 - Programmable mask (PSMKR register)
 - The masked value is compared bit per bit with the match register (PSMAR)
 - The result of the comparison can be ANDed or ORed.
 - Interrupt is generated when match is detected (Stop on Match Flag)
- Automatically stops when a match is detected



A specific mode has been implemented in the Quad-SPI interface to autonomously poll the status registers in the external Flash memory. The Quad-SPI interface can also be configured to periodically read a register in the external Flash memory. The returned data can be masked to select the bits to be evaluated. The selected bits are compared with their required values stored in the match register. The result of the comparison can be treated in two ways: in ANDed mode, if all the selected bits are matching, an interrupt is generated. In ORed mode, if one of the selected bits is matching, an interrupt is generated. When a match occurs, the Quad-SPI interface can stop automatically.

Memory-mapped mode 9

Simple extension of memory into the project

Low-power management

- Prefetch for XiP
- External Flash memory seen as an internal one with wait states
 - Read operations are automatically generated upon AHB access
 - Frame & opcode defined during IP configuration as for Indirect mode
- Pin nCS is held low and clock is stopped to stall the Quad-SPI bus and relaunch a sequential read if needed
- Timeout counter to release pin nCS High for low power



The Quad-SPI memory interface also has a Memory-mapped mode. The main application benefit introduced by this mode is the simple integration of an external memory extension thanks to their being no difference between the read accesses of internal or externally-connected memories, except the number of wait states. This mode is only suitable for read operations and the external Flash memory is seen as internal one with wait states included to compensate for the lower speed of the external memory. The maximum size supported by this mode is limited to 256 Mbytes.

The prefetch buffer supports execution in place, therefore code can be executed directly from the external memory without having to download it into the internal RAM.

This mode also supports SIOO mode (Send Instruction Only Once) supported by certain Flash memories, which

allows the controller to send an instruction only once and to remove the instruction phase for following accesses.

Delayed data sampling

10

Useful when signals are delayed due to PCB layout

- The sampling clock can be shifted by additional $\frac{1}{2}$ clock cycle
 - Supported only in SDR mode
- The output data can be shifted by $\frac{1}{2}$ system clock cycle
 - Supported only in DDR mode



Delayed data sampling allows users to compensate for the delay of the signals due to constraints on the PCB layout optimization. It allows applications to shift the data sampling time by an additional $\frac{1}{2}$ clock cycle when operating in SDR mode. In DDR mode, the output data can be shifted by $\frac{1}{2}$ system clock cycle to relax hold constraints.

Interrupt event	Description
Timeout	Set when timeout occurs.
Status match	Set in Automatic Polling mode when the masked received data matches the corresponding bits in the match register.
FIFO threshold	Set in Indirect mode when the FIFO threshold has been reached.
Transfer complete	Set in Indirect mode when the programmed number of data has been transferred or in any mode when the transfer has been aborted.
Transfer error	Set in Indirect mode when an invalid address is being accessed.

- DMA requests can be generated in Indirect mode when FIFO threshold is reached.



The Quad-SPI memory interface has 5 interrupt sources: Timeout, Status Match when the masked received data matches the corresponding bits in the match register in Automatic Polling mode, FIFO Threshold, Transfer Complete and Transfer Error. DMA requests can be generated in Indirect mode when the FIFO threshold is reached.

Low-power modes 12

Mode	Description
Run	Active.
Sleep	Active. Peripheral interrupts cause the device to exit Sleep mode.
Low-power run	Active.
Low-power sleep	Active. Peripheral interrupts cause the device to exit Low-power sleep mode.
Stop 0/Stop 1	Frozen. Peripheral registers content is kept.
Stop 2	Frozen. Peripheral registers content is kept.
Standby	Powered-down. The peripheral must be reinitialized after exiting Standby mode.
Shutdown	Powered-down. The peripheral must be reinitialized after exiting Shutdown mode.



The Quad-SPI memory interface is active in Run, Sleep, Low-power run and Low-power sleep modes. A Quad-SPI interrupt can cause the device to exit Sleep or Low-power sleep modes. In Stop 0, Stop1 or Stop2 mode, the Quad-SPI is frozen and its registers content is kept. In Standby or Shutdown mode, the Quad-SPI is powered-down and it must be reinitialized afterwards.

Performance & power consumption

13

- Data fetch performance comparison

- Time to read a 10-Kbyte table from external Quad-SPI Flash / internal Flash / internal SRAM
- Code execution from internal Flash memory

Conditions	External QSPI	Internal Flash	Internal SRAM
CPU @ 80 MHz – QSPI SDR 4 lanes @ 40 MHz	257 μ s	152 μ s	88 μ s
CPU @ 48 MHz - QSPI DDR 4 lanes @ 48 MHz	214 μs	227 μ s	147 μ s

External Quad-SPI: Micron N25Q256A13EF840E / XiP Mode - Internal Flash: ART enable - Compiler: IAR v7.30.1.7746

- Power consumption of the STM32L4 during this benchmark (External Flash memory excluded)

Conditions	External QSPI	Internal Flash	Internal SRAM
CPU @ 80 MHz – QSPI SDR 4 lanes @ 40 MHz	21 mA	12.74 mA	14.11 mA
CPU @ 48 MHz - QSPI DDR 4 lanes @ 48 MHz	14.6 mA	8.79 mA	8.64 mA

External Quad-SPI: Micron N25Q256A13EF840E / XiP Mode - Internal Flash: ART enable - Compiler: IAR v7.30.1.7746

The external Flash consumption depends on the Quad-SPI Flash device.

We measured ~4.7 mA @ 80 MHz/SDR and ~6.1 mA @ 48 MHz/DDR on data read benchmark for the selected part.



To measure the performance of the Quad-SPI memory interface, we use a simple benchmark reading a 10-Kbyte table from the Quad-SPI Flash memory, internal Flash memory and internal SRAM. The code itself is executed from the internal Flash memory. In Single Data Rate mode at 80 MHz, the internal Flash memory is 40% faster than the external Quad-SPI Flash memory. But in Double Data Rate mode at 48 MHz, the Quad-SPI takes advantage of its prefetch mechanism and is a little bit faster than the internal Flash memory! The STM32L4's power consumption is higher when using the Quad-SPI Flash memory because of I/O toggling.

- Combining the execution time and the power consumption of the data fetch benchmark, we can determine the energy budget required

Conditions	Ext. QSPI (STM32 only)	Ext. QSPI (incl. external Flash)	Int. Flash	Int. SRAM
CPU @ 80 MHz - QSPI SDR 4 lanes @ 40 MHz	17.81 μ J	21.80 μ J	6.38 μ J	4.10 μ J
CPU @ 48 MHz - QSPI DDR 4 lanes @ 48 MHz	10.31 μJ	14.62 μ J	6.58 μ J	4.19 μ J

External Quad-SPI: Micron N25Q256A13EF840E / XiP Mode - Internal Flash: ART enable - Compiler: IAR v7.30.1.7746
 The external Flash consumption depends on the Quad-SPI Flash device.
 We measured ~4.7 mA @ 80 MHz/SDR and ~6.1 mA @ 48 MHz/DDR on data read benchmark for the selected part.



Combining the execution time and the power consumption during the benchmark, we can obtain an overall picture of the energy budget. The DDR mode is the most optimized in terms of energy consumption for a given task.

- Execution comparison

- Even if the Quad-SPI interface is optimized for data transfers. It's possible to execute code from the external Quad-SPI Flash memory

Conditions	External QSPI
CPU @ 80 MHz – QSPI SDR 4 lanes @ 40 MHz	0.469 CM/MHz
CPU @ 48 MHz - QSPI DDR 4 lanes @ 48 MHz	0.754 CM/MHz

External Quad-SPI: Micron N25Q256A13EF840E / XiP Mode - Internal Flash: ART enable - Compiler: IAR v7.30.1.7746

- The consumption is equivalent to the data fetch benchmark under the same conditions.

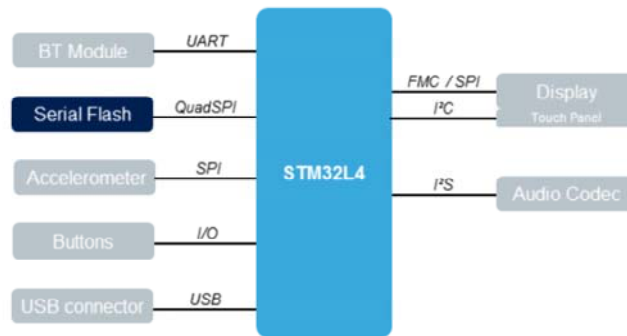


Even if the Quad-SPI interface is optimized for data transfers, it's possible to execute code from the external Quad-SPI Flash memory. The CoreMark results are better when using DDR mode. The consumption is equivalent to the data fetch benchmark under the same conditions.

Application examples

16

- Wearable application including connectivity and user interface:



- External Quad-SPI can store graphical (icons, fonts...etc...) and audio data required for user interfaces.



Wearable applications require low-power management functions together with a high-quality user interface. This can be achieved using the STM32L4's Quad-SPI interface to store in an external Flash memory all the graphical content needed including background images, high resolution icons, or fonts to support multiple languages. Additional audio data for ringtones can also benefit from the large space offered by an external Flash memory. The low pin-count needed to drive such devices allows for a highly optimized system integration.

Related peripherals 17

- Refer to these peripherals trainings linked to this peripheral:
 - RCC (Quad-SPI clock control, Quad-SPI enable/reset)
 - Interrupts (Quad-SPI interrupt mapping)
 - DMA (Quad-SPI data transfer)
 - GPIO (Quad-SPI input/output pins)



You can refer to the Peripherals training slide related to RCC, interrupts, DMA and GPIO for additional information.

- For more details, please refer to following resources
 - AN4760 - Quad-SPI (QSPI) interface on STM32 microcontrollers



For more details, please refer to application note
AN4760, Quad-SPI interface on STM32 microcontrollers.
Thank you