Improved Memory Throughput Using Serial NOR Flash
NOR Flash memory remains the preferred non-volatile technology for discrete memories in embedded systems. For some applications, NOR usage is migrating away from the Parallel NOR Bus to products based on the lower pin count Serial Peripheral Interface (SPI) to optimize the memory subsystem. Many systems using NOR based SPI memories have reached 108MHz bus clock rates using a QuadIO (x4) interface to achieve a 54MB/s sustained read throughput while retaining compatibility with the original interface specified over 25 years ago. The latest SPI-NOR product releases are incorporating a DDR read operation and reaching throughputs of up to 80MB/s. As system level read throughput requirements continue to increase a number of strategies have been deployed to optimize the SPI interface for higher performance. This paper describes both system level and memory device strategies that have been deployed to allow higher SPI read throughputs.

HISTORY
The Serial Peripheral Interface (SPI) was introduced in the early 1980s by Motorola and continues to be popular throughout the embedded market. The original interface (Figure 1) has an efficient four pin (CS#, SCK, SI, SO) bus structure that rapidly achieved market dominance over the proprietary I²C (Inter-Integrated Circuit from Philips Semiconductor) and Microwire (from National Semiconductor) interfaces introduced during the same period. Initial EEPROM products for all three of the interfaces used low cost eight pin DIP and SOIC packages.
The early EEPROM offerings had limited operating frequencies by today’s standards: 1MHz for SPI and Microwire, and 100KHz for I2C. While the initial offerings had modest clock rates, by the time NOR flash had emerged as the dominant NVM technology in the mid 90s it was clear that read throughputs would become a first order value proposition.

The SPI bus has evolved over time improving read throughputs by increasing both clock rates and bus width. Bus width increased from the original x1 to x2 and finally a x4 interface (see Figure 2). The evolution of the SPI bus width has been accomplished by retasking the original six SPI bus signals to support the x2 and the x4 bus widths without increasing the overall pin count. Today’s NOR based QuadIO SPI memories commonly reach 108MHz clock rates to achieve a 54MB/s sustained read throughput. Clock rates of 133MHz are starting to appear on some SPI memories. The SPI bus interface is used on virtually all higher density NOR Flash offerings.

**COMPARISON WITH ALTERNATIVE NOR FLASH MEMORY INTERFACES**

Two of the more significant criteria used when considering NOR Flash devices is the sustained read throughput and the number of pins used for the bus interface. Figure 3 compares different NOR Flash bus interfaces and their respective read throughputs. The legacy parallel interfaces (Async, Page and ADP Burst) all require over 40 pins with an address/data multiplexed version of the parallel bus (ADP) having around 30 pins. Historically the parallel interfaces have provided higher read throughputs but recent advances in SPI bus performance have significantly closed the throughput gap.

The read throughput comparison in Figure 3 shows that SPI NOR exceeds the abilities of the Asynchronous NOR bus and approaches the level of the Page Mode interface. This comparable performance to Async and Page products is especially relevant for the many embedded chipsets that do not support either of the Burst NOR interfaces. Recent product announcements of 133MHz QuadIO
(66MB/s) and 80MHz DDR-QuadIO (80MB/s) tilt the performance in favor of the SPI interface wherever the Parallel Burst and Page Mode interfaces are not available.

**SYSTEM LEVEL IMPROVEMENTS**

**Direct CPU Read Accesses**

Host systems have historically interfaced with SPI Flash memories through an integrated peripheral SPI controller. To accomplish a read operation the host software will:

1. Load the target address into the controller address register
2. Load the Read command into the command register
   a. This loading would automatically initiate the read transaction on the SPI bus
3. Poll the SPI controller status register until the target data has been output by the SPI memory and captured by the host SPI controller
4. Extract the target data from the data buffer in the SPI controller

This process is dramatically slower than what occurs during a read from a parallel Flash memory device where the external memory is mapped directly into the CPU address space. In the parallel usage case the CPU simply reads from the target address and the data is returned without the need for additional
Improved Memory Throughput Using Serial NOR Flash

Software intervention. This process is a requirement for XiP operation where op-code fetches must be performed directly for efficient code execution.

Several recently released SoC products have included a bimodal SPI controller strategy that maintains the legacy peripheral access infrastructure while also allowing read operations to be performed directly from the CPU address map. This invention has led to the rapid adoption direct execution of code out of SPI memories. Direct mapping into the CPU memory map eliminates throughput bottlenecks that exist during transfers through the SPI peripheral controller.

**Dual SPI Interfaces**

In a few high performance applications the need for increased read throughput and high density support has caused SoC manufacturers to develop dual channel QuadIO SPI interfaces (*Figure 5 - Dual SPI Interface*). The two channels are operated simultaneously in QuadIO mode to double the read throughput while only increasing the interface by six signals. The increased pin count (from 6 to 12 pins) is still much lower than the 40+ signals that would be required for a parallel NOR interface.

**Device Level Improvements**

Device level behavior has also been pressured by the need for higher read throughputs. The first order requirement of increasing the clock rate has been implemented but a number of other parameters and characteristics have also received attention to make the higher data rates viable in a real world system.

**Bus Timing Enhancements**

Two of the more significant timing parameters that have received scrutiny are the clock to data valid time (tV) and the data hold after clock time (tHO). These two parameters describe how long data is valid on the bus. tV describes when data becomes valid after a clock edge and tHO describes...
Improved Memory Throughput Using Serial NOR Flash

how long data will remain valid after a clock edge. Much work has been done to minimize $t_{VH}$ and maximize $t_{HO}$ to extend the data valid period for operation at higher clock frequencies (shorter clock periods).

Shift from 3V to 1.8V Operating Voltages
Device operating voltages are starting to transition from the legacy 3V supply to the 1.8V voltage level. This migration has been largely driven by the use of SPI devices in cell phones where the lower operating voltage is attractive from a power consumption perspective. A peripheral advantage to the lower operating voltage is that signal swings on the bus interface are reduced when operating with lower voltages. This smaller voltage swing between logic states means shorter transition times which are necessary at higher operating frequencies. Current 1.8V offerings are starting to appear with operating frequencies of 133MHz with the possibility of even higher frequencies.

Output Drive Strength Control
One emerging trend to maximize signal integrity is to allow the output drive strength to be optimized in the target environment. This capability has long been part of the high speed DRAM world and is essential to maximizing signal integrity at higher data rates. Typical implementations provide four settings that are configurable in-system by the host processor. Environmental problems related to capacitive loading, trace impedance and trace length can be mitigated by adjustment of the output drive strength. Figure 6 shows the output drive capabilities from one of the devices that support drive strength control.

Figure 7
PROTOCOL IMPROVEMENTS

Significant focus has also been placed on minimizing the protocol overhead on the SPI bus. One advantage that parallel (NOR) busses have is the ability to instantly identify the transaction that needs to be performed. Serial devices incur additional latency because the command to perform and the target address is presented in a serial (or “semi-serial”) fashion. This serial process requires several clock cycles just to fully describe the operation to perform.

In the multi-IO versions of the SPI protocol the target address and sometimes the command is presented in a multi-bit wide manner to minimize the number of clock cycles required to initiate an operation. Other strategies to minimize command, address and data transfer overhead are described below.

Burst Types

Continuous burst read mode has been available since the first SPI-EEPROM devices were available. A read burst starts at the target address and data continue to be clocked out of the device from sequential addresses (Figure 8 – Continuous (Non-Wrapped) Burst of Eight).

More recently SoC products are able to execute code directly out of SPI Flash memories. For cached systems a wrapped read burst is available in many SPI-NOR devices to efficiently fill a cache line. In a wrapped burst the critical data from the target address is output first and subsequent data values are output until the last data value within the burst length is output. The next value that is output wraps back to the first address in the burst length. The burst continues in a wrapped burst until all data within the burst length is output (Figure 9 – Wrapped Burst of Eight). The length of the burst read is often chosen to be the same as the size of the target cache line. Wrapped burst are commonly available in 8B, 16B, 32B, 64B lengths.

PROTOCOL MINIMIZATION DURING BUS TRANSACTIONS

Innovation has also occurred in optimization in the basic SPI bus transaction protocol. Reducing the number of clock cycles used to specify the bus transaction minimizes the inherent latency disadvantage of a serial interface. To maintain backwards compatibility with the x1 interface the
Improved Memory Throughput Using Serial NOR Flash

Legacy Quad Output Read sequence sends the command and address information to the device using only the IO0 signal as shown in Figure 10 – Quad Output Read (32 clocks for Command/Address). Once the Quad Output Read command and target address has been received by the memory device a number of latency clocks are required to retrieve the target data from the array. Once this initial latency has been satisfied the target data will be output in a nibble-wide manner on IO0-IO3.

The Quad IO Read command sequence takes advantage of the four IOs to transfer the target address from the host to the memory in a nibble-wide manner. Once the command and address has been specified the memory retrieves the target data from the array. Once the data is available it is output in a nibble-wide manner. This nibble-wide transfer of the target address reduces the number of clocks required to specify the command and target address (from 32 clocks down to 14 clocks). Note that after the target address has been identified there are two clock cycles used to specify eight mode bits.

On many recent product offerings the eight mode bits are used to indicate whether the next bus transaction can be assumed to be the same as the current transaction. This ability to implicitly specify
a new transaction eliminates the eight bit command sequence portion of a read transaction (Figure 12 – Quad IO Read with Implied Read Command (6 clocks for Command/Address)). When the system wants to switch back to an explicit command protocol the mode bits are set in a manner indicating an exit from implied command mode. The further reduction of eight clock cycles brings the total number of cycles required to fully describe a read operation down to six clocks (from 24 using the legacy serial interface).

**DOUBLE DATA RATE (DDR)**

The next step in transaction compression was the introduction of Double Data Rate (DDR) transfers during both the address/mode and data portions of a read transaction (Figure 13 – DDR Quad IO Read). The command needs to remain Single Data Rate (SDR) to support legacy protocol constraints but once the device recognizes a DDR request the remaining transaction can be performed in a DDR fashion.

DDR read transactions also supported the implied command usage of the mode bits that allows the elimination of the eight bit command at the beginning of a read transaction. The DDR read transaction using the command elimination strategy reduces the command/address overhead to only three clocks (Figure 14 – DDR Quad IO Read with Implied Command).
Improved Memory Throughput Using Serial NOR Flash

PREAMBLE FOR DATA TRAINING

Capturing read data is a fundamental problem for common clock memory interfaces. As data is returned from the memory device data placement is skewed significantly with respect to the clock. This is not a significant problem at lower frequencies where the data from the memory has enough time to reach the host prior to the capturing clock edge but as the clock period shrinks it becomes harder to return data in a timely manner. The problem is compounded when the data is presented in a DDR fashion with a new data value output by the memory every half clock cycle. Modern DRAM and NAND memories have dealt with this issue by including a source synchronous clocking signal that allows the memory to toggle a data strobe with the same characteristic skew present in the output data. The strobe returns to the host and is used to indicate when the target read data is present on the bus.

The problem for the SPI bus is that all of the six legacy bus signals are used for the high performance Quad IO Read operation. An alternative host data capture strategy is to incorporate a Data Learning Pattern (DLP) in the read sequence that will help to indicate when the target data is valid on the bus. The DLP is presented during the unused Dummy Cycles that occur immediately before the target data is output by...
Improved Memory Throughput Using Serial NOR Flash

The DLP is used by the host controller to identify the characteristic skew between the common clock and the data output of the memory. While the DLP sequence is presented the host determines the optimal skew required to successfully capture the target data. All of the read data for a particular transaction is captured with the same skew that was indicated during the DLP training. The DLP training process is performed during every read transaction to assure that all system level and device level skews are understood by the host prior to data capture.

DDR read transactions using the DLP also supported the implied command usage of the mode bits that allows the elimination of the eight bit command at the beginning of a read transaction. The DDR read transaction with DLP using the command elimination strategy also reduces the command/address overhead to three clocks (Figure 16 – DDR Quad IO Read with Data Learning Pattern and Elimination of Command).

FUTURE IMPROVEMENTS
The need for higher read throughputs is emerging in many applications and NVM manufactures continue to develop strategies to meet these increasing demands. While the SPI bus has some room for read bandwidth increases any dramatic improvement will require a fundamental reevaluation of the legacy SPI interface.

BOND PAD LOCATIONS
Signal assignments on the original eight lead DIP and SOIC packages has historically mandated the relative positions of the signals on an SPI memory die. The legacy pad placement may be attractive for wire bonding but is non-optimal from a signal skew
perspective. Significant care needs to be taken during the design of high speed SPI memories to minimize the IO to IO skew while simultaneously maximizing the data valid period.

While leaded packages (SOIC) are still driving SPI volumes a significant number of products are shipping in FBGA packages. Signal assignment in FBGA packages is more flexible than for leaded packages and it is likely that future on-die signal placement will be optimized for higher operating speeds without the legacy packaging constraints.

**A WIDER BUS?**

During the evolution of the SPI bus from x1 to x2 to x4 the legacy six signal interface (CS#, SCK, SO, SI, WP, HOLD) has been fully retasked for improved bus throughput (CS#, SCK, IO[3:0]). No more signals are available to improve bus performance. One of the obvious possibilities for improvement is a departure from the six pin constraint with an increase in the signal count to support an x8 bus width (CS#, SCK, IO[7:0]).

An x8 SPI interface has a few of drawbacks that are worth consideration.

1. In modern production test environments many devices are tested simultaneously. An increase in the number of pins will result in less parallelism and cause in a higher per unit test cost.
2. The higher pin count will require a new package, the legacy 8 lead devices will not support a x8 interface. A SO16 package is popular for SPI-NOR Flash memory devices but there is an associated higher cost for the larger package.
3. The move to a x8 bus will only result in a doubling of bus throughput. The resulting throughput might be attractive in the immediate future but there would be little enthusiasm for taking the next step to a x16 bus width.
4. SPI controllers on the host SoC would need to be upgraded to support the increased bus width and an additional four pins allocated for the new interface.
5. One issue in the SPI-NOR market is that there are significant differences in the bus protocol between the different manufacturers. The basic subset of x1, x2 and x4 read operations have been grandfathered in but a new interface with the required new commands would need alignment between manufacturers if an x8 interface were to gain market traction.

Even though there are obstacles for an upgraded SPI bus protocol the need for improved read throughput will cause a higher performance interface to be developed. This need for a next generation SPI interface might be an activity for an industry standards organization like JEDEC. SPI standardization through JEDEC is an activity that would likely be well received by both customers and manufacturers.

**CONCLUSION**

SPI NOR will continue to be the low cost leader due to minimal pin count, low cost packaging, highly parallel testing infrastructure and perhaps most importantly an efficient bus interface that minimizes die area. Fundamental SPI bus constraints set the stage for a next generation NOR interface that will “rebalance”
low pin count with higher read throughput. Even after a low pin count next generation, high performance interface comes to market the legacy NOR based SPI offerings will continue to be viable for a wide swath of applications. At this point there is every reason to believe that the SPI bus will enjoy another 25 years of life as an NVM interface.