AN1090

NoBL™: The Fast SRAM Architecture

Associated Project: No
Associated Part Family: All NoBL™ SRAMs
Software Version: None
Related Application Notes: None

Abstract

AN1090 describes the operation of NoBL™ SRAMs and outlines how it is suitable for networking applications.

Introduction

Processors in high-performance communication equipments and networking applications demand high-speed memories. The type of memory required is determined by the system architecture, the application, and the processor used. System performance suffers if the memory sub-system cannot satisfy the processor requirements.

This application note describes the Cypress NoBL SRAMs architecture designed to improve memory sub-system performance.

NoBL SRAM Description

NoBL stands for No Bus Latency. NoBL SRAMs have been specially designed to eliminate the bus turn around delay associated with switching between Read and Write operations. These devices are also known by different trademark name zero bus turnaround (ZBT).

NoBL architecture eliminates the number of unused (or 'dead') cycles on the bus between Read and Write operations thereby improving bus utilization to 100 percent making it suitable for networking applications, which have frequent READ/WRITE transitions.

NoBL SRAMs are offered in two flavors namely Pipelined and Flow through. The Pipelined option is suitable for applications where frequency is a critical and Flow-through option is suitable for applications where latency is critical.

For pipelined option, the data read from the SRAM is available two cycles after the address is clocked in. For flow-through option, the data read from the SRAM is available as single cycle after the address is clocked in; there is a single cycle between address and data.

NoBL SRAM Operation

The timing diagram of a Pipelined NoBL SRAM in a Read/Write/Read/Write sequence is illustrated in Figure 1.

In clock cycle #1, the address for a read access is latched into the SRAM. Because of the internal pipeline register, the SRAM provides data in clock cycle #3. The write access can be initiated in clock cycle #2 as shown. The write data for the corresponding address is provided in clock cycle #4. All accesses are completely symmetrical (three clock cycles to complete a READ or WRITE). Therefore all accesses can be fully pipelined with no cycle lost between a read and a write.

The timing diagram of a Flow-through NOBL SRAM is shown in Figure 2. A read operation is initiated in cycle #1 and the SRAM drives the data in cycle #2. A write operation can be initiated in cycle #2 and the corresponding data provided in cycle #3. As is the case with Pipelined NOBL SRAM, both read and write operations take the same number of cycles (two cycles for read or write). NoBL SRAMs use a signal, the Advance/$\overline{\text{Load}}$ pin (ADV/$\overline{\text{LD}}$). When ADV/$\overline{\text{LD}}$ is asserted LOW, a Read or Write command is given to the SRAM, depending upon the state of the Read/Write pin ($\overline{\text{WE}}$). A Deselect command may also be executed if one of the three chip enables is inactive. When ADV/$\overline{\text{LD}}$ is HIGH, a burst command is performed. For NoBL SRAM a write operation is performed by using a $\overline{\text{WE}}$ pin and Byte Write ($\overline{\text{BWx}}$) signals.

Figure 1. Pipelined NoBL SRAM Timing Diagram

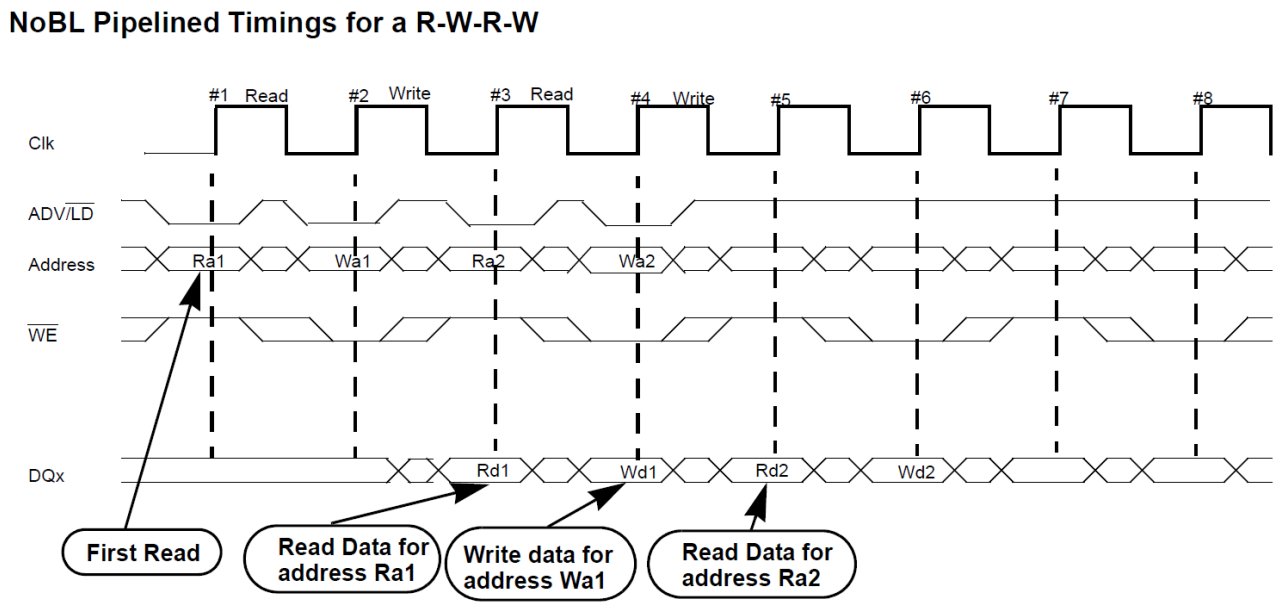
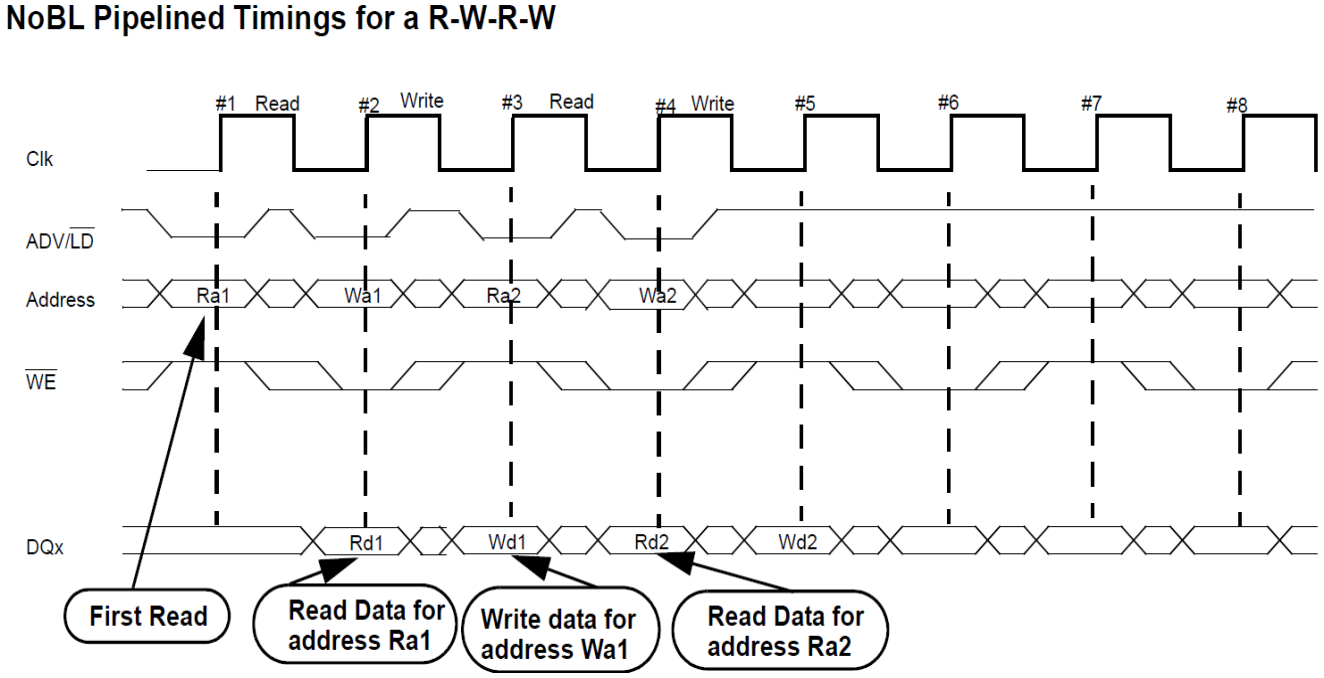


Figure 2. Flow-through NoBL Timing Diagram



Bus Efficiency

Bus efficiency is a metric used to measure the efficiency of a device transferring data over a bus. In the case of an SRAM, this figure shows the number of cycles that are used up when transferring back-to-back READ/WRITE data.

Bus efficiency = Data transfer cycles / total number of cycles.

For an operation such as an R-W-R-W, the maximum bus efficiency is achieved when data is transferred once every clock. In other words, 100 percent bus efficiency occurs when the data is transferred on every clock cycle regardless of the operation. The NoBL SRAM is designed to complete a data transfer on every cycle, so it has a bus efficiency of 100 percent.

This translates to an increase in available bandwidth even when there are reads followed by writes or writes followed by reads.

Figure 3. Detailed Timing of Pipelined NoBL SRAM

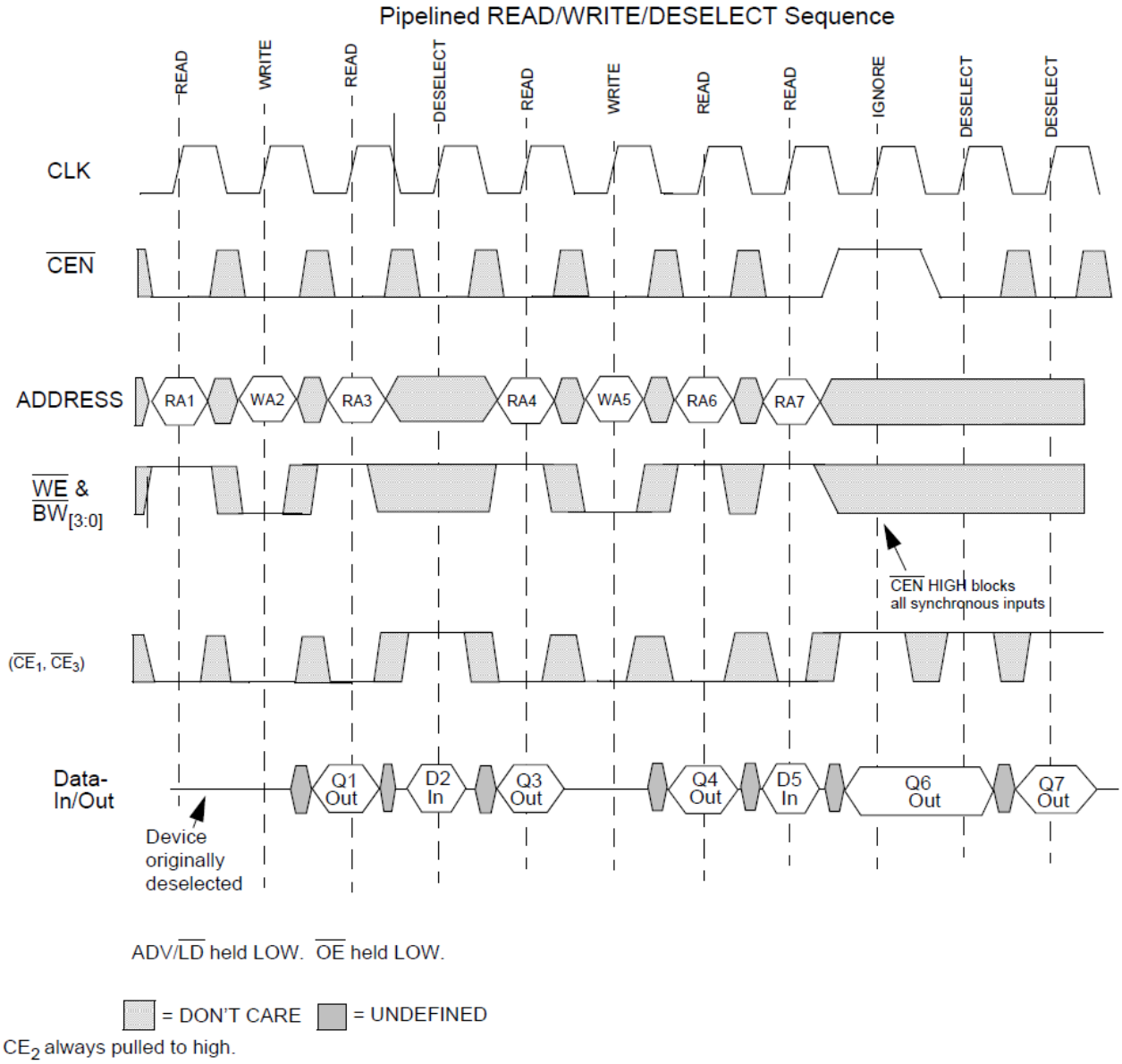
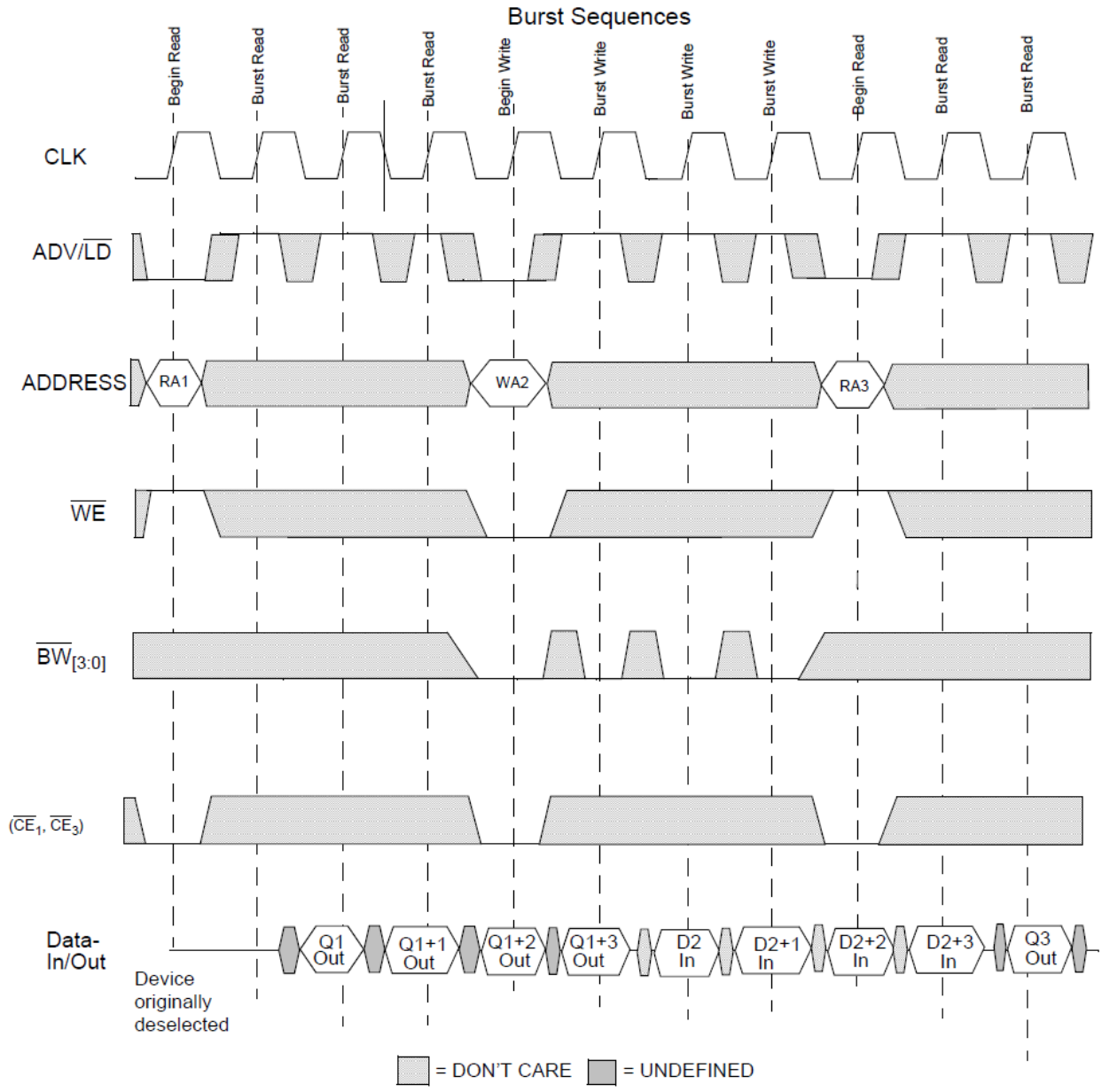


Figure 4. Burst Timing for the Pipelined NoBL Device



CE_2 is always pulled high.

Figure 3 and Figure 4 show the timings of a pipelined NoBL SRAM.

New access cycles are controlled by the ADV/$\overline{LD}$ signal. Advance/Load input used to advance the on-chip address counter or to load a new address. When this input is HIGH at the rising edge of the clock (and $\overline{CEN}$ is asserted LOW) the internal burst counter is advanced. When this signal is LOW at clock rise, a new address can be loaded through the address lines into the device for an access.

A read cycle is initiated when the $\overline{WE}$ signal is high on the rising edge of the clock. In the subsequent cycles, if the

ADV/$\overline{LD}$ is high, the device starts a burst access shown in Figure 4.

The NoBL device starts a write access if the $\overline{WE}$ is LOW on a rising edge of the clock with the device selected.

The sequence of the burst counter is determined by the MODE input signal. A LOW input on MODE selects a linear burst mode; a HIGH selects an interleaved burst sequence. Both burst counters use A0 and A1 in the burst sequence, and wrap-arounds when incremented sufficiently. The sequence of operations for a burst access is as follows. As a new address is loaded to the SRAM by setting ADV/$\overline{LD}$, $\overline{CEN}$, $\overline{CE_1}$, $\overline{CE_3}$ LOW, and CE_2 HIGH. In the subsequent clock cycles setting ADV/$\overline{LD}$ HIGH will increment the internal burst counter regardless of the state of CE or $\overline{WE}$ pins. $\overline{WE}$ is latched at the beginning of a burst cycle. Therefore, the type of access (Read or Write) is maintained throughout the burst sequence. Table 1 shows the interleaved burst sequence, while Table 2 shows the linear burst sequence.

Table 1. Interleaved Burst Sequence

First Address	Second Address	Third Address	Fourth Address
Ax+1, Ax	Ax+1, Ax	Ax+1, Ax	Ax+1, Ax
00	01	10	11
01	00	11	10
10	11	00	01
11	10	01	00

Table 2. Linear Burst Sequence

First Address	Second Address	Third Address	Fourth Address
Ax+1, Ax	Ax+1, Ax	Ax+1, Ax	Ax+1, Ax
00	01	10	11
01	10	11	00
10	11	00	01
11	00	01	10

Considerations when using NOBL SRAMs

1. t_{CHZ} and t_{CLZ}: t_{CHZ} parameter specifies the time it takes for the NoBL device to place its output drivers into a high-impedance state after the rising edge of the clock. t_{CLZ} parameter specifies the time it takes for the NoBL device to start driving data onto the data bus (a low-impedance state). Table 3 shows the maximum and minimum timings of the t_{CHZ} and t_{CLZ} as specified on the datasheet.

   Table 3. Values of t_{CHZ} and t_{CLZ}

Parameter	Min	Max
t_{CHZ}	1.5	3.5
t_{CLZ}	2.5	

   This appears as if the device is specified to allow data contention between SRAMs sharing a common data bus (due to the overlap of $t_{CHZ(max)}$ and $t_{CLZ(min)}$). This is not the case. The specifications for the two parameters are guaranteed over the entire process, temperature, and voltage range. $t_{CHZ(max)}$ is seen at slow corner of the process, high temperature, and low operating voltage. $t_{CLZ(min)}$ is seen at the opposite operating extreme outside of process variations (fast process, low temperature, high voltage).

   Obviously these two extremes do not exist on the same board at the same time. The NoBL device is designed to drive the bus into High-Z before Low-Z under all operating conditions with approximately 1 ns of delta between the two timings, regardless of processing variations. Therefore, contention on the data bus does not occur between NoBL SRAMs.

2. Chip Selects on NoBL SRAM

   The Chip selects on the NoBL SRAM operates in ways different from that of a synchronous pipelined SRAM. The NoBL SRAM has three chip selects $\overline{CE_1}$, CE_2, and CE_3. On a Sync pipelined SRAM the $\overline{ADSP}$ signal is ignored when the chip selects are inactive.

   In the NoBL SRAMs, the CE pins are sampled on the rising edge of the clock. None of the pins on the NoBL SRAMs are masked by any other input. Therefore, all chip enables need to be active to select the device, and any of the three can deselect the device.

3. $\overline{OE}$ Control

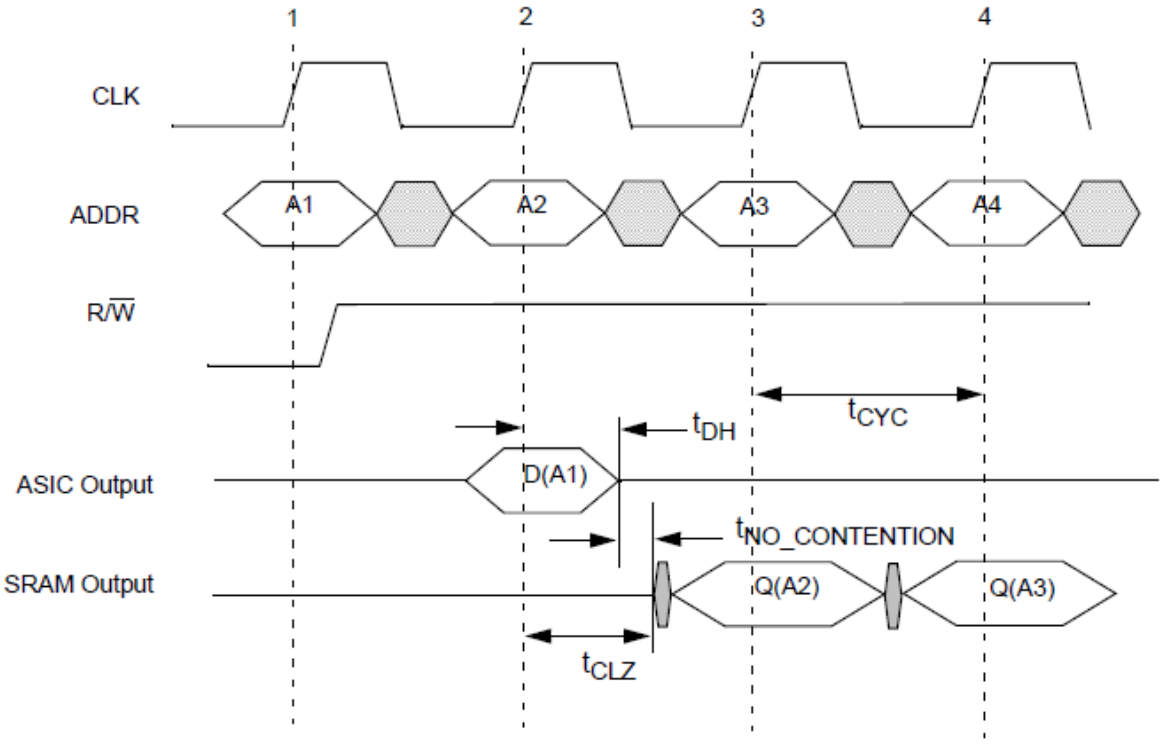
   The NoBL device is a common I/O device. This implies that data should not be driven into the device while the outputs are active. The output enable ($\overline{OE}$) can be de-asserted HIGH before presenting data to the DQ_0–DQ_{31} inputs. Doing so will make the output

drivers tristate. However, the internal logic recognizes when a write is initiated, and synchronously disables the output drivers in order to allow the presentation of the write data. This feature greatly simplifies write sequences and, in most cases, eliminates the need to use $\overline{OE}$ during writes.

4. Bus Contention

   One of the concerns of system designers is bus contention, especially at high frequencies. Of particular concern is the shift from a Write command to a Read command with no bus dead cycles. The Write being performed by the SRAM controller must go into High-Z before the SRAM output drivers turn on for the next Read cycle.

Figure 5. Bus Contention can Occur if the Turn-off Time of the ASIC is Longer than $t_{NO\_CONTENTION}$



Bus contention cannot be completely eliminated. Figure 5 shows how bus contention could occur. When performing a Write cycle, the data being driven to the SRAM (in this case from an ASIC) must meet the hold time of t_{DH} or 0.5 ns. In order to guarantee 0.5 ns of hold, to compensate for temperature, V_{DD} variation and clock skew, and to allow for timing margin, the ASIC must drive the bus longer than 0.5 ns. If this time exceeds $t_{NO\_CONTENTION}$ after the hold time is met, bus contention may occur. The length of time that an ASIC will drive the bus will be a function of the process technology of the ASIC and the output driver used. It is possible that there could be some bus contention between the ASIC trying to turn-off and the SRAM trying to turn on. Certainly t_{CLZ} of 1.5 ns is a worst-case condition for the SRAM, and under most cases of temperature and voltage, it will be longer than 1.5 ns. However, the 1.5-ns value must be used for worst-case calculations.

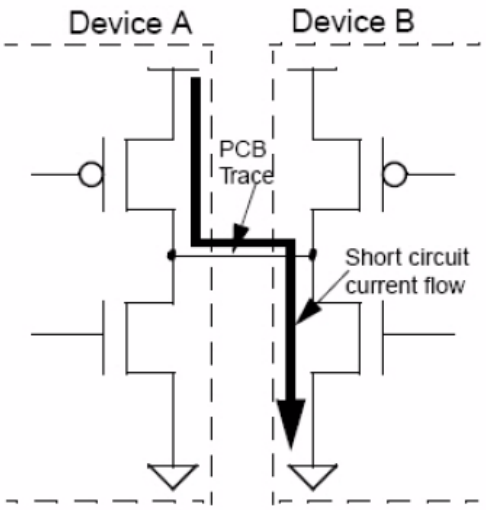
Hence it will be impossible to completely avoid bus contention under all conditions of temperature and voltage. In the following paragraphs we will analyze the impact to the device.

Suppose that the ASIC goes into High-Z 2.5 ns after the clock rises. This means that under worst-case conditions there is 1 ns of bus contention. Assuming the NoBL SRAM is running at 133 MHz, or a 7.5-ns clock period (t_{CYC}). This contention only occurs when going from a Write cycle to a Read cycle. In the worst case this occurs once every two cycles. So the worst-case bus contention occurs 1 ns out of every two clock cycles (15 ns), or 6.7 percent of the total cycle.

On the average, half of the DQ pins will be in bus contention with the SRAM (because the data bits can be at either a 1 or 0 value). So for an x36 SRAM, we will calculate the bus contention assuming that 18 pins will be in contention (on average). If we assume that during the period of contention the resistance of the SRAM driver is 50 Ω and that of the ASIC is 50 Ω, then there is a 100 Ω path between power and ground as shown in Figure 6. Note that this assumption is only an approximation because the resistance of both the SRAM driver and ASIC driver is changing over time as one turns off and the other turns on. Given V_{DD} (Max) of 3.6 V, Current during contention = 3.6 / (50 + 50) = 36 mA.

36 mA will be sourced between power and ground for each I/O in contention.

Figure 6. Devices in Contention



Power dissipated in the SRAM due to contention = 0.36^2*50 = 0.065 W.

For 18 I/Os total power dissipated = 18*0.065= 1.17 W

At first glance this appears to be an unacceptable amount of power. However, it only occurs for 6.7 percent of the total cycle.

Power dissipated without bus contention = Core power + I/O switching power.

Core power= V_{DD}(Max) x I_{DD}(Max) = 3.6*0.35 = 1.26 W

I/O switching power assuming 50 percent reads: = ½*f *C*V^2

where, f = 133 MHz

C = 20 pF*36 for 36 I/Os

V = 3.6 V

Therefore I/O switching power

= ½*133*10^6*20*10^{-12}*36*3.6*3.6*0.5 = 0.31 W

Total power = 1.26 + 0.31 = 1.57 W

Adding in the power of the bus contention for 6.7 percent of the time, 1.57 + (0.067*1.17) = 1.65 W, or a minimal increase.

The effect on junction temperature:

Operating Junction temperature T_J = T_A + θ_{JA}*P where P is the power dissipated by the SRAM.

T_A (Max) = 70 ºC.

θ_{JA} = 25 ºC/W

Without bus contention, T_J= 70 + 25*1.57 = 109 ºC

With bus contention T_J= 70 + 25*1.65 = 111 ºC

Therefore the junction temperature increased by 2 ºC that is acceptable.

In other words, it is acceptable to run with small amounts of bus contention. There will not be driver damage due to bus contention. The NoBL SRAM is designed with separate power pins for I/Os and core so the core power supply remains isolated from the I/O ring and any current surges that occur on the I/O. The extra current will cause a small IR and LdI/dt voltage drop to the I/O ring, but not to the core logic. This architecture prevents voltage drops to the core of the SRAM due to small amounts of contention.

Note that the above calculation assumes that bus contention occurs once every two cycles that is worst case. It also assumes that the resistance of the drivers is constant. Lastly, it assumes that the ASIC is at worst-case turn-off at the same time the SRAM is at worst-case turn-on. Typically these occur at opposite extremes of current and temperature, thus making the above scenario unlikely.

Application

As described in the earlier sections, applications using back-to-back READ-WRITE operations would benefit significantly from the NoBL SRAM.

The NoBL SRAMs eliminate data latency and provide maximum memory bandwidth utilization.

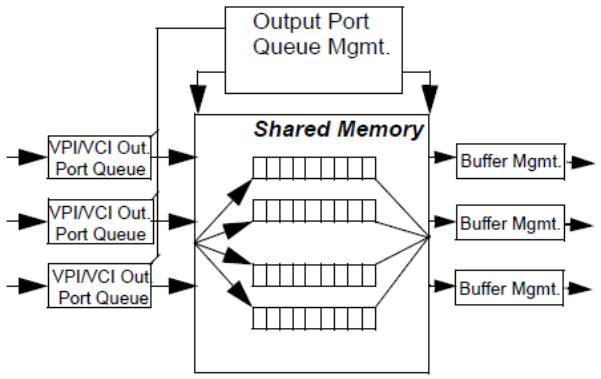
An ATM switch application is used as an example to show the improvement in system performance by using NoBL SRAMs.

ATM switches are applications that require high memory throughput. There are three ways of implementing an ATM switch: Shared memory, Self-routing fabric, Shared backplane.

The Shared memory architecture is one where a large shared memory is used to buffer the incoming cells before being routed to one of the output ports. Figure 7 shows a typical shared memory architecture of an ATM switch. The size of the shared memory depends on the number of cells that have to be buffered for each of the output ports.

The data rate supported by the switch determines the width of the data bus for the shared memory and the frequency of operation of the memory.

Figure 7. A Shared Memory ATM Switch



Most of the shared memory switches use SRAM's. For this discussion we use a shared memory module, which can store up to 26 K cells (1 ATM Cell = 53 Bytes) with a target data rate of 19.2 Gbps. Most of the ATM operations involve continuous Writes and Reads of ATM cells.

The most important consideration in using a certain type of memory in this application is its ability to provide the data rate of 19.2 Gbps. There are several ways of achieving the data rate required for such an application. One way of achieving the desired data rate without running the SRAM at very high-speed data rate is to use wider data bus widths. Another way of achieving the higher data rate is to run the synchronous SRAM's at a higher clock frequency.

The clock frequency required can be calculated using the formula.

Frequency = Data rate / (bus efficiency × width of the data bus)

In this application, let us assume that we use a bus width of 192-bits.

Solution Using NoBL devices

One of the solutions available for such applications is the NoBL SRAM. These devices does not need a turnaround time between a write and a read operation.

The pipelined version of the device is designed to have a standard offset of two cycles for a read and a write.

By using this device, the bus utilization can be improved to 100 percent.

To operate this block at 19.2 Gbps, the frequency of operation comes to 19.2 Gbps / (1.0 × 192 bits) = 100 MHz

Conclusion

The NoBL architecture eliminates the wait periods between read and write, and utilizes the I/O bus close to 100%. This dramatically improves the bandwidth in a given system.

Document History

Document Title: NoBL™: The Fast SRAM Architecture - AN1090

Document Number: 001-26399

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	1420883	NJY	09/12/2007	Obtain spec # for note to be added to spec system. This note had no technical updates. Kindly replace existing .pdf file on cypress.com
*A	3023522	NJY	09/09/2010	Minor edits to maintain consistency with the new application note template.
*B	3346000	OSN	08/16/2011	Corrected WE signal in Figure 5. Added bandwidth calculation in page 6 Elaborated bus contention section in pages 9, 10, and 11. Updated to latest template.
*C	3560900	NJY	03/26/2012	Modified Abstract Modified description in Introduction section Removed comparison references of Standard Synchronous SRAMs Converted application note from FrameMaker to Word template.

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

Automotive	cypress.com/go/automotive
Clocks & Buffers	cypress.com/go/clocks
Interface	cypress.com/go/interface
Lighting & Power Control	cypress.com/go/powerpsoc cypress.com/go/plc
Memory	cypress.com/go/memory
Optical Navigation Sensors	cypress.com/go/ons
PSoC	cypress.com/go/psoc
Touch Sensing	cypress.com/go/touch
USB Controllers	cypress.com/go/usb
Wireless/RF	cypress.com/go/wireless

PSoC® Solutions

psoc.cypress.com/solutions

PSoC 1 | PSoC 3 | PSoC 5

Cypress Developer Community

Community | Forums | Blogs | Video | Training

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

Phone : 408-943-2600
Fax : 408-943-4730
Website : www.cypress.com