caches and other applications that operate using bursts. An SSRAM contains one or more control signals that defines whether a memory cycle uses an externally supplied address or an internally latched address and counter. When a burst transfer is desired, the memory controller asserts a control signal to load the internal burst counter and then directs the SSRAM to use that incrementing count value for the three subsequent cycles. Bursts are supported for both reads and writes. The two-bit burst counter can be configured in one of two increment modes: linear and interleaved. Linear increment is a simple binary counter that wraps from a terminal value of 11 back to 00. Bursts can be initiated at any address, so, if the burst begins at A[1:0] = 10, the counter will count 10, 11, 00, and 01 to complete the burst. Interleaved mode forces the data access pattern into two pairs where each pair contains an odd and even address with A[1] held constant as shown in Table 8.2. Interleaving can benefit implementations that access words in specific pairs.

TABLE 8.2 SSRAM Interleaved Burst Addressing

Initial Value of A[1:0] Supplied Externally	Second Address Generated Internally	Third Address Generated Internally	Fourth Address Generated Internally
00	01	10	11
01	00	11	10
10	11	00	01
11	10	01	00

Flow-through and pipelined SSRAMs fall into two more categories: normal and *zero-bus turnaround* ®(ZBT)*. Normal SSRAMs exhibit differing read and write latencies: write data can be asserted on the same cycle as the address and write enable signals, but reads have one to two cycles of latency, depending on the type of device being used. Under conditions of extended reads or writes, the SSRAM can perform a transfer each clock cycle, because the latency of sequential commands (all reads or all writes) remains constant. When transitioning from writing to reading, however, the asymmetry causes idle time on the SSRAM data bus because of the startup latency of a read command. The read command is issued in the cycle immediately following the write, and read data becomes available one or two cycles later. If an application performs few bus turnarounds because its tends to separately execute strings of reads followed by writes, the loss of a few cycles here and there is probably not a concern. However, some applications continually perform random read/write transactions to memory and may lose necessary bandwidth each time a bus turnaround is performed.

ZBT devices solve the turnaround idle problem by enforcing symmetrical delays between address and data, regardless of whether the transaction is a read or write. This fixed relationship means that any command can follow any other command without forced idle time on the data bus. Flow-through ZBT devices present data on the first clock edge following the corresponding address/command. Pipelined ZBT SSRAMs present data on the second clock edge following the corresponding address/command as shown in Fig. 8.12. As with normal SSRAMs, higher clock frequencies are possible with pipelined versus flow-through ZBT devices, albeit at the expense of additional read latency.

ZBT SSRAMs provide an advantage for applications with frequent read/write transitions. One example is a single-clock domain FIFO implemented using a discrete SSRAM and control logic. A ge-

* ZBT and Zero Bus Turnaround are trademarks of Integrated Device Technology, Inc., and the architecture is supported by Micron Technology, Inc. and Motorola Inc.

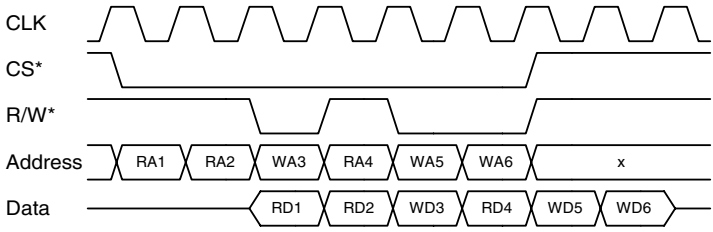

FIGURE 8.12 Pipelined ZBT SSRAM read/write timing.

neric FIFO must be capable of sustained, interleaved reads and writes, which results in frequent bus turnaround delays when using a normal SSRAM. ZBT SSRAM devices are manufactured by companies including Integrated Device Technology, Micron Technology, and Motorola. Cypress Semiconductor manufactures functionally equivalent SSRAMs under the trademark NoBL. Other manufacturers offer equivalent devices with differing naming schemes.

SSRAMs are very popular in high-performance computing and networking applications. Computers with large secondary and tertiary caches use SSRAM to hold lines of data. Networking equipment makes extensive use of SSRAMs for buffering and lookup table applications. SSRAM devices are commonly available in densities ranging from 2 to 16 Mb in 16-, 18-, 32-, and 36-bit wide data buses. The nine-bit bus multiples are used in place of eight-bit multiples for such purposes as the storage of parity and flag bits.

8.4 DDR AND QDR SRAM

SSRAM transitioned to a DDR interface to increase bandwidth in the same general manner as SDRAM. DDR SRAM devices are fully pipelined and feature fixed burst transfer lengths of two or four words to enable a less complex single-rate address/control interface. With the data bus running at twice the effective frequency of the address bus, a burst size of two guarantees that random access transfers can be issued in any order without falling behind the data interface's higher bandwidth. Burst length is fixed by the particular device being used. A burst length of four words simplifies applications such as some caches that operate using four-word transactions, although no inherent throughput advantage is gained. As with a DDR SDRAM, special clocking techniques must be employed to enable the design of reliable interfaces at effective data rates in the hundreds of megahertz. A DDR SRAM accepts a primary pair of complementary clocks, K and K*, that are each 180° out of phase with each other. Address and control signals are registered on the rising edge of K, and write-data is registered on the rising edges of both clocks. An optional secondary pair of clocks, C and C*, must be same frequency as K/K* but can be slightly out of phase to skew the timing of read-data according to an application circuit's requirements. A small degree of skewing can ease the design of the read capture logic. If such skewing is not necessary, C/C* are tied high, and all output (read) timing is referenced relative to K/K*. The SRAM automatically recognizes the inactivity on C/C* and chooses K/K* as the causal output clock. A pair of output echo clocks, CQ and CQ*, are driven by the SRAM in phase with read data such that both the echo clocks and read data are timed relative to C/C* or K/K*. These echo clocks are free running and do not stop when read activity stops. This combined clocking scheme is illustrated in Fig. 8.13. The read capture logic may choose to use the echo clocks as source-synchronous read clocks, or it may use an alternate scheme and not use the echo clocks at all. An alternative scheme could be to skew C/C* such that returning read data is in