*10.3.5 Synchronous SRAM

synchronous SRAM (SSRAM)

A new variety of SRAM, called a *synchronous SRAM (SSRAM)* ("S-S-ram"), still uses latches internally but has a clocked interface for control, address, and data. As shown in Figure 10-26, internal edge-triggered registers AREG and CREG are placed on the signal paths for address and control. As a result, an operation that is set up before the rising edge of the clock is performed internally during a subsequent clock period. Register INREG captures the input data for write operations, and depending on whether the device has "pipelined" or "flow-through" outputs, register OUTREG is or is not provided to hold the output data from a read operation.

late-write SSRAM with flow-through outputs

The first variety of SSRAM to be introduced was the *late-write SSRAM with flow-through outputs*. For a read operation, shown in Figure 10-27(a), the control and address inputs are sampled at the rising edge of the clock, and the

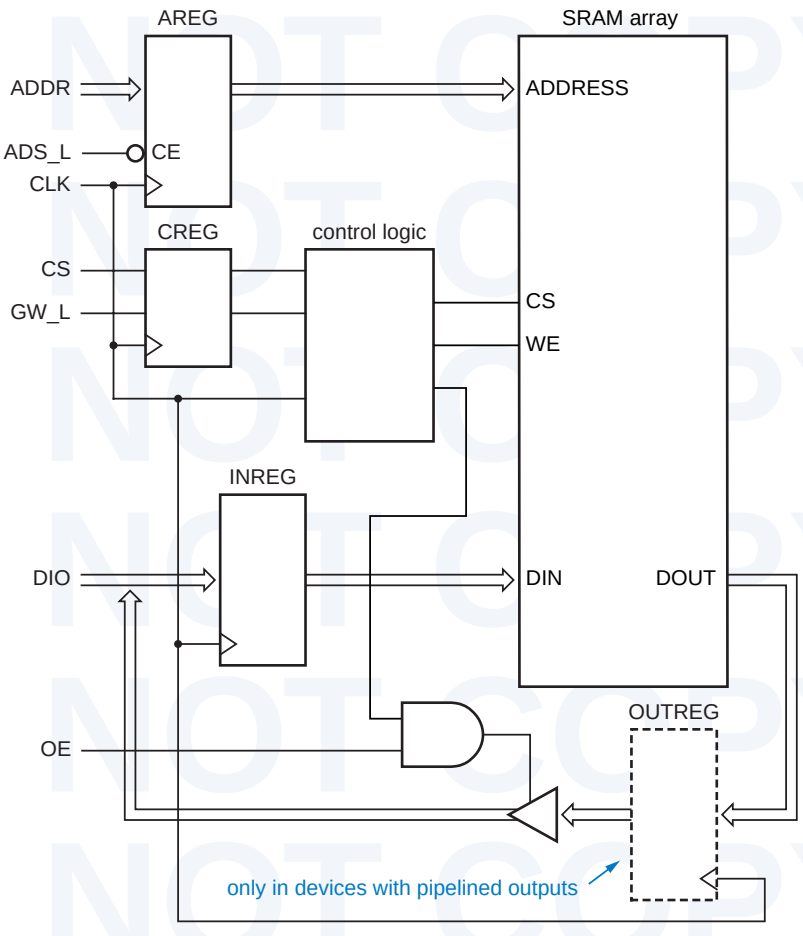


Figure 10-26
Internal structure of a synchronous SRAM.

internal address register AREG is loaded only if ADS_L is asserted. During the next clock period, the internal SRAM array is accessed and read data is delivered to the device's DIO data-bus pins. The device also supports a burst mode, in which data at sequence of addresses is read. In this mode, AREG behaves as a counter, eliminating the need to apply a new address at each cycle. (The control signals that support burst mode are not shown in Figures 10-26 or 10-27.)

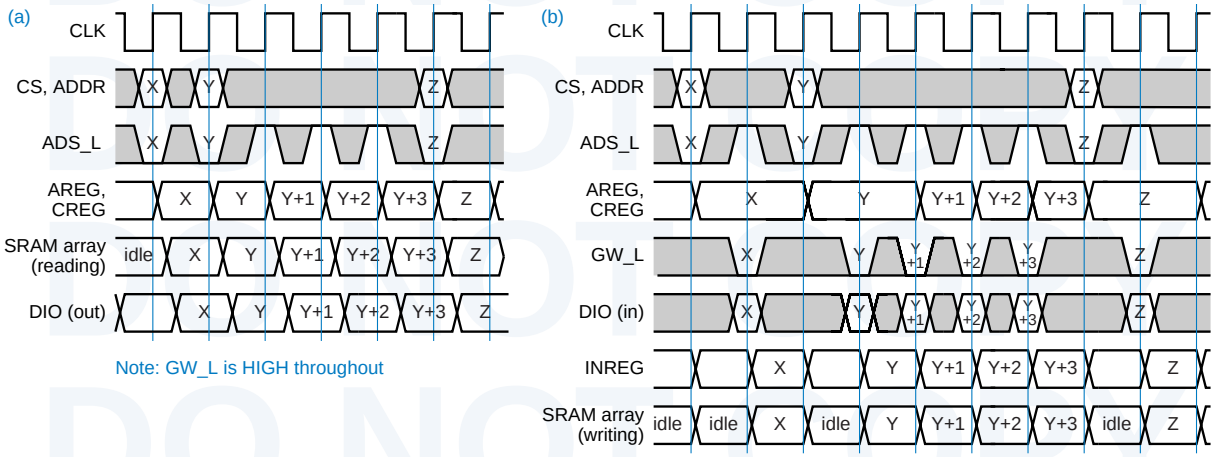
For a write operation, shown in (b), the write data is stored temporarily in an on-chip register INREG, which is sampled one clock tick *after* the address register is loaded. Therefore, ADS_L must be inhibited for at least one tick after loading the address, so that the address in AREG is still valid when the write takes place. The write takes place during the clock period following the edge on which the "global write" control signal GW_L is asserted. As with reading, the device has a burst mode where a sequence of addresses can be written without supplying a new address.

Note that the "late-write" protocol makes it impossible to write to two different, nonsequential addresses in successive clock periods; the SRAM array is idle for one clock period between writes (except in burst mode). From the point of view of internal chip capabilities, this behavior is not necessary. However, the late-write protocol was designed this way to match the bus protocols of microprocessors that use these SSRAMs in their cache subsystems.

late-write SSRAM with pipelined outputs

A *late-write SSRAM with pipelined outputs* is like the previous version, except that a register OUTREG is placed between the SRAM array output and the device output for read operations. As shown in Figure 10-28, this delays the read output data at the device pins until the beginning of the next clock period,

Figure 10-27 Timing behavior for a late-write SSRAM with flow-through outputs: (a) read operations; (b) write operations.



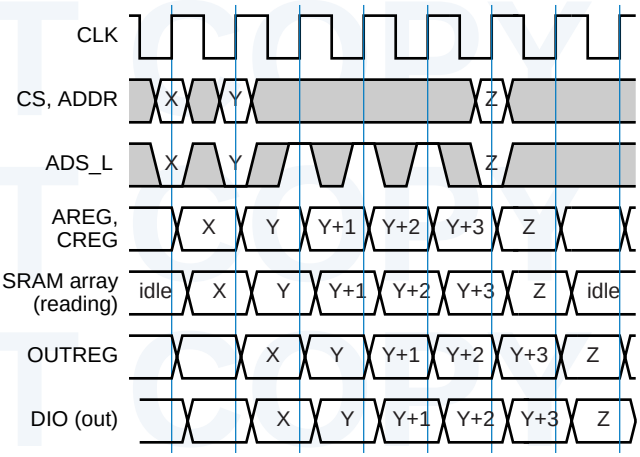


Figure 10-28 Read-timing behavior for a late-write SSRAM with pipelined outputs.

but it also provides the benefit that the data is now valid for almost the entire clock period. The write cycle behaves the same as with flow-through outputs. Compared to flow-through outputs, pipelined outputs provide much better setup time for the device received the read data, and thus may allow operation at higher clock frequencies.

As we showed in Figure 10-26, conventional SSRAMs share the same pins for both input data and output data. During a given clock period, the data I/O pins can be used for reading or writing but not both. If you study the pattern of data-bus and SRAM-array use in both styles of late-write SSRAM, you'll find cases where it's not possible to initiate a read one clock cycle after initiating a write or vice versa, due to resource conflict (see Exercise 10.32). Thus, late-write SSRAMs suffer a *turn-around penalty*, a clock period in which the internal SRAM array must be idle when a read is followed by a write or vice versa.

turn-around penalty

zero-bus-turn-around (ZBT) SSRAM

The turn-around penalty is eliminated in so called *zero-bus-turn-around (ZBT) SSRAMs*. The timing for a *ZBT SSRAM with flow-through outputs* is shown in Figure 10-29. The type of operation (read or write) is selected by a control signal R/$\overline{W}$ that sampled at the same clock edge as the address. Regardless of whether the operation is a read or a write, the DIO bus is used during the next clock period to transfer the read or write data. As a result, there is no data-bus-usage conflict, as long as OE is controlled properly to avoid bus-fighting between successive cycles. However, if a write is followed by a read, both operations would like to use the SRAM array during the same clock period. To avoid this resource conflict, the write operation is deferred until the next available SRAM-cycle. This opportunity occurs when either another write operation or no operation is initiated on the address and control lines.

Although a ZBT SSRAM can use access the internal SRAM array on every clock cycle, this performance improvement is not without a price. While a write operation is pending, the write address and related information must be stored in another register, WAREG, since AREG is reused by other operations; this costs

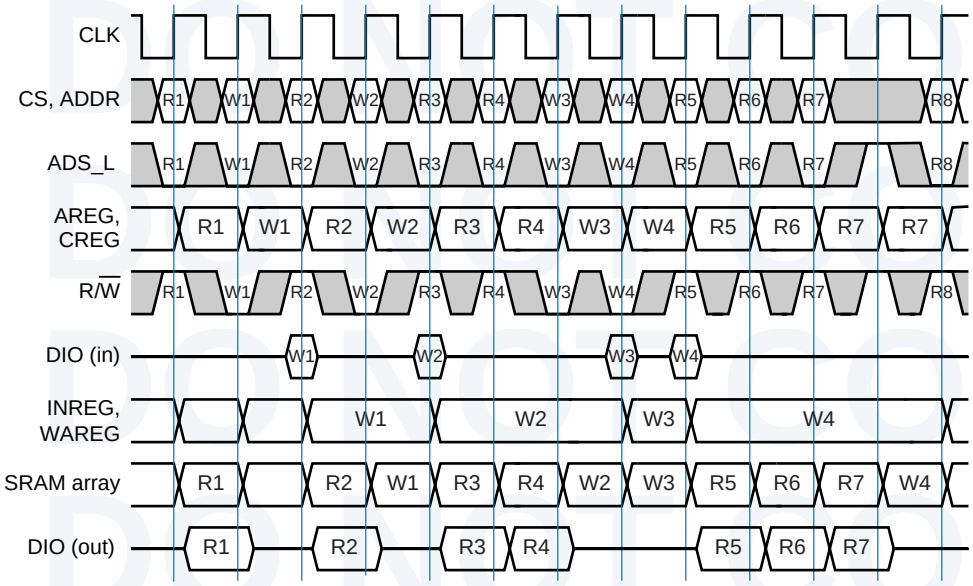


Figure 10-29 Timing behavior for a ZBT SSRAM with flow-through outputs.

chip area. More significantly for some applications, a write operation may be deferred indefinitely if it is immediately followed by a continuous series of read operations. This anomaly may require tricky controller design to detect the case where one of these read operations attempts to access the address that was just written, since the value stored in the SRAM array is "stale"!

A *ZBT SSRAM with pipelined outputs* adds OUTREG to the read data path but is otherwise similar to the previous device. In this device, both reads and writes use the DIO bus during the second clock period following the clock edge in which the operation was initiated. As in the previous device, writes to the internal SRAM array are deferred until an available cycle, so that reads can take precedence. Timing is shown in Figure 10-30. As implied by the timing, two levels of internal registers are needed for write address and data, since up to two writes may be deferred while a sequence of reads occurs.

Synchronous SRAM access protocols are very beneficial in high-speed systems. For example, address, control, and write inputs can be applied with more-or-less conventional setup and hold times with respect to the system clock, and read data on pipelined output pins is available for almost a complete clock cycle. Very importantly, the designer does not have to worry about the tricky circuits and timing paths that are otherwise needed to enable conventional SRAM latch-style operation.

SSRAMs were available in 1999 with clock frequencies as high as 166 MHz. Note that among the four styles of SSRAM that we described, no single one is the "best." The best SSRAM is the one that best fits the bus protocol and other requirements of the system in which it is used.

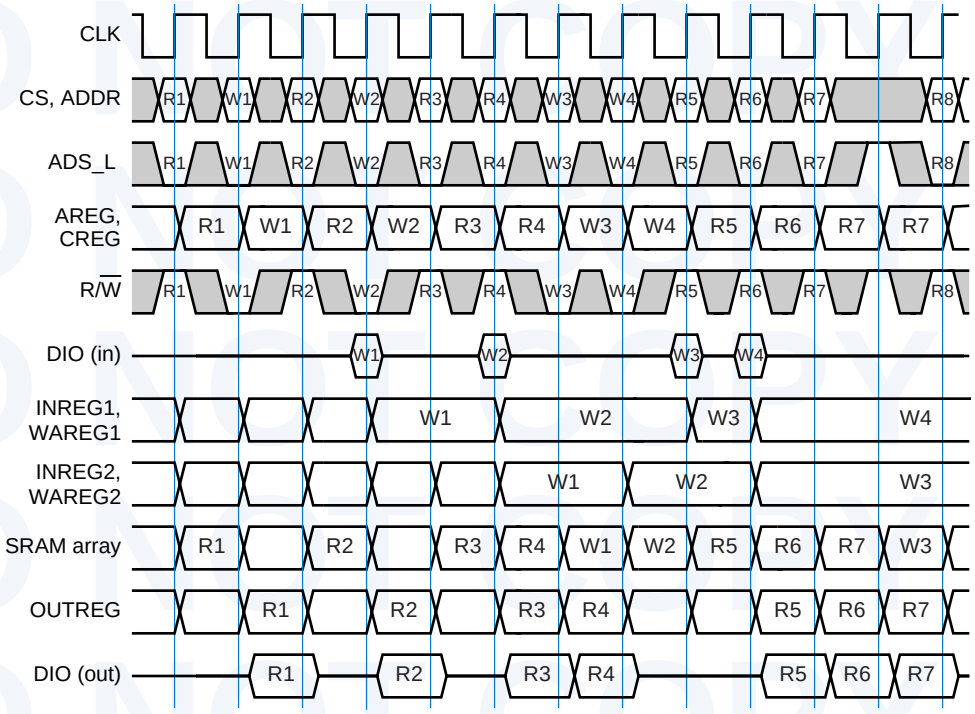


Figure 10-30 Timing behavior for a ZBT SSRAM with pipelined outputs.

10.4 Dynamic RAM

The basic memory cell in an SRAM, a *D latch*, requires four gates in a discrete design, and four to six transistors in a custom-designed SRAM LSI chip. In order to build RAMs with higher density (more bits per chip), chip designers invented memory cells that use as little as one transistor per bit.

10.4.1 Dynamic-RAM Structure

dynamic RAM (DRAM)

It is not possible to build a bistable element with just one transistor. Instead, the memory cells in a *dynamic RAM (DRAM)* store information on a tiny capacitor accessed through a MOS transistor. Figure 10-31 shows the storage cell for one bit of a DRAM, which is accessed by setting the word line to a HIGH voltage. To store a 1, a HIGH voltage is placed on the bit line, which charges the capacitor through the "on" transistor. To store a 0, LOW voltage on the bit line discharges the capacitor.

precharge

sense amplifier

To read a DRAM cell, the bit line is first *precharged* to a voltage halfway between HIGH and LOW, and then the word line is set HIGH. Depending on whether the capacitor voltage is HIGH or LOW, the precharged bit line is pulled slightly higher or slightly lower. A *sense amplifier* detects this small change and recovers a 1 or 0 accordingly. Note that reading a cell destroys the original