

MHz. Despite this, BEDO memory is no longer used, mainly because of the lack of chipsets to support his use. Intel considered that the memory of type EDO is no longer viable, so they did not implement support of BEDO memory into their chipsets. Several important memory manufacturers had put considerable time and funds into the development of SDRAM, and were not interested to support the BEDO memory.

#### 4.4.6.5. SDRAM

All DRAM memories that have a synchronous interface are known generically as SDRAM (*Synchronous* DRAM). These memories include PC100 SDRAM, PC133 SDRAM, DDR SDRAM, CDRAM, ESDRAM, and others. However, the type of memory that most often is called SDRAM is the JEDEC (*Joint Electron Device Engineering Council*) standard synchronous DRAM. JEDEC is a committee of the *Electronic Industries Association* (EIA) in the U.S.A., that elaborates electrical standards for packaging, pin-out, and other features of semiconductor devices.

#### Principle of SDRAM Memory

Synchronous DRAM is entirely different in its architecture and controlling methods from asynchronous DRAM. The first difference is the configuration of the SDRAM memory, which utilizes a multi-bank architecture. Typical SDRAM modules have 2 or 4 banks per module, allowing one bank to precharge while the others are read or written to. Therefore, not only the precharge times are masked, but multiple rows can be simultaneously accessed in each bank of memory.

The second difference is that the SDRAM memory can operate in burst mode for 1 bit, 2 bits, 4 bits, 8 bits, and a full page. In burst mode, several locations with consecutive addresses are transferred in each memory transaction. The burst mode is advantageous because instructions and data are read in sequential order most of the time. For example, when an L2 cache memory is present, memory blocks with fixed size are transferred, containing words with consecutive addresses. The burst mode is controlled by a mode register that can be set at power-on and changed during operation. This register controls the burst transfer type (sequential or interleaved), the burst transfer length and the *CAS* latency.

Another factor that distinguishes synchronous DRAM from asynchronous DRAM is the method of control. Asynchronous DRAM is controlled by the chipset based on the timing relationship between the  $\overline{RAS}$ ,  $\overline{CAS}$ ,  $\overline{WE}$  and  $\overline{OE}$  signals. Synchronous DRAM is controlled by commands that are placed on the bus, which are interpreted on the rising edge of the clock signal.

Below we define the main signals of an SDRAM memory chip and the functions of these signals.

- *CLK* (*Clock*) is the clock signal provided by the memory controller and is used to synchronize input commands, data and addresses with the rising edge

of the clock. The rising edge of the clock signal initiates the command decoding and execution. The clock signal also increments the counter used for burst operations and controls the output registers. An SDRAM memory module uses two or four clock lines.

- *CKE (Clock Enable)* activates and deactivates the *CLK* signal. When *CKE* is inactive, the *CLK* signal is no longer used by the SDRAM. By deactivating the clock, the *CKE* signal initiates the *Power Down* mode, *Self Refresh* mode or *Clock Suspend* mode. When the *CLK* signal is deactivated, the input buffers are turned off to save power, and an internal clock is used to continue any on-going command.
- $\overline{CS}$  (*Chip Select*) controls all accesses to the SDRAM memory. It can be used to select the appropriate chip of a memory module.
- $\overline{RAS}$  (*Row Address Strobe*),  $\overline{CAS}$  (*Column Address Strobe*),  $\overline{WE}$  (*Write Enable*) are signals with the same function as for asynchronous DRAM memories.
- $DQ [w-1, 0]$  are the bidirectional data lines.
- *DQM (DQ Mask)* is used to control the data lines. For read accesses, *DQM* controls the data output buffers, representing an output enable signal for these accesses. For write accesses, it can mask the data from being written to the memory array. The *DQM* signal performs the role of the  $\overline{OE}$  signal, which is not used for the SDRAM memory.
- $A [m-1, 0]$  represent the address lines. Some address pin definitions change as a function of the memory array size.
- $BA [1, 0]$  represent the memory bank address lines and select the bank to which a command is being applied.

The first SDRAM modules contained only two clock lines, but it was soon determined that this was insufficient. This created two different module types, with 2 and 4 clock lines, which are not compatible with each other. In a module with 2 clock lines, two chips are accessed in each clock cycle. In a module with 4 clock lines, four chips are accessed in each clock cycle. In order to operate correctly, it is necessary to know the type of the modules required by a certain motherboard. Today, the most used modules are those with 4 clock lines. By using four clock lines instead of two, the overall capacitive load for each clock line will be reduced, resulting in lower rise and fall times of the voltage.

Although the timings were theoretically supposed to be 5-1-1-1 at 66 MHz, many of the original SDRAM modules only operated with timings of 6-2-2-2 when run in pairs, mainly because the chipsets (such as i430VX, SiS5571) had difficulties to coordinate the accesses between modules. The i430TX chipset of Intel and later chipsets corrected these deficiencies, and an EEPROM chip was added to the standard. This chip, called SPD (*Serial Presence Detect*), contains information about the SDRAM

module, such as timing settings. The chipset could read these settings from the SDRAM module, and therefore the module can operate more reliably on a larger number of motherboards. Unfortunately, the SPD EEPROM memory was either not included on many modules, or not read by the motherboards. These problems were corrected later.

The speed of SDRAM memories is expressed in MHz, rather than nanoseconds. In this way there is a correspondence between the bus speed and the memory speed, because theoretically the SDRAM memory allows the operation without wait states, at a rate of one access per clock cycle, after an initial latency. One indication of the memory speed is the clock cycle time ( $t_{CLK}$ ), which is usually marked on the memory chips. For example, values such as -12, -10, or -8 found on an SDRAM chip indicate the minimum clock cycle time for that component, in ns. A value of -10 means that the maximum clock frequency of the memory is 100 MHz.

The original SDRAM modules either used 83-MHz chips (12 ns) or 100-MHz chips (10 ns), but these were only rated for operation on a 66-MHz bus. These SDRAM modules are now called PC66, to differentiate them from those conforming to Intel's PC100 and PC133 specifications.

### SDRAM Commands

An SDRAM command is determined by a certain combination of the  $\overline{CS}$ ,  $\overline{RAS}$ ,  $\overline{CAS}$ , and  $\overline{WE}$  signals. At every rising edge of the  $CLK$  signal, the command is latched and then will be executed. Command signals no longer need to be interpreted based on the timing relationship between them, but only at the rising edge of the clock signal.

Table 4.1 contains the status of the command and address signals for the main SDRAM commands. The following abbreviations are used in this table: H: high level; L: low level; V: valid data input; X: don't care value. The number of address lines is for a 128-Mbits SDRAM chip with four banks.

In the following we describe the main commands of an SDRAM memory.

**Command Inhibit.** This function prevents new commands from being executed by the memory, regardless of whether the clock signal is enabled or not.

**No Operation.** The *No Operation* (NOP) command is used to activate a particular memory chip and then to place it in the idle state. For example, this command can be used to delay the read operation by a few clock cycles in order to give the memory enough time to provide the requested data. By inserting NOP commands, the memory enters a wait state instead of the CPU. Therefore the CPU can resume other operations after it requests a particular data, instead of being idle during the entire reading process.

**Table 4.1.** Main SDRAM commands.

Command	CS	RAS	CAS	WE	DQM	A11	A10	BA[1:0]	A[9:0]
Command Inhibit	H	X	X	X	X	X	X	X	X
No Operation	L	H	H	H	X	X	X	X	X
Activate	L	L	H	H	X	V	V	V	V
Read	L	H	L	H	L/H	V	L	V	V
Read with Auto Precharge	L	H	L	H	L/H	V	H	V	V
Write	L	H	L	L	L/H	V	L	V	V
Write with Auto Precharge	L	H	L	L	L/H	V	H	V	V
Burst Terminate	L	H	H	L	X	X	X	X	X
Precharge Selected Bank	L	L	H	L	X	V	L	V	X
Precharge All Banks	L	L	H	L	X	X	H	X	X
Auto Refresh	L	L	L	H	X	X	X	X	X
Mode Register Set	L	L	L	L	X	L	L	V	V

**Activate.** This command selects a particular memory bank, addressed by the bank address pins, and activates a row in the selected bank. Read and write operations can only be initiated on this activated row after the minimum  $t_{\text{RCD}}$  time is elapsed from the activate command.

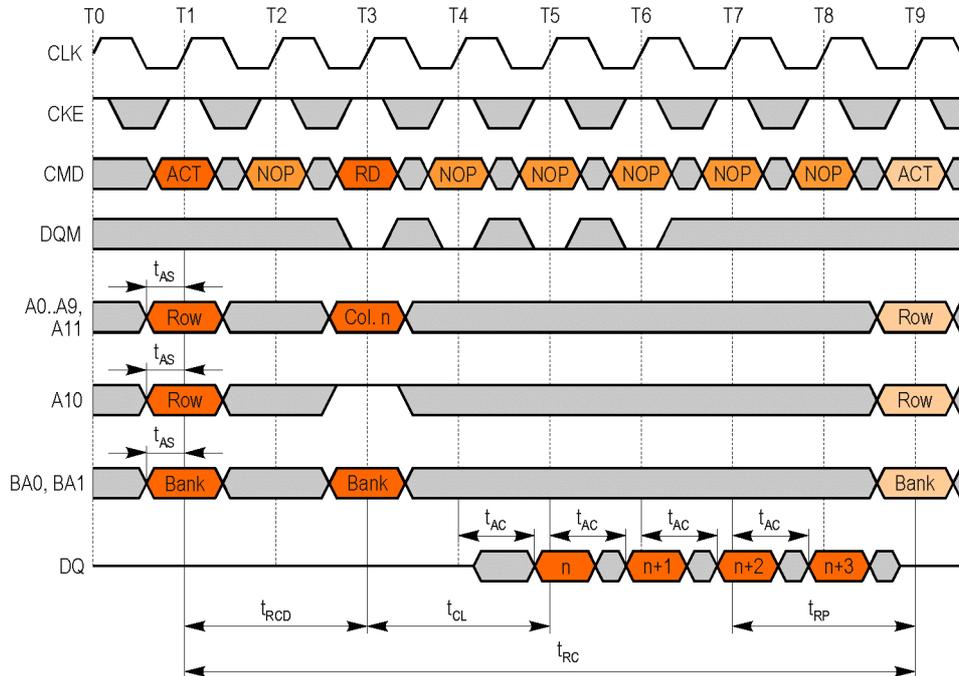
**Read, Write.** The *Read* command is used after a row activate command to initiate a burst read access to a particular row. The column address pins select the starting column. The *Write* command is used to initiate a burst write access after a row activate command. The length of the burst transfer and the *CAS* latency will be determined by the values programmed during the *Mode Register Set* command.

**Read/Write with Auto Precharge.** These commands combine a read or write operation with an automatic precharge of an individual bank without an explicit precharge command. The advantage of these commands is that the precharge is performed at the earliest time within a burst transfer.

Figure 4.15 shows the execution of a *Read with Auto Precharge* command.

At the beginning of the read operation, the row address is placed on the address lines  $A0..A11$  and the *Activate* command is placed on the command lines. The bank address lines  $BA0$ ,  $BA1$  are selecting the bank which should be activated. The addresses must be maintained for the  $t_{\text{AS}}$  (*Address Setup*) time before the rising edge of the clock. Then the column address is placed on the  $A0..A9$ ,  $A11$  address lines, so that after the *RAS*-to-*CAS* delay ( $t_{\text{RCD}}$ ) the address signals must be stable. The *Read* command is placed on the command lines and the  $A10$  address line is set to the high state to enable the auto precharge. After the *CAS* latency, the first word of memory is being read and placed in the output buffers. The next subsequent words of data are then

read on each rising edge of the clock. While the third word of data is read, the auto precharge begins and continues for a time  $t_{RP}$ . When the auto precharge finishes, another *Activate* command can be issued.



**Figure 4.15.** Execution of the *Read with Auto Precharge* command.

**Burst Terminate.** This command is used to terminate burst transfers. The most recently activated read or write command will be terminated.

**Precharge Selected Bank.** This command indicates to the active memory bank to recharge itself in order to be ready for the next access.

**Precharge All.** All banks are precharged at the same time when this command is issued.

**Auto Refresh.** The *Auto Refresh* command refreshes the SDRAM array explicitly. Refresh addresses are generated by the internal refresh controller and are incremented automatically after each refresh.

**Mode Register Set.** This command loads the mode register with information on the burst length, the CAS latency, and the order of accesses within a burst transfer (sequential or interleaved). The burst length is the maximum number of columns that can be accessed with a read or write command, and can be up to 8 columns.

### Types of SDRAM Memories

The original SDRAM memories could not operate properly at frequencies above 83 MHz. Some of the SDRAM modules even became unstable at a bus frequency of 83 MHz. To solve this problem, in 1998 Intel introduced the PC100 specification as a guideline to manufacturers for building modules that would function properly on a system bus at 100 MHz, using Intel's i440BX chipset. With the PC100 specification, Intel elaborated a number of guidelines for trace lengths and widths, spacing between them, number of PCB layers, programming specifications for the EEPROM memory (SPD), etc. An SDRAM memory conforming to this specification is called **PC100** memory. Ideally, the PC100 memory can achieve a 4-1-1-1 timing. At a frequency of 100 MHz, there are  $100 \cdot 10^6$  clock cycles per second, and a maximum of 1 word (8 bytes) can be transferred in each cycle. The maximum theoretical bandwidth is therefore 800 MB/s.

There is some confusion regarding the components that a memory module conforming to the PC100 specification should contain. There are several modules being sold as conforming to the PC100 specification, but they do not operate reliably at 100 MHz. While the chip speed rating is used most often to determine the overall performance of the chip, a number of other timings are very important. For example, the *RAS*-to-*CAS* delay or the *CAS* latency play an important role in determining the maximum bus speed the module will operate on to achieve a 4-1-1-1 timing.

Later on, Intel elaborated the specifications for the **PC133** and **PC150** memories, which can operate with stability at 133 MHz and 150 MHz, respectively. At 133 MHz, the maximum theoretical bandwidth is  $8 \cdot 133 = 1064$  MB/s, while at 150 MHz, this bandwidth is  $8 \cdot 150 = 1200$  MB/s.

### Performance Factors

As described in Section 4.4.5, the latency of a synchronous memory is usually designated by three digits, such as 3:2:2. These digits indicate the *CAS* latency, the *RAS*-to-*CAS* delay and the *RAS* precharge time, respectively, expressed in clock cycles.

One of the performance factors of the SDRAM memory is the *CAS* latency. The *CAS* latency is the delay, expressed in clock cycles, between the registration of the *Read* command and the availability of the first word of data. Usually, this latency is 2 or 3 clock cycles. According to the SDRAM specifications, the *CAS* latency can have a maximum value of 3 clock cycles.

When a burst read cycle is initiated, the address is sent to the bus and the  $\overline{RAS}$  and  $\overline{CS}$  signals are held in the logical 0 state on the next clock cycle (rising edge of the *CLK* signal), thereby activating the amplifiers of the memory bank. After the *RAS*-to-*CAS* delay, the  $\overline{CAS}$  and  $\overline{CS}$  signals are held in the logical 0 state. After the column access time ( $t_{CA}$ ), the first word of data is placed on the output lines and can be retrieved at the next clock cycle. The basic rule is that the product between the *CAS* latency ( $t_{CL}$ ) and the clock cycle time ( $t_{CLK}$ ) must be equal or greater than the

column access time:  $t_{CL} \cdot t_{CLK} \geq t_{CA}$ . This means that the column access time is the limiting factor for the CAS latency.

For example, if the system bus is running at 133 MHz (with a clock cycle time of 7.5 ns), in order to have a CAS latency of 2 clock cycles, the column access time would have to be less than 15 ns:  $2 (t_{CL}) \cdot 7.5 (t_{CLK}) = 15$  ns. Since reducing the CAS latency reduces the number of available clock cycles for the memory to place the data into the output buffers, the higher the bus frequency is, the higher the CAS latency has to be. Table 4.2 shows the allowed operating frequencies for a 64-MB SDRAM memory module from Micron Technology, when the CAS latency is 2 or 3 clock cycles.

**Table 4.2.** Allowable operating frequencies for an SDRAM module, depending on the CAS latency.

Speed	Allowable operating frequency	
	CAS Latency = 2	CAS Latency = 3
-7G	–	≤ 143 MHz
-7E	≤ 133 MHz	≤ 143 MHz
-75	≤ 100 MHz	≤ 133 MHz
-8E	≤ 100 MHz	≤ 125 MHz

For this memory module, a CAS latency of 2 clock cycles can be maintained up to a frequency of 133 MHz of the bus, while a CAS latency of 3 clock cycles can be maintained up to a frequency of 143 MHz.

When the absolute timings of an SDRAM memory and the clock frequency are known, the components of the  $x:y:z$  designation can be easily determined. Suppose a memory chip with the following timings:  $t_{CA} = 20$  ns,  $t_{RCD} = 20$  ns,  $t_{RP} = 20$  ns. With a clock frequency of 100 MHz ( $t_{CLK} = 10$  ns), the values of the three components would be  $x = t_{CA} / t_{CLK} = 2$ ,  $y = t_{RCD} / t_{CLK}$ ,  $z = t_{RP} / t_{CLK} = 2$ . Therefore, the memory designation would be 2:2:2 for a 100-MHz clock. With a clock frequency of 133 MHz ( $t_{CLK} = 7.5$  ns), the values of  $x$ ,  $y$ , and  $z$  would be  $20 / 7.5 = 2.67$ , which must be rounded up to 3, resulting a memory designation of 3:3:3.

Another performance factor of the SDRAM memory is the *clock access time*,  $t_{AC}$ . This is the time interval from the rising edge of the clock until the data is available in burst mode (Figure 4.15). The maximum value of this time interval must be lower than the clock cycle time, because some time is needed for signal stabilization (1–2 ns) and data transfer to the output pins (1–2 ns). For example, considering that a total time of 2.5 ns is needed for signal stabilization and data transfer, with a 100-MHz clock the maximum value of the clock access time is  $t_{AC} = 10 - 2.5 = 7.5$  ns, while with a 133-MHz clock this value is  $t_{AC} = 7.5 - 2.5 = 5$  ns.

Finally, another performance factor is the *random cycle time*,  $t_{RC}$ . This is the time between two successive activations of a memory bank for a burst operation (Figure 4.15). Typical values of the random cycle time are between 50 ns and 70 ns.

### Real Bandwidth

To determine the real bandwidth of the whole memory system, we need to take into account the following latencies:

1. The latency to transfer the address and the command over the front side bus (FSB) from the CPU to the chipset (1 clock cycle).
2. The latency to transfer the address and the command from the chipset to the memory module (1 clock cycle).
3. The *RAS-to-CAS* delay, needed to charge the appropriate row (2-3 clock cycles).
4. The *CAS* latency, needed to access the appropriate column (2-3 clock cycles).
5. The latency to transfer the data to the output buffer of the memory (1 clock cycle).
6. The latency to transfer the data from the output buffer of the memory to the CPU, via the chipset (2 clock cycles).

This is the total latency needed to transfer the first word (8 bytes). A PC100 SDRAM memory with a *CAS* latency of 2 and a timing of 4-1-1-1 will have a total latency of 9 clock cycles to transfer the first word. That is, 5 extra clock cycles are added to obtain the total latency. Two cycles are added to transfer the address and the command from the CPU to the chipset and the memory module, one cycle to transfer the data to the output buffer of the memory and two cycles to transfer the data to the chipset and the CPU. In each of the next clock cycles, one word can be read.

We consider a PC100 SDRAM memory designated as 2:2:2 (the *CAS* latency, the *RAS-to-CAS* delay, and the *RAS* precharge time are all equal to 2 clock cycles). We analyze the situation when a cache miss occurs, and a memory read is needed to fill a line of the cache memory. The size of a cache line is considered to be 32 bytes (4 words). Unlike the memory read operations, the write operations are not critical, because the CPU does not have to wait for a write operation to terminate. Writes can be buffered, so that the CPU can continue its work while the chipset performs the transfer of data from the write buffer to the main memory when the memory bus is less saturated.

Table 4.3 indicates the various latencies (in clock cycles) and the situations when they occur in order to read 32 bytes from the main memory.

**Table 4.3.** Latencies and maximum bandwidths for different situations.

Situation	Statistical chance	SDRAM latency	Total latency	Latency to transfer 32 bytes	Maximum bandwidth
Page hit	55 %	$t_{CL} = 2$	7	$7-1-1-1 = 10$	320 MB/s
Page miss	40 %	$t_{RCD} + t_{CL} = 4$	9	$9-1-1-1 = 12$	266.6 MB/s
Page miss and activation of a new row	5 %	$t_{RP} + t_{RCD} + t_{CL} = 6$	11	$11-1-1-1 = 14$	228.5 MB/s

The third column indicates the latency of the SDRAM memory for the situation described in the first column. For example, in case of a page miss the latency of the SDRAM memory includes the *RAS-to-CAS* delay (2 clock cycles) and the *CAS* latency (2 clock cycles). The fourth column contains the total system latency, which includes the SDRAM latency and the 5 extra clock cycles. For this example, where the read operation is performed to fill a line of the cache memory, even for a page hit the maximum real bandwidth can only reach 320 MB/s, which represents 40% of the maximum theoretical bandwidth (800 MB/s).

Table 4.4 contains a comparison between the maximum bandwidth of the PC133 SDRAM memory (with a *CAS* latency of 2 or 3) and the PC100 SDRAM memory (with a *CAS* latency of 2), in case of a page hit.

**Table 4.4.** Comparison between PC133 and PC100 SDRAM memories.

SDRAM	Total latency	Latency to transfer 32 bytes	Maximum bandwidth	Increase over PC100, CL2
PC133, CL2	7	7-1-1-1 = 10	426.5 MB/s	33 %
PC133, CL3	8	8-1-1-1 = 11	387.7 MB/s	21 %

#### 4.4.6.6. HSDRAM

The *High Speed* SDRAM (HSDRAM) uses higher quality memory chips in order to raise the frequency above that of standard SDRAM. Some chips can operate more reliably and with lower latencies at 133 MHz, and others can operate at frequencies above 133 MHz, such as 150 MHz or 166 MHz. The HSDRAM memory is compatible with standard SDRAM, while improving system stability and performance. It provides faster clock access time ( $t_{AC}$ ), shorter random access time ( $t_{RAC}$ ), and faster random cycle time ( $t_{RC}$ ). The HSDRAM memory was intended to be used in high-performance systems, including personal computers, workstations, servers, communication switches, DSP systems, and 3D graphics systems. Later on, the higher quality HSDRAM memory chips were included in other memory architectures, such as ESDRAM or DDR SDRAM.

As examples of HSDRAM memory chips, we present the main characteristics of the SM3603 and SM3604 memory devices, which were manufactured by Enhanced Memory Systems in 1999:

- Operation latencies at 133 MHz: 3:2:2 (*CAS* latency, *RAS-to-CAS* delay, *RAS* precharge time);
- Clock access time:  $t_{AC} = 4.6$  ns;
- Random access time:  $t_{RA} = 34.6$  ns;
- Random cycle time:  $t_{RC} = 52.5$  ns;
- Programmable *CAS* latency (1, 2, 3).