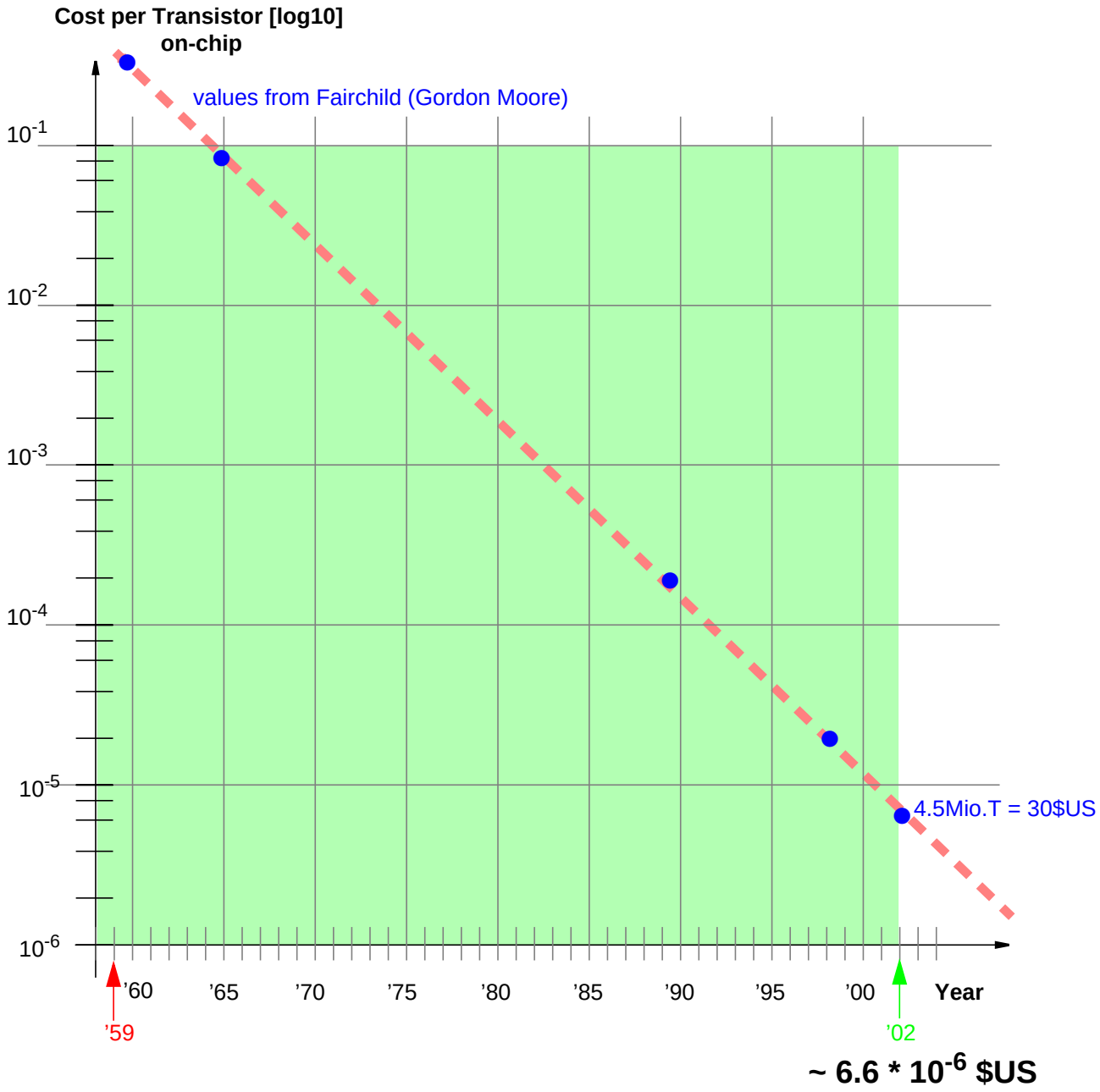
Memory Technology

Development



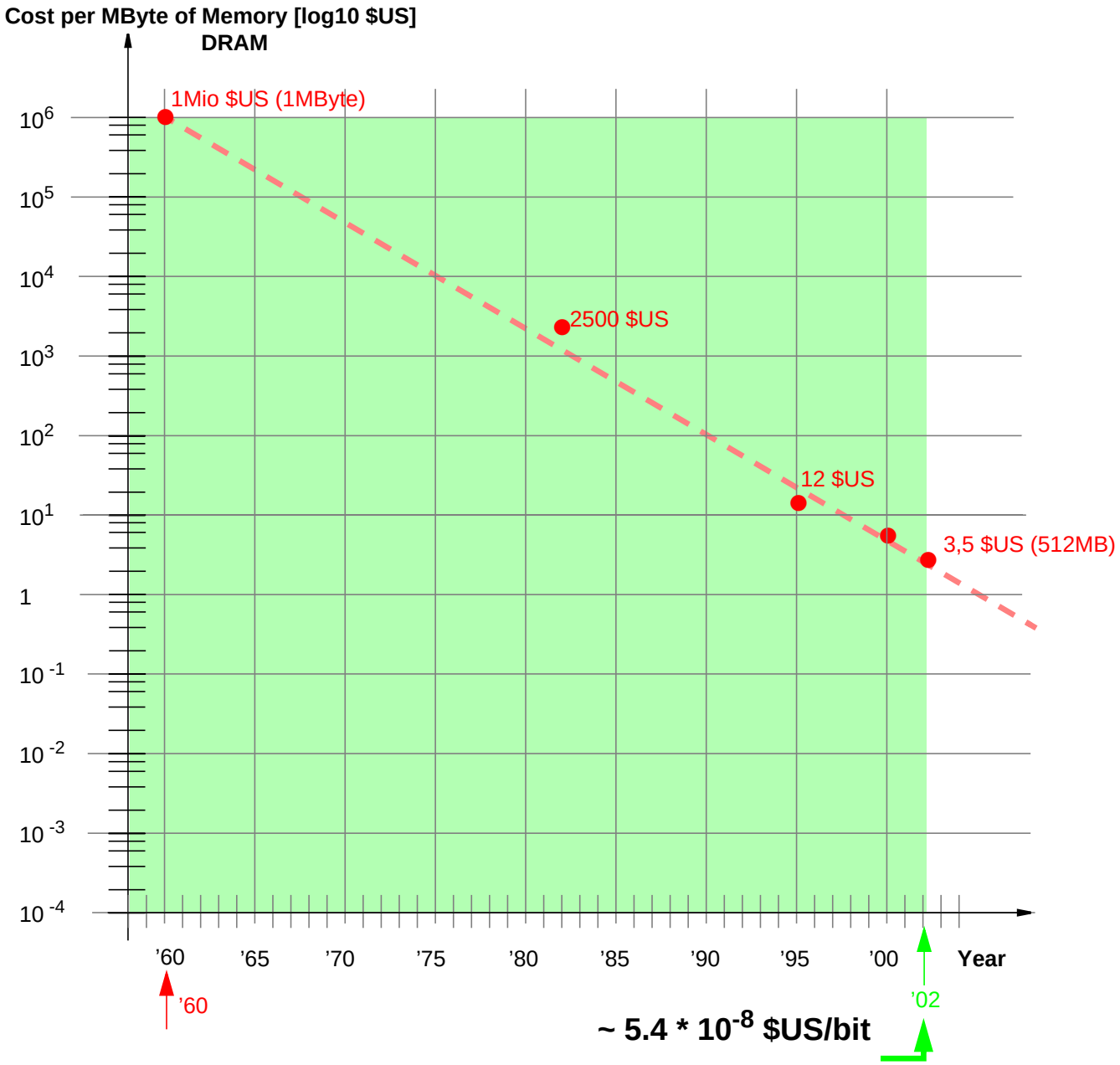
year 2002: 4.5Mio. transistors in 0.18µm CMOS technology on a 5x5mm die with BGA package ~ 30 $US = 6.6 x 10^{-6} $US per transistor; standard cell design

Memory Technology

Cost development of DRAMS

Stone, Harold S., High-Performance Computer Architecture:

"Memory chips have been quadrupling capacity every two to three years. The manufacturing cost per chip is usually constant per chip, regardless of the memory capacity per chip. When a new memory chip that has four times the capacity of its predecessor is introduced, a typical strategy is to sell it at four or five times the price of its predecessor. Although the price per bit is about equal for new and old technologies, the newer technology leads to less expensive systems because of having only one fourth the number of memory chips."



some data

1982	16kbit SRAM	~80 $US	chip mit 16Kbit 16Kx1
1982	64kbit DRAM	~20 $US	chip mit 64Kbit 64Kx1
1995	1Mbit SRAM	~50 $US	chip mit 1Mbit 1Mbx1 oder 256Kx4
1995	4Mbit DRAM	~6 $US	chips mit 4Mbit 4Mbx1 oder 1Mx4

Main Memory

The main memory is the lowest level in the semiconductor memory hierarchy. Normally all data objects must be present in this memory for processing by the CPU. In the case of a demand-paging memory, they are fetched from the disk storage to the memory pages on demand before processing is started. Its organization and performance are extremely important for the system's execution speed.

The advantages of DRAMs in respect of cost and integration density are much more important than the faster speed provided by SRAMs.

	SRAM	DRAM
access time	**10 - 100 ns**	**t_{rac} 60 - 100 ns t_{cac} 20 - 30 ns t_{cyc} 110 - 180 ns**
power consumption	**200 - 1300 mW**	**300 - 600 mW**
memory capacity	**< 1 Mbit**	**< 4 Mbit**
price	**$ 50/Mbit**	**$ 6/Mbit**

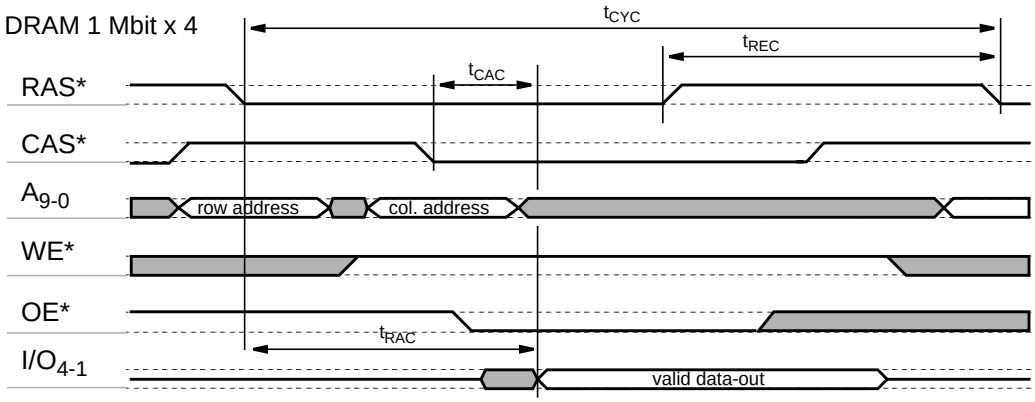
t_{rac} row-address access time
t_{cac} column-address access time
t_{cyc} cycle time

Important parameters of the DRAM are:

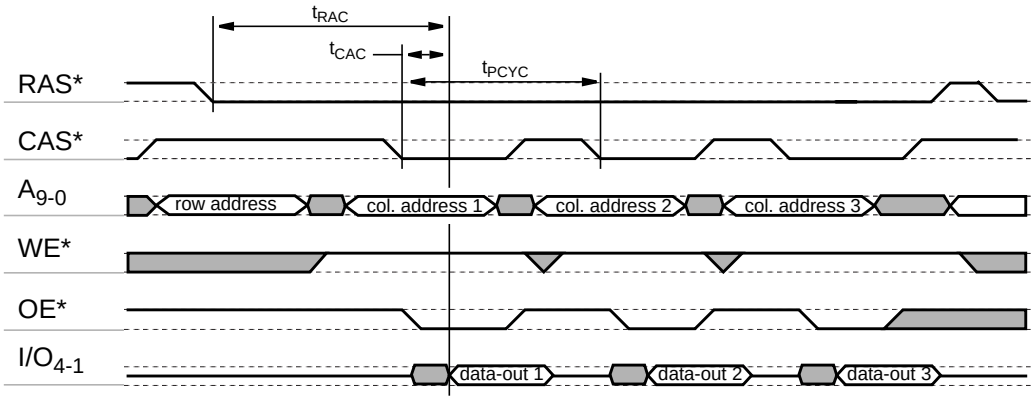
- RAS access time - t_{RAC}; typ. 80ns
  the time from row address strobe (RAS) to the delivery of data at the outputs on a read cycle
- CAS access time - t_{CAC}; typ. 20ns
  the time from column address strobe (CAS) to the delivery of data at the outputs on a read cycle
- RAS recovery time - t_{REC}; typ. 80ns
  after the application of a RAS pulse, the DRAM needs some time to restore the value of the memory cell (destructive read) and to charge up the sense lines again, until the next RAS pulse can be applied, this time allows the chip to 'recover' from an access
- RAS cycle time - t_{CYC}; typ. 160ns
  is the sum of the RAS access time and the recovery time and defines the minimum time for a single data access cycle
- CAS cycle time - t_{PCYC}; typ. 45ns
  the time from CAS being activated to the point at which CAS can be reactivated to start a new access within a page

Main Memory

The address lines of DRAMs are multiplexed to save pins and keep the package of the chip small. The partitioning of the address into the Row Address RA and the Column Address CA necessitates a sequential protocol for the addressing phase. The names ‘row’ and ‘column’ stem from the arrangement of the memory cells on the chip. The RA is sampled into the DRAM at the falling edge of RAS*, then the address is switched to the CA and sampled at the falling edge of CAS* (CAS*↓).



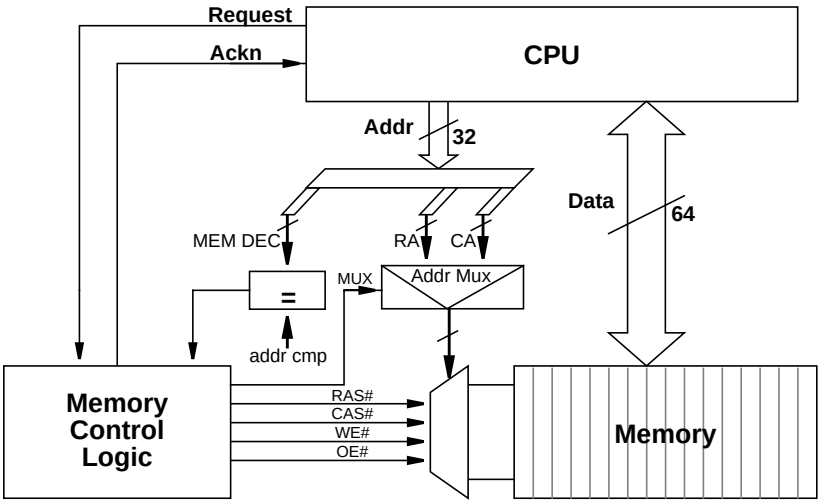
single read cycle

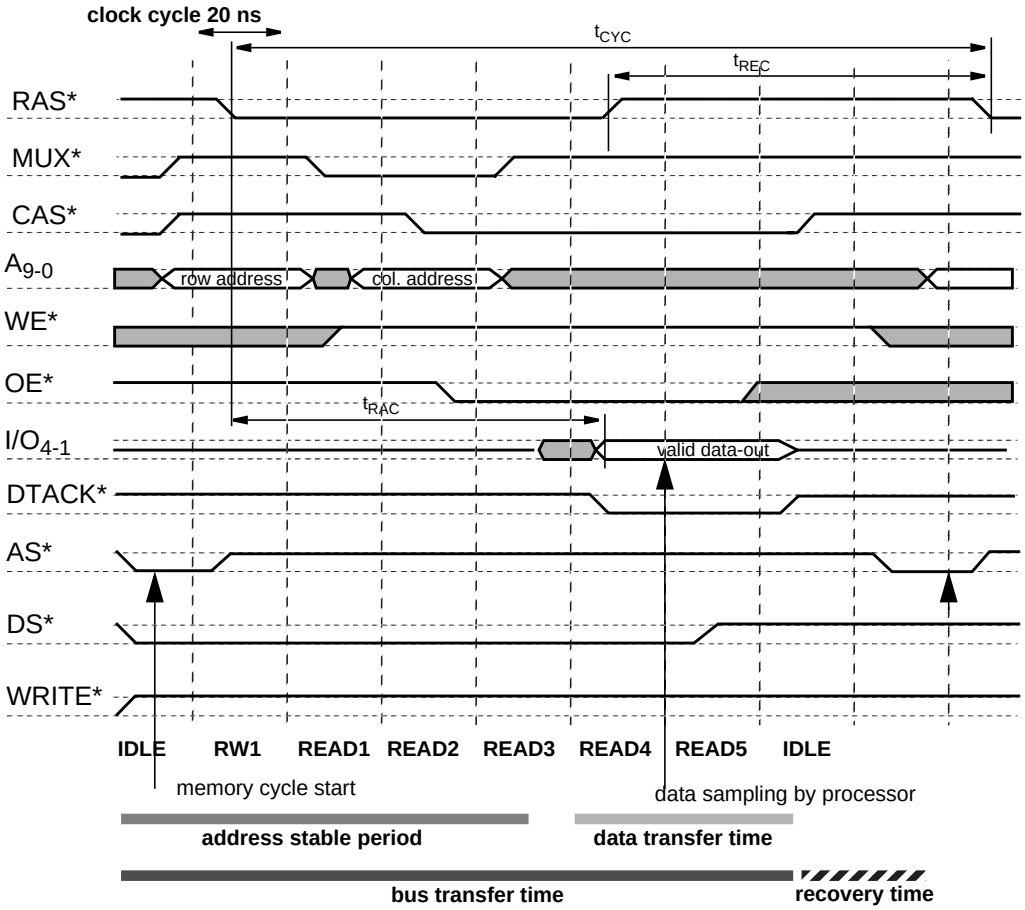


page mode cycle

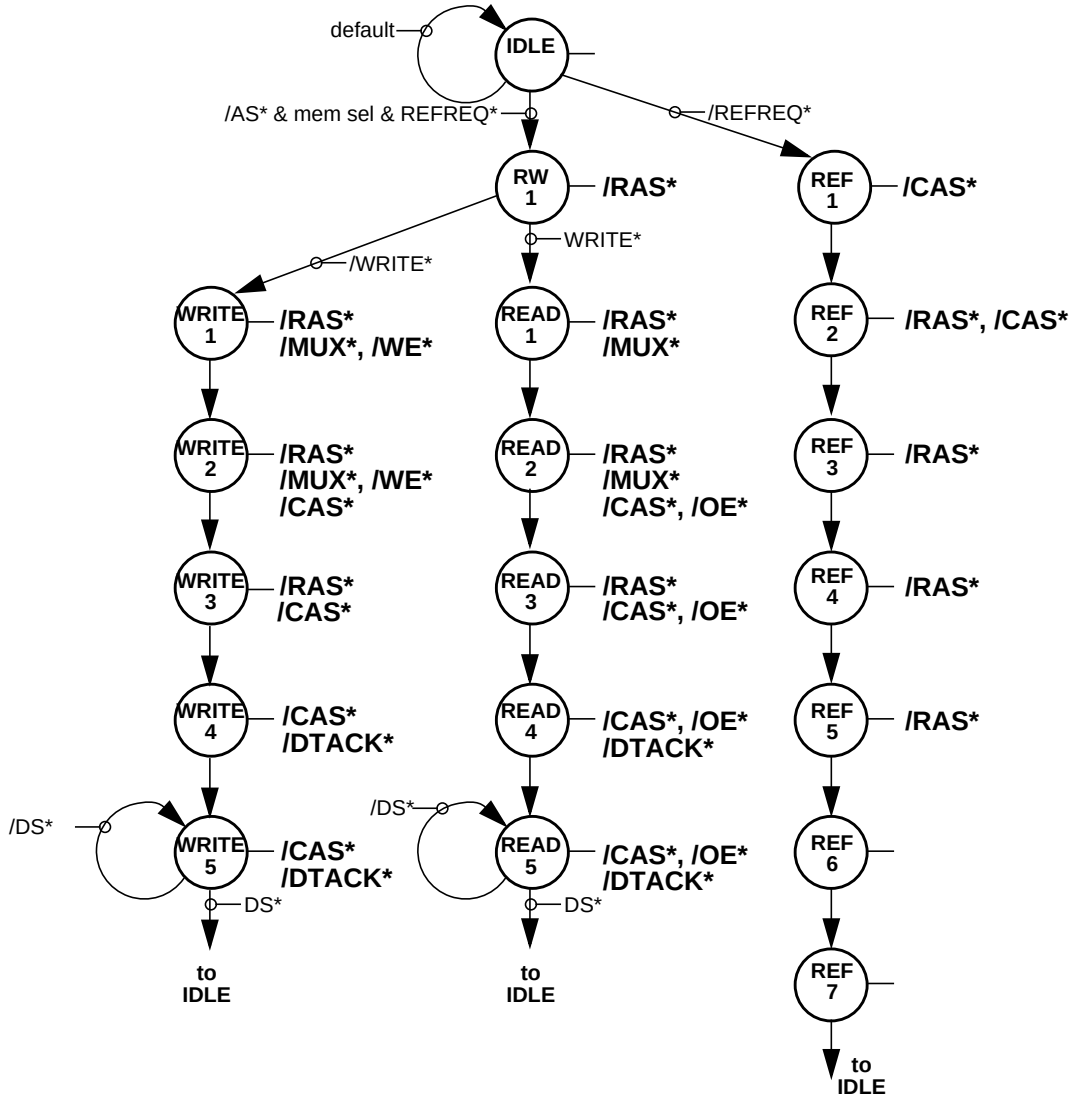
Word-Wide Memories

The simplest form of memory organization is the word-wide memory, matching the bus width of the external processor interface.





State Machine of Simple DRAM



Word-Wide Registered Memories

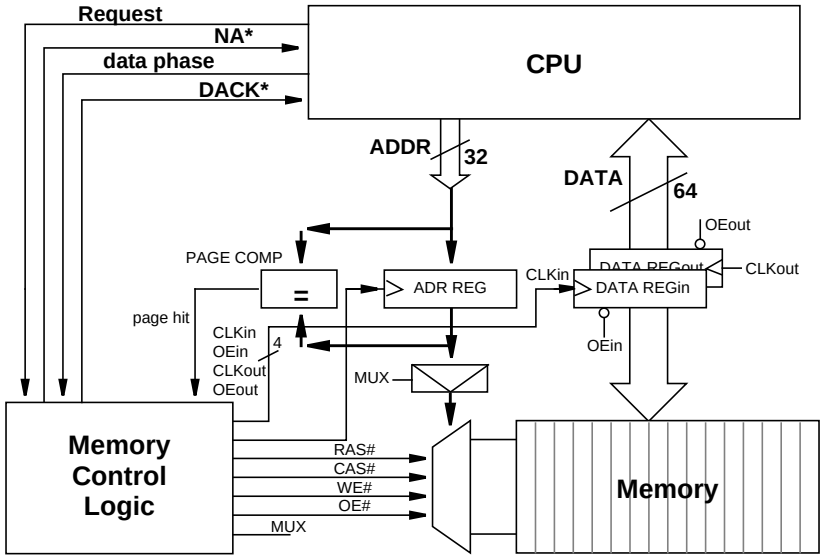
High-performance processors need more memory bandwidth than a simple one-word memory can provide. The access and cycle times of highly integrated dynamic RAMs are not keeping up with the clock speed of the CPUs.

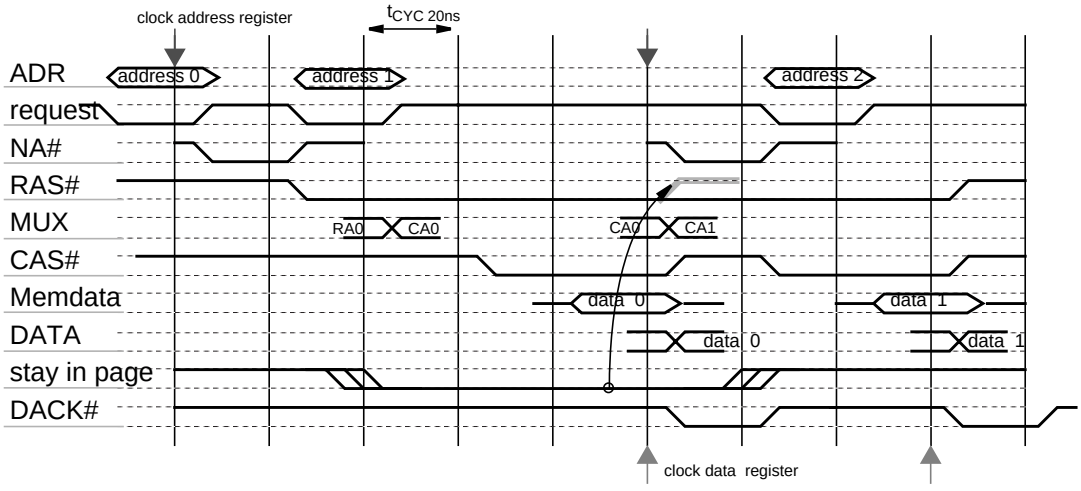
Therefore, special architectures and organizations must be used to speed-up the main memory system.

The design goal is to transport one data word per clock via the external bus interface of the CPU to or from the main memory.

This sort of performance cannot be obtained by a one-word memory.

The two basic principles for enhancing speed - **pipelining** and **parallel processing** - must also be applied to the main memory. Pipelining of a one-word memory involves attempting to divide the memory access operation into suboperations and to overlap their execution in the different stages of the pipeline. The subdivision of the memory cycle into two suboperations - the addressing phase and the data phase - allows pipelining of the bus interface and the memory system. If there are separate handshake signals for address and data, several transfers can be active at different phases of execution.





New DRAM Architectures

New generations of special DRAM devices are being designed to support fast block-oriented data transfers and page-mode accesses. Four different types of devices are under development by various manufacturers.

Type of dynamic RAM	Enhanced	Cache	Synchronous	Rambus
I/O width, bits	X1,X4	X4	X4,X8,X9	X8,X9
Data rate, single hit, MHz	67	50-100	50-100	500
First-access latency Cache/bank hit, ns Cache/bank miss, ns	 15-20 35-45	 10-20 70-80	 30-40 60-80	 36 112
Cache-fill bandwidth, MB/s	7314	114	8533 (a)	9143 (a)
Cache/bank size, bits	2048	8192	4096 (a)	8192 (a)
Area penalty, percent (b)	5	7	5-10	10-20
Output level	CMOS/TTL	CMOS/TTL	CMOS/TTL, GTL/CTT	600mV swing, terminated
Access method	Asynchronous DRAM-like	Synchronous proprietary	Synchronous pulsed/RAS	Synchronous proprietary
Access during refresh	Yes	Yes	Undecided	No
Pin count/package	28/SOJ	44/TSOP	44/TSOP	32/VSMP
Density, bits	4M	4M	16M	4M

SOJ= small-outline J-lead package; TSOP= thin small-outline package;
VSMP= nonstandard vertically mounted package

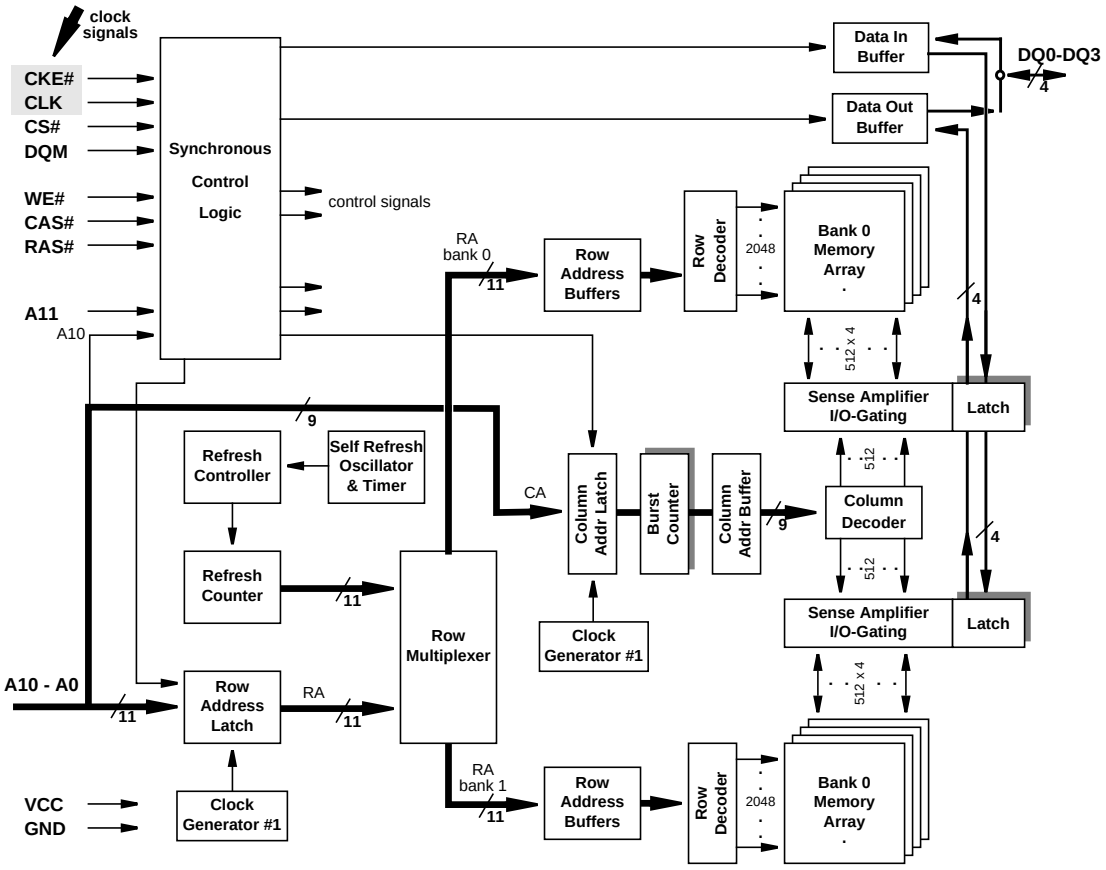
(a) Synchronous and Rambus DRAMs store data in sense amplifier latches, not in separate synchronous RAM caches.

(b) Area penalty is relative to the manufacturer's standard die size, so that the figures are not directly comparable.

Synchronous DRAM

The **SDRAM** device latches information in and out under the control of the system clock. The information required for a memory cycle is stored in registers; the SDRAM can perform the request without leaving the CPU idle at the interface. The device responds after a programmed number of clock cycles by supplying the data. With registers on all input and output signals, the chip allows pipelining of the accesses. This shortens its average access time and is well suited to the pipelined interfaces of modern high-performance processors like the i860XP. The interface signals are common CMOS levels, which appear to restrict the data rate to 100Mbit/s (1bit devices). A JEDEC approval procedure is currently in progress.

Two internal memory banks support interleaving (see Section 4.3.2) and allow the precharge time of one bank to be hidden in an access of the other bank. In the same way, the refresh can be directed to the second bank while accessing the first one. The built-in timer for the refresh period and the refresh controller can hide the refresh inside the chip. The 512 x 4 sense amplifier can hold the data of one page like a cache, and all accesses to the data within the page are very fast.



Functional Block Diagram of 16 Mbit Synchronous DRAM

The 16Mbit SDRAM contains 4 banks which are not explicitly marked !!!

Burst Mode Memory

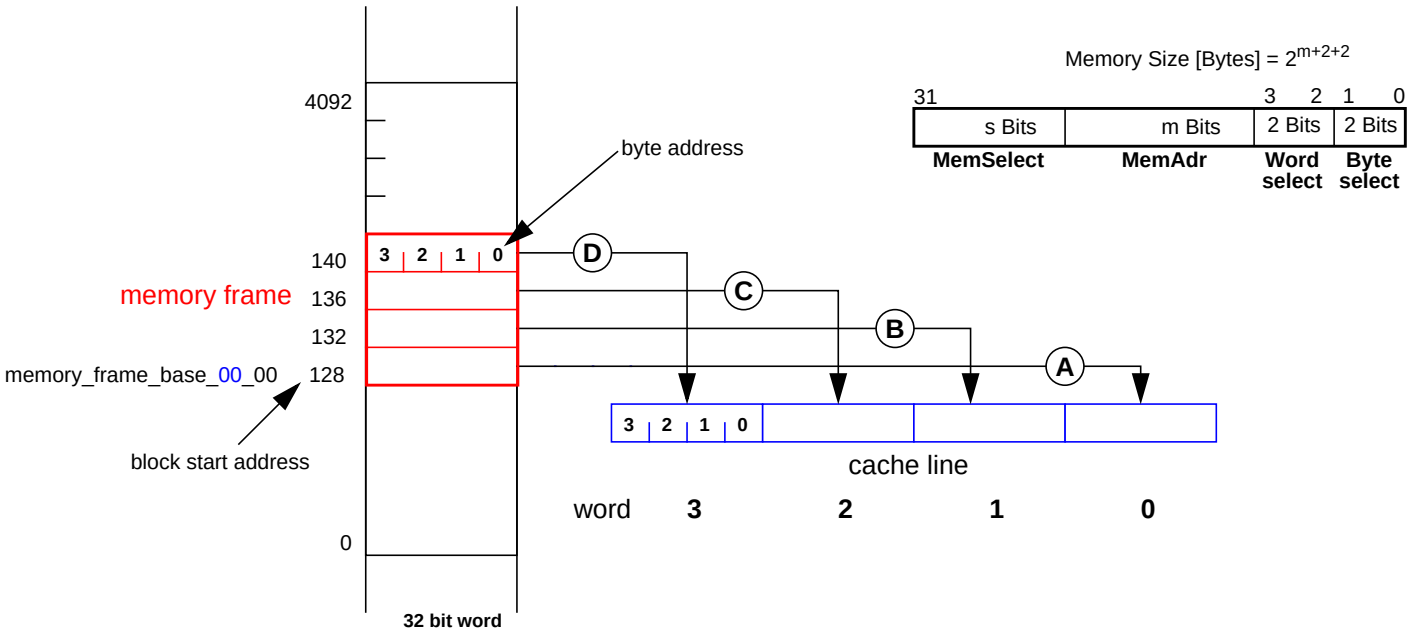
Bei einem Zugriff auf den Hauptspeicher werden mehr Daten geholt, als die Wortbreite des Speichers liefert. Bei diesem "Burst Transfer" werden nacheinander mehrere (2^n ; typically n=2 or 3) Werte gelesen oder geschrieben.

Vorteil: Durch die Ankündigung (burst mode control signal) eines solchen Transfers ist es möglich, die weiteren Daten vorausschauend und im Pipeline-Modus zu holen und damit eine wesentlich höhere Datentransferleistung zu erbringen. Hierbei wird der "page mode" des Speichers ausgenutzt.Die Bezeichnung des Transfers erfolgt häufig nach folgender Syntax: (L:B:B:B) - (5:1:1:1). Die Notation bedeutet eine Startlatenz von 5 Takten gefolgt von weiteren Daten jeden weiteren Takt.

Es gibt unterschiedliche Festlegungen für die Adreßsequenz innerhalb eines Burst-Zugriffs.

- linearer Burst A B C D
- modulo Burst
  - upcounting A B C D | B C D A | C D A B | D A B C
  - interleaved A B C D | B A D C | C D A B | D C B A

Probleme entstehen, wenn die Startadresse des Burst Cycles nicht auf einer Startadresse liegt, die ’aligned’ ist, oder der Burst die page der Speichers (oder der MMU) überschreitet. Aus diesem Grund werden oft Einschränkungen bei der Adreßsequenz vorgenommen.



Die Startadresse sollte bei einem Cache Line Fill möglichst das Wort sein, welches als erstes vom Prozessor benötigt wird. Dadurch enstehen Burst cycles mit ’missaligned’ Startadressen. Dafür verwendet man dann meist einen modulo Burst Zugriff.

Interleaved Memories

The next step is to apply parallel processing to the main memory. This solution has long been employed in high-performance vector supercomputers such as the CRAY series in the form of memory interleaving.

The parallel processing made use of interleaving requires partitioning of the memory system into parallel memory banks, which are controlled by a local bank controller. The global interleave controller checks and controls the interaction between the CPU and memory banks. The number of memory banks defines the order of interleaving (the CRAY-1 memory system for example, contains 32 banks and is therefore described as 32-way-interleaved). Usually the number of banks for interleaving is to the power of 2.

The one-word memory with data and address registers forms the basic hardware structure of each bank.

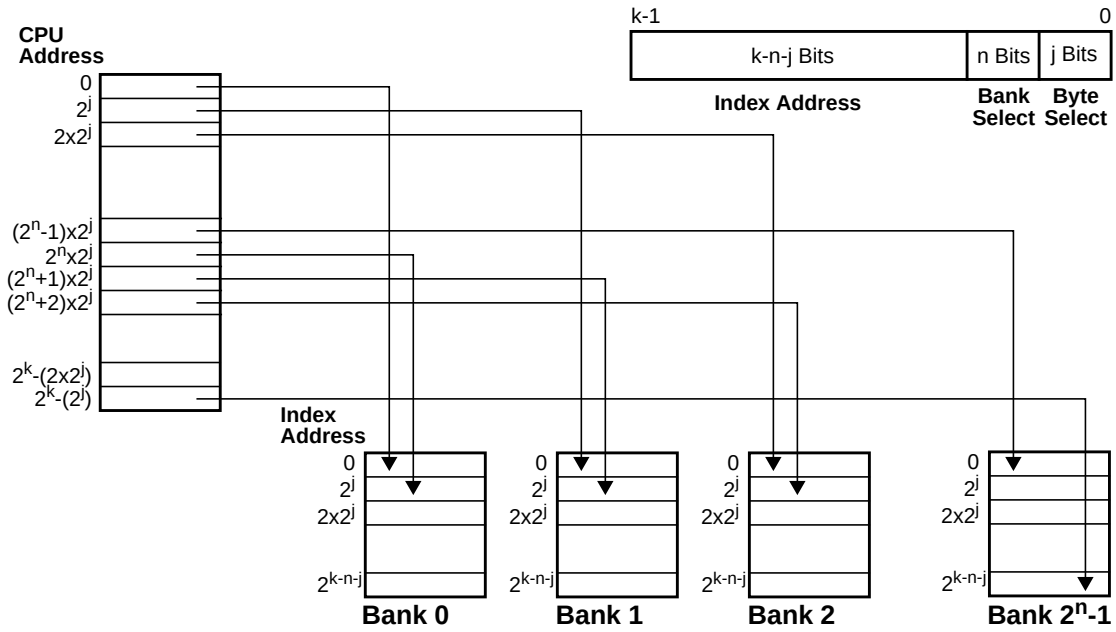
Two basic forms of interleaving can be distinguished:

- low-order interleaving
- high-order interleaving

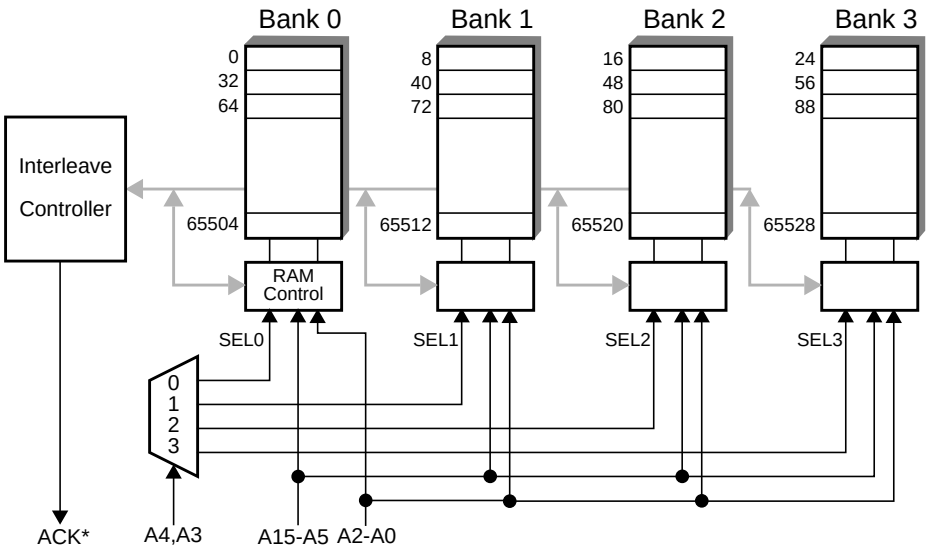
Low-order interleaving assumes that the least significant bits of the address A are used to distinguish the banks. The selection of a bank B is performed by the modulo function B = A mod n, where n is the number of banks. The performance gain achieved by a low-order interleaved memory depends on the address pattern applied to the memory and on the number of banks. A linear sequence of the addresses, selecting one bank only for every n^{th} access, increases the available bandwidth by n, compared with a word-wide memory. However, if the access function references the same bank, the bandwidth is equal to that of the word-wide memory. Depending on the access function, the performance gain lies between these two extremes. The burst-mode access fetching four consecutive (or specially sequenced) data values also fits in well with low-order interleaving. The fetch can be performed as one access to all banks in parallel, and the sequential data transport from the registers to the external bus interface is controlled by the data-path controller of the memory system. The memory can execute a new request in the addressing phase while the data phase is active. This requires that the microprocessor overlaps or pipelines the address and data phase on the bus.

High-order interleaving uses the most significant bits of the memory address to select the banks. For this structure, the next memory address must be ‘very far away’ from the previous one and should have distinct high-order bits, so that the access can be scheduled to different banks. An address pattern of this sort is highly application-dependent, and this makes the utilization of high-order interleaving rather difficult.

Low-Order Interleaved Memories

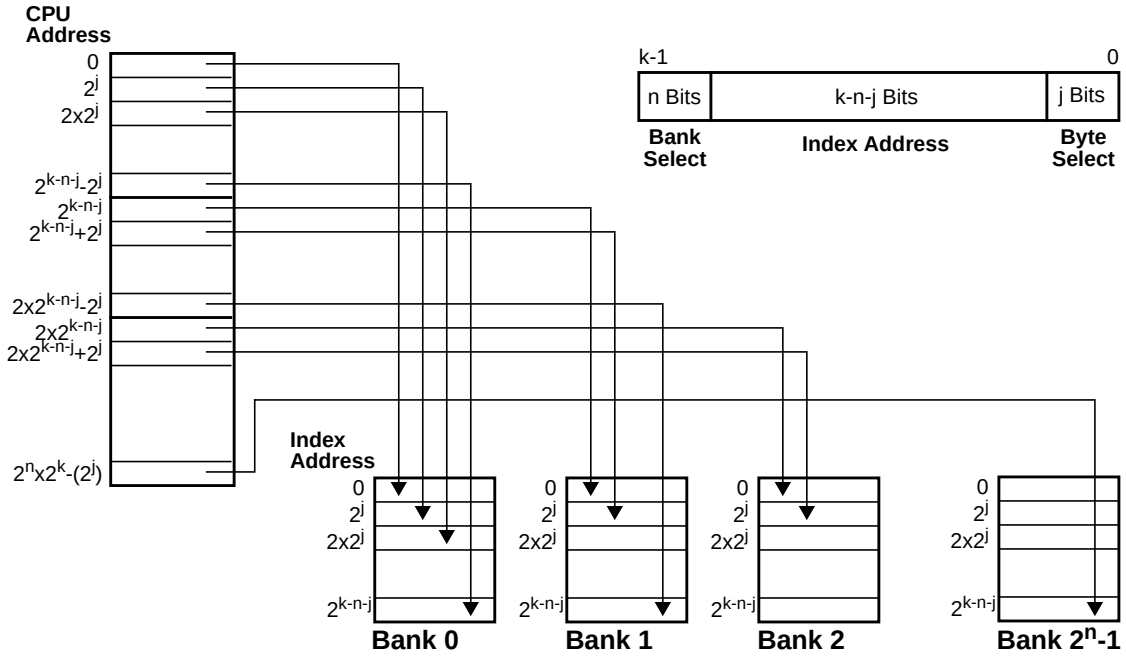


mapping from CPU address to physical memory bank

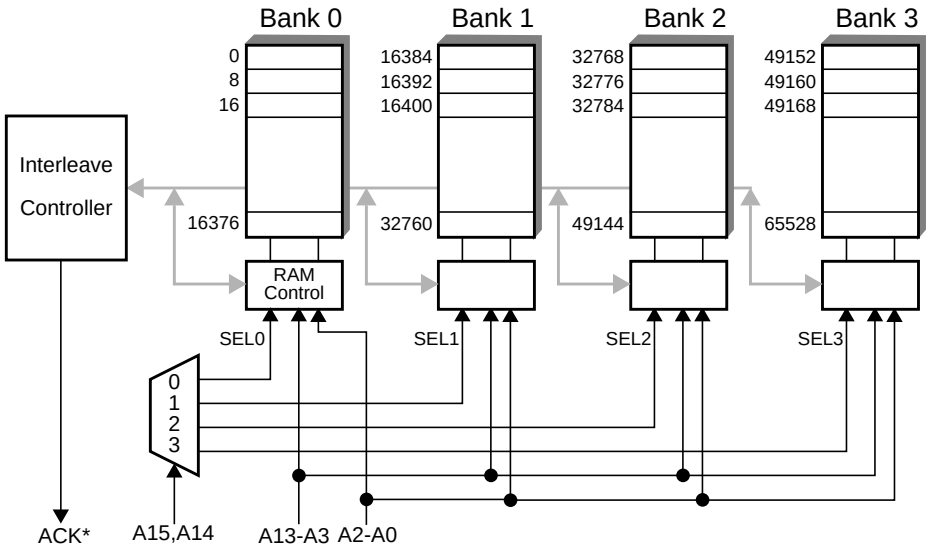


hardware structure of the addressing path (no data path)

High-Order Interleaved Memories



mapping from CPU address to physical memory bank



hardware structure of the addressing path (no data path)

Four-Bank Low-Order Interleaved Memories