

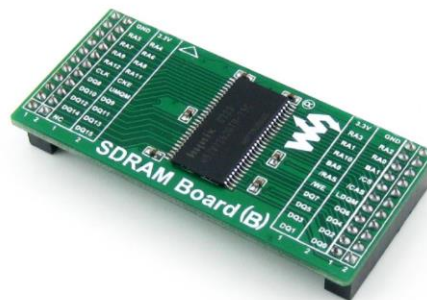
SDRAM

1. Introduction to SDRAM

Synchronous Dynamic Random Access Memory is a widely used storage device at present. It has the advantages of low price, fast data reading speed, and large storage capacity. Synchronize means that it adds a dual bank function on the basis of DRAM and so that it can read data faster.[1] Because SDRAM uses capacitors to store data, and capacitor discharge will cause data loss. Therefore, it is necessary to constantly refresh the memory cells of the SDRAM to keep data from being lost. This is also why it is called 'dynamic'.

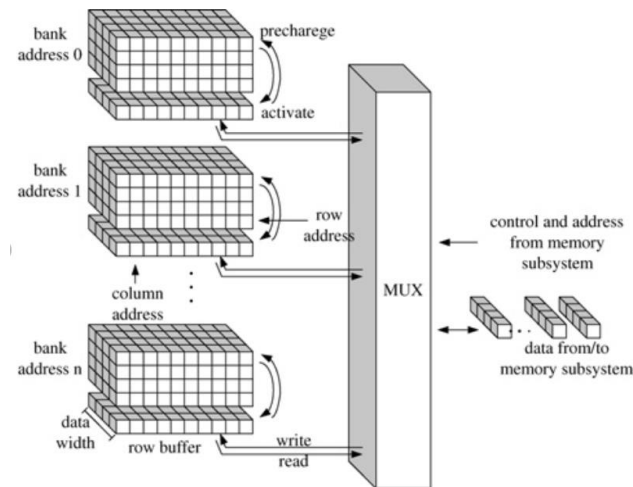
The history of SDRAM can be traced back to SRAM. SRAM (Static Random Access Memory) uses six transistors to form a memory cell. [2] However, the expensive price of transistors hindered the development of SRAM. The subsequent SDRAM used only one transistor and one capacitor in a memory cell, which greatly reduced the cost. However, the discharge phenomenon of the capacitor makes it a volatile memory, and the capacitor that stores data usually loses its stored data every 64ms. The device needs to refresh each storage unit of the sdram to ensure that the data will not be lost. The subsequent birth of DDR SDRAM uses a pre-stored double data method to enable it to transmit data on both the rising and falling edges of the driving clock. This makes the data throughput of DDR (double data rate) SDRAM double the amount of SDRAM (When the system clock frequency remains unchanged). The subsequent DDR2, DDR3, and the latest DDR4 technologies have optimized the data reading method to achieve lower working voltage, lower energy consumption and faster data reading speed.

Below shows the outlook of an sdram module:[3]



2. Structure of SDRAM

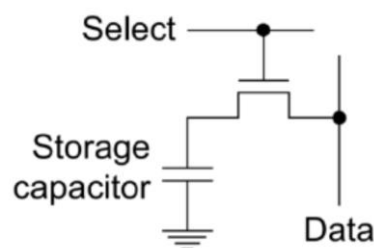
The diagram below shows the structure of an SDRAM[4]



Usually the storage array inside the SDRAM is divided into multiple banks to better manage them. At the same time, different banks can use the same address to refer to different memory cells, which greatly reduces the required address line bit width. When the data to be written arrives at the interface, the bank address needs to be given to specify which bank the data will be written to.

A bank of SDRAM is an array composed of many memory cells, and each memory cell has its unique address (row and column address) in the bank. When storing data into SDRAM, the address line needs to give the address where the data is stored. Generally, SDRAM needs to be activated to the active state in advance. In this state, the address port needs to give the row address of the stored data, and the bank port needs to give the bank address where the stored data is located. When the corresponding row of memory cells is activated, the SDRAM will enter the write state. At this time, the address port needs to give the column address of the stored data. Then, the location of the stored data in the SDRAM will be determined, and the data on the data line will be recorded.

The structure of a memory cell of SDRAM is shown in the diagram below:[5]



It can be seen from the figure that when a memory cell is selected, its corresponding address line will also be pulled high, so that the transistor is turned on, and the amount of charge stored in the capacitor in the memory cell will be determined by the stored data. Multiple such cells will form a 'word'. The data bit width of general SDRAM is 16 bits, which also means that their data line bit width is also 16 bits. The storage capacity of a SDRAM system is shown in the formulas below:

Number of storage units

$$= \text{number of rows} \times \text{number of columns} \times \text{number of banks}$$

Number of storage bits

$$= \text{input data width} \times \text{number of rows} \times \text{number of columns} \\ \times \text{number of banks}$$

3. Pre-charge

Unlike the refresh operation, SDRAM also requires a pre-charge operation. After a data is written, if the device needs to address another row, the current activated row needs to be closed, and at the same time, the voltage of all row address lines needs to be charged to $V_{cc}/2$ and wait for the next row address's arrival. This operation is called pre-charging. When data comes in, its voltage will be compared with $V_{cc}/2$. If it is greater than $V_{cc}/2$, it will be treated as data '1'. Conversely, if it is less than $V_{cc}/2$, it will be treated as data '0'. Most of the mainstream SDRAMs currently on the market support automatic pre-charge operation.

4. Burst write/read

The addressing process of SDRAM will consume a lot of time. Therefore, if the data being written is continuous, writing data continuously can save a lot of time. When a row is input, SDRAM can choose to continuously write data to multiple column addresses (just add one to the column address, and then write the next data), which reduces the number of addressing. Of course, there are many situations that will interrupt the write/read operation, such as: 1. The data has been written/read. 2. All storage units in this row have been used up and the next row needs to be activated. At this time, the system state machine will jump to the ACT state. 3. Refresh request signal is received. The system needs to refresh all SDRAMs every period of time to ensure that data will not be lost. Therefore, the priority of read/write operations will be lower than refresh operations.

5. SDRAM workflow

Used SDRAM manual: [HY57V281620ET-H 中文资料 - 百度文库 \(baidu.com\)](#)

The SDRAM used in this project is HY57V281620ET-H. Since the SDRAM has multiple different working states, the SDRAM driver of this project will use a finite state machine in the top level entity to arbitrate the different states to issue commands in different states (such as refresh, pre-charge, Read and write operations, etc.). A working state transition diagram of SDRAM is as follows[6]:

