

CHƯƠNG 2: CẤU TRÚC PHẦN CỨNG HỆ THỐNG NHÚNG

Bài 4: Bộ nhớ

cuu duong than cong. com

cuu duong than cong. com

Tổng quan

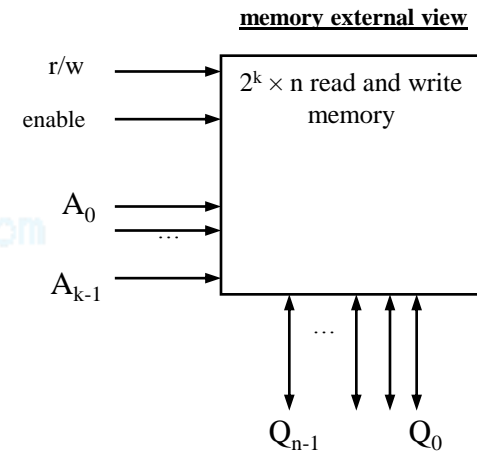
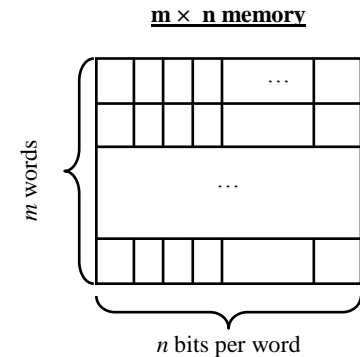
- Khả năng ghi dữ liệu và chất lượng lưu trữ của bộ nhớ
- Các kiểu bộ nhớ chung
- Ghép bộ nhớ
- Phân cấp bộ nhớ và Cache
- RAM cải tiến

Giới thiệu

- Chức năng của các hệ nhúng
 - Xử lý
 - Bộ xử lý
 - Biến đổi dữ liệu
 - Lưu trữ
 - Bộ nhớ
 - Khôi phục dữ liệu
 - Truyền thông
 - Bus
 - Chuyển dữ liệu

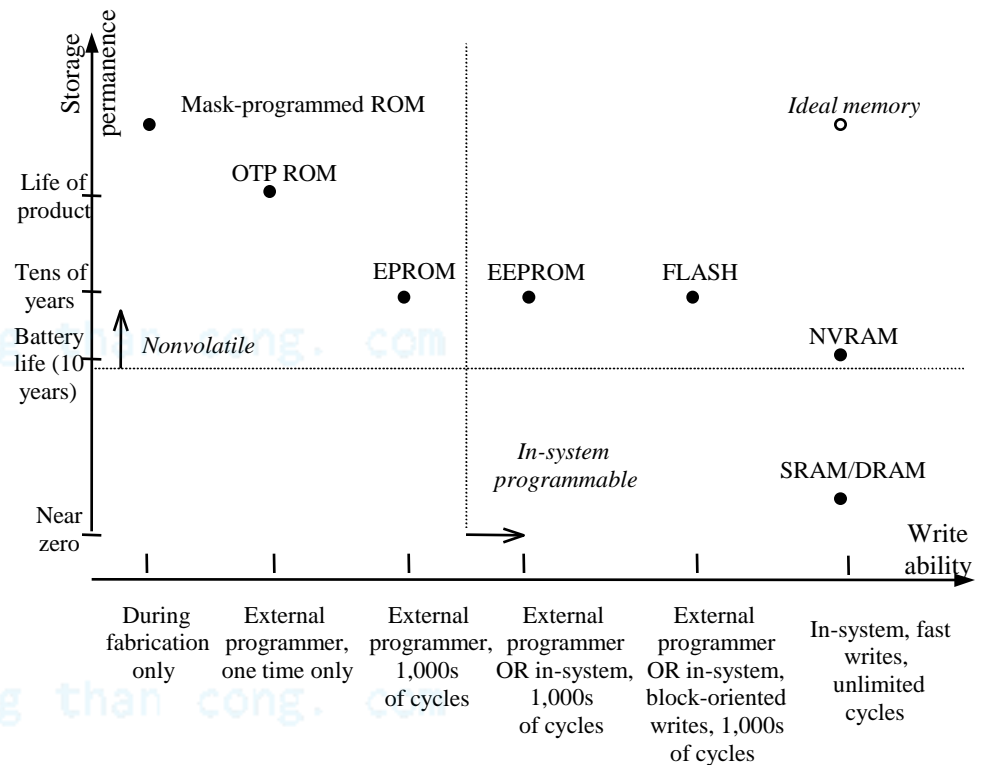
Bộ nhớ: khái niệm cơ bản

- Lưu trữ số lượng bit lớn
 - $m \times n$: m từ mỗi từ n bit
 - $k = \log_2(m)$ tín hiệu địa chỉ đầu vào
 - Hoặc $m = 2^k$ từ
 - VD: bộ nhớ 4,096 x 8:
 - 32,768 bits
 - 12 tín hiệu địa chỉ
 - 8 tín hiệu I/O
- Truy cập bộ nhớ
 - r/w: lựa chọn đọc hoặc ghi
 - enable: chỉ cho phép khi tích cực
 - multiport: đa truy cập tới nhiều vị trí khác nhau đồng thời



Khả năng ghi/chất lượng lưu trữ

- Phân biệt ROM/RAM truyền thống
 - ROM
 - Chỉ đọc, bit được lưu trữ không cần nguồn
 - RAM
 - Đọc và ghi, mất dữ liệu nếu không có nguồn
- Phân biệt truyền thống đã thay đổi
 - ROMs cải tiến có thể ghi
 - e.g., EEPROM
 - RAMs có thể lưu trữ không cần nguồn
 - e.g., NVRAM
- Khả năng ghi
 - Tốc độ để một bộ nhớ được ghi
- Chất lượng lưu trữ
 - Khả năng mà bộ nhớ lưu trữ dữ liệu sau khi nó được ghi



Khả năng ghi và chất lượng lưu trữ của bộ nhớ

Khả năng ghi

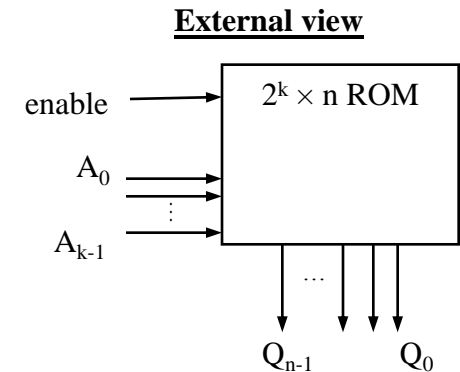
- Phạm vi của “khả năng ghi”
 - Mức cao
 - Bộ xử lý ghi vào bộ nhớ một cách đơn giản và nhanh chóng
 - e.g., RAM
 - Mức trung bình
 - Bộ xử lý ghi vào bộ nhớ, nhưng chậm
 - e.g., FLASH, EEPROM
 - Mức thấp hơn
 - Các thiết bị đặc biệt, “bộ lập trình” phải được sử dụng để ghi bộ nhớ
 - e.g., EPROM, OTP ROM
 - Mức thấp
 - bits chỉ được lưu trữ trong quá trình sản xuất
 - VD: ROM lập trình được bằng mặt nạ
- Bộ nhớ có thể lập trình được trong hệ thống
 - Có thể được ghi bởi bộ xử lý trong hệ thống nhúng
 - Các bộ nhớ loại này có khả năng ghi cao hoặc trung bình

Chất lượng lưu trữ

- Phạm vi chất lượng lưu trữ
 - Loại cao
 - Loại không bao giờ mất bit
 - VD: mask-programmed ROM
 - Loại trung bình
 - Có khả năng lưu trữ bit nhiều ngày, nhiều tháng, hoặc nhiều năm sau khi tắt nguồn
 - VD: NVRAM
 - Loại trung bình thấp
 - Có khả năng lưu trữ bit khi có nguồn cung cấp
 - VD: SRAM
 - Loại thấp
 - Mất bit gần như ngay sau khi được ghi
 - VD: DRAM
- Bộ nhớ không thay đổi được
 - Lưu trữ bit ngay cả khi không được cấp nguồn
 - Chất lượng lưu trữ cao hoặc trung bình

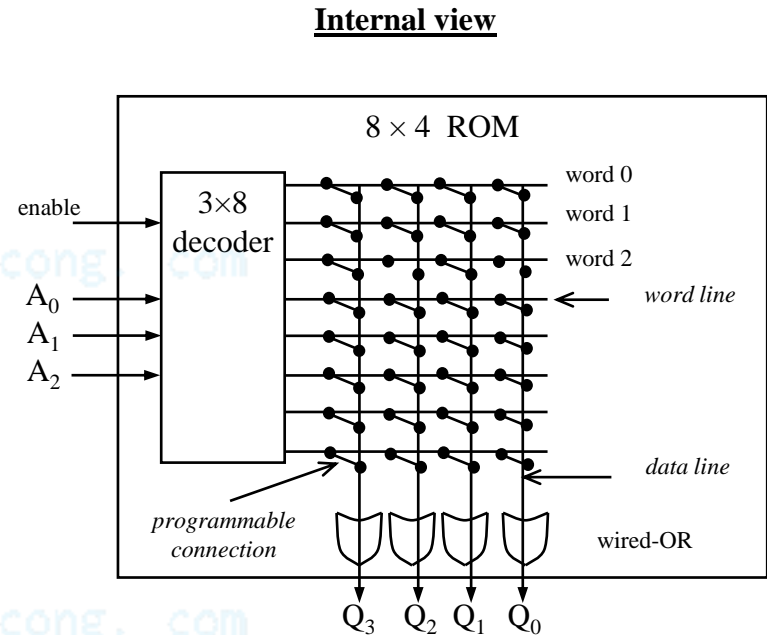
ROM: Bộ nhớ “chỉ đọc”

- Là bộ nhớ không thay đổi được
- Có thể đọc nhưng không thể ghi bởi một bộ xử lý trong hệ thống nhúng
- Thường được ghi bằng cách “lập trình”, trước khi tích hợp trong hệ thống nhúng
- Sử dụng
 - Lưu trữ chương trình phần mềm cho bộ xử lý chức năng chung
 - Lệnh trong chương trình có thể là 1 hoặc nhiều từ nhớ trong ROM
 - Lưu trữ các dữ liệu cố định
 - Thực hiện các mạch tổ hợp



Ví dụ: 8 x 4 ROM

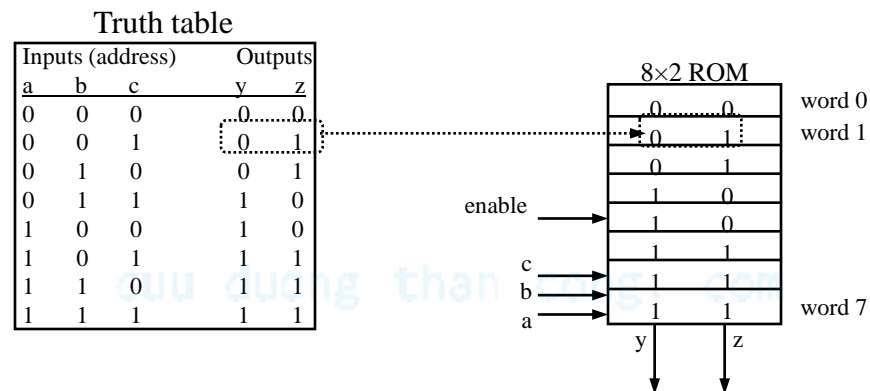
- Hàng ngang = từ
- Hàng đứng = dữ liệu
- Các đường kết nối ở giao điểm
- Bộ giải mã nối đường dẫn của từ số 2 là 1 nếu địa chỉ đầu vào là 010
- Đường dữ liệu Q_3 và Q_1 đặt là 1 bởi vì có một kết nối “được lập trình” với đường của từ số 2
- Từ số 2 không được kết nối với đường dữ liệu Q_2 và Q_0
- Đầu ra là 1010



Thực hiện mạch tổ hợp

- Bất cứ mạch tổ hợp nào với n hàm có cùng k biến đều có thể được thực hiện bằng ROM $2^k \times n$

cuu duong than cong. com



ROM lập trình bằng mặt nạ

- Các kết nối được “lập trình” khi sản xuất
 - Thiết lập các mặt nạ
- Khả năng ghi thấp nhất
 - Chỉ ghi một lần
- Chất lượng lưu trữ cao nhất
 - Bít không bao giờ thay đổi
- Được sử dụng điển hình cho các thiết kế cuối cùng của hệ với dung lượng lớn
 - Giá NRE cao nếu sản xuất với số lượng ít

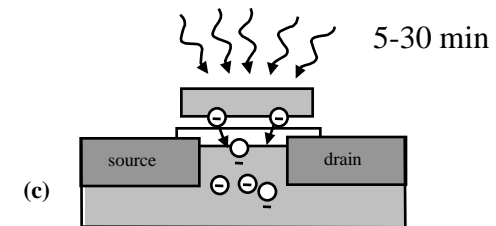
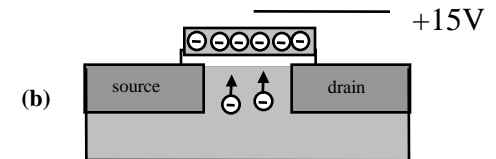
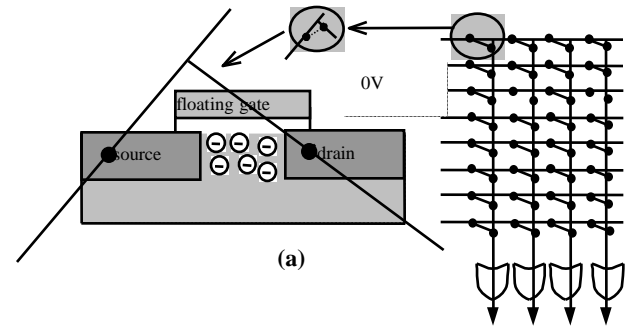
OTP ROM: ROM lập trình một lần

- Các kết nối được “lập trình” sau khi sản xuất bởi người sử dụng
 - Người sử dụng cung cấp nội dung yêu cầu lưu trữ trong ROM
 - Nội dung được đưa vào thiết bị gọi là bộ lập trình ROM
 - Mỗi kết nối có thể lập trình được là một cầu chì
 - Bộ lập trình ROM phá vỡ các cầu chì khi không muốn duy trì kết nối
- Khả năng ghi rất thấp
 - Diễn hình ghi chỉ một lần và yêu cầu thiết bị lập trình ROM
- Chất lượng lưu trữ rất cao
 - Bit không thay đổi trừ khi kết nối với bộ lập trình
- Thường sử dụng trong các sản phẩm cuối cùng
 - Rẻ, khó thay đổi nội dung

EPROM: ROM lập trình ghi xóa

Phần tử có thể lập trình là một transistor MOS

- Transistor có cổng “floating” bao quanh bởi chất cách điện
- (a) Điện tích âm hình thành một kênh giữa nguồn và máng lưu giữ mức logic “1”
- (b) Điện áp dương lớn ở “cổng” làm cho các điện tích âm di chuyển ra khỏi kênh và duy trì trong cổng “floating” lưu trữ mức logic 0
- (c) (Xóa) chiếu tia UV trên bề mặt cổng “floating” làm cho các điện tích âm trở lại kênh từ cổng “floating” lưu trữ mức logic 1
- (d) Một IC EPROM có một cửa sổ để tia UV có thể chiếu qua



Khả năng ghi tốt

- Có thể tẩy xóa và lập trình hàng nghìn lần

Chất lượng lưu trữ giảm

- Chương trình kéo dài khoảng 10 năm nhưng sẽ bị suy giảm do nhiễu bức xạ và từ trường

Điện hình được sử dụng trong quá trình phát triển sản phẩm

EEPROM: ROM có thể ghi xóa bằng điện

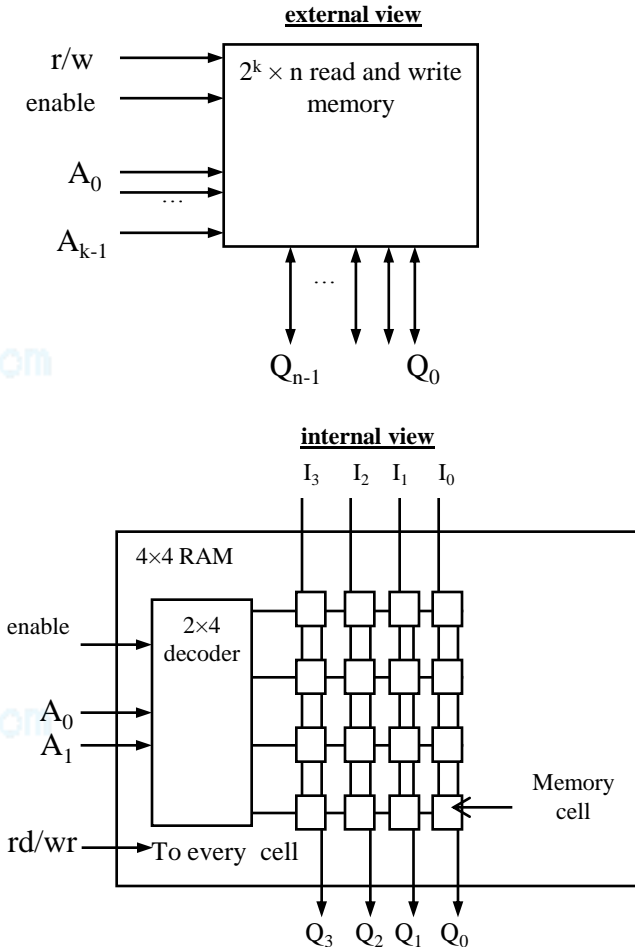
- Lập trình và xóa bằng điện
 - Sử dụng mức điện áp cao hơn bình thường
 - Có thể lập trình và xóa từng từ riêng
- Khả năng ghi tốt
 - Có thể lập trình trong hệ thống với mạch tích hợp để cung cấp điện áp cao hơn mức thông thường
 - Bộ điều khiển tích hợp trong bộ nhớ thường dùng để ẩn các chi tiết từ người sử dụng bộ nhớ
 - Ghi rất chậm do xóa và lập trình
 - Chân “busy” biểu thị với bộ xử lý là EEPROM vẫn đang trong quá trình ghi
 - Khả năng xóa và lập trình có thể lên tới 10 nghìn lần
- Chất lượng lưu trữ tương tự như EPROM (khoảng 10 năm)
- Thuận tiện hơn EPROMs, nhưng đắt hơn

Bộ nhớ Flash

- Mở rộng của EEPROM
 - Nguyên lý công “floating” tương tự
 - Chất lượng lưu trữ và khả năng ghi tương tự
- Xóa nhanh
 - Nhiều ô nhớ được xóa đồng thời, chứ không chỉ một ô nhớ tại một thời điểm
 - Các khối nhớ có kích thước khoảng vài nghìn byte
- Khả năng ghi từng từ có thể chậm hơn
 - Toàn bộ khối phải được đọc, từ được cập nhật, sau đó toàn bộ khối được ghi
- Sử dụng với các hệ thống nhúng lưu trữ dữ liệu lớn trong bộ nhớ
 - VD: camera số, điện thoại di động

RAM: Bộ nhớ “Truy cập ngẫu nhiên”

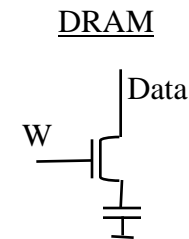
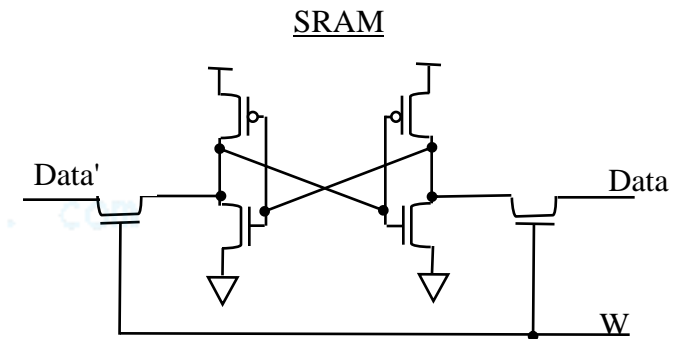
- **Thường là bộ nhớ “volatile”**
 - Dữ liệu bị mất khi không có nguồn cấp
- **Khi và đọc dễ dàng trong hệ thống nhúng trong quá trình làm việc**
- **Cấu trúc bên trong phức tạp hơn ROM**
 - Một từ nhớ có vài ô nhớ, mỗi ô nhớ chứa một bit nhớ
 - Mỗi đường dữ liệu vào và ra kết nối tới mỗi ô nhớ trong cột của nó
 - rd/wr Kết nối tới tất cả các ô nhớ
 - Khi hàng được “enable” bởi bộ giải mã, mỗi ô nhớ có mức logic lưu trữ bit dữ liệu đầu vào khi chân rd/wr biểu thị ghi hoặc đầu ra lưu trữ bit khi chân rd/wr biểu thị đọc



Các kiểu RAM cơ bản

- SRAM: RAM tĩnh
 - Ô nhớ dùng flip-flop để lưu trữ bit
 - Yêu cầu 6 transistors
 - Giữ dữ liệu khi có nguồn cấp
- DRAM: RAM động
 - Ô nhớ dùng transistor MOS và tụ để lưu trữ bit
 - Gọn nhẹ hơn SRAM
 - Yêu cầu “Refresh” do tụ bị dò
 - Các ô nhớ của từ được “refresh” khi đọc
 - Tốc độ “refresh” thường khoảng 15.625 microsec.
 - Truy cập chậm hơn SRAM

memory cell internals



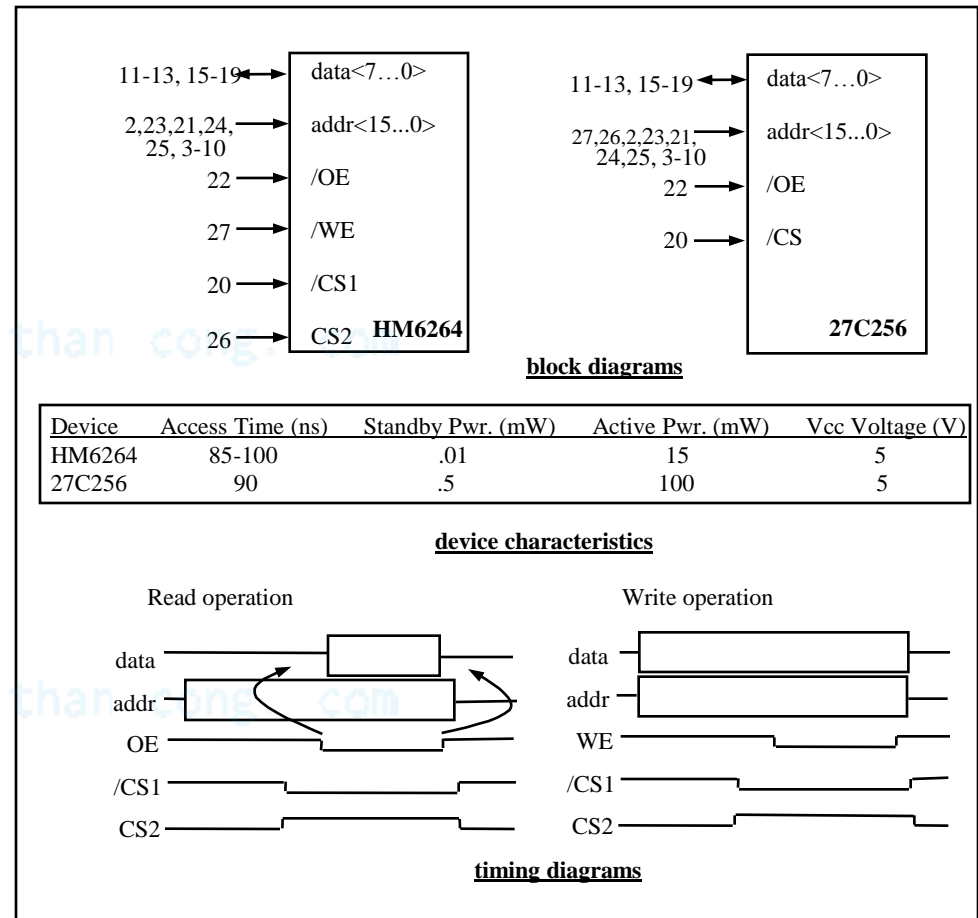
Các biến thể của RAM

- PSRAM: RAM giả tĩnh
 - DRAM với bộ điều khiển “refresh” tích hợp bên trong bộ nhớ
 - Giá thấp và mật độ lưu trữ cao hơn so với SRAM
- NVRAM: Nonvolatile RAM
 - Lưu trữ giữ liệu ngay cả khi không cấp nguồn
 - RAM có nguồn dự phòng
 - SRAM với battery được kết nối vĩnh cửu
 - Ghi nhanh như đọc
 - Không giới hạn số lần ghi
 - SRAM với EEPROM hoặc flash
 - Lưu trữ toàn bộ nội dung của RAM trên EEPROM hoặc flash trước khi ngắt nguồn

Ví dụ:

Thiết bị HM6264 & 27C256 RAM/ROM

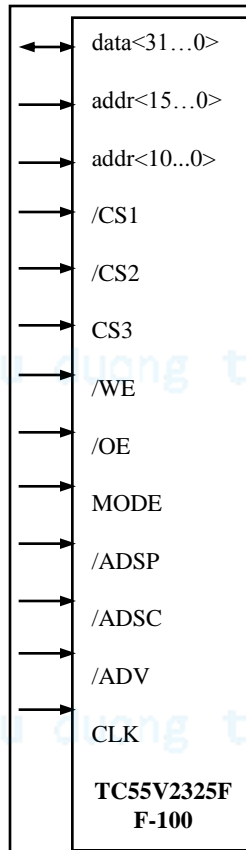
- Là thiết bị giá thấp, mật độ thấp
- Thường dùng trong bộ hệ thống nhúng dựa trên vi điều khiển 8-bit
- Hai số đầu thể hiện kiểu thiết bị
 - RAM: 62
 - ROM: 27
- Các số tiếp theo biểu thị dung lượng tính theo kilobits



Ví dụ:

Thiết bị nhớ TC55V2325FF-100

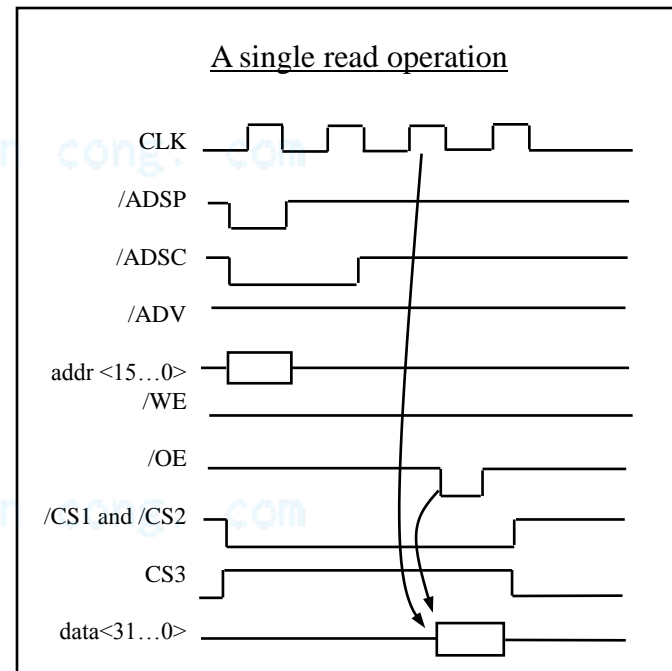
- Thiết bị nhớ SRAM 2-megabit
- Thiết kế để giao tiếp với bộ xử lý 32-bit
- Có khả năng đọc ghi tuần tự nhanh



block diagram

Device	Access Time (ns)	Standby Pwr. (mW)	Active Pwr. (mW)	Vcc Voltage (V)
TC55V23 25FF-100	10	na	1200	3.3

device characteristics

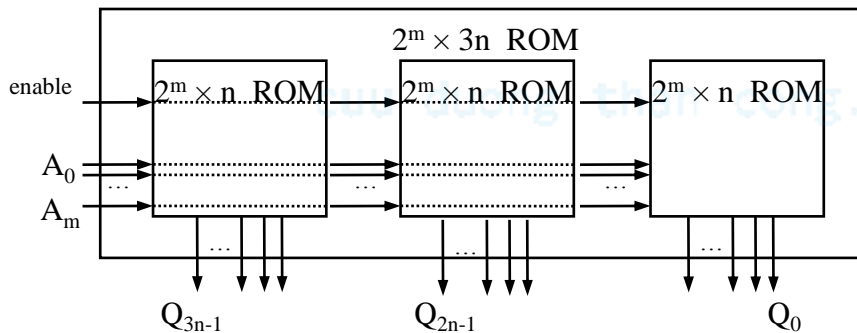


timing diagram

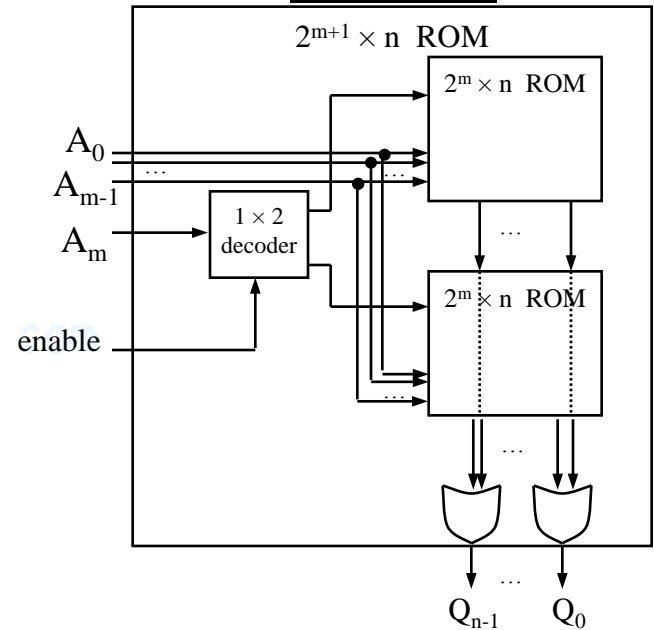
Ghép bộ nhớ

- Kích thước bộ nhớ yêu cầu thường khác với kích thước thiết kế của bộ nhớ
- Khi bộ nhớ thiết kế sẵn lớn hơn yêu cầu, chúng ta chỉ cần bỏ các địa chỉ nhớ ở vùng cao và đường dữ liệu ở vùng cao
- Khi bộ nhớ thiết kế sẵn nhỏ hơn yêu cầu, chúng ta cần ghép một vài bộ nhớ nhỏ hơn thành một bộ nhớ lớn
 - Kết nối kề nhau để tăng độ rộng từ nhớ
 - Kết nối tầng để tăng số từ
 - Dùng các đường địa chỉ vùng cao để lựa chọn bộ nhớ nhỏ hơn sử dụng bộ giải mã
 - Kết hợp cả hai khi cần tăng số từ cũng như độ rộng từ

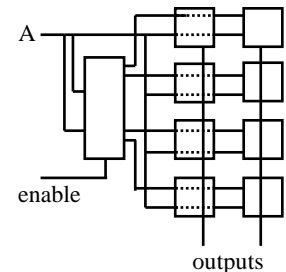
Tăng độ rộng từ nhớ



Tăng số từ nhớ

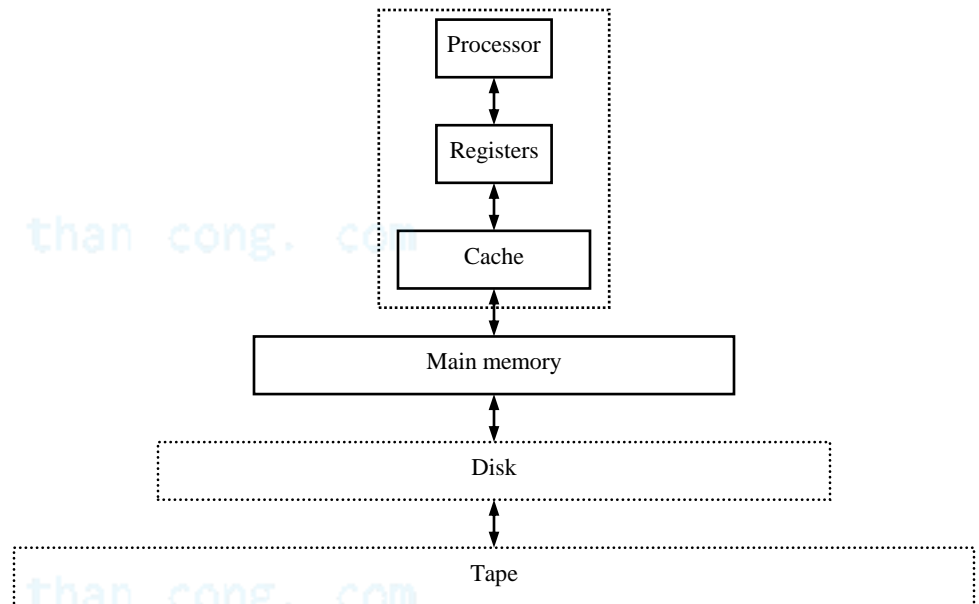


Tăng số từ cũng như độ rộng từ nhớ



Phân cấp bộ nhớ

- Chúng ta muốn bộ nhớ rẻ, truy cập nhanh
- Bộ nhớ chính
 - Dùng bộ nhớ dung lượng lớn, rẻ, chậm để lưu trữ toàn bộ chương trình và dữ liệu
- Cache
 - Dùng bộ nhớ nhỏ, đắt tiền và nhanh để lưu trữ phần “copy” của phần dữ liệu truy cập thuộc bộ nhớ lớn
 - Có thể có nhiều mức cache



Cache

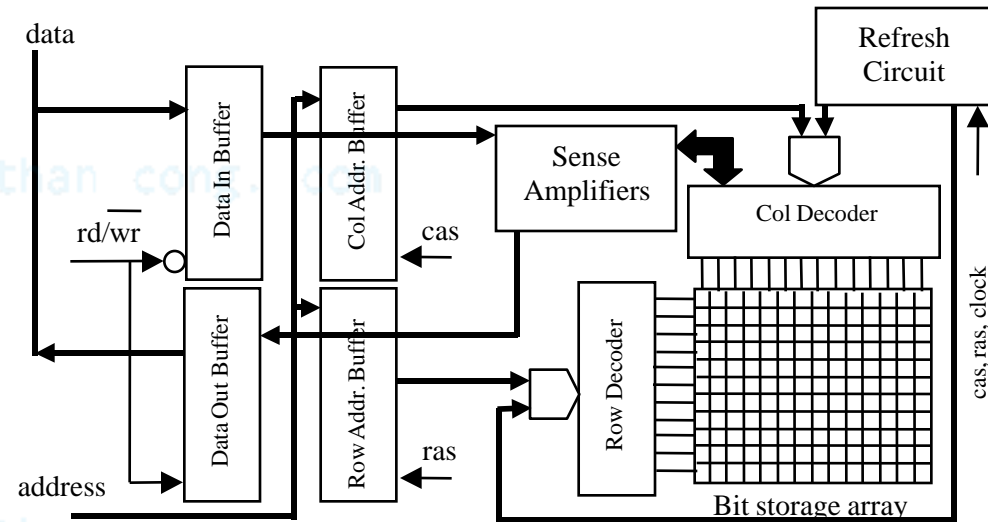
- **Thường được thiết kế dùng SRAM**
 - Nhanh hơn nhưng đắt hơn DRAM
- **Thường đặt trên cùng chip với bộ xử lý**
 - Không gian hạn chế, vì vậy có dung lượng nhỏ hơn nhiều so với bộ nhớ chính bên ngoài chip
 - Truy cập nhanh hơn (thường là 1 chu kỳ đồng hồ so với vài chu kỳ đồng hồ so với bộ nhớ ngoài)
- **Một số thiết kế cache**
 - Bản đồ cache, cơ chế thay thế, và kỹ thuật ghi

RAM cải tiến

- DRAMs thường được sử dụng như bộ nhớ chính trong bộ xử lý của hệ thống nhúng
 - Dung lượng lớn, giá thành thấp
- Các biến thể chính của DRAMs
 - Cần tương thích với tốc độ của bộ xử lý
 - FPM DRAM: DRAM kiểu trang nhanh
 - EDO DRAM: DRAM có đầu ra dữ liệu mở rộng
 - SDRAM/ESDRAM: DRAM đồng bộ và đồng bộ mở rộng

DRAM cơ bản

- Bus địa chỉ ghép giữa các phần tử hàng và cột
- Địa chỉ hàng và cột được chốt, tuần tự, bằng các tín hiệu *ras* và *cas*, tương ứng
- Mạch “refresh” có thể bên trong hoặc bên ngoài DRAM



Vấn đề tích hợp DRAM

- SRAM dễ dàng tích hợp trên một chip như bộ xử lý
- DRAM khó hơn
 - Sự khác biệt giữa thiết kế DRAM và mạch tổ hợp
 - Mục tiêu của người thiết kế mạch tổ hợp:
 - Giảm điện dung ký sinh để giảm trễ truyền lan và mức tiêu thụ công suất
 - Mục tiêu của người thiết kế DRAM:
 - Tạo ra điện dung để lưu trữ thông tin
 - Quá trình tổ hợp gặp khó khăn

Đơn vị quản lý bộ nhớ (MMU)

- Chức năng của MMU
 - Thực hiện “refresh” DRAM, giao tiếp bus và điều phối
 - Thực hiện việc chia sẻ bộ nhớ
 - Chuyển đổi địa chỉ nhớ từ bộ xử lý sang địa chỉ nhớ vật lý của DRAM
- Các CPUs thường có bộ MMU tích hợp sẵn
- Bộ xử lý chức năng đơn có thể sử dụng để xây dựng MMU