

1 Lezione del 06-11-24

1.1 Collegamento al bus e maschere

Resta da capire da dove vengono gli indirizzi di lettura usati nell'accesso alla SRAM. Questi si trovano su un **bus indirizzi**, il cui valore è impostato dal processore ogni volta che vuole effettuare un accesso. Ad esempio, per un modulo di RAM da $256\text{M} \times 8$ bit, quindi con 28 ingressi, si avrà che bisogna di montare banchi di memoria a partire da $0xE0000000$ fino a $0xFFFFFFFF$.

L'ingresso di select verrà quindi generato a partire dalla parte alta dell'indirizzo, fatto passare attraverso una certa maschera, facendo quindi corrispondere una certa impostazione dei bit più significativi (in questo caso 4) al segnale di select. Visto che avevamo detto si parte da $0xE0000000$, vogliamo $0xE = B1110$, quindi serve la rete combinatoria (maschera):

$$/s = \overline{a_{31}} + \overline{a_{30}} + \overline{a_{29}} + a_{28}$$

Con un montaggio di questo tipo possiamo usare il select per scegliere quale banco di memoria RAM è associato a quale parte dello spazio indirizzabile. Questo tipo di configurazione giustifica inoltre il fatto che il bit di select viene impostato con ritardo rispetto agli indirizzi: visto che deve essere calcolato sulla base di quest'ultimi, risentirà del ritardo della rete combinatoria che lo genera.

1.2 Memorie ROM a sola lettura

Le memorie a sola lettura, dette ROM, sono effettivamente reti combinatorie: l'uscita è costante qualunque siano gli stati passati. Vengono montate nello spazio di memoria assieme alla RAM, e rappresentano la parte **non volatile** (persistente) dello spazio di memoria stesso (abbiamo visto come contengono le prime istruzioni eseguite dal processore). La loro struttura interna ricalca quella della memoria RAM, privata della circuiteria necessaria alla lettura, e che usa generatori di costante al posto dei D-latch.

Una ROM può essere realizzata attraverso il MSU (*Modello Strutturale Universale*), collegando in maniera ortogonale file di OR alle uscite di un decoder (come ci è concesso dall'espansione di Shannon), nei punti in cui vogliamo che un'uscita corrisponda a un certo ingresso. Nella pratica, si usano più spesso porte NOR per ragioni elettroniche, e quindi si collegano le linee nel caso un'uscita corrisponde alla negazione dell'ingresso. Questo circuito viene solitamente stampato su un singolo chip di silicio, il cui costo fisso di progettazione è giustificato solo nel caso di produzione su larga scala. Conviene quindi realizzare delle alternative, le **ROM programmabili**.

1.2.1 ROM programmabili

In una memoria ROM programmabile, le porte NOR sono tutte attaccate alle linee degli AND del decoder. Si possono disabilitare selettivamente alcune di queste porte NOR per effettivamente **programmare** la memoria contenuta nella ROM.

Possiamo individuare delle categorie per queste reti:

- Le **OTPROM** (*One Time Programmable ROM*) vengono realizzate attraverso questa tecnologia, è la loro programmazione risulta quindi **distruttiva** (una volta programmato un bit non si può più rimuovere).

- Una tecnologia più sofisticata è rappresentata dalle **EPROM** (*Erasable programmable ROM*). Queste vengono realizzate attraverso transistor a field-effect. La scrittura della EPROM può essere ripetuta sottoponendola a una luce ultravioletta, e quindi cancellando tutti i dati, per poi riscrivere altri.

Di una EPROM ci interessano:

- **Endurance:** quante riscritture successive può supportare (solitamente dalle 10K alle 100K volte);
 - **Data retention:** il periodo per cui si può fare affidamento sui dati contenuti in una EPROM (solitamente dai 10 ai 100 anni).
- Infine, le **EEPROM** (*Electrically Erasable Programmable ROM*) permettono la riprogrammazione direttamente attraverso segnali elettrici, sul chip già montato nello spazio di memoria. Potremmo pensare che EEPROM e RAM sono effettivamente equivalente. Ci sono invece alcune differenze, che sono:
 - L'EEPROM è persistente, mentre la RAM è volatile;
 - Il numero di volte in cui si può riprogrammare una EEPROM è comunque limitato;
 - Il tempo di riprogrammazione di una EEPROM è maggiore del tempo di lettura della RAM;
 - Le tensioni che si usano nella programmazione di una EEPROM (12V-18V) sono maggiori dei 5V (o 3.3V) che richiede la RAM.

1.3 Il linguaggio Verilog

Per descrivere le reti logiche fa comodo adottare una **notazione testuale**. Finora abbiamo usato disegni o espressioni algebriche: adesso introduciamo un **linguaggio di descrizione hardware**, il **Verilog**. Questo linguaggio è più **compatto**, e può essere **interpretato** automaticamente da una macchina, permettendoci di effettuare prove (in modo simile a come avevamo introdotto coi diagrammi di temporizzazione).