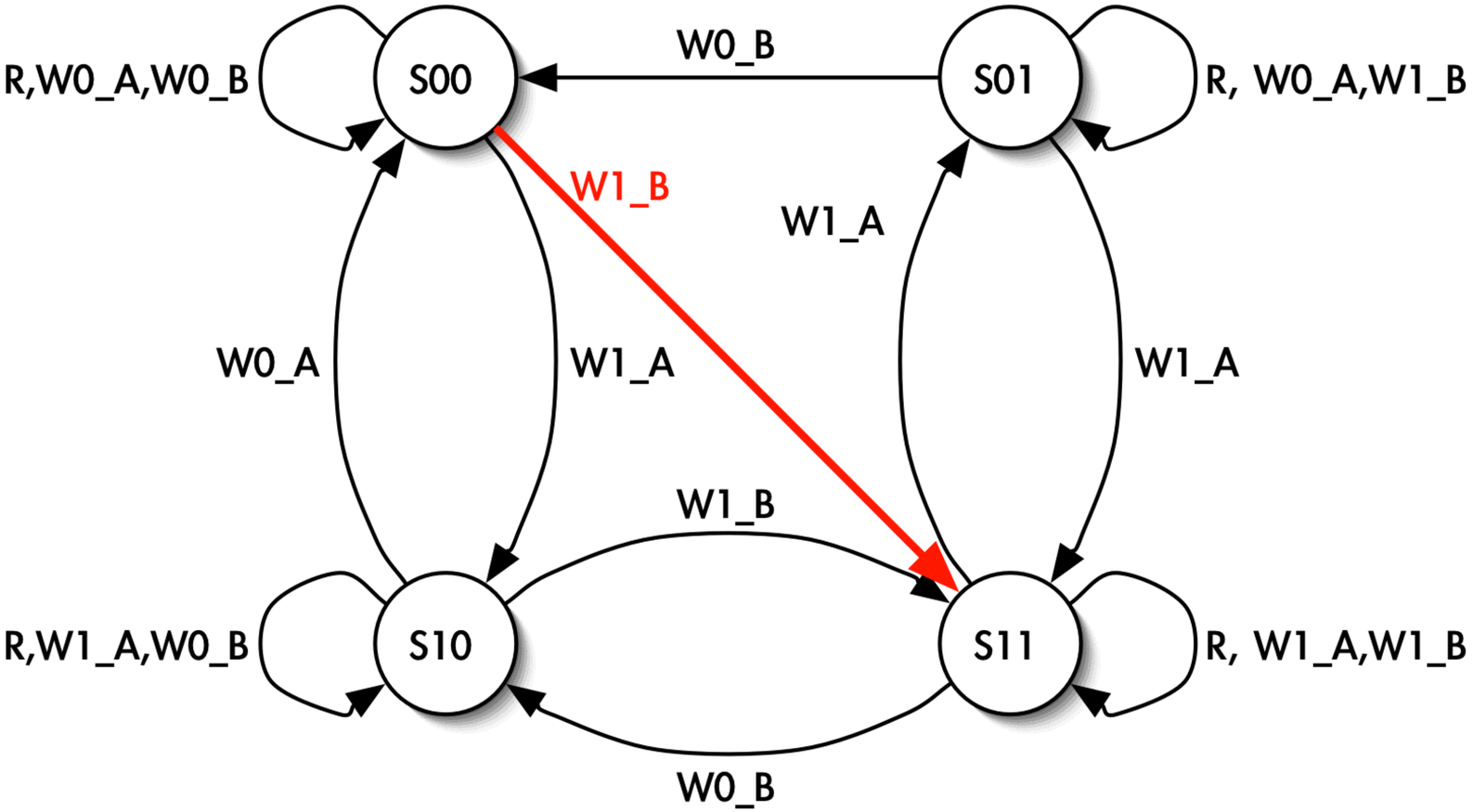
HW 03-1 Coupling Fault

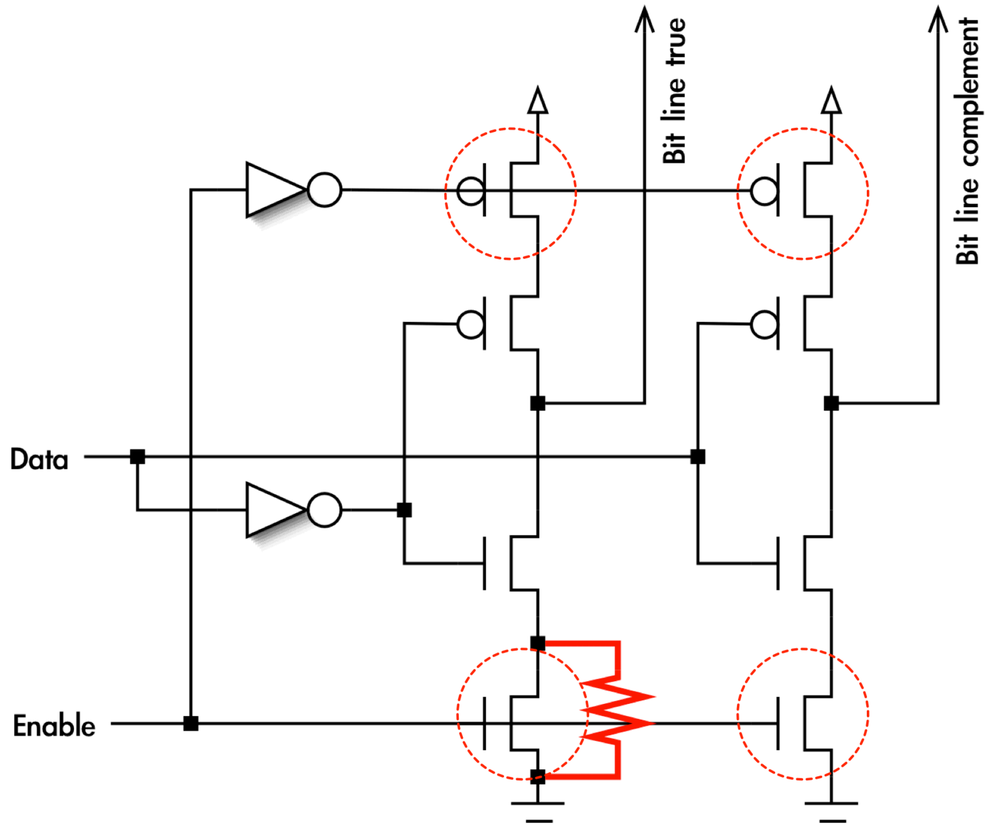
Let A and B be two adjacent memory cells. Find one test pattern (that we discussed in the classroom) that detects the following coupling fault. Show how your pattern detects the fault.



- First, set AB to 00.
- Then, apply r0w1.
  - If A < B, detected w/ decreasing address order.
  - If A > B, detected w/ increasing address order.
  - Detection:
    - fault-free: 00 => 01 => 11
    - faulty: 00 => 11
  - Example: March C- $\left\{ \Updownarrow (w0);\ \Uparrow (r0,w1);\ \Uparrow (r1,w0);\ \Downarrow (r0,w1);\ \Downarrow (r1,w0);\ \Updownarrow (r0) \right\}$

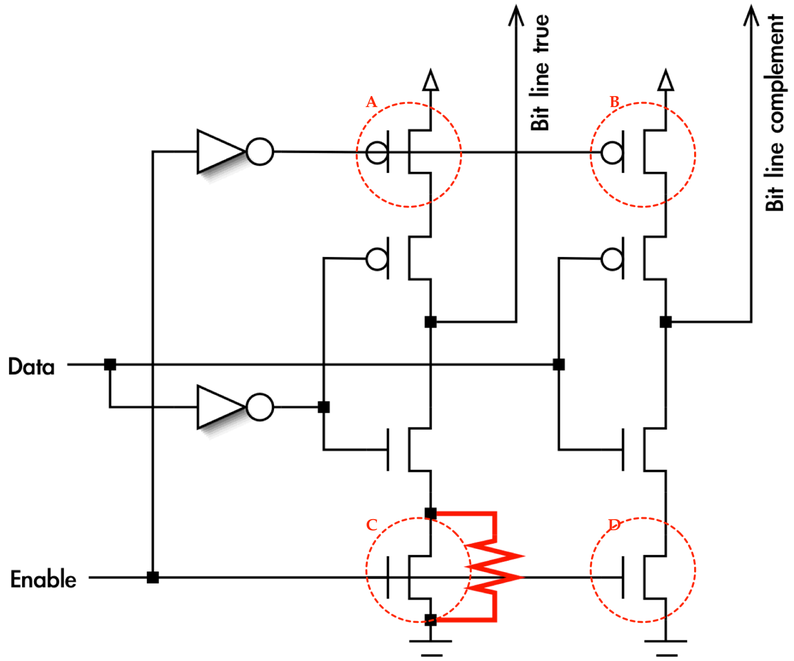
HW 03-2 False Write Through

- Advise a test pattern that detects the short faults at the four circled transistors. Show how the faults are detected.



Assume that "Data" stays at its last value unless a new write operation is performed.

1. The true bit line PMOS stuck-on (site A) will prevent the true bit line from being pulled down. Thus, it can be detected by the r0 operation following a w1 operation (to set Data to 1) *at the same column* that sets "Data" to 1, e.g., (r0 … w1). Note that the address order should be *row first*.
2. The complement bit line NMOS stuck-on (site D) will prevent the complement bit line from being pulled up, which is similar to a recharge fault. This can be activated by w1 followed by r0 (at the same column) — the voltage at the complement bit line may be lower than that in the true bit line due to the imperfect pre-charge. The detecting element is (r0 … w1).
3. The complement bit line PMOS stuck-on (site B) will prevent the complement bit line from being pulled down. To activate it, we must set Data to 0 (by w0). An r1 may fail because the complement bit line may not be sufficiently low (compared to the true bit line). The detecting element is {r1 … w0} — row first.
4. The analysis is similar to case 2. The detecting element is (r1 … w0).



Many memory test patterns have the elements to detect these faults, for example, March C-.

$$\{ \Updownarrow (w0); \Uparrow (r0,w1); \Uparrow (r1,w0); \Downarrow (r0,w1); \Downarrow (r1,w0); \Updownarrow (r0) \}$$

Note that the address order has to be *row-first*.