Enter Rita
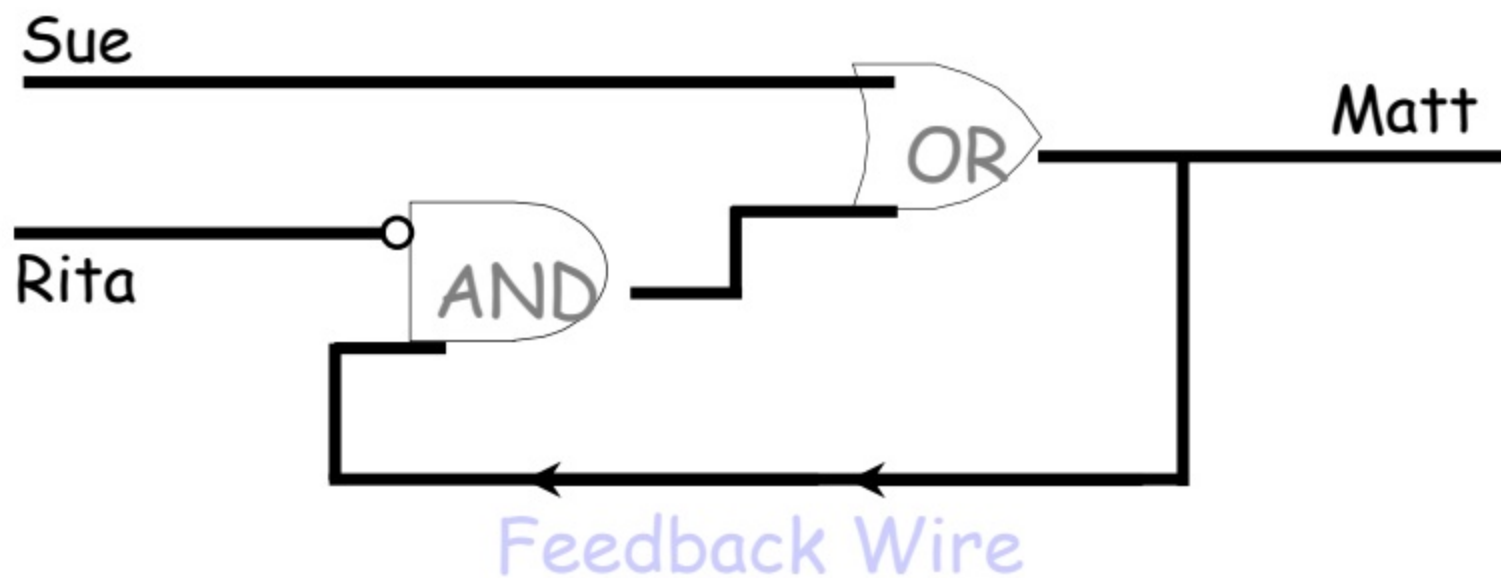
- Matt doesn't like Rita

- Matt decides to go to the party if Sue decides to go OR if he (Matt) already feels like going, UNLESS Rita decides to go.

Enter Rita

- Matt doesn't like Rita

- Matt decides to go to the party if Sue decides to go OR:

  - If he (Matt) already feels like going AND Rita decides NOT to go.
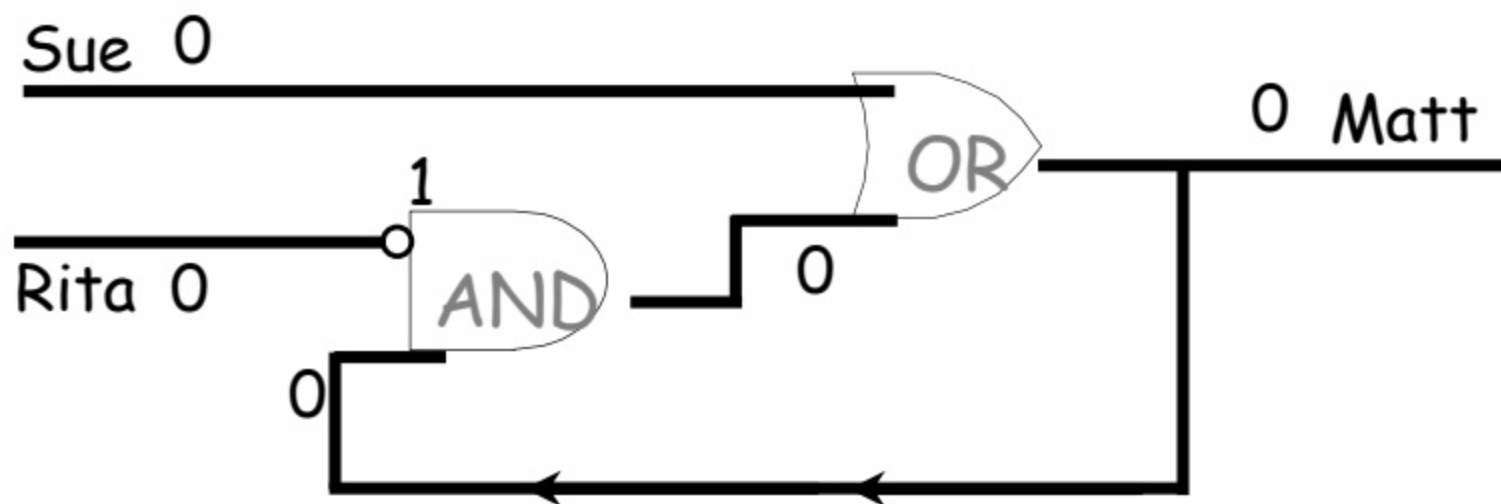
Enter Rita

- Matt doesn't like Rita

- Matt decides to go to the party if Sue decides to go OR:

  - If he (Matt) already feels like going AND Rita decides NOT to go.

Sue

Rita

AND

OR

Matt

Feedback Wire
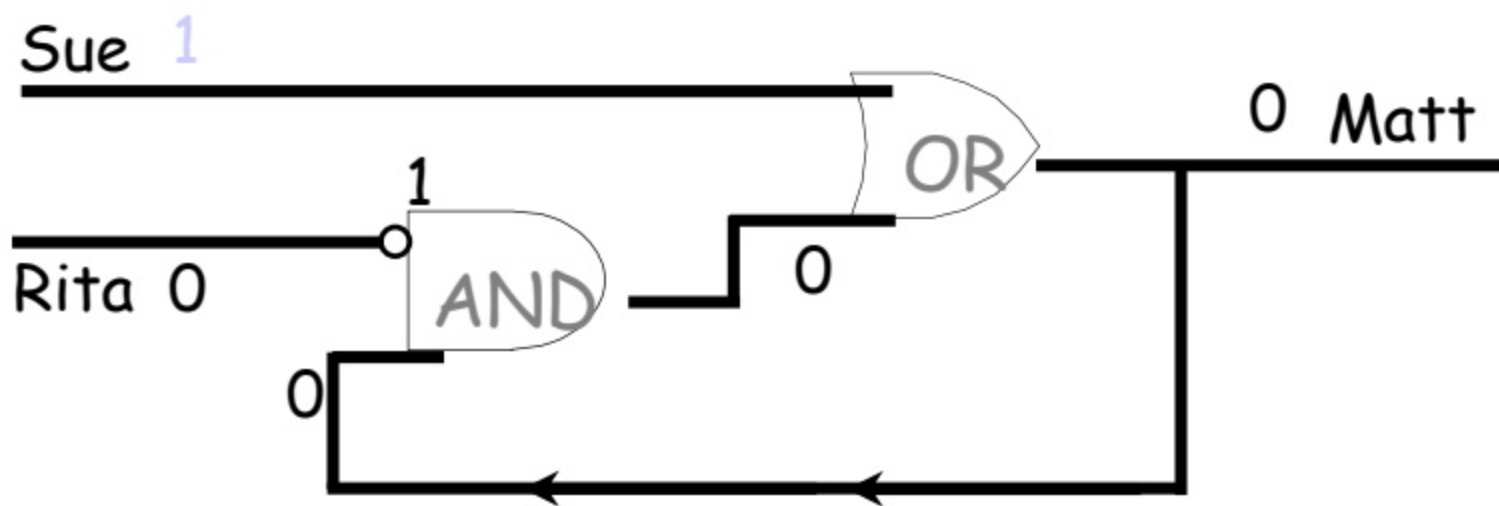
Enter Rita

- Matt doesn't like Rita

- Matt decides to go to the party if Sue decides to go OR:

  - If he (Matt) already feels like going AND Rita decides NOT to go.

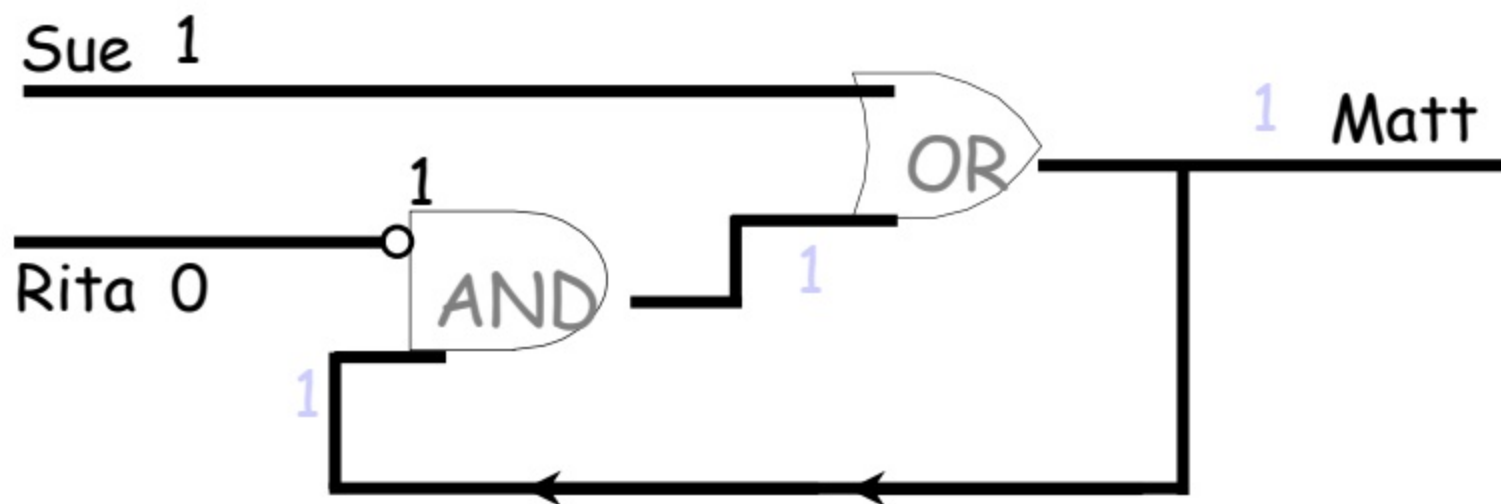Sue 0

Rita 0

1

AND

0

0

OR

0 Matt

Enter Rita

- Matt doesn't like Rita

- Matt decides to go to the party if Sue decides to go OR:

  - If he (Matt) already feels like going AND Rita decides NOT to go.

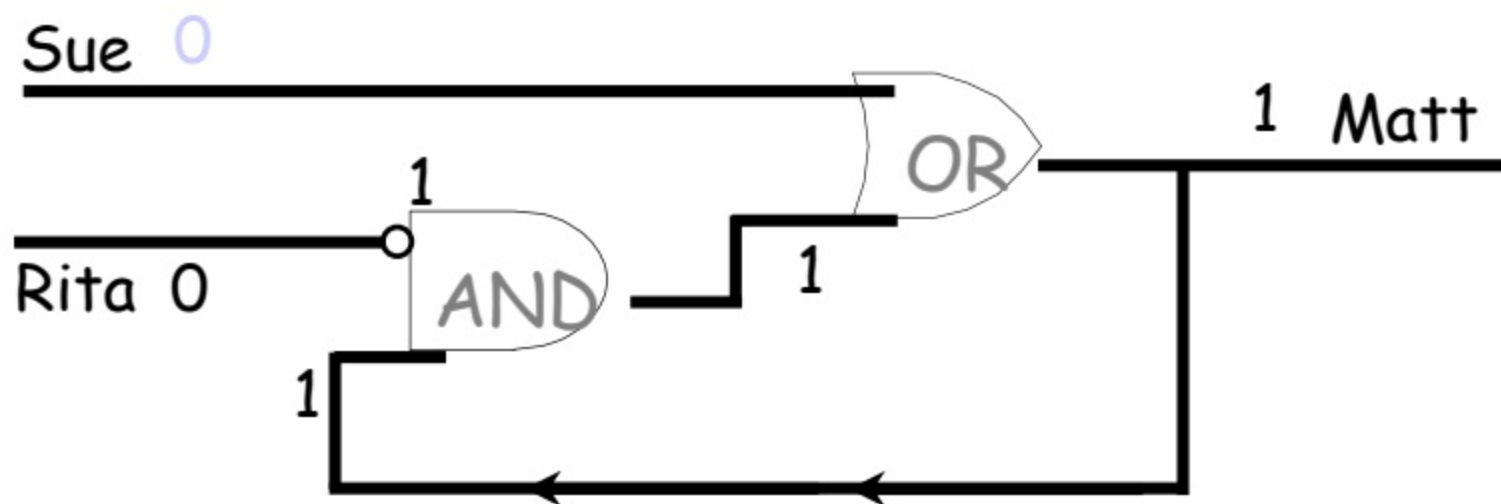Sue 1

1

Rita 0

AND

OR

0

0 Matt

0

Enter Rita

- Matt doesn't like Rita

- Matt decides to go to the party if Sue decides to go OR:

  - If he (Matt) already feels like going AND Rita decides NOT to go.

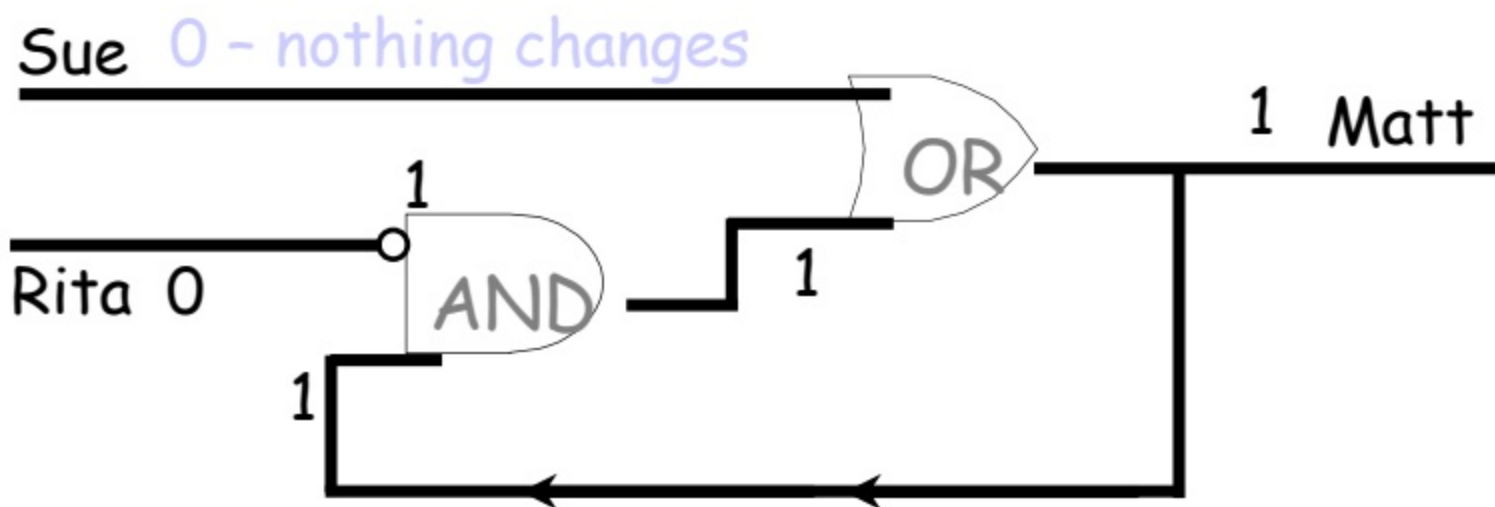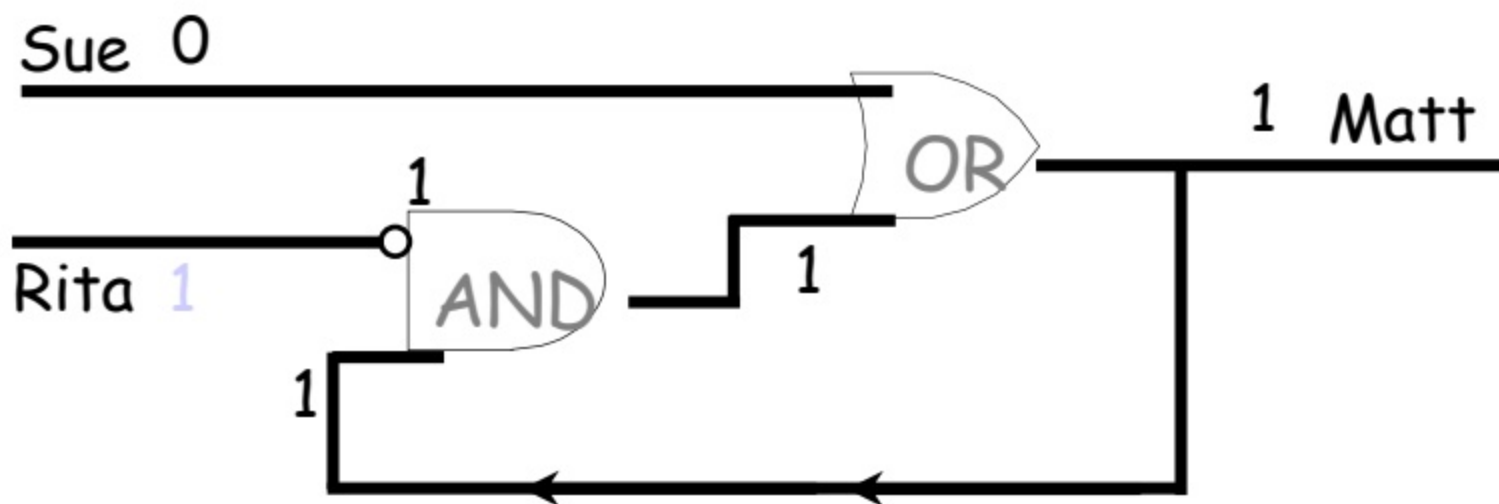Sue 1

Rita 0

1

AND

1

OR

1

1 Matt

Enter Rita

- Matt doesn't like Rita

- Matt decides to go to the party if Sue decides to go OR:

  - If he (Matt) already feels like going AND Rita decides NOT to go.

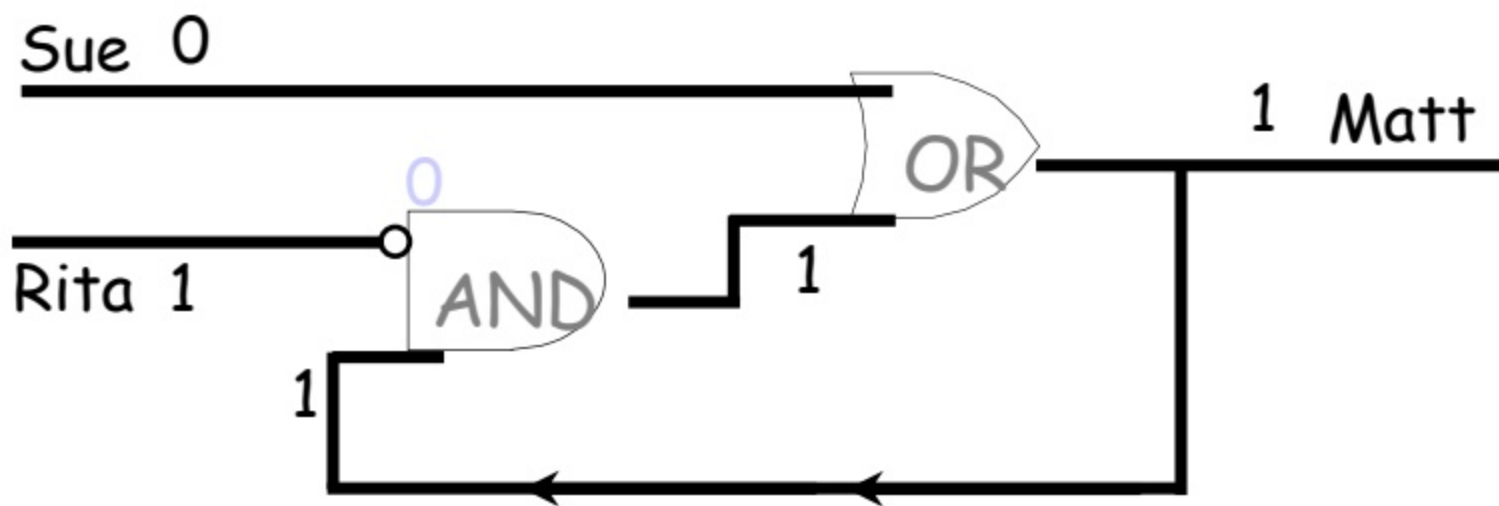Sue 0

1 Matt

1

OR

Rita 0

1

AND

1

Enter Rita

- Matt doesn't like Rita

- Matt decides to go to the party if Sue decides to go OR:

  - If he (Matt) already feels like going AND Rita decides NOT to go.

Sue 0 – nothing changes
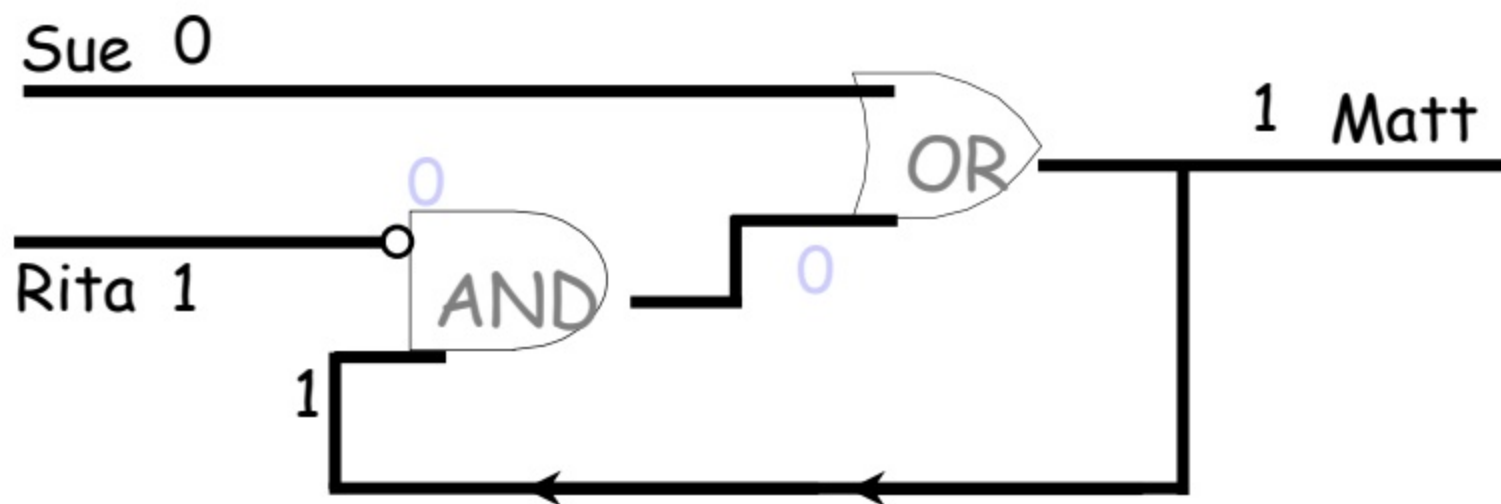
1 Matt

1

Rita 0

AND

1

OR

Enter Rita

- Matt doesn't like Rita

- Matt decides to go to the party if Sue decides to go OR:

  - If he (Matt) already feels like going AND Rita decides NOT to go.

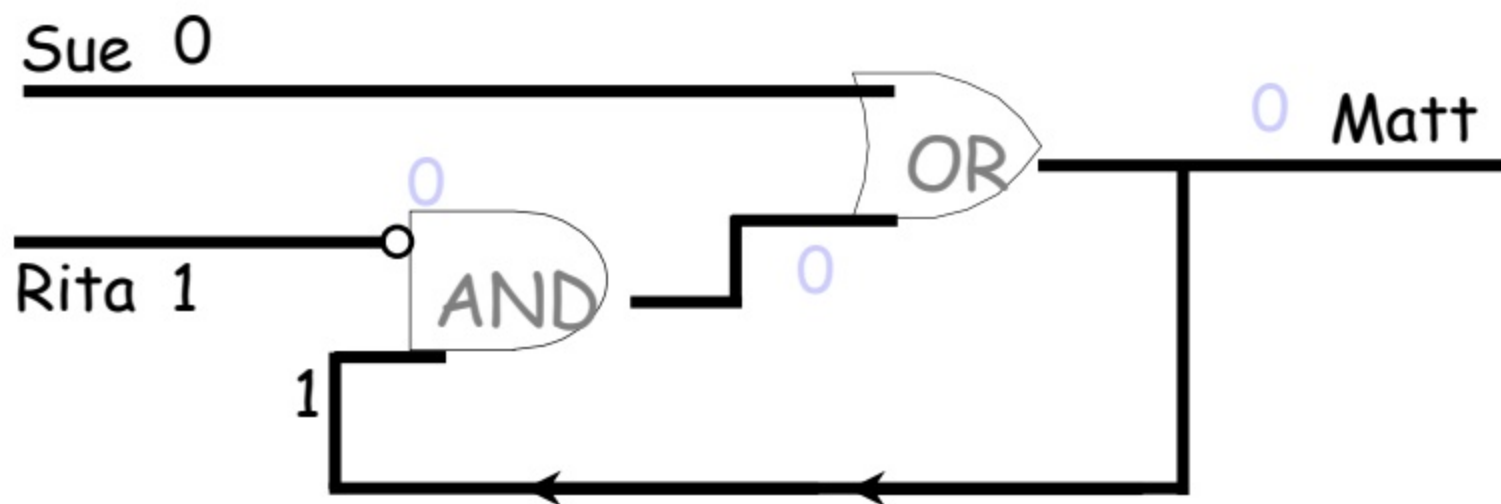Sue 0

Rita 1

1

OR

AND

1

1

1

1 Matt

Enter Rita

- Matt doesn't like Rita

- Matt decides to go to the party if Sue decides to go OR:

  - If he (Matt) already feels like going AND Rita decides NOT to go.

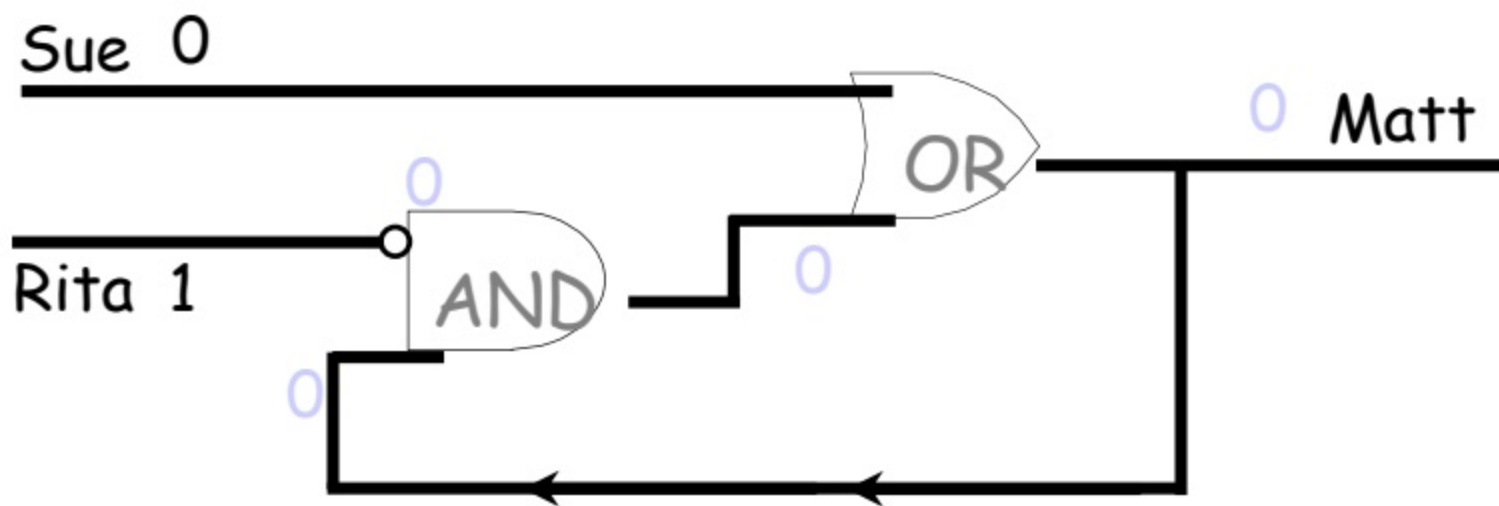Sue 0

0

Rita 1

1

AND

1

OR

1 Matt

Enter Rita

- Matt doesn't like Rita

- Matt decides to go to the party if Sue decides to go OR:

  - If he (Matt) already feels like going AND Rita decides NOT to go.

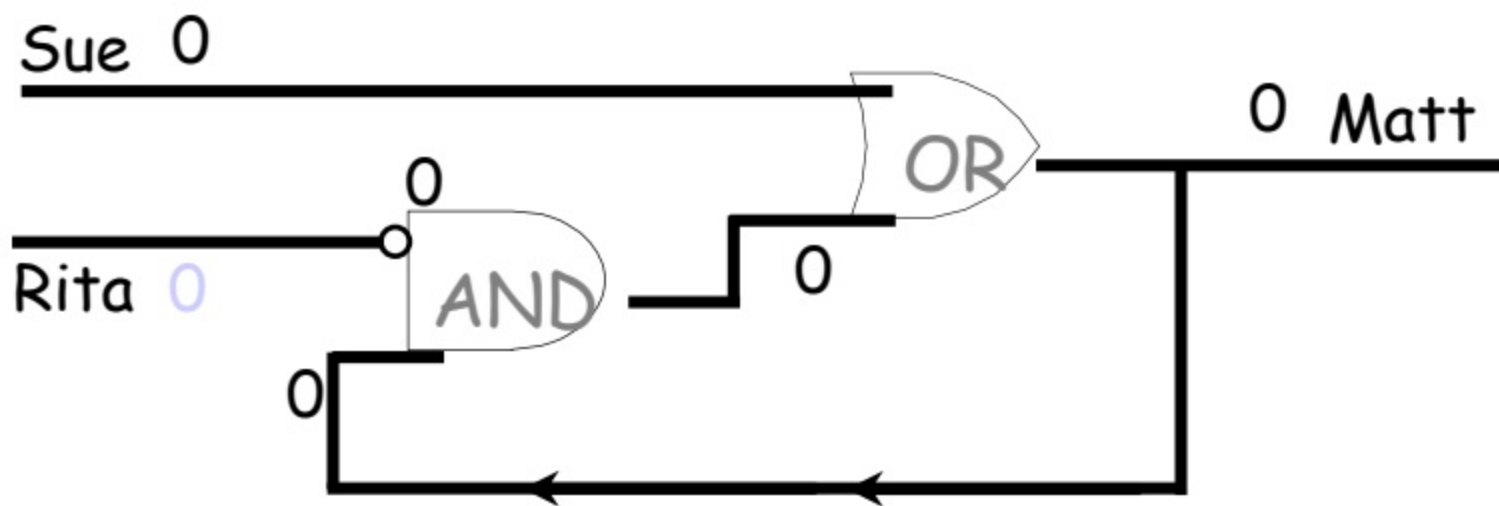Sue 0

Rita 1

0

AND

1

0

OR

1 Matt

Enter Rita

- Matt doesn't like Rita

- Matt decides to go to the party if Sue decides to go OR:

  - If he (Matt) already feels like going AND Rita decides NOT to go.

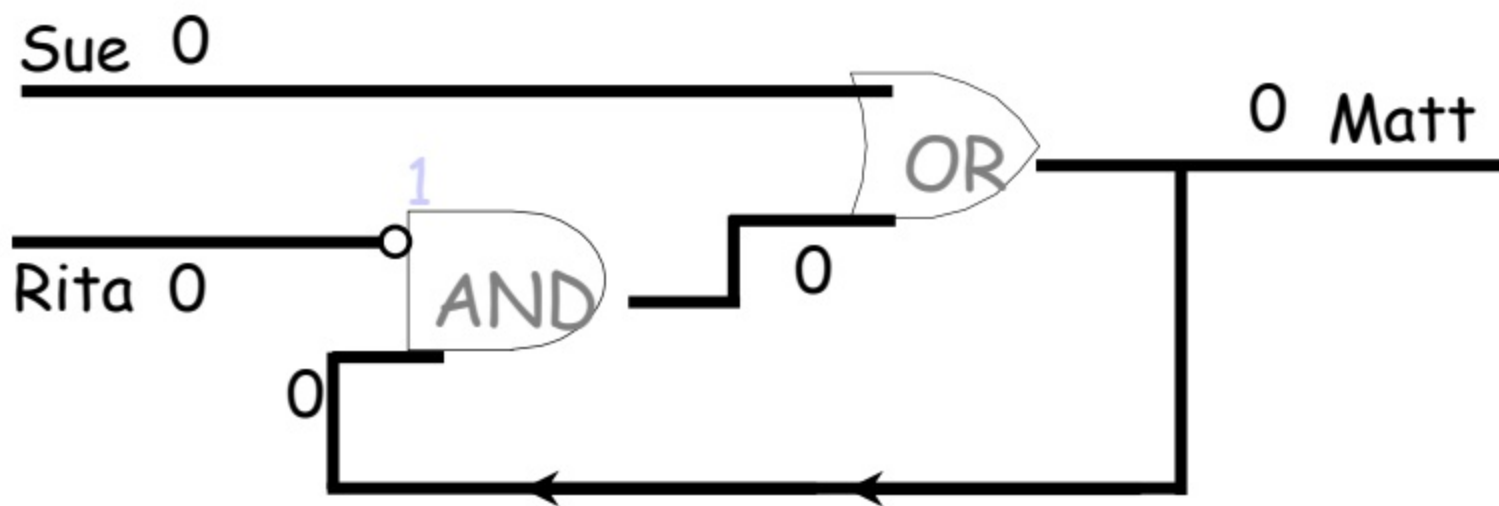Sue 0

0 Matt

0

OR

0

Rita 1

AND 0

1

Enter Rita

- Matt doesn't like Rita

- Matt decides to go to the party if Sue decides to go OR:

  - If he (Matt) already feels like going AND Rita decides NOT to go.

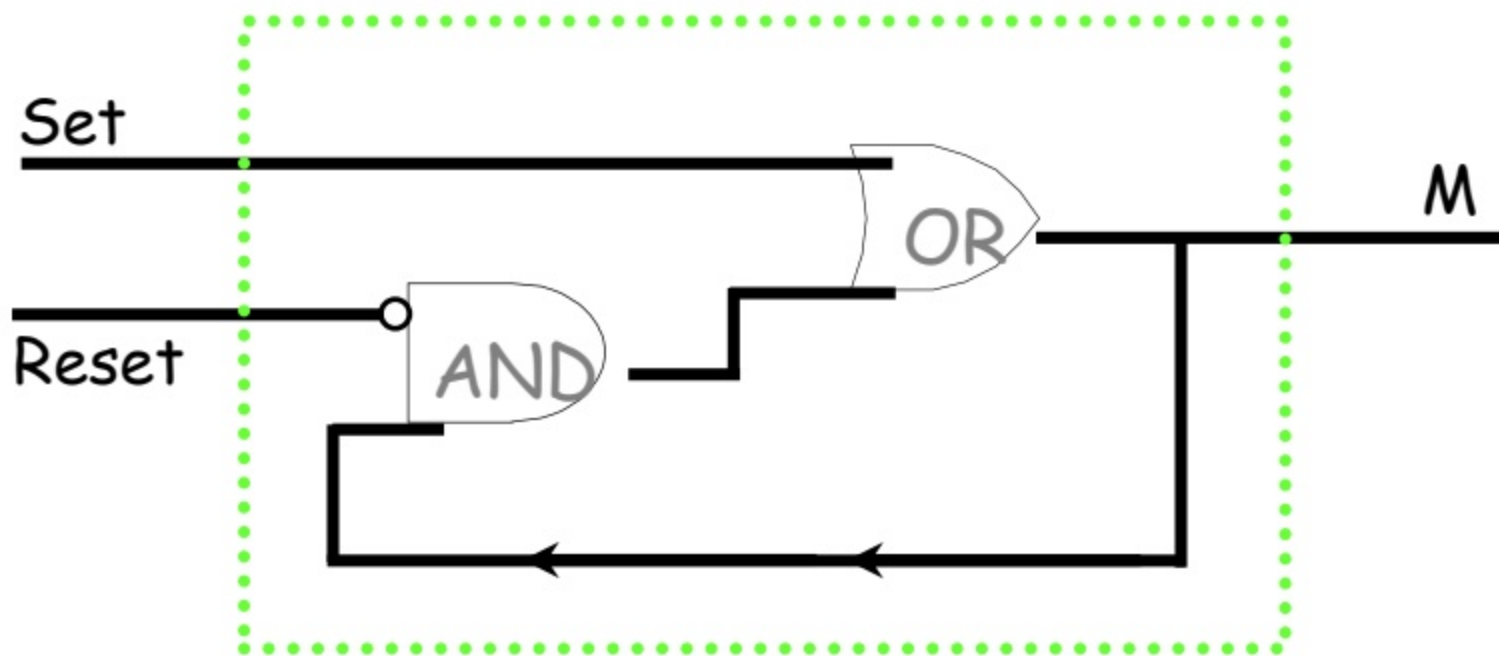Sue 0

Rita 1

0

0 AND

0

0 OR

0 Matt

Enter Rita

- Matt doesn't like Rita

- Matt decides to go to the party if Sue decides to go OR:

  - If he (Matt) already feels like going AND Rita decides NOT to go.

Sue 0

0 Matt

0

OR

0

Rita 0

AND

0

Enter Rita

- Matt doesn't like Rita

- Matt decides to go to the party if Sue decides to go OR:

  - If he (Matt) already feels like going AND Rita decides NOT to go.
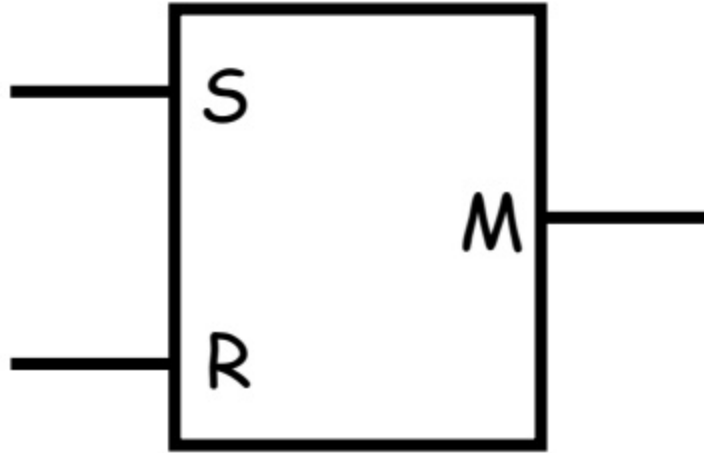
Sue 0

Rita 0

1

AND

0

0
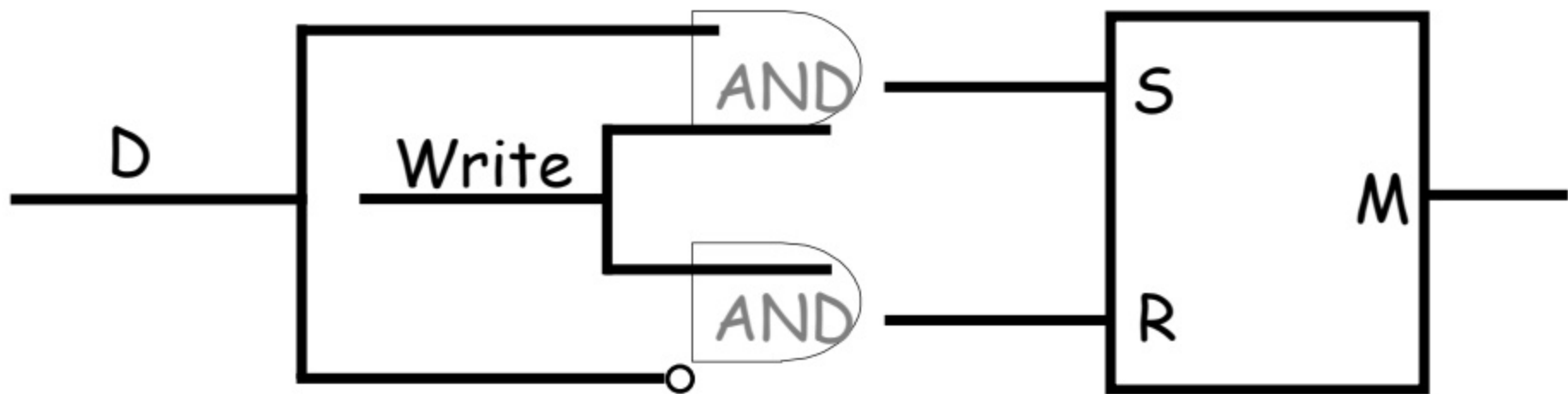
OR

0 Matt

0

The Flip-Flop

- M becomes 1 if Set is turned on

- M becomes 0 if Reset is turned on

- Otherwise (if Set and Reset are both 0), M just remembers its value
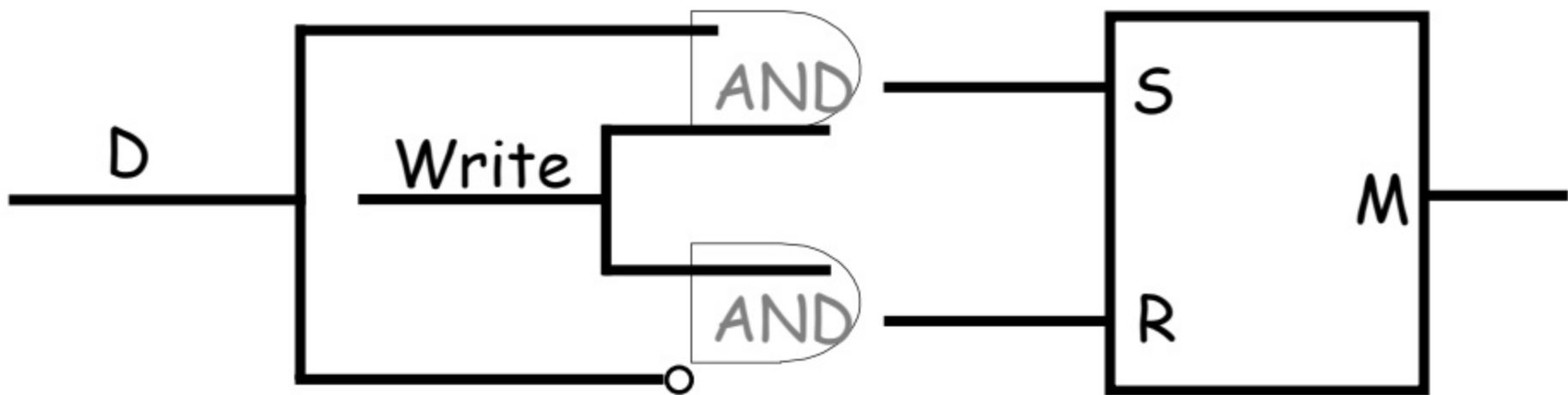
The Flip-Flop

- M becomes 1 if Set is turned on

- M becomes 0 if Reset is turned on

- Otherwise (if Set and Reset are both 0),
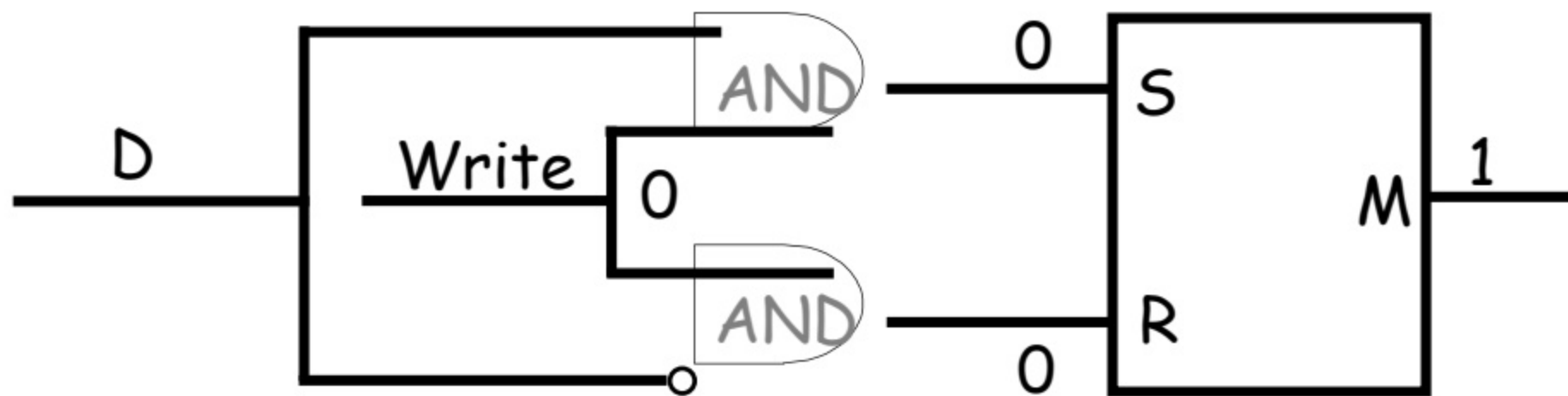 M just remembers its value

The Data Flip-Flop

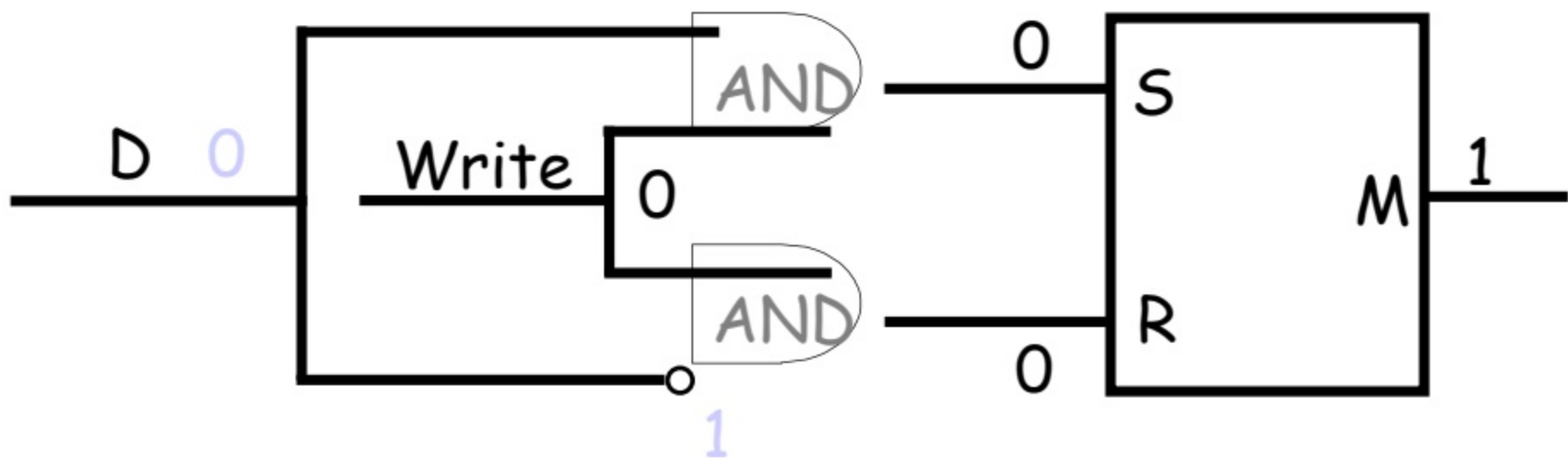- Nothing happens unless Write = 1

The Data Flip-Flop

- Nothing happens unless Write = 1

- If Write = 1, then M becomes set to D

- Once Write = 0 again, M just keeps its value. (It ignores D.)

Using a Data Flip-Flop
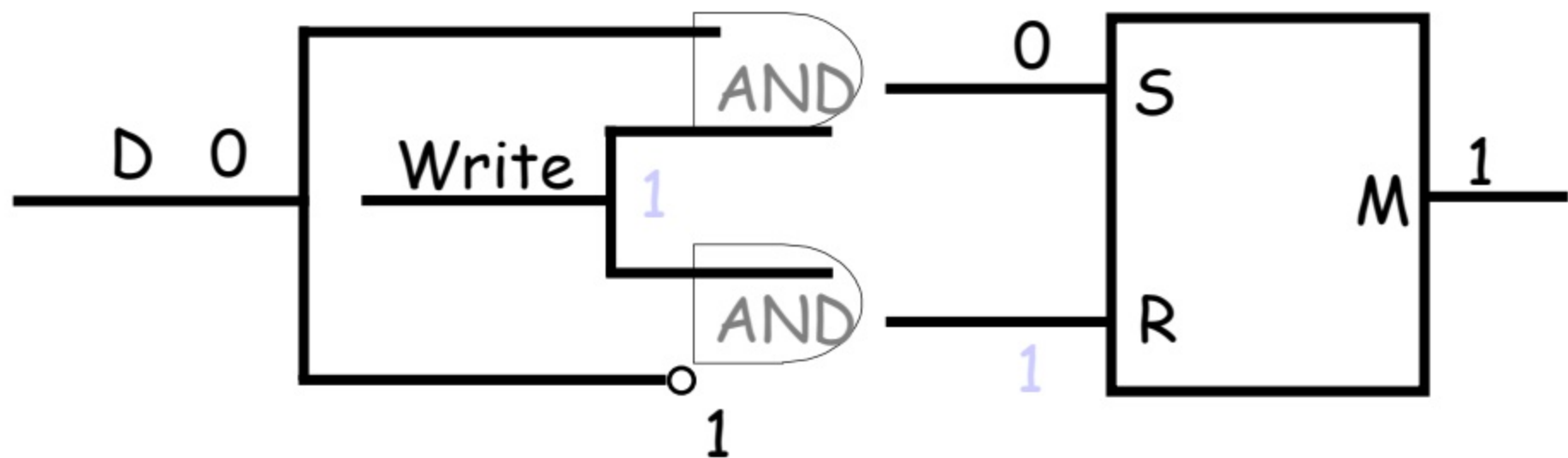

- Initially, Write = 0. Let's say M = 1.
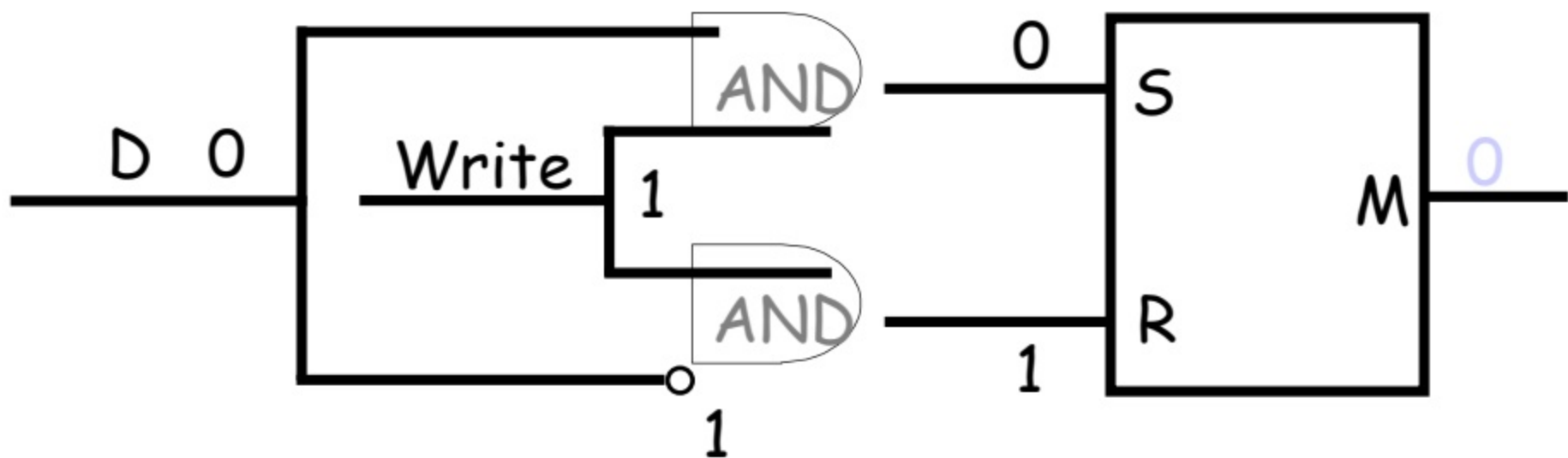
Using a Data Flip-Flop

- Initially, Write = 0.   Let's say M = 1.

- First, set D to desired value, say 0.

Using a Data Flip-Flop

D 0 Write 1 AND 0 S M 1
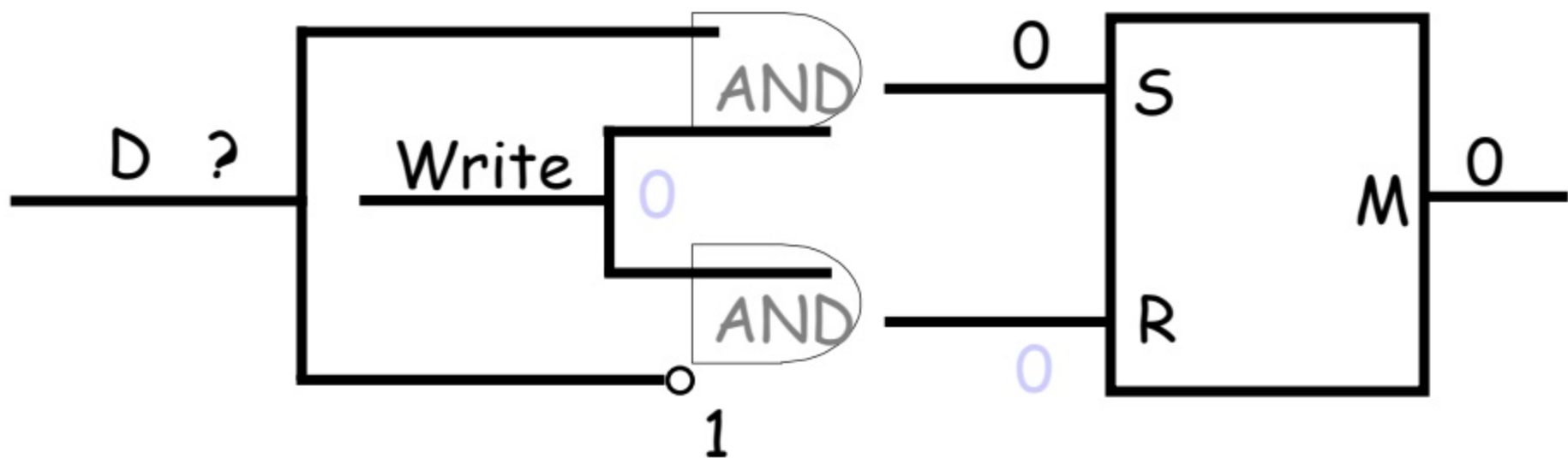
AND 1 R 1

- Initially, Write = 0.   Let's say M = 1.

- First, set D to desired value, say 0.

- Then, set Write to 1.
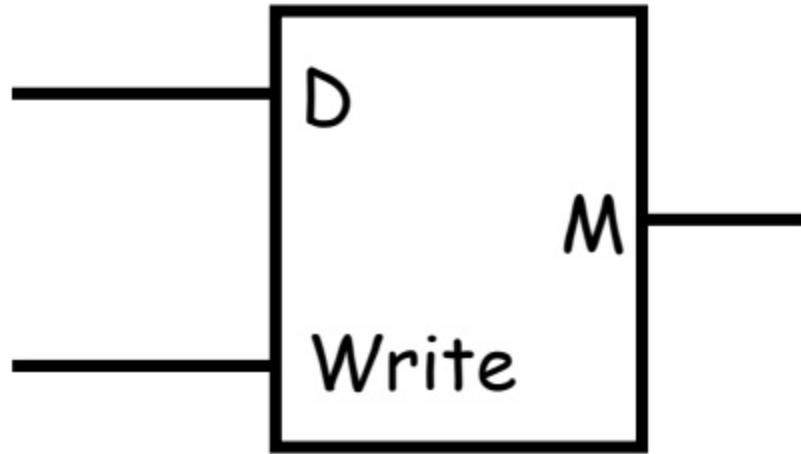
Using a Data Flip-Flop

- Initially, Write = 0. Let's say M = 1.

- First, set D to desired value, say 0.

- Then, set Write to 1.

- This causes M to be reset to 0.
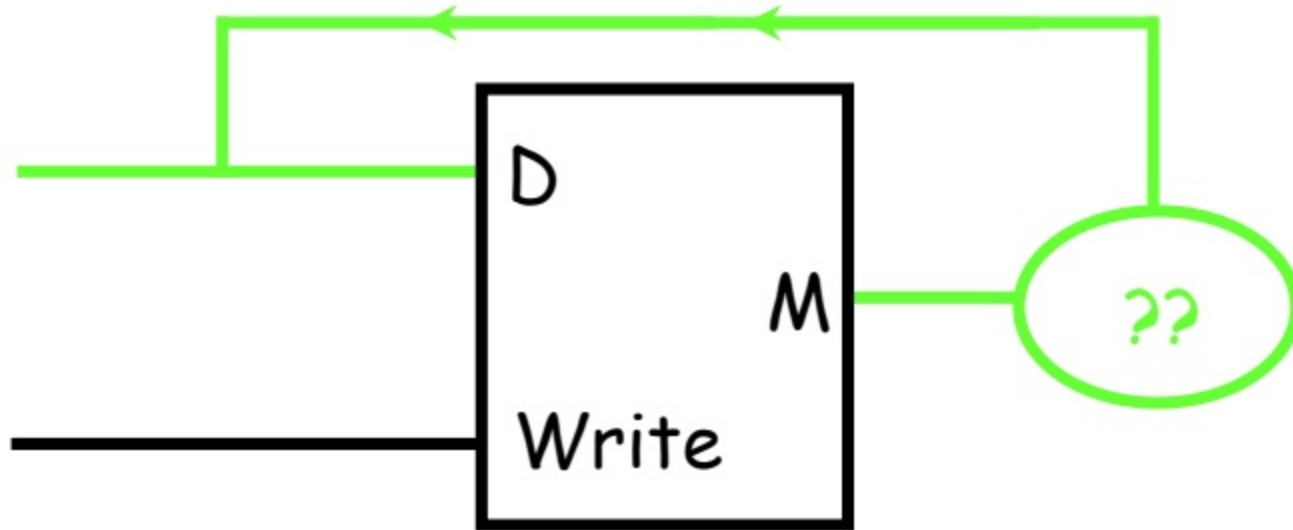
Using a Data Flip-Flop

- Initially, Write = 0.   Let's say M = 1.

- First, set D to desired value, say 0.

- Then, set Write to 1.

- This causes M to be reset to 0.

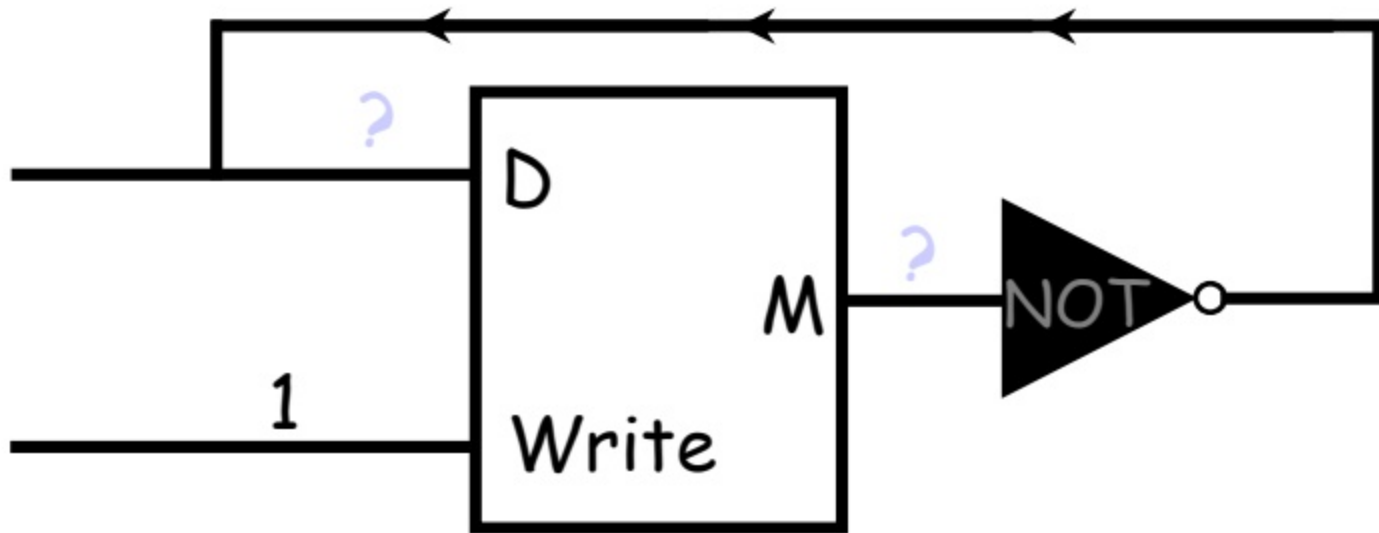- Finally set Write back to 0. Now D irrelevant.

The Data Flip-Flop

- If Write = 0, M just keeps its value. (It ignores D.)

- If Write = 1, then M becomes set to D

A subtle problem

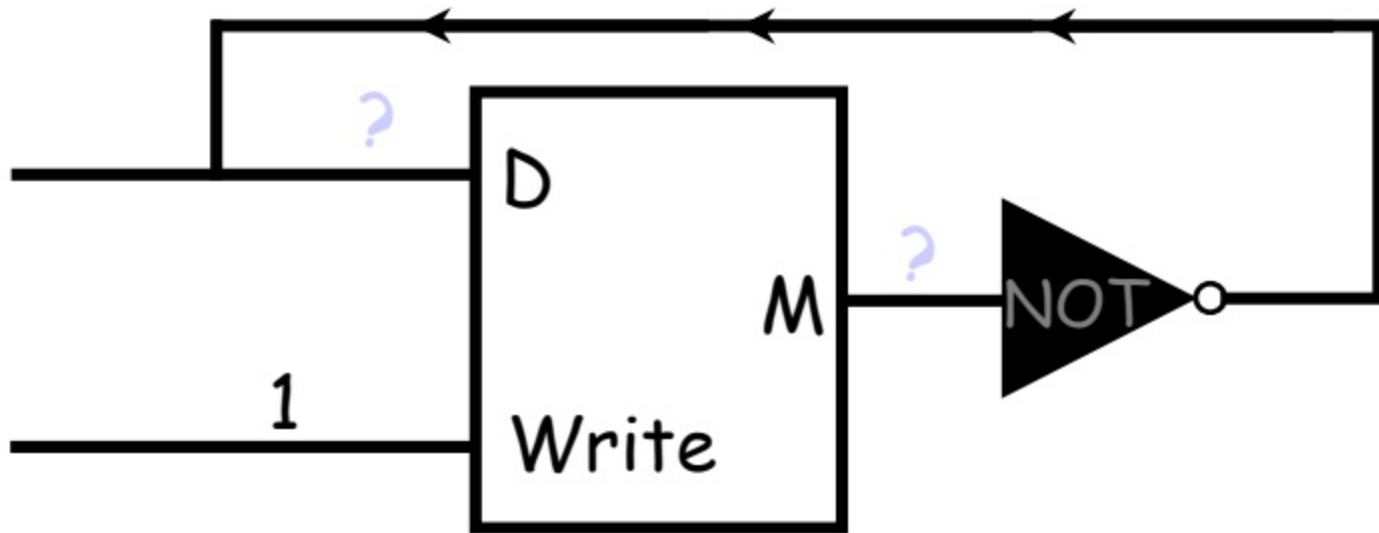
- When Write = 1, then M = D.

- If we have some feedback between M and D, then circuit could go haywire.
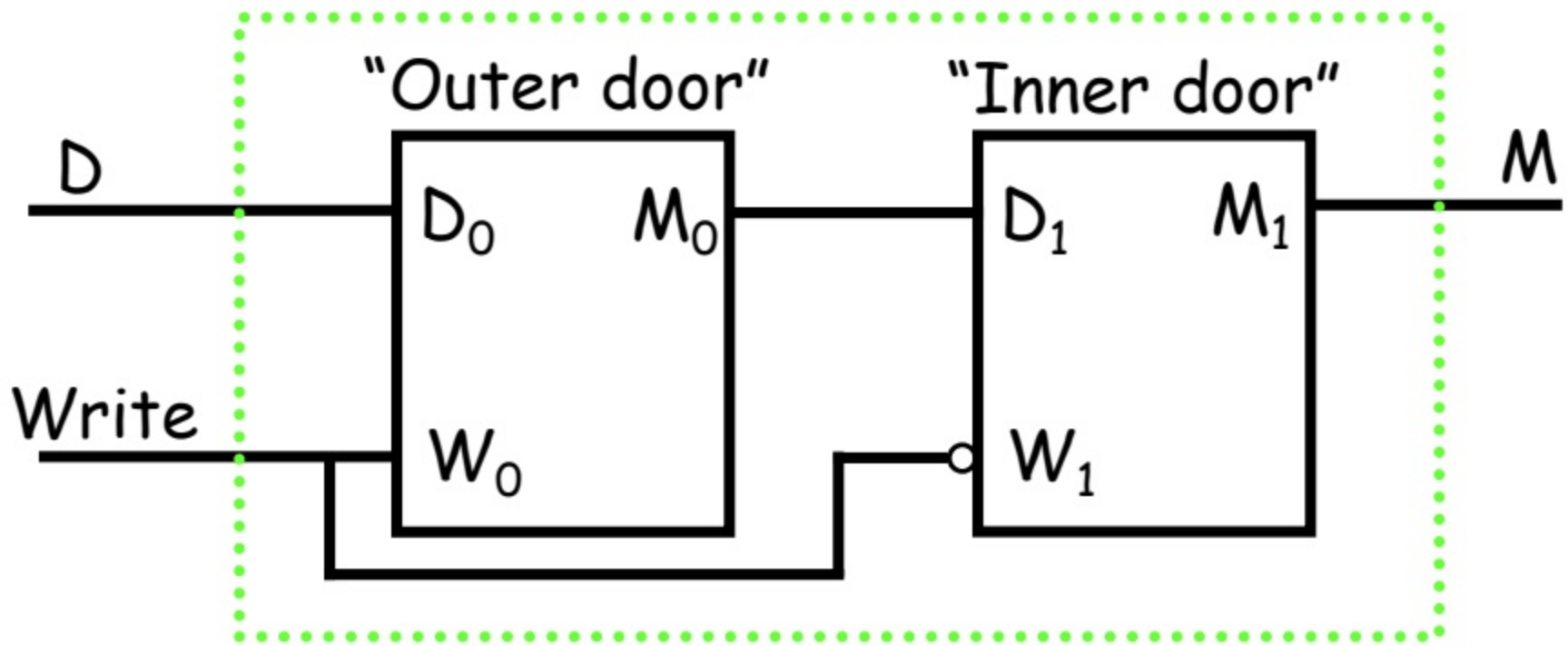
A subtle problem

- For example, suppose NOT gate connects M and D.

- When Write = 1,
  M and D keep changing. We have no control.
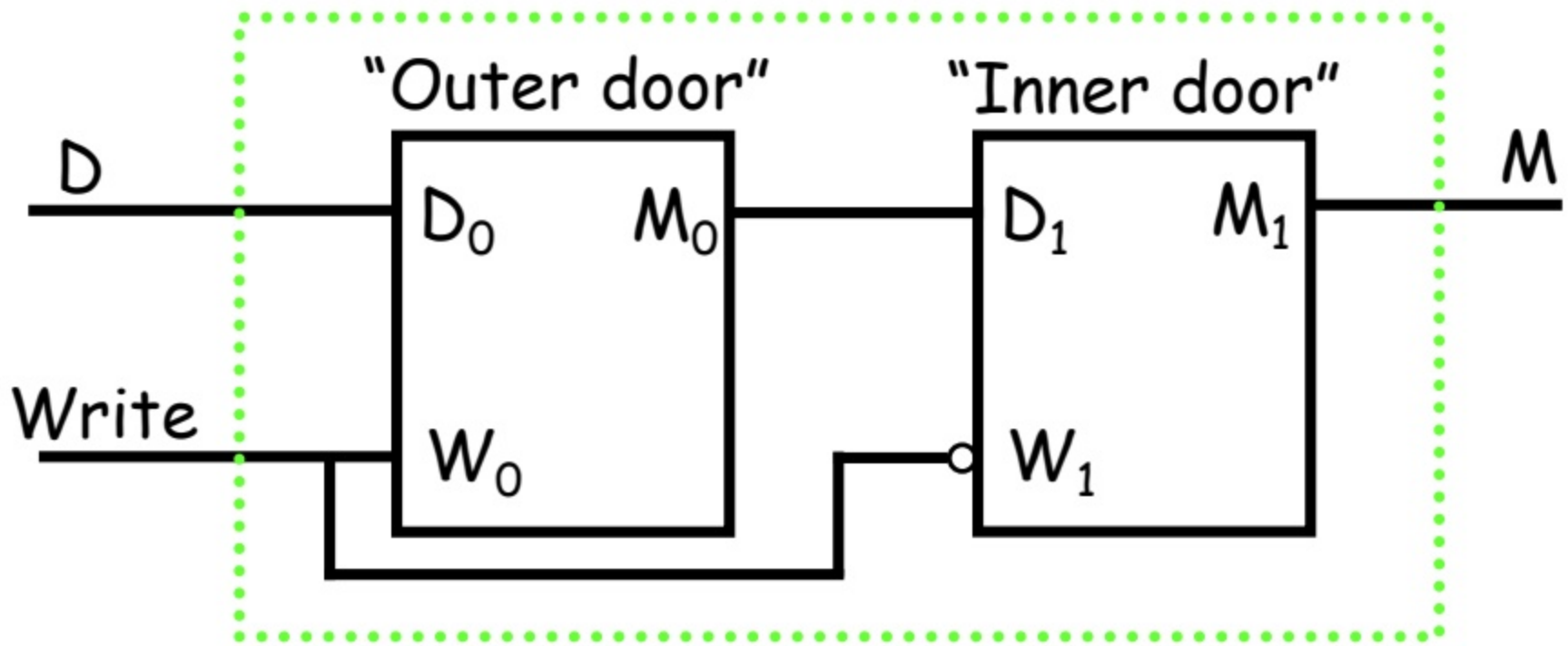
A subtle problem

- We want to control the feedback, so that each time we set Write to 1 and then back to 0, M stores only the last value of D

- (In this case, M should invert itself once each time we set Write to 1 and back to 0)
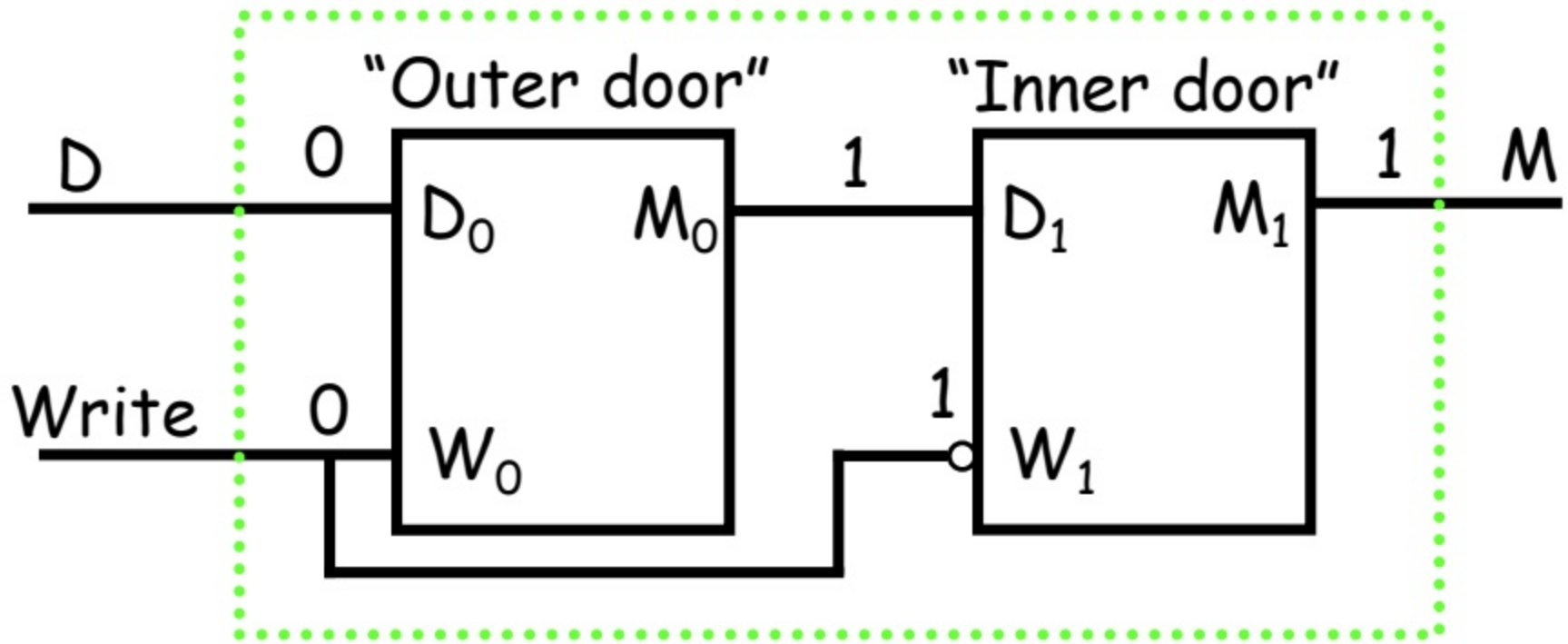
"Airlock" Flip-Flop

- Two-Stage System to prevent feedback loop.
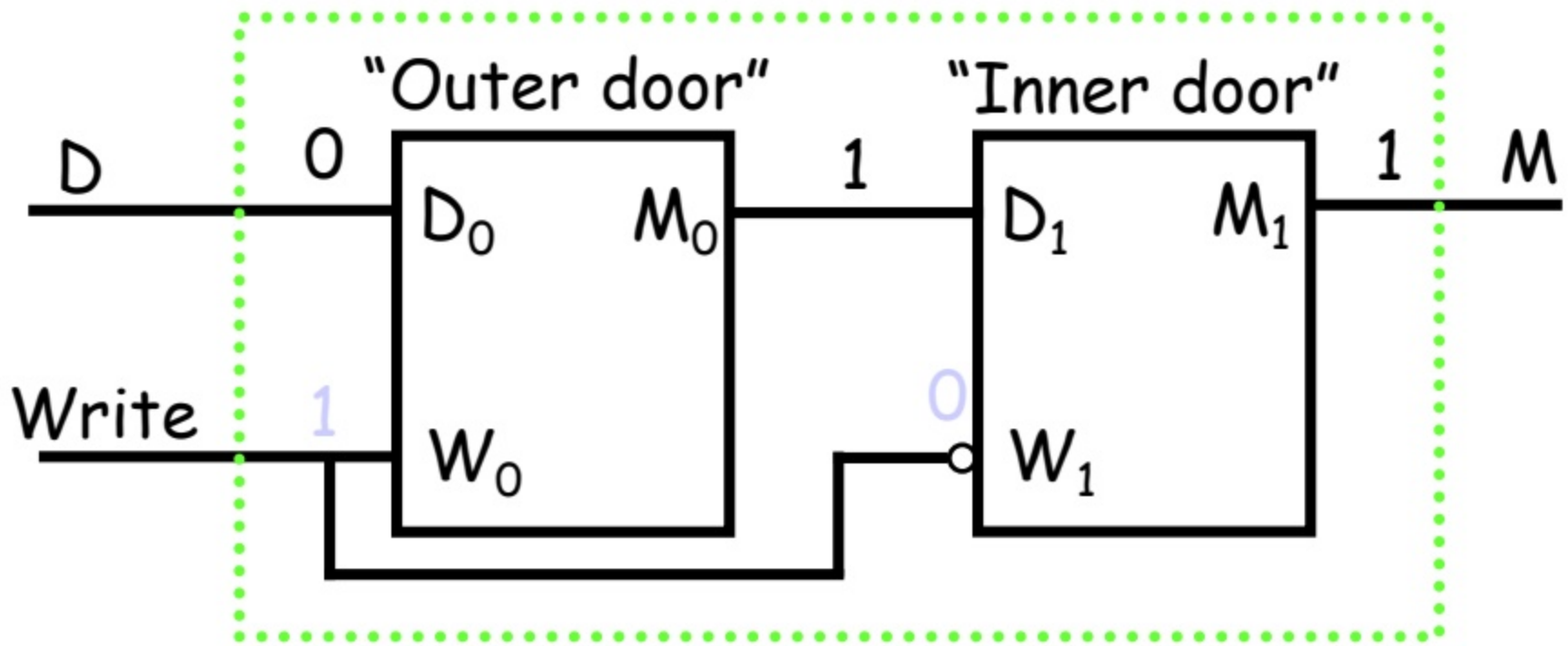
"Airlock" Flip-Flop

"Outer door"   "Inner door"

D —— D_0 M_0 —— D_1 M_1 —— M

Write —— W_0 W_1

- We start with Write = 0.

- Let's say D is always NOT M.
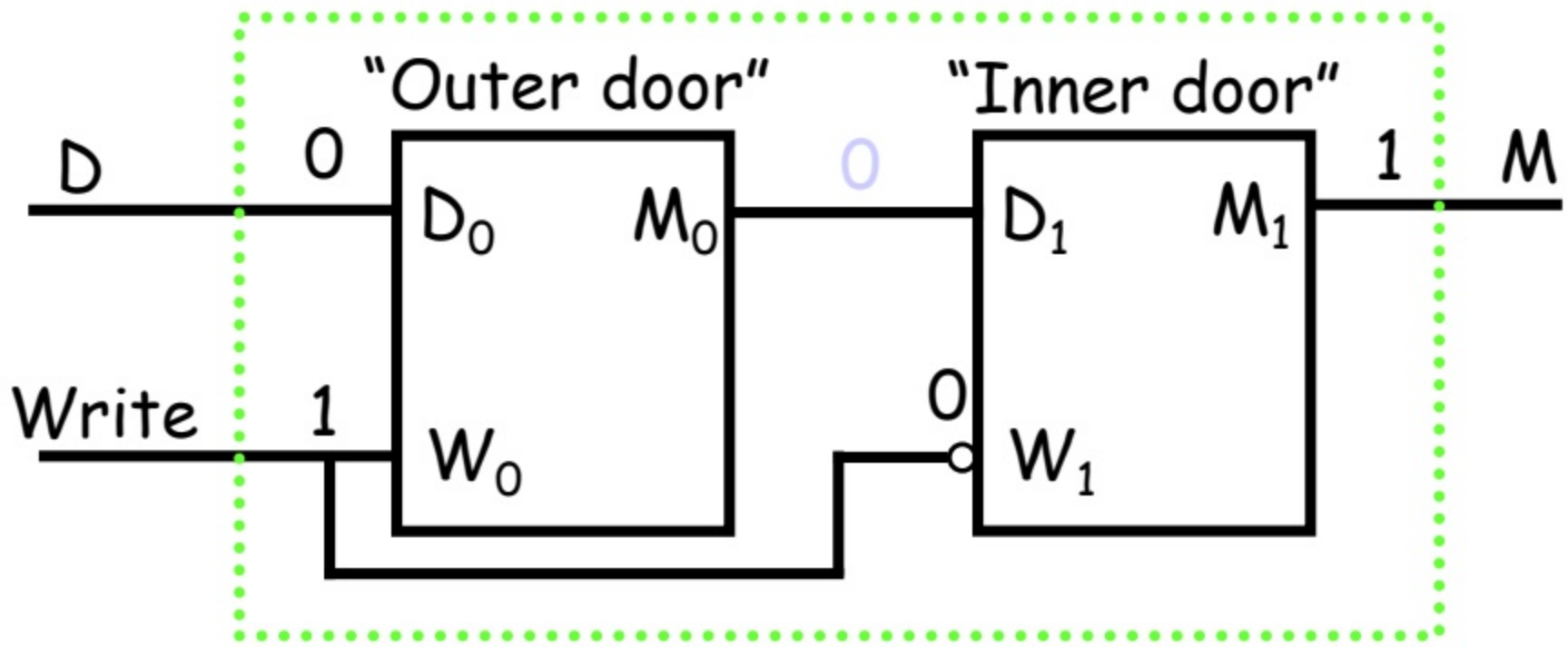 Start with D = 0, M = 1.

"Airlock" Flip-Flop

- We start with Write = 0.

- Let's say D is always NOT M.
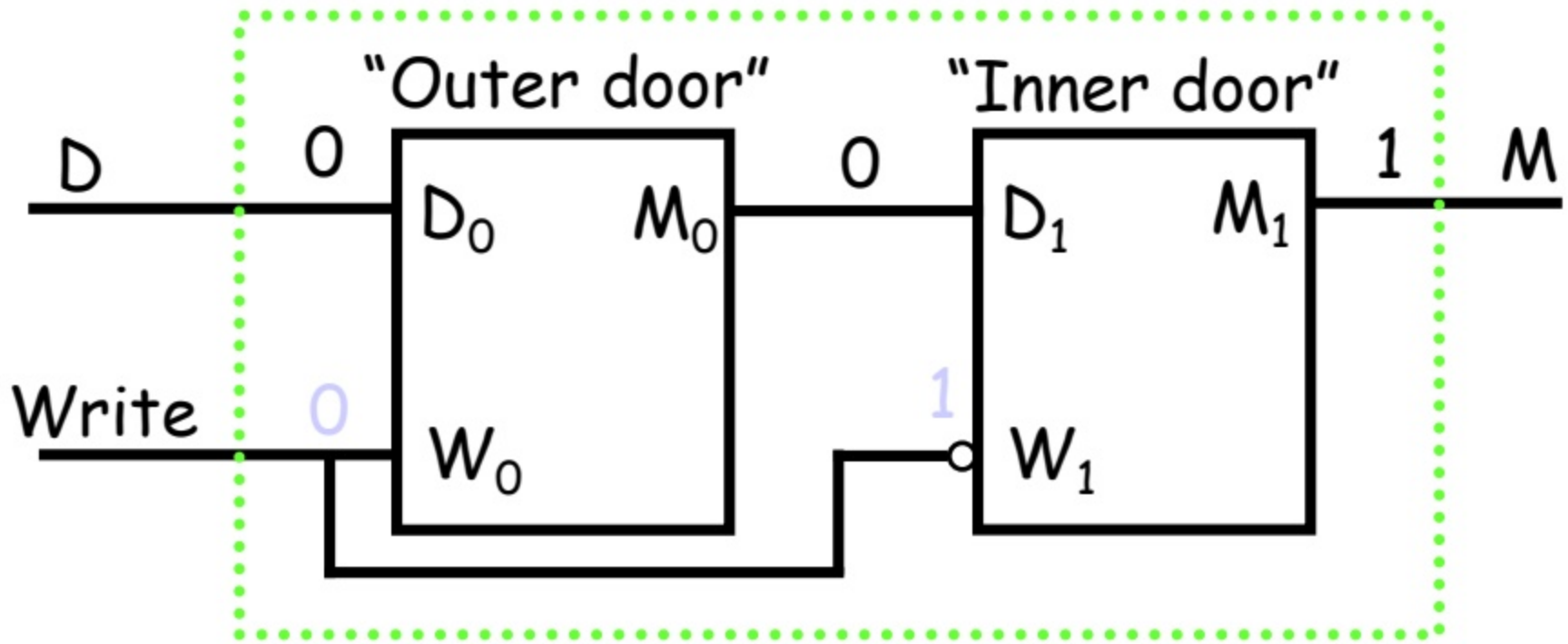 Start with D = 0, M = 1.

"Airlock" Flip-Flop

- Want to store D in memory.
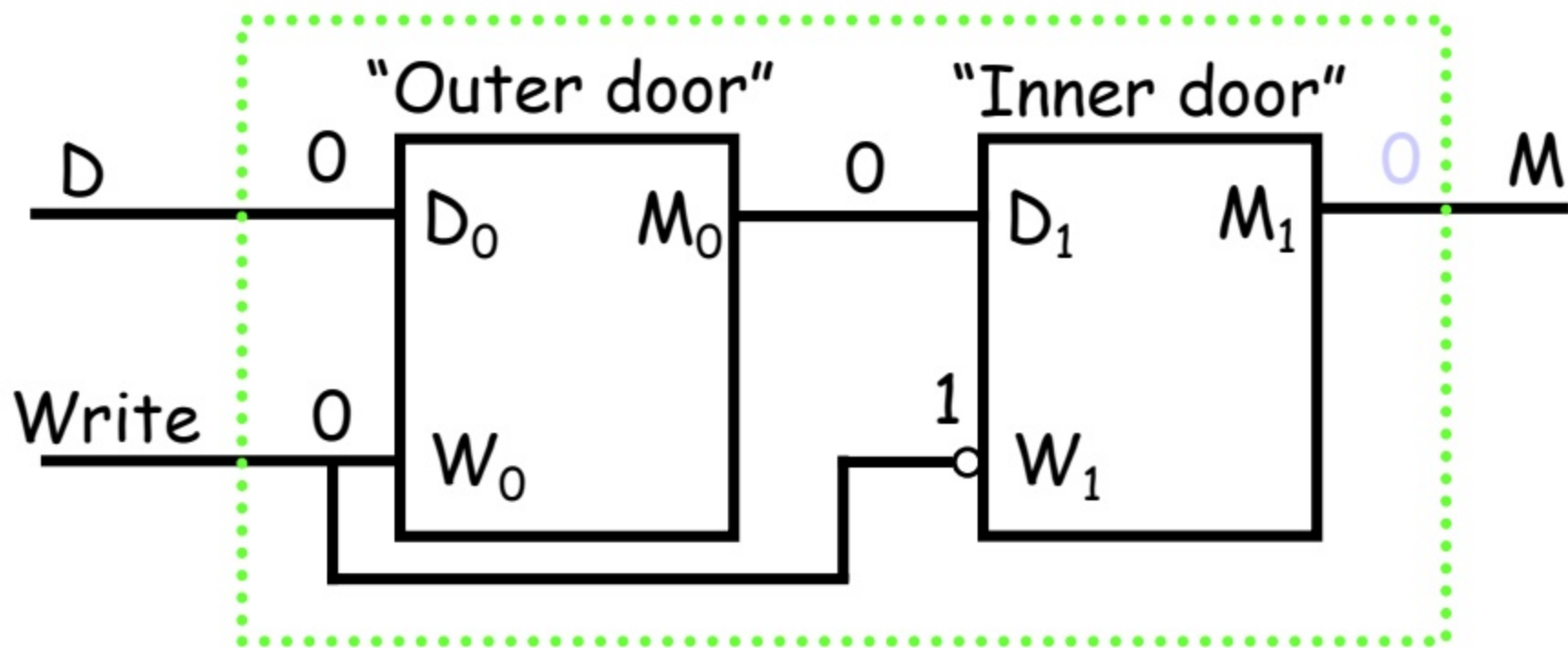- Set Write to 1

"Airlock" Flip-Flop

- Want to store D in memory.

- Set Write to 1

- "Outer" flip-flop sets $M_0 = D_0 = 0$

- "Inner" flip-flop ignores D_1 since $W_1 = 0$

"Airlock" Flip-Flop

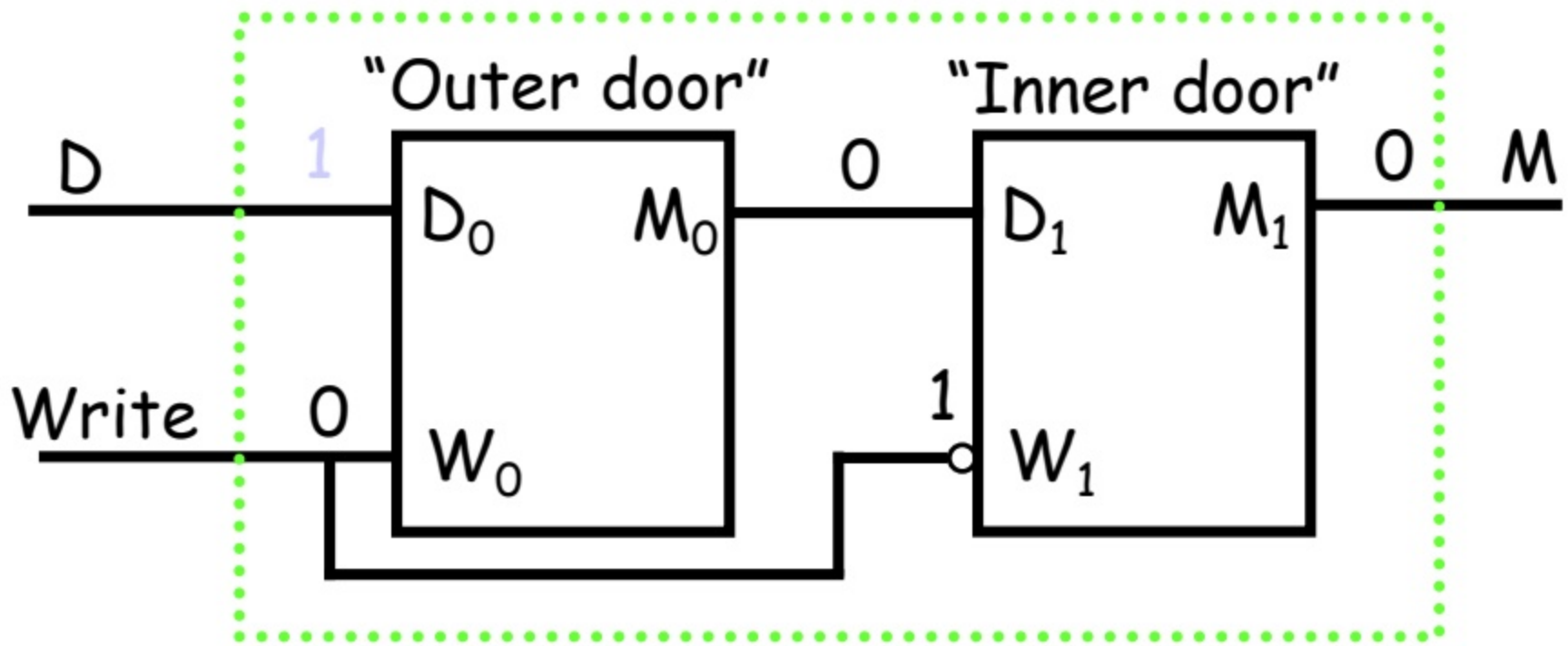"Outer door" "Inner door"

D ——— 0 ——— D_0 M_0 ——— 0 ——— D_1 M_1 ——— 1 ——— M

Write ——— 0 ——— W_0 1 W_1

- Now, set Write back to 0

"Airlock" Flip-Flop

- Now, set Write back to 0
- Now "Inner" flip-flop sets $M = D_1 = 0$

"Airlock" Flip-Flop

"Outer door" "Inner door"

D —— 1 —— D_0 M_0 —— 0 —— D_1 M_1 —— 0 —— M
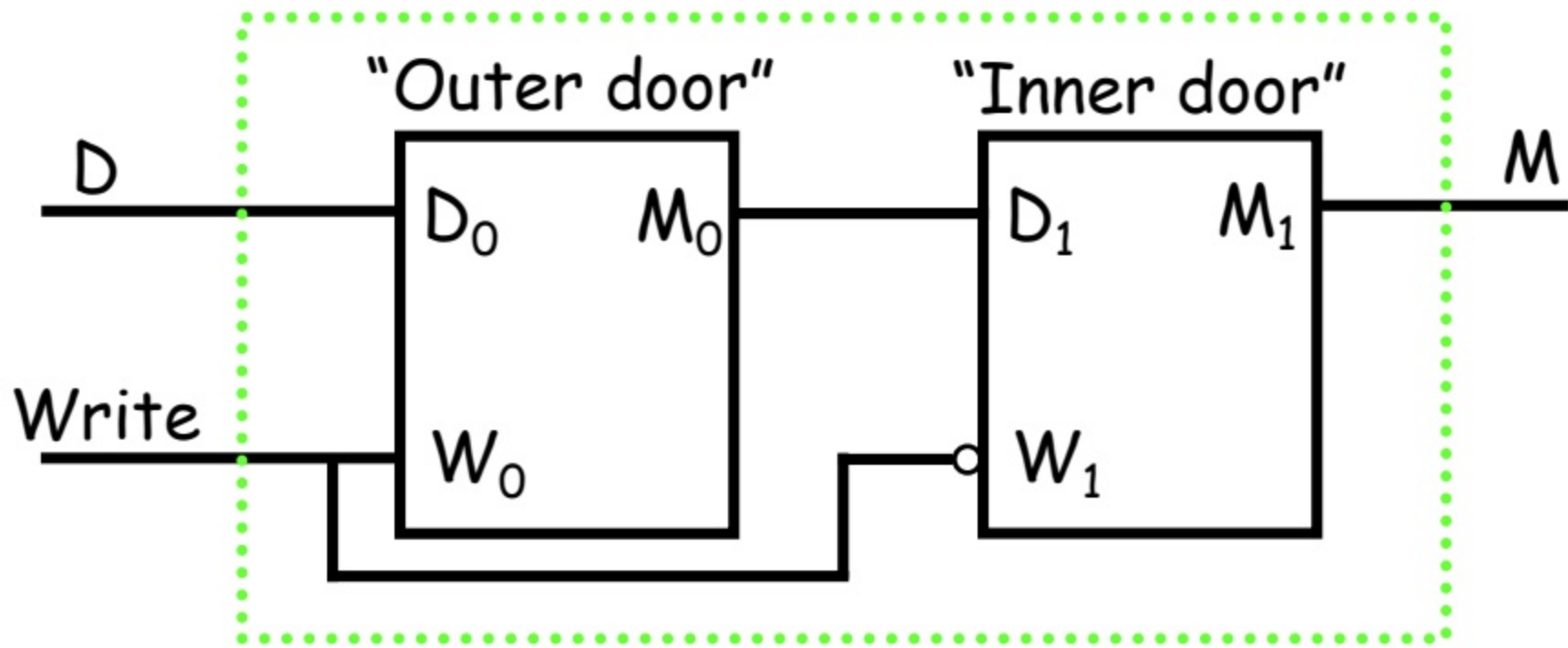
Write —— 0 —— W_0 —— 1 —— W_1

- Because of feedback, D might change to (NOT M) , which is 1

- But Write = 0, so "Outer" flip-flop ignores D, and so M_0 stays 0.

"Airlock" Flip-Flop

"Outer door"   "Inner door"

D —1—→ D_0 M_0 —0—→ D_1 M_1 —0—→ M

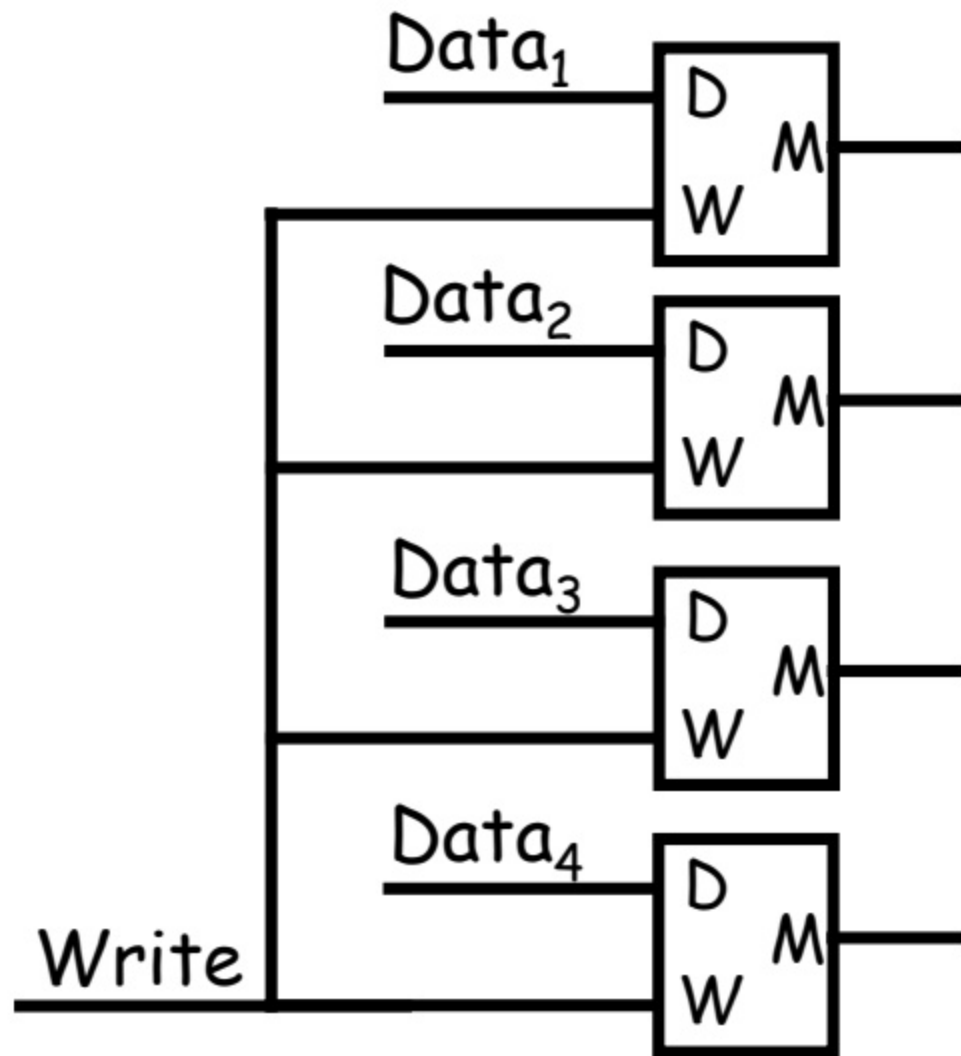Write —0—→ W_0 1 ○ W_1
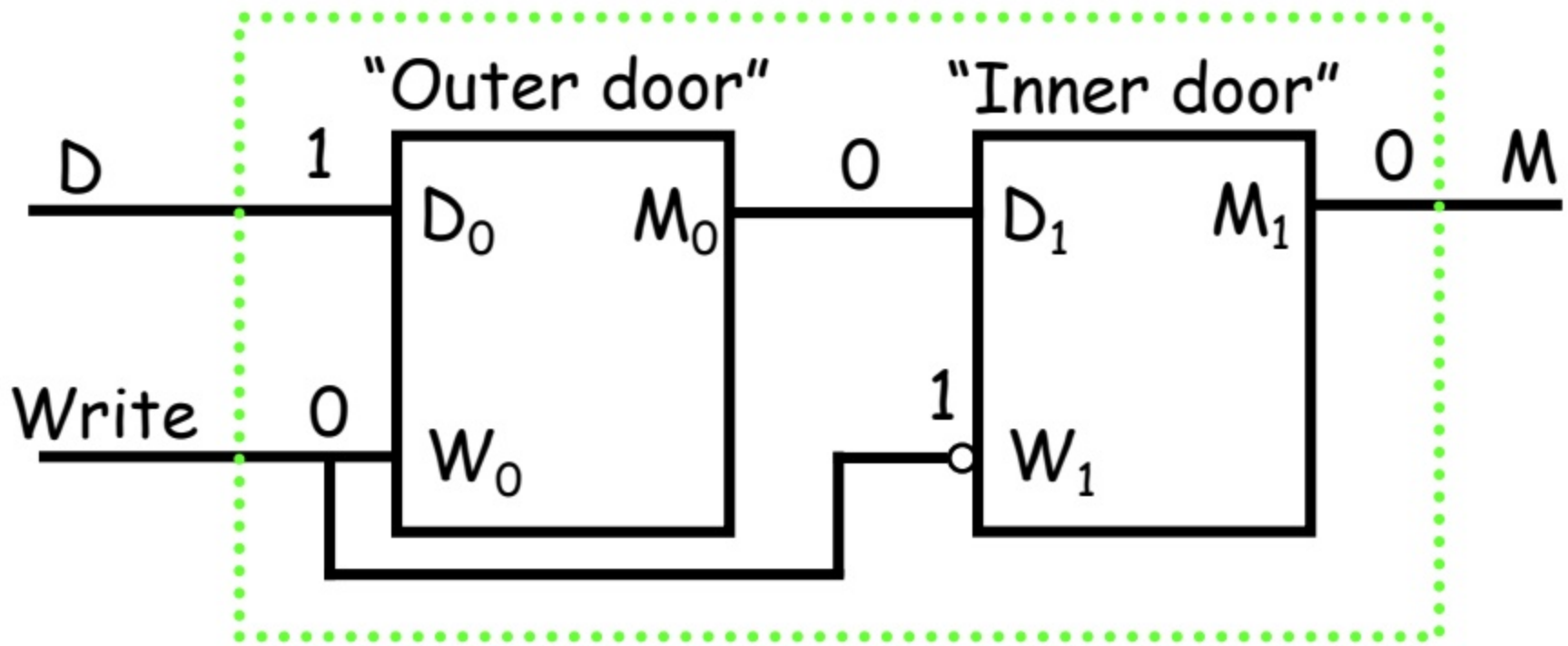
- So memory does not change until we "toggle" Write.

- ("toggle" means change from 0 to 1 or vice versa)

"Airlock" Flip-Flop

- This is Real Memory!

Memory "Register": 4 bits

Review

- We have used the Universal method to build
  - ALU
  - Memory

- Next steps
  - State machines to computer with memory
  - Building the computer
  - Writing a program to use the computer