Discussion

- What did you learn from the Boole/Clarke “proof” of the existence of God?
- How convincing was it?

Part1: Wrapup of Boolean logic and Combinational circuits

COS 116

3/16/2006

Instructor: Sanjeev Arora

Pickup: HW3, Blogging Assignment.

Combinational circuit for binary addition

	25	11001
+	29	11101
	54	110110

- Desired: circuit for adding any two *N*-bit integers

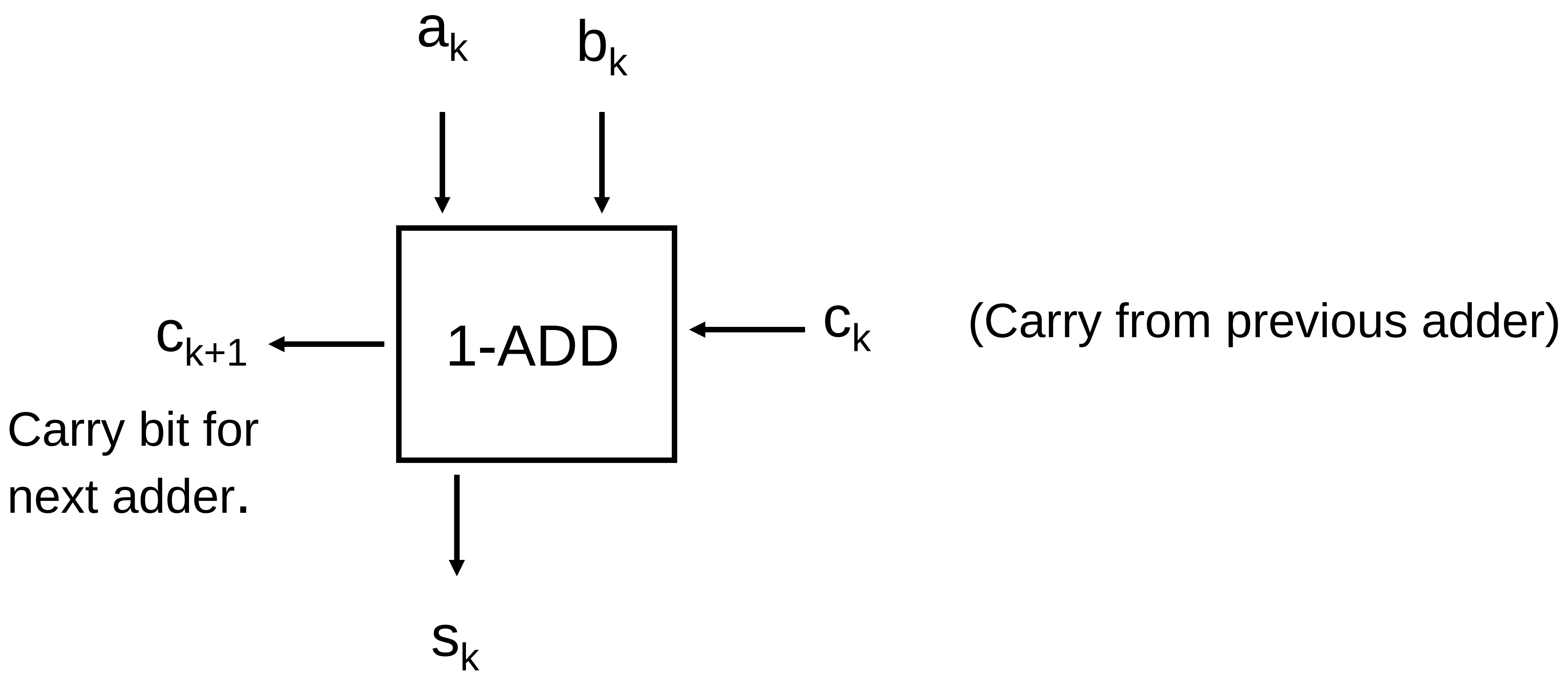
Modular design

$$
\begin{array}{lcccccc}
 & & c_{N-1} & c_{N-2} & \dots & c_1 & c_0 \\
 & & a_{N-1} & a_{N-2} & \dots & a_1 & a_0 \\
+ & & b_{N-1} & b_{N-2} & \dots & b_1 & b_0 \\
\hline
 & s_N & s_{N-1} & s_{N-2} & \dots & s_1 & s_0
\end{array}
$$

Carry bits

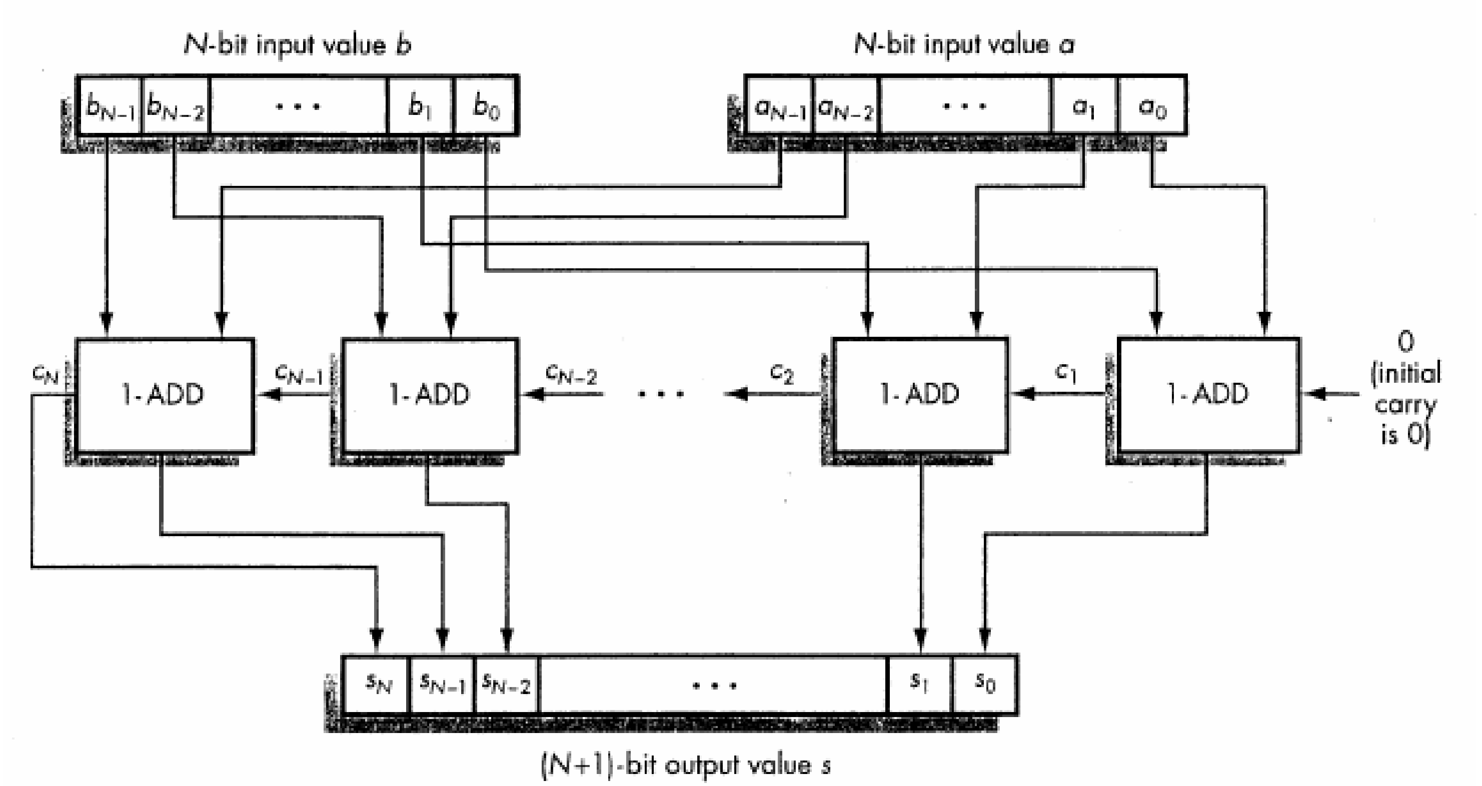
Need *N* 1-bit adders

1-bit adder



Do yourself: Write truth table, circuit.

A Full Adder (from handout)



Memory in boolean circuits

COS 116
3/16/2006
Instructor: Sanjeev Arora

Going beyond combinational circuits

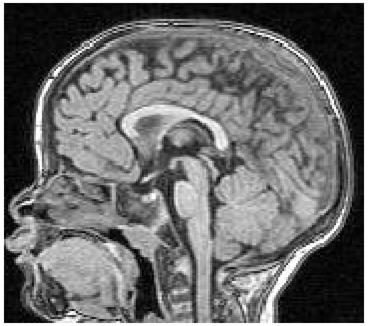
- Need 2-way communication between circuits (i.e. need cycles!)

CPU

Ethernet card

- Need memory (scratchpad)

What do you understand by ‘memory”?

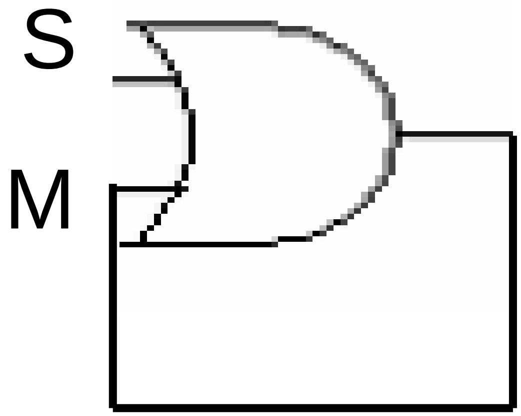


How can you tell that a 1-year old child has it?

Behaviorist’s answer: His/her actions depend upon past events.

Matt likes Sue but he doesn't like changing his mind

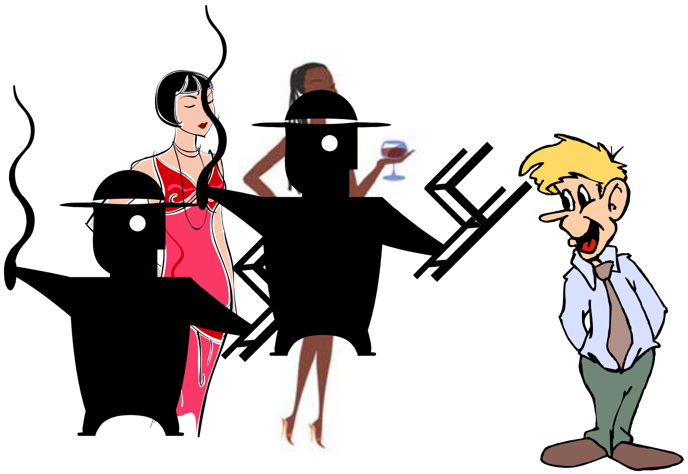
- Represent with a circuit: Matt will go to the party if Sue goes or if he already wanted to go

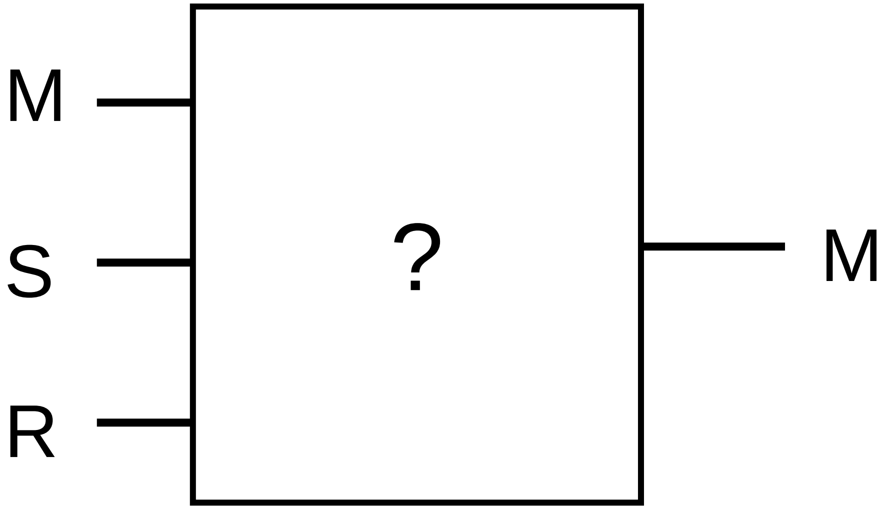


Is this well-defined?

Enter Rita

- Matt will go to the party if Sue goes OR if the following holds: if Rita does not go and he already wanted to go



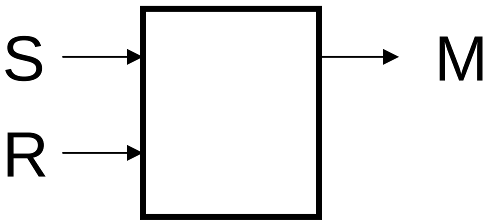


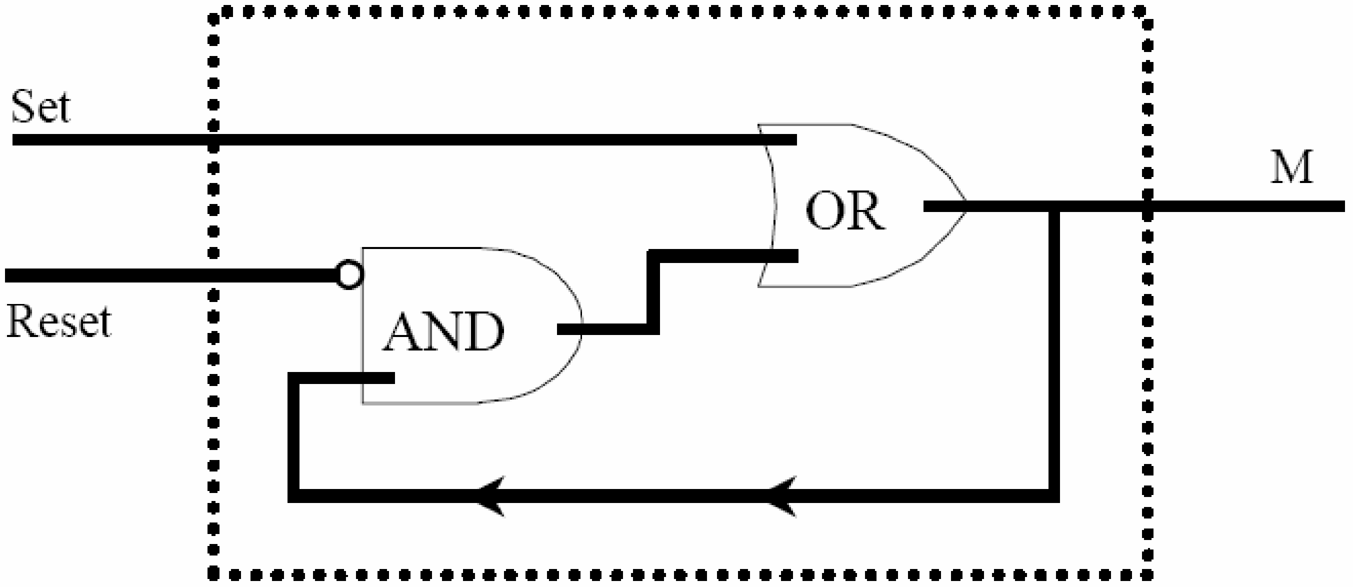
R, S: “**control**” inputs

What combination of R, S changes M?

Flip-Flop

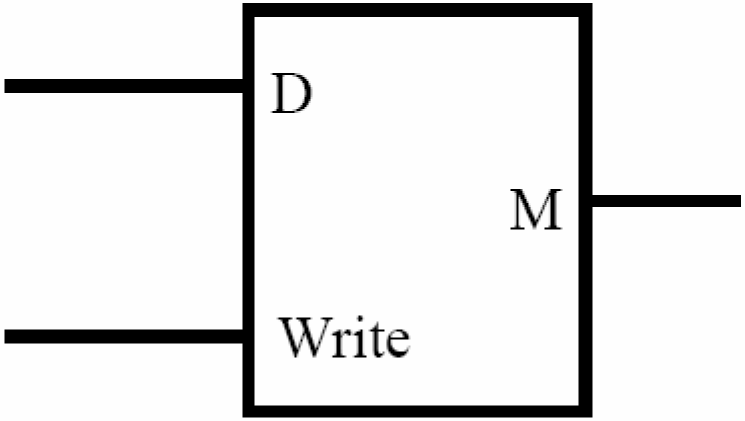
Flip-Flop





- M becomes 1 if Set is turned on
- M becomes 0 if Reset is turned on
- Otherwise (if both are 0), M just remembers its value

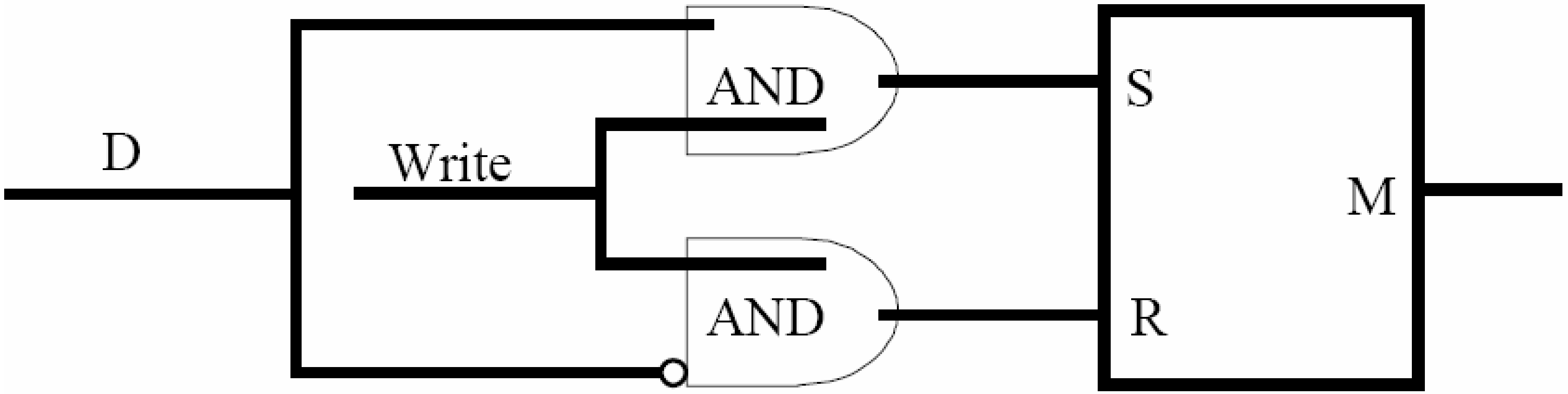
Desired: more convenient form of memory



- If Write = 0, M just keeps its value. (It ignores D.)
- If Write = 1, then M becomes set to D

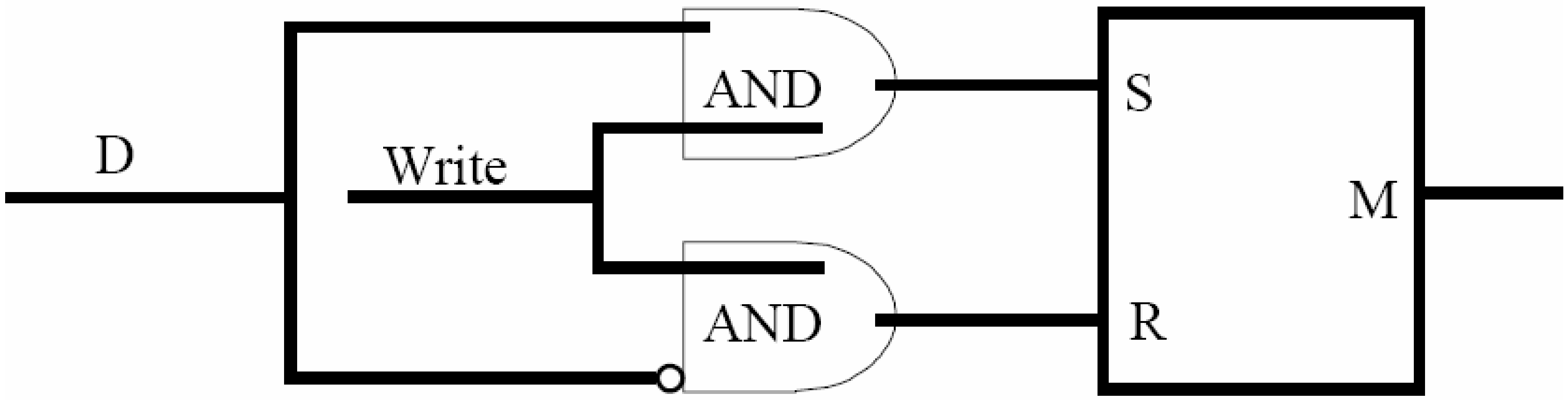
"Data Flip-Flop" or "D flip flop."

Design of D Flip Flop



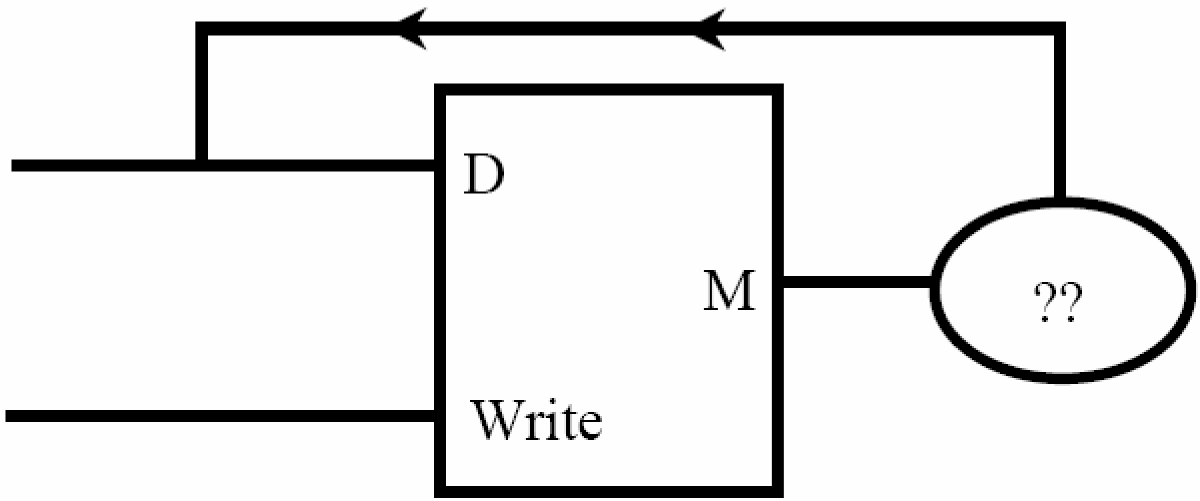
- Nothing happens unless Write = 1

The D 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.)

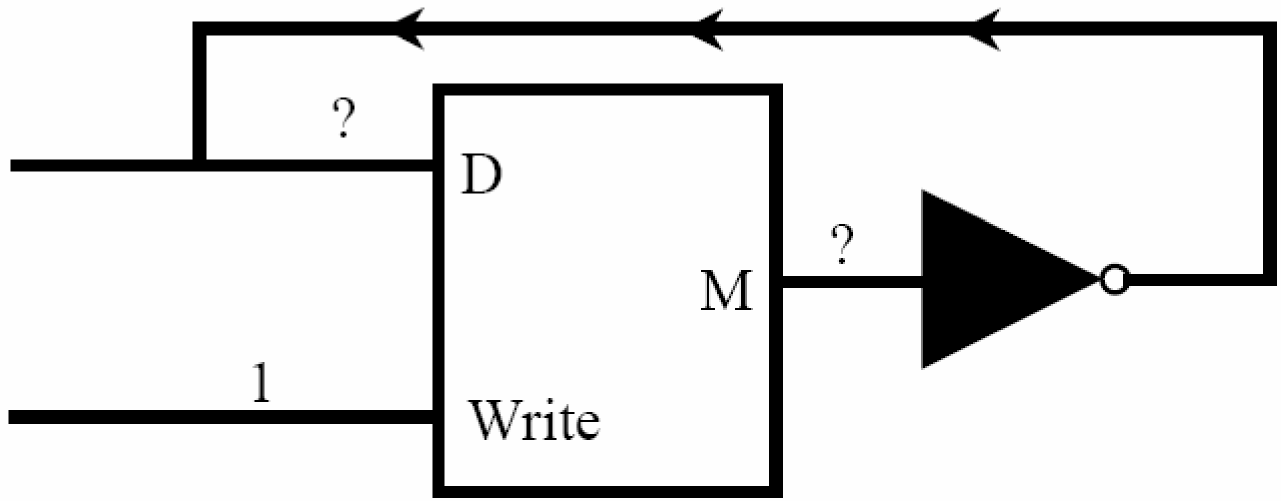
A Subtle Problem



"Race condition."

- When Write = 1, then M = D.
- If we have some feedback between M and D, then circuit could go haywire.

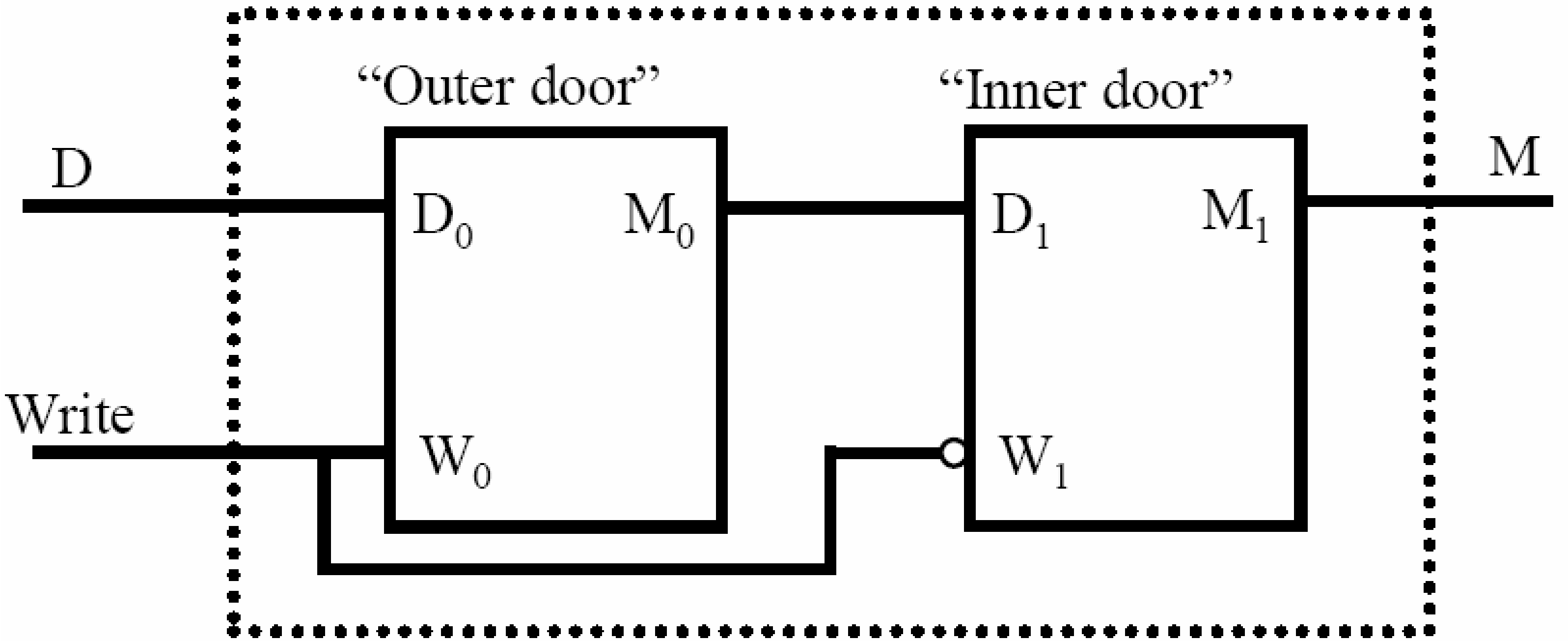
Example



- For example, suppose a NOT gate connects M and D.
- When Write = 1, M and D keep changing. We have no control.

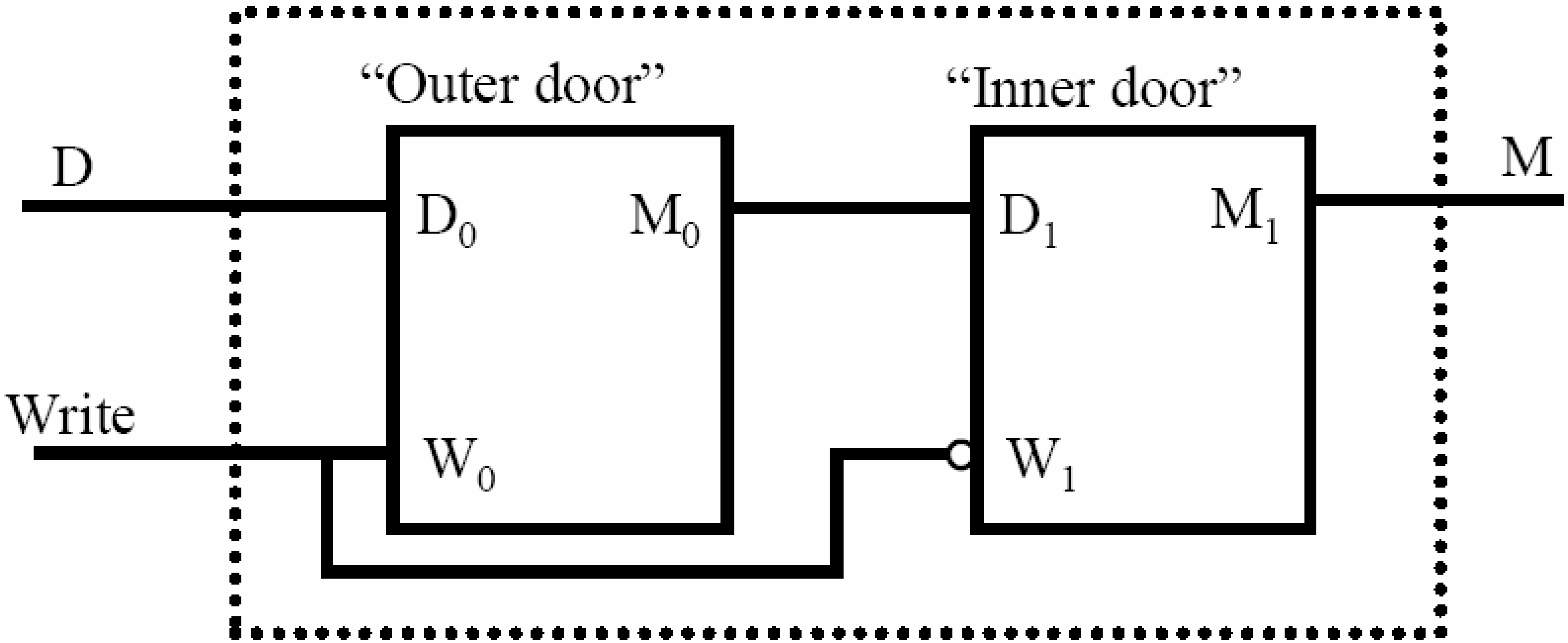
Desired: M should invert only **<u>once</u>** when we make Write =1

The "Airlock" Flip-Flop



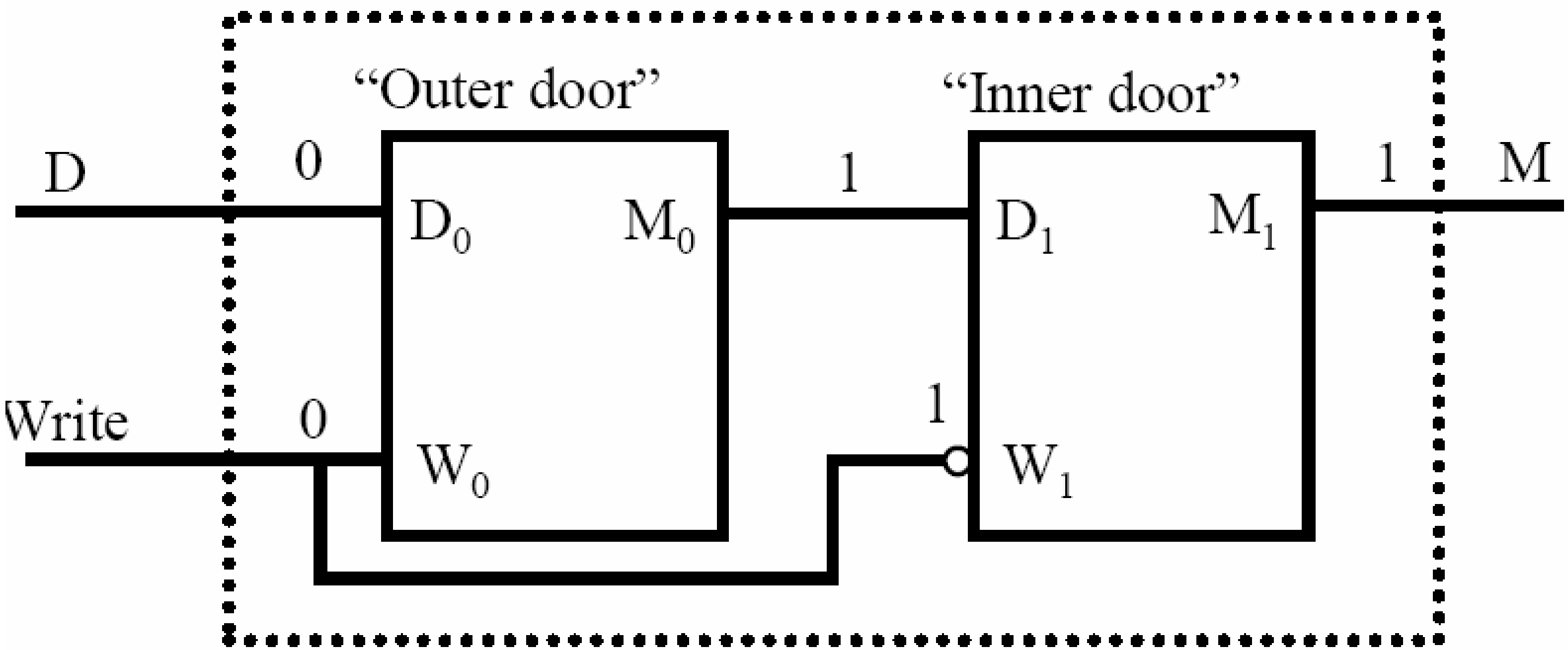
- Two-Stage System to prevent D ever passing through directly to M (W_0,W_1 connected by NOT, so never 1 at the same time)

The “Airlock” Flip-Flop



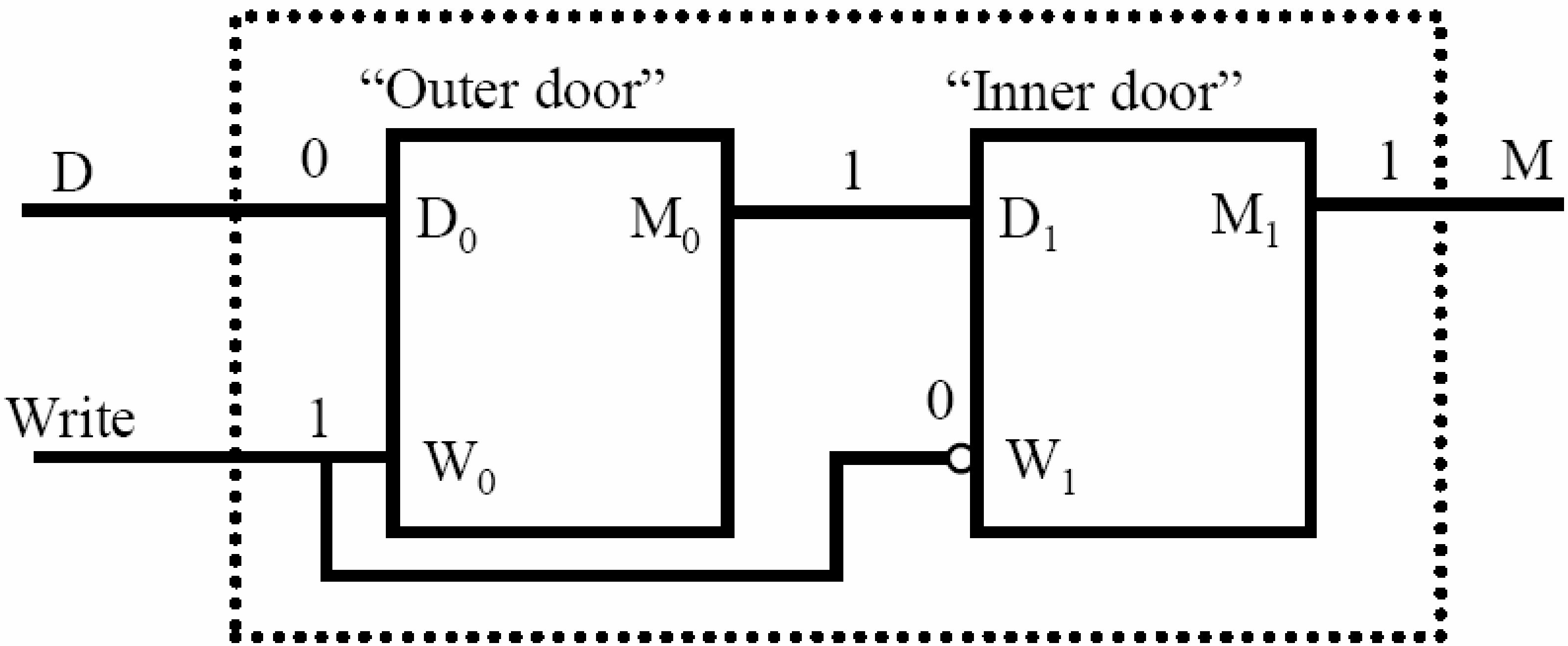
- We start with Write = 0.
- Let’s say D is always NOT M; i.e. connected by NOT gate. Start with D = 0, M = 1.

The “Airlock” Flip-Flop



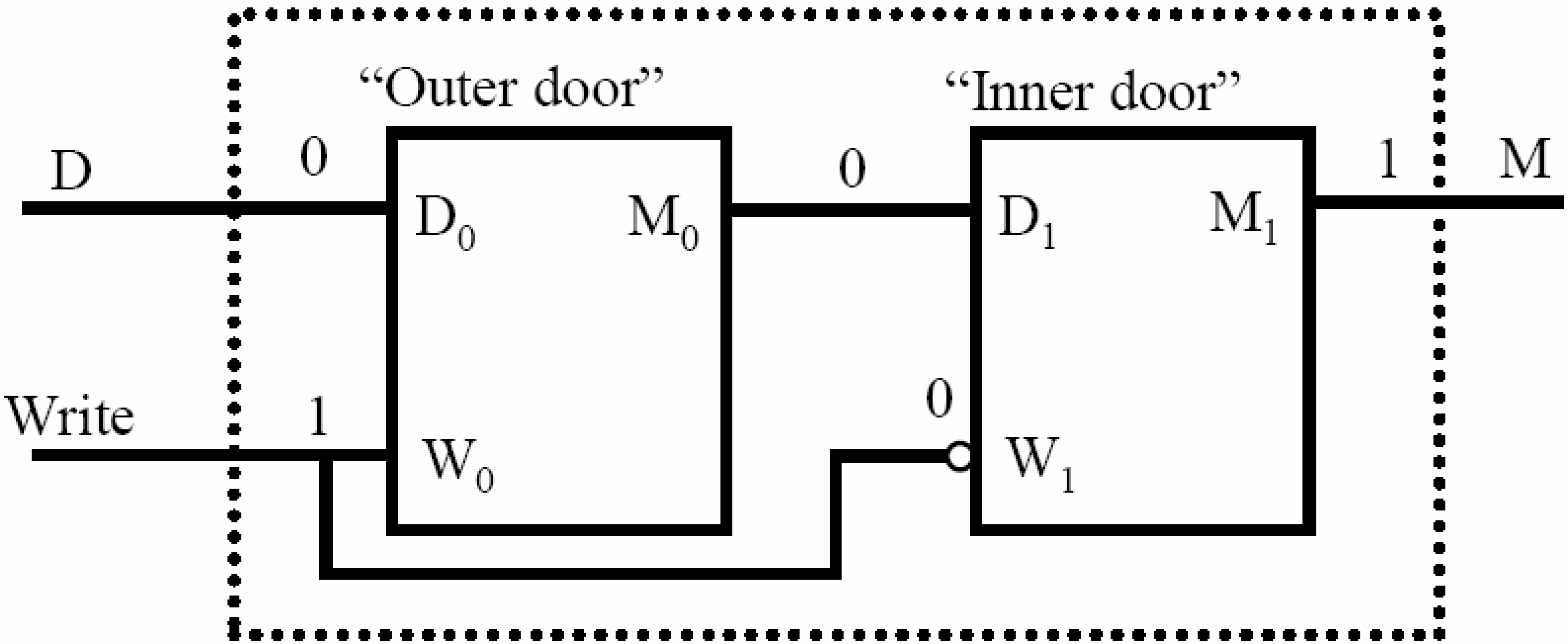
- We start with Write = 0.
- Let’s say D is always NOT M; i.e. connected by NOT gate
  Start with D = 0, M = 1.

The “Airlock” Flip-Flop



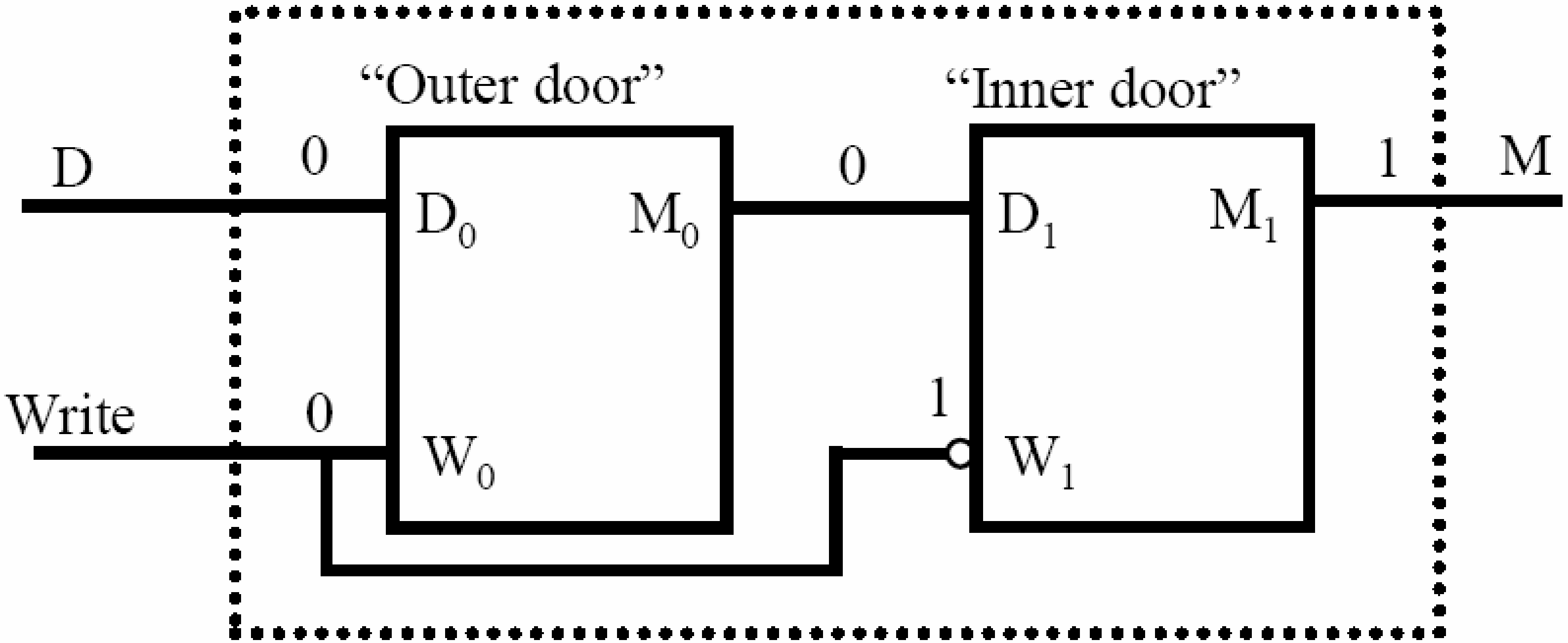
- Want to store D in memory.
- Set Write to 1

The "Airlock" Flip-Flop



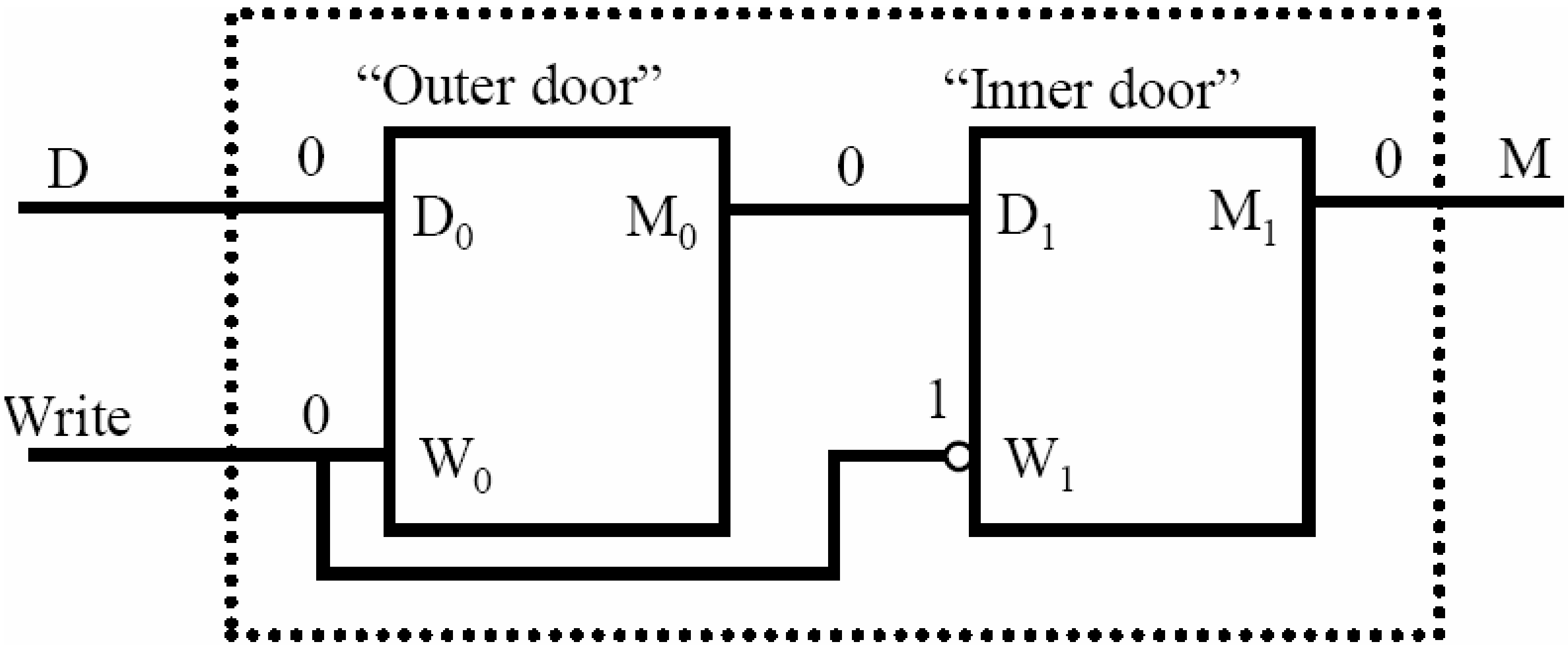
- Want to store D in memory.
- Set Write to 1
- "Outer" flip-flop sets $M_0 = D_0 = 0$
- "Inner" flip-flop ignores D_1 since $W_1 = 0$

The “Airlock” Flip-Flop



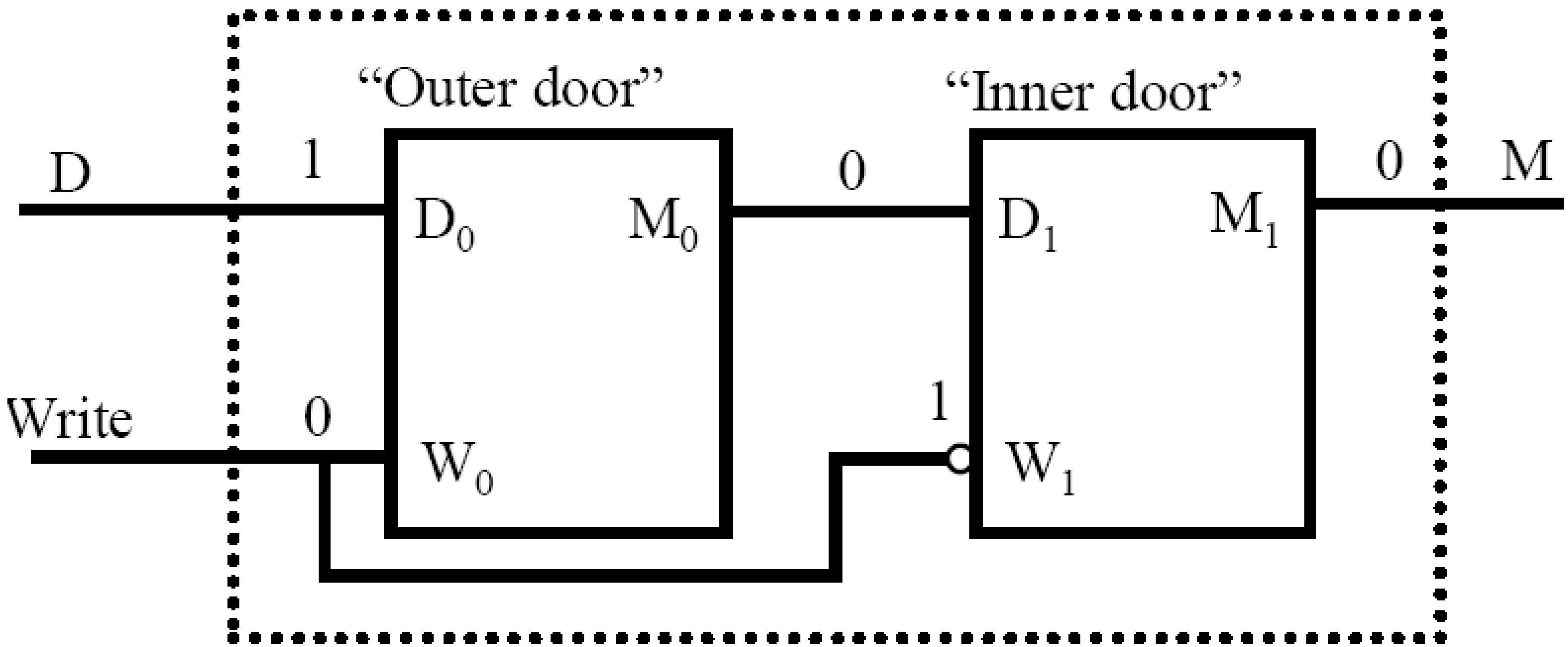
- Now, set Write back to 0

The “Airlock” Flip-Flop



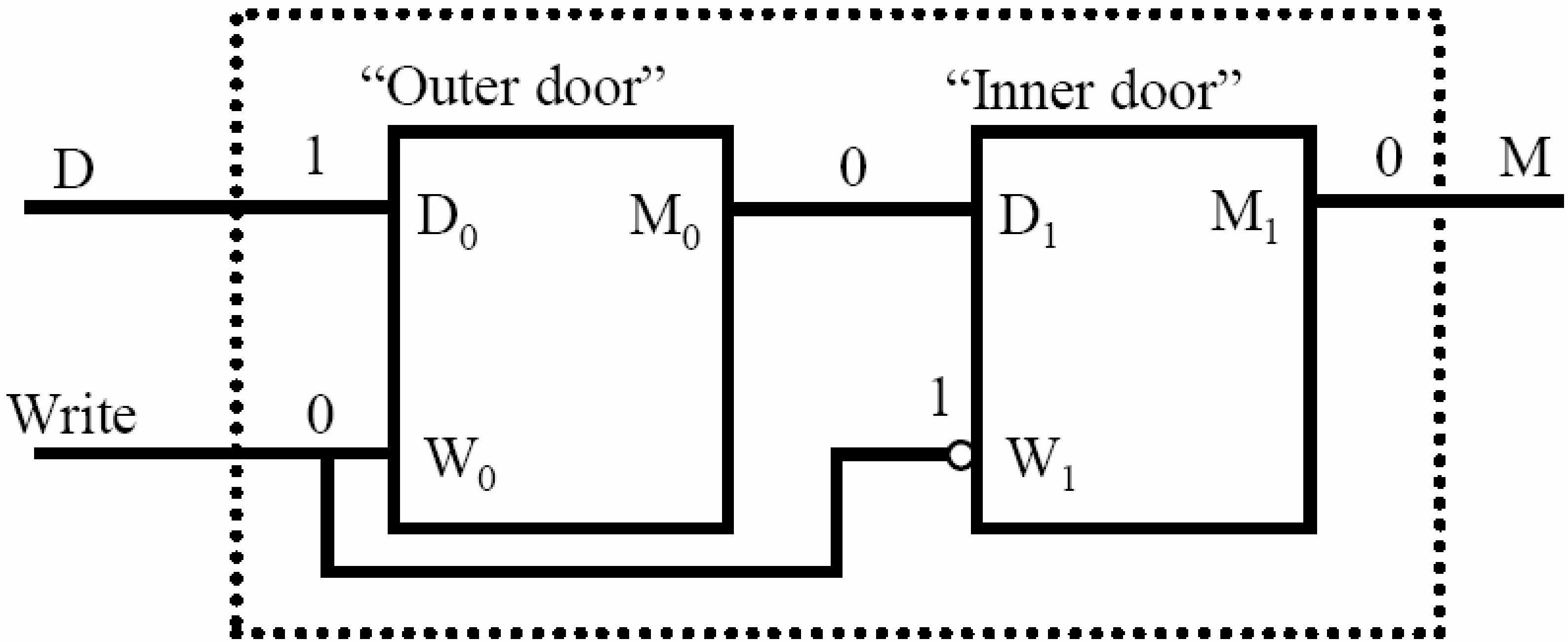
- Now, set Write back to 0
- Now “Inner” flip-flop sets $M = D_1 = 0$

The “Airlock” Flip-Flop



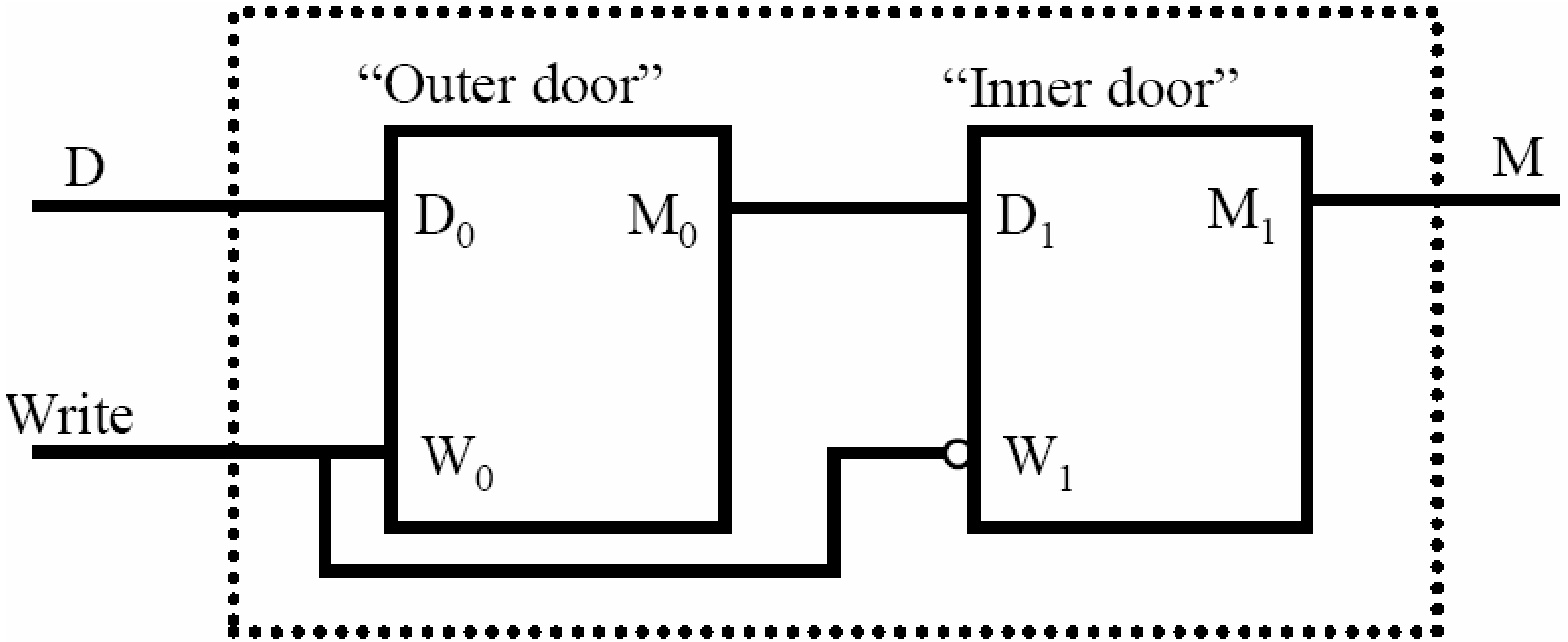
- Because of feedback, D might change to (NOT M) , which is 1
- But Write = 0, so “Outer” flip-flop ignores D, so M_0 stays 0.

The “Airlock” Flip-Flop



- So memory does not change until we “toggle” Write.
- (“toggle” means change from 0 to 1 or vice versa)

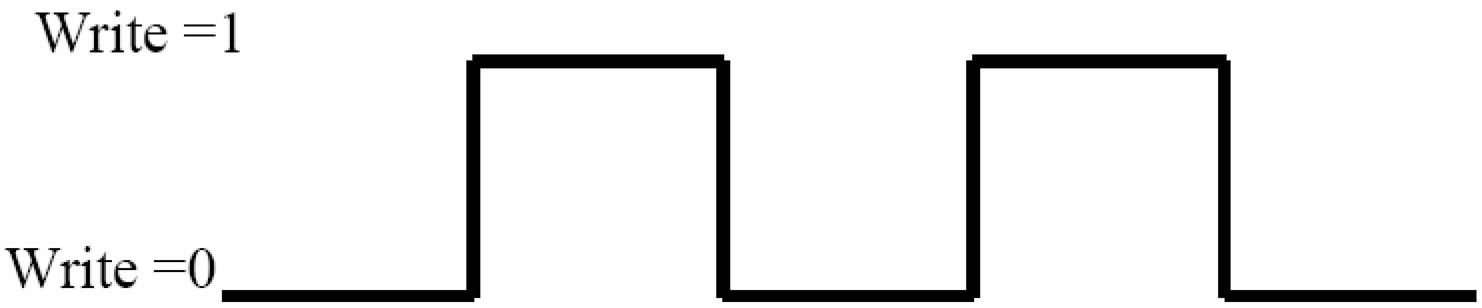
The “Airlock” Flip-Flop



- This is Real Memory!

What controls the “Write” signal?

- Often, the system clock!
- “clock” = device that sends out a fluctuating voltage signal that looks like this



“Computer speed” often refers to the clock frequency (e.g. 2.4GHz)

Next time

Finite State Machines and Clocked Circuits

Memory "Register": 4 bits