Sequential and Clocked Circuits; Finite State Machines

3/28/2006

COS 116

Instructor: Sanjeev Arora

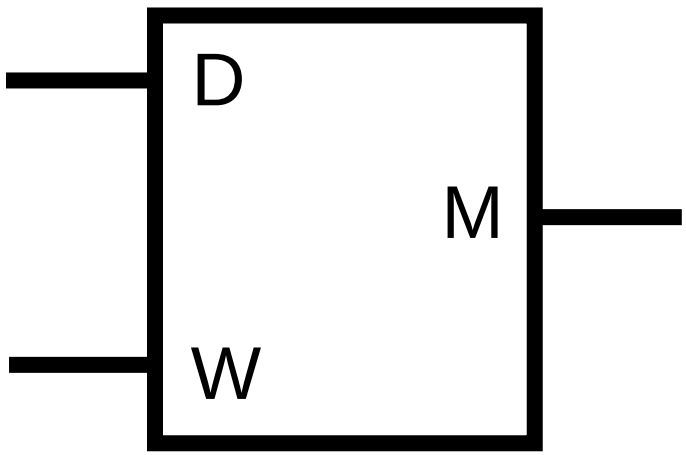
Sequential Circuits (Recap.)

- Circuits with AND, OR and NOT gates.
- Cycles ***are*** allowed.
- Can exhibit “memory”.
- May exhibit instabilities (saw last time).

Recap: D Flip Flop

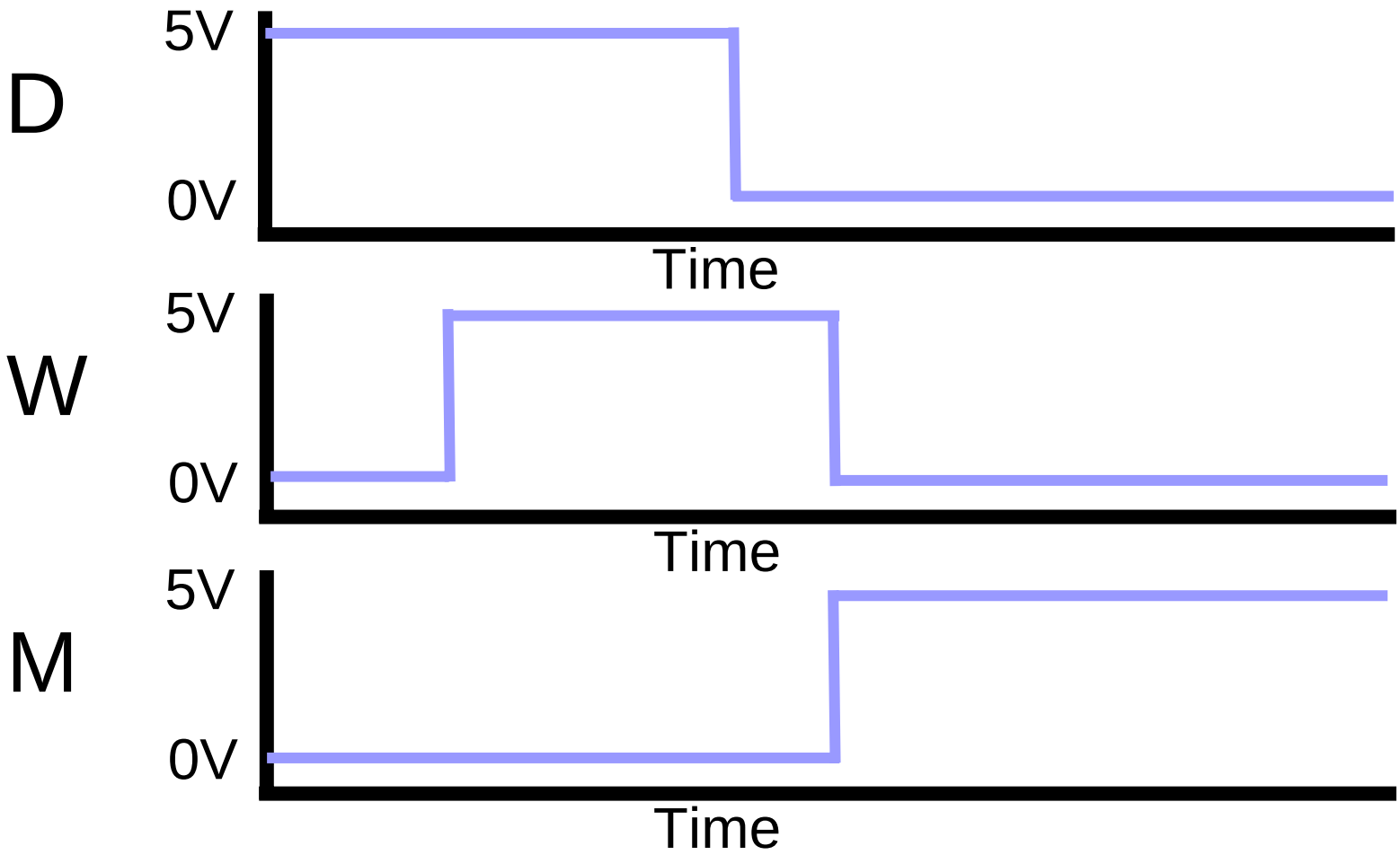
a.k.a. “Airlock”, “Master-Slave”

Basic Memory Block – stores 1 bit.



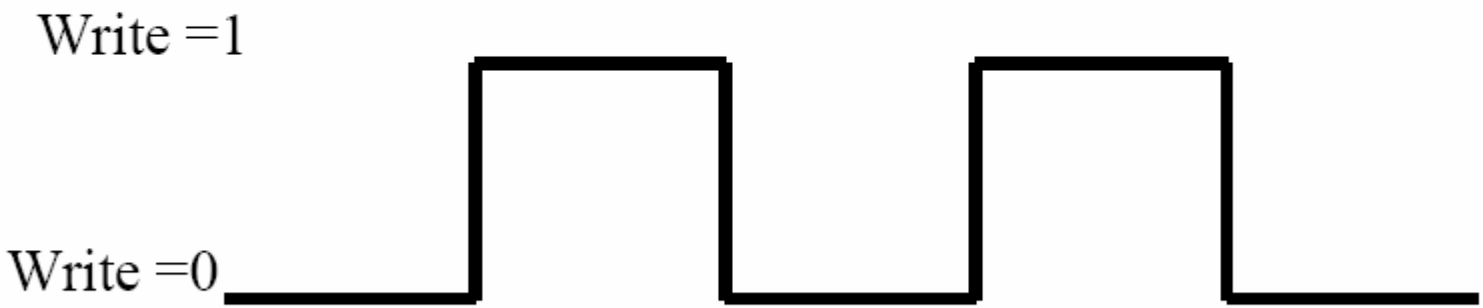
If we “toggle” the write input (setting it 1 then setting it 0) then M acquires the value of D.

Timing Diagram



What controls the “Write” signal?

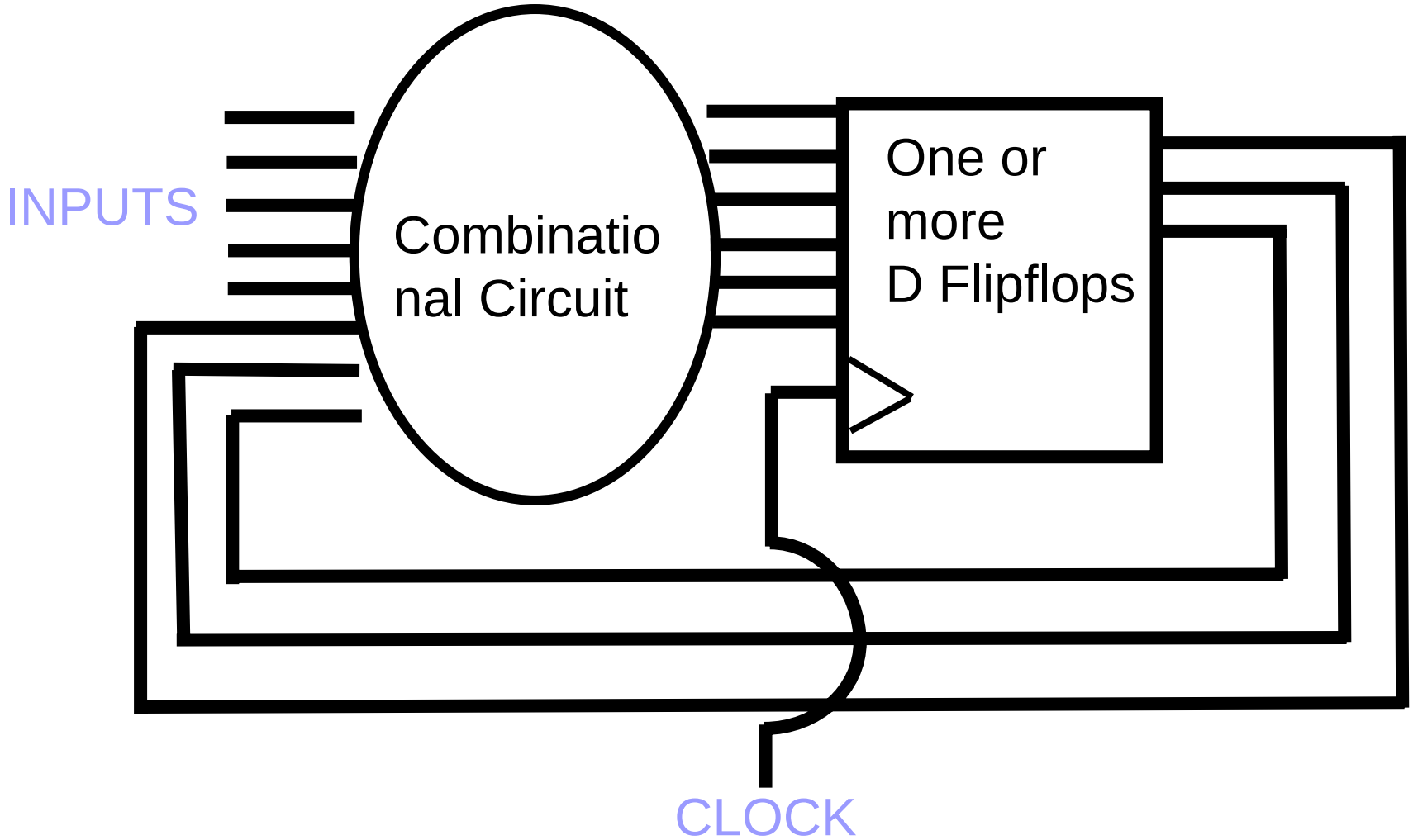
- Often, the system clock!
- “clock” = device that sends out a fluctuating voltage signal that looks like this



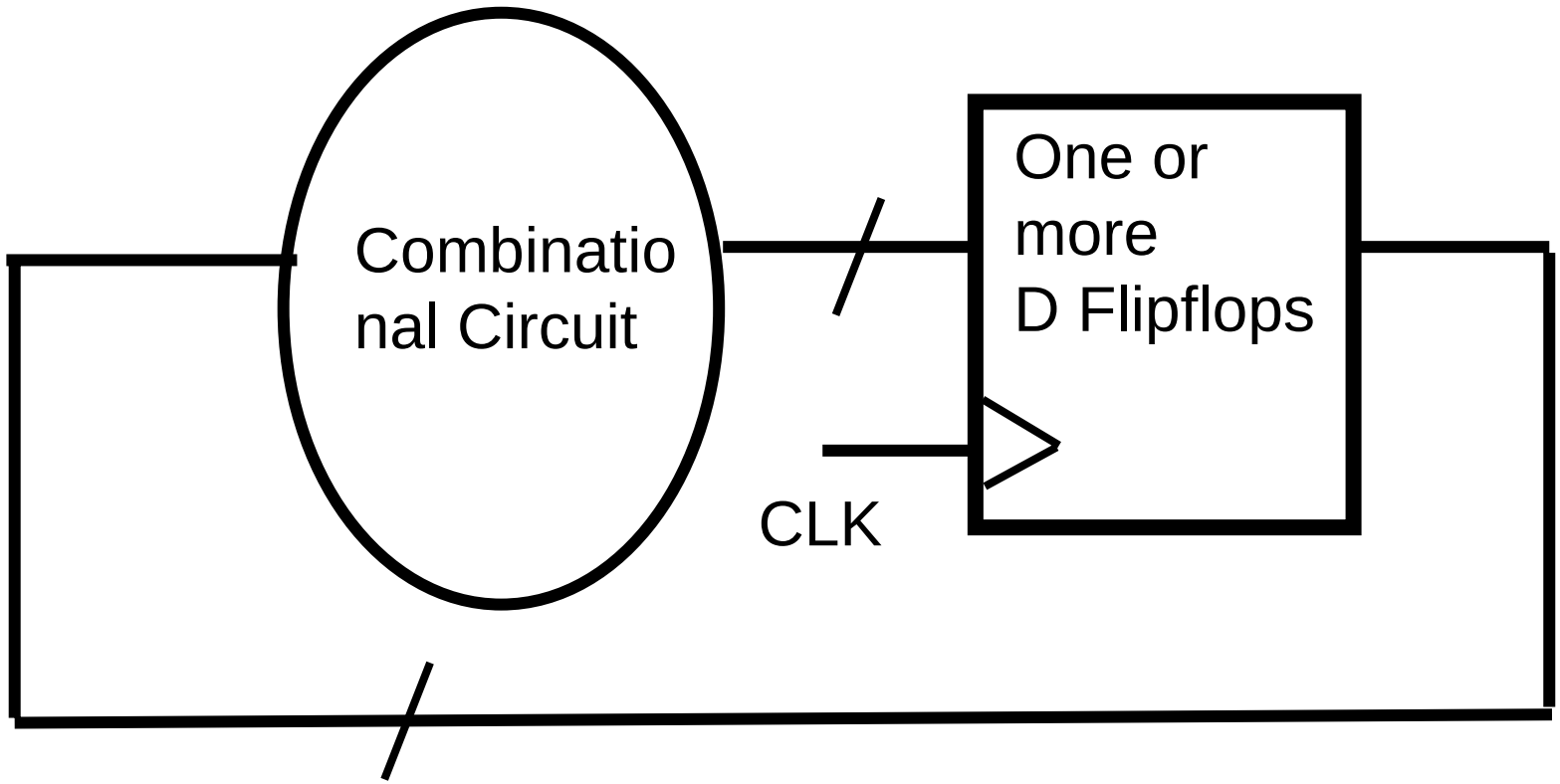
“Computer speed” often refers to the clock frequency (e.g. 2.4GHz)

Synchronous Sequential Circuit

(aka Clocked Sequential Circuit)



Shorthands



Clock Speeds

Heinrich Hertz
1857-94

1974	Intel 8080	2 MHz (Mega = Million)
1981	Original IBM PC	4.77 MHz
1993	Intel Pentium	66 MHz
2005	Pentium 4	3.4 GHz (Giga = Billion)

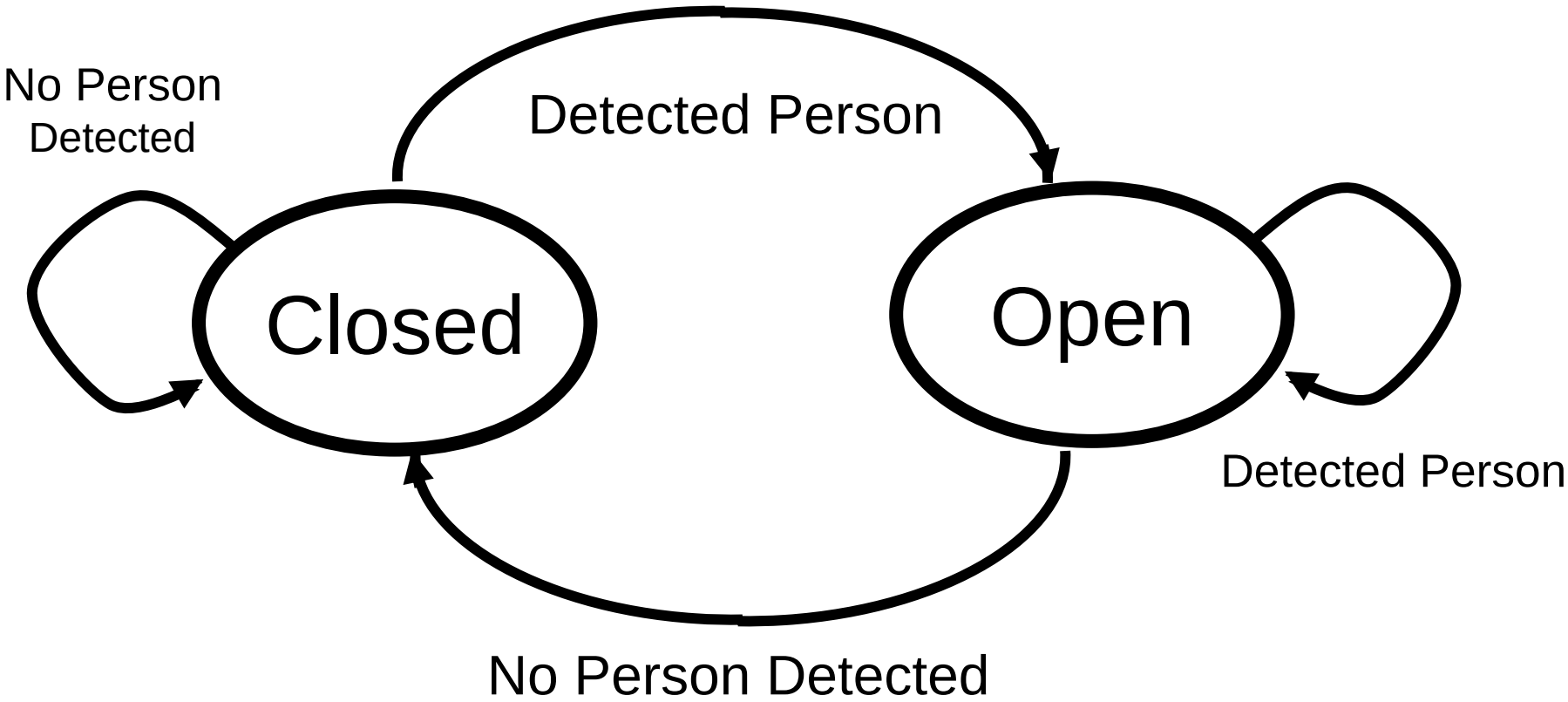
Distance traveled by light during 1 clock cycle of Pentium 4

4 inches

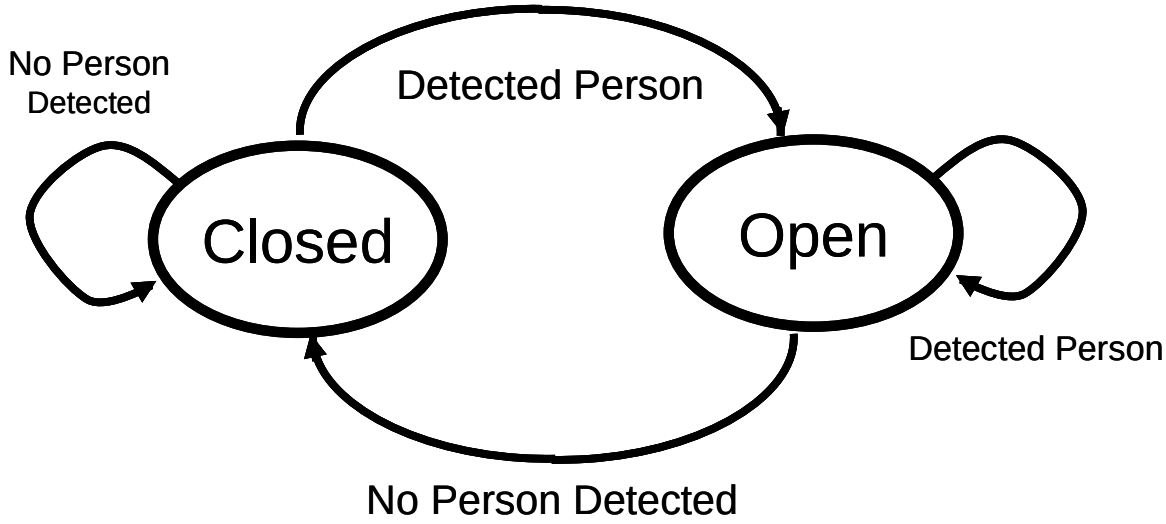
What limits clock speed?

Finite State Machines

State diagram for automatic door



Implementing as synchronous circuit



INPUT

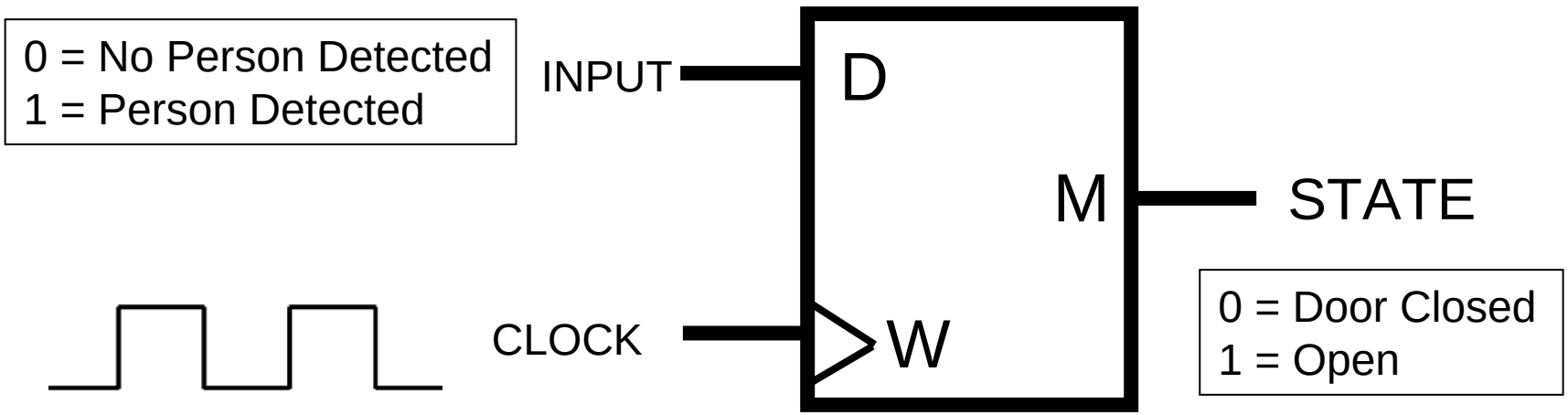
0 = No Person Detected
1 = Person Detected

STATE

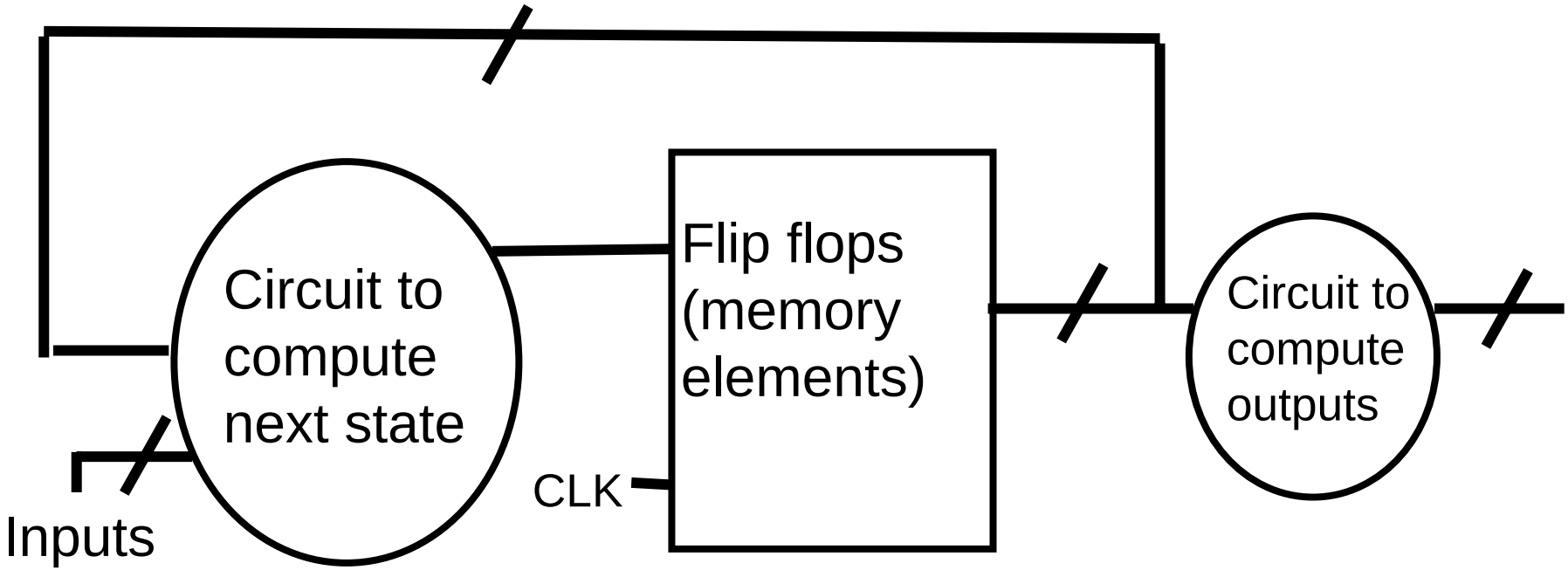
0 = Door Closed
1 = Open

Input	Present State	Next State
0	0	0
1	0	1
0	1	0
1	1	1

Implementation



Moore FSM (see handout)



K Flip flops allow FSM to have 2^K states

Other examples of FSMs

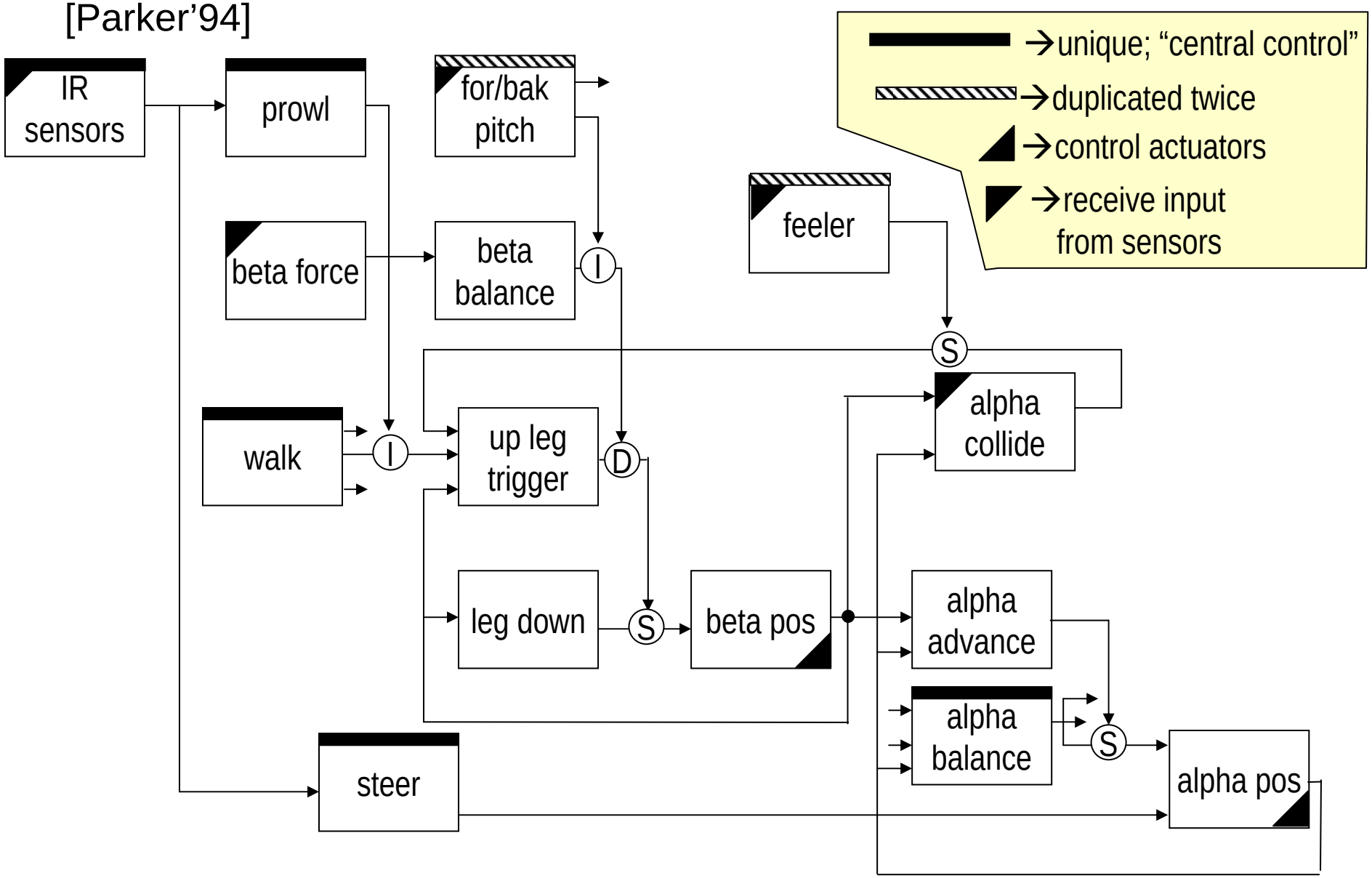
- Sisyphus

- Brook's Genghis (51 FSMs) (see p. 46 in our text)

- Human Soul a la Aquinas (see Handout)

Portion of Genghis AFSM Network

[Parker'94]



Finally...

How computers execute programs.

Scribbler Control Panel Program

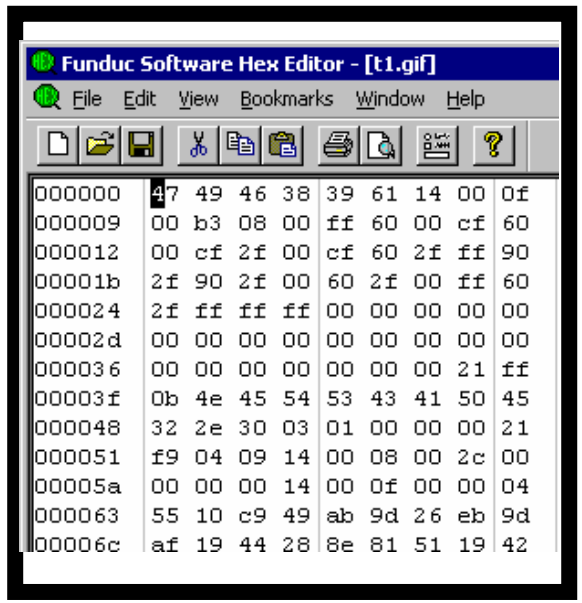
```
If <Obstacle on Either Side> Then
{
  Play Sound for 1s at Frequency 440Hz
}
Else
{
  LED: ON, ON, ON
}
END
```

F5

“Download to Robot”

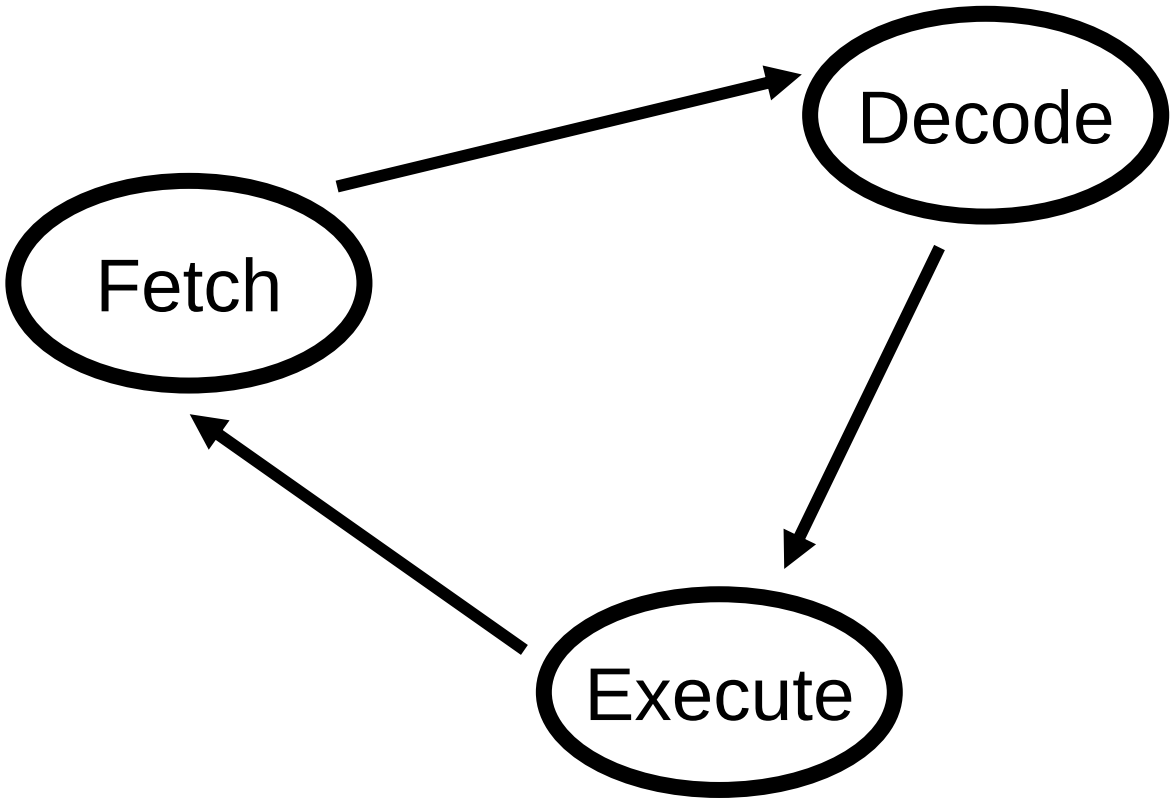
(Compilation)

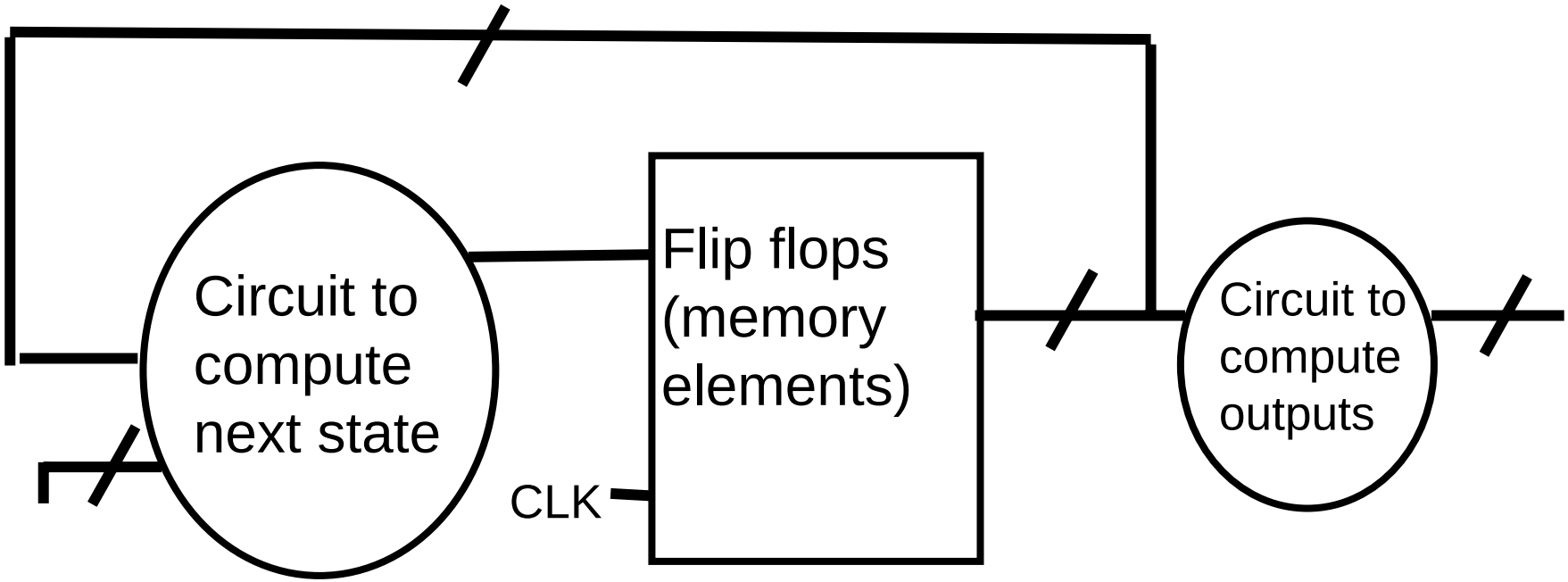
Machine Executable Code



Meet the little green man…

The Fetch – Decode – Execute FSM

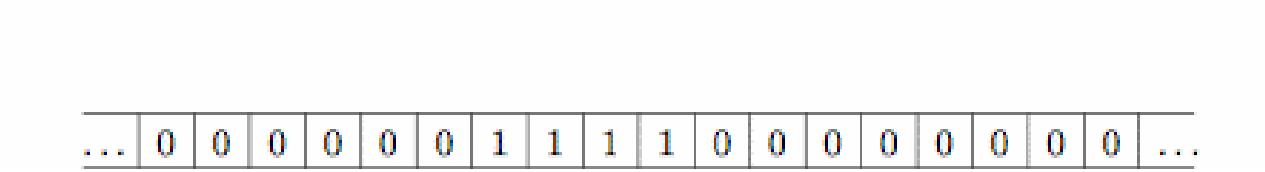




…….....01101000001101111010101111……

Program stored in machine memory; each instruction represented by say 64 bits

Discussion: How would you implement a Turing-Post program with a digital circuit?



1.PRINT 0
2. GO RIGHT
3. GO TO STEP 1 if 1 SCANNED
4. GO TO STEP 1 if 0 SCANNED
5. STOP

Assume “PRINT” and “SCAN” as basic operations