IST 338 in April – *big picture*

I've got *all* my
eyes on this view!

The view
from here…

Final Projects

Picobot
vPool
TextID

CS Practice

IST 338 in April – *big picture*

I've got *all* my eyes on this view!

The view from here…

is over here:

CS in practice

Final Projects

- Picobot
- vPool
- TextID

CS Applications

- population analysis + algorithms
- graphics / media / games …
- feature-based modeling + classification
- robots and other fun stuff (today…)

CS 5

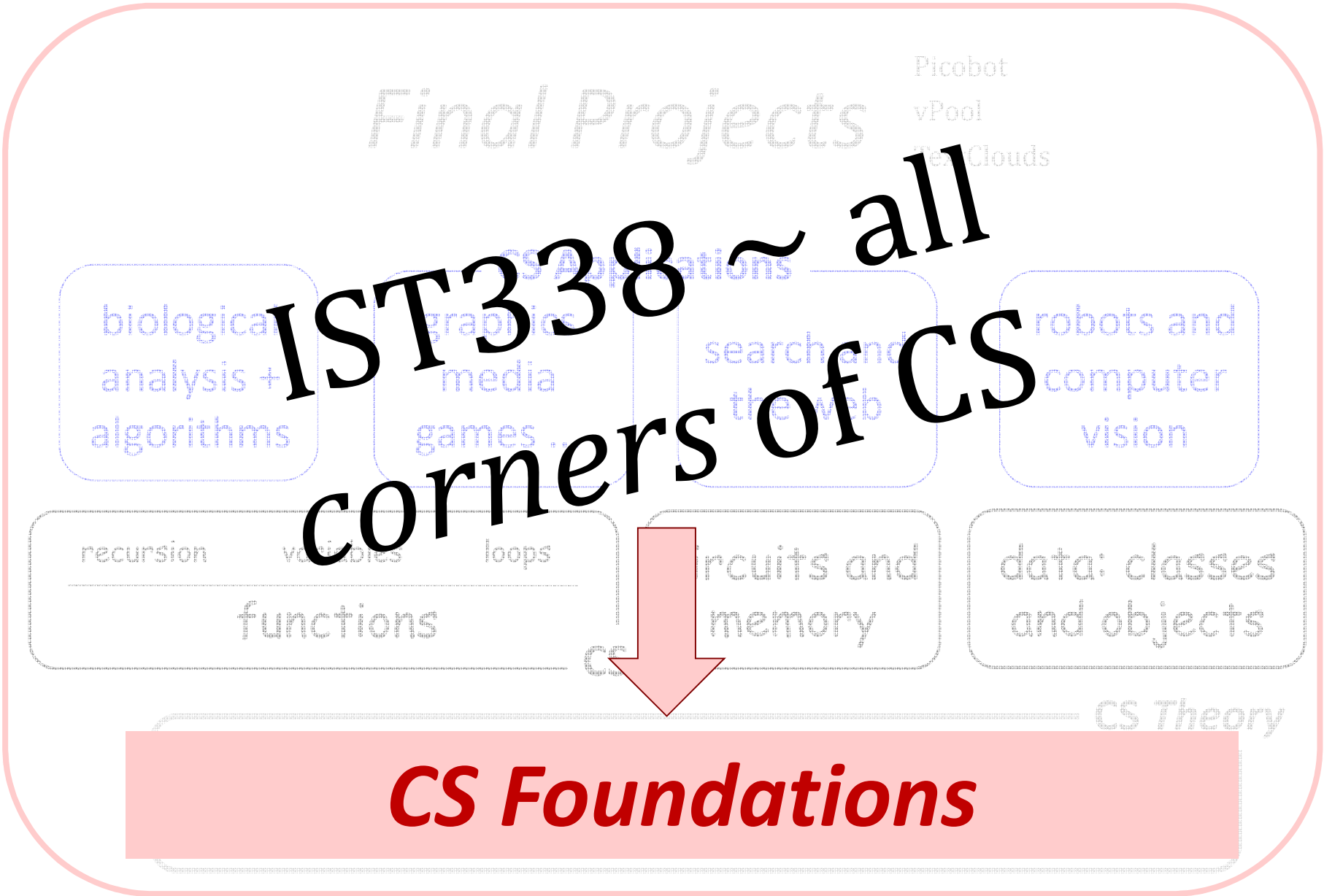
- recursion, variables, loops — functions
- circuits and memory
- data: classes and objects

CS Theory

CS Foundations

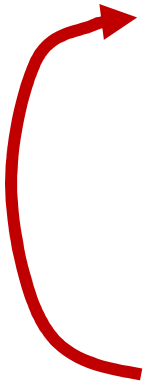
What ***can*** be computed…
… and how efficiently?

CS5 in Dec. ~ Big picture



IST338 ~ all corners of CS

The next two weeks in IST338...



(1) define "computer" precisely

(2) define "compute" precisely

(3) see what computers *provably* can't compute

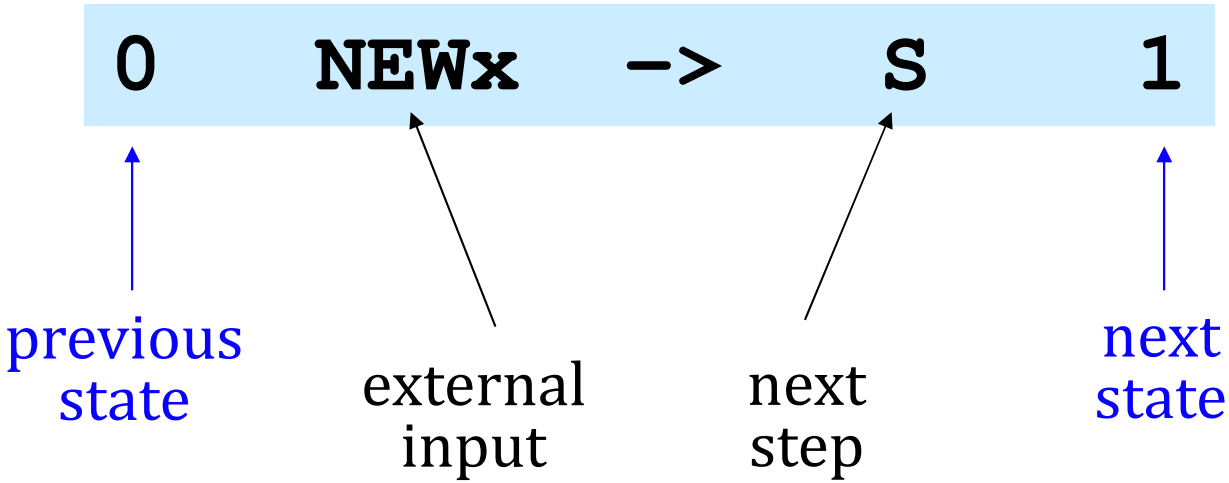
(4) go to step (1) and define things *better...*

(5) ... *or*, time will run out.

Aargh!

CS Foundations

Unifying idea: **State**

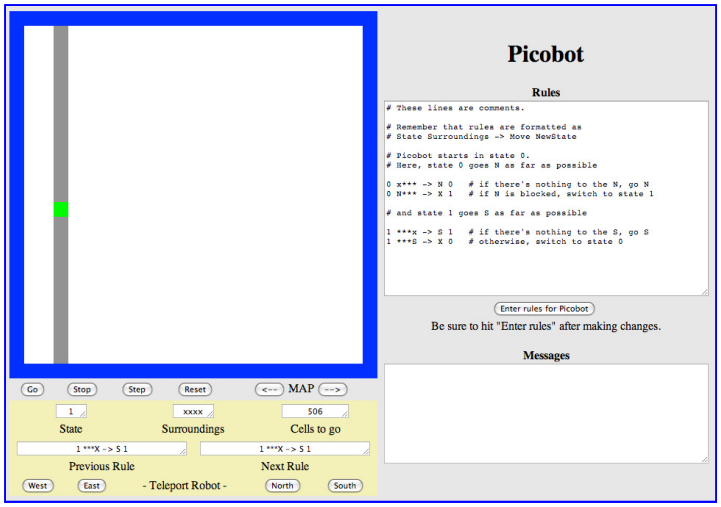


The ***state*** of a computation is

all the internal information
needed to take the next step

and for Picobot, *next step* is taken literally!

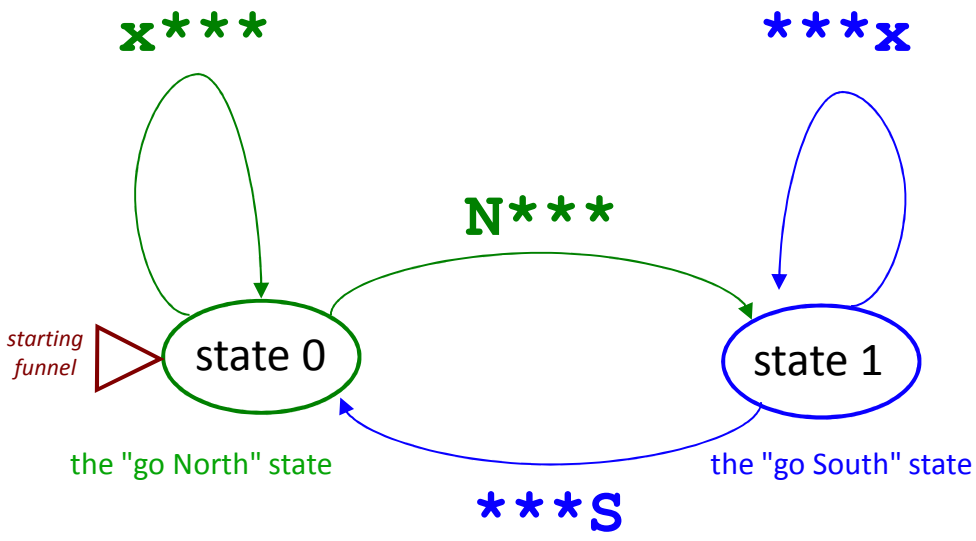
states as ***subtasks***



state pattern -> move new state

```
0 x***  ->  N 0
0 N***  ->  X 1
1 ***x  ->  S 1
1 ***S  ->  X 0
```

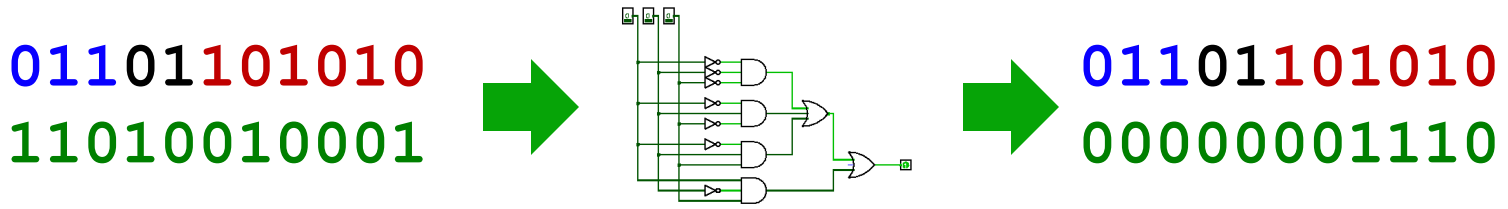
each circle represents a different robot state



transitions move from state to state

Computation is a deliberate sequence of state-changes

this doesn't seem very *meaningful*



bits before

bits after

A *model* of computation: FSM

Finite State Machine

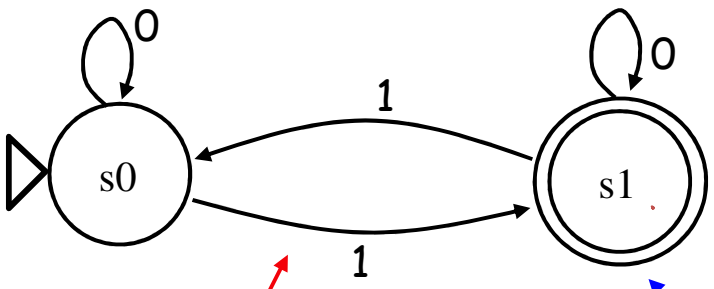
Example input

input sequence

read left-to-right

100101

Example FSM



start state

"input funnel"

transitions

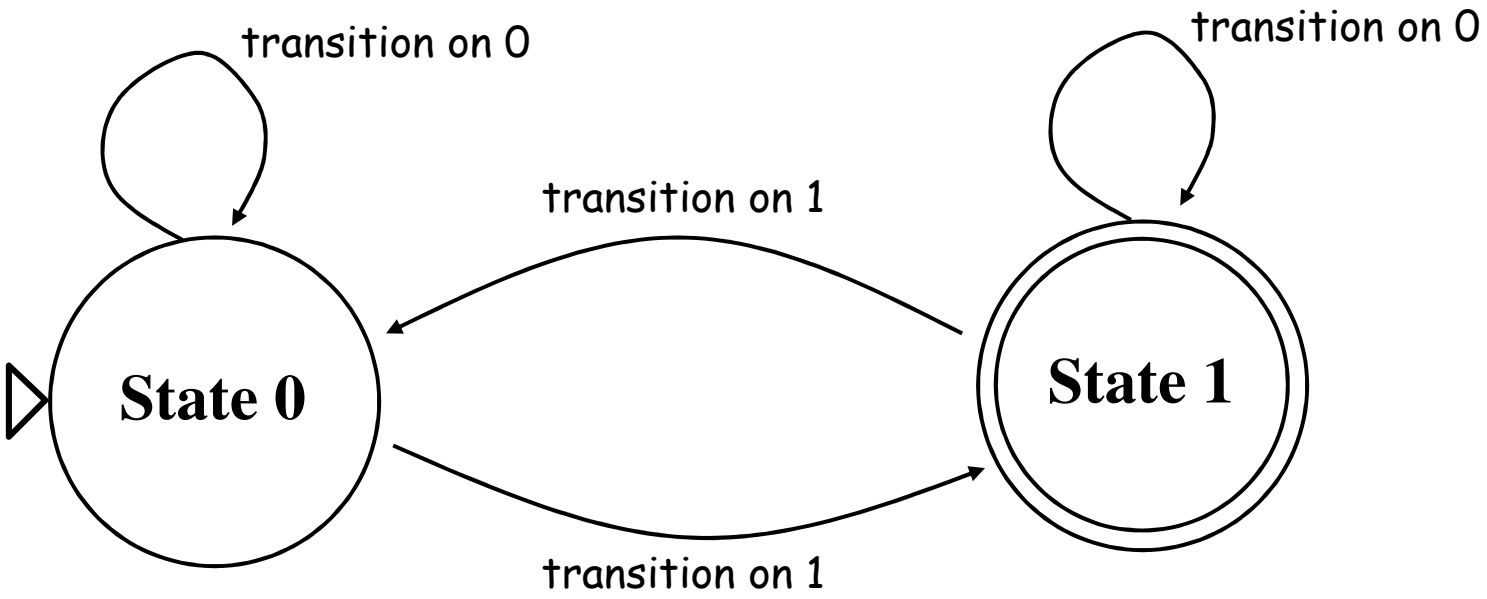
"where to go"

labeled by input !

accepting states

double circled

FSM: *Finite state machine*



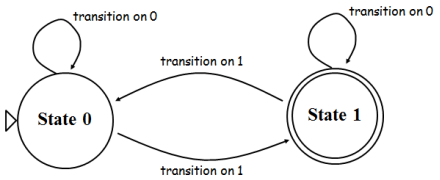
another input sequence
always left-to-right

0010111

What does each state MEAN ?

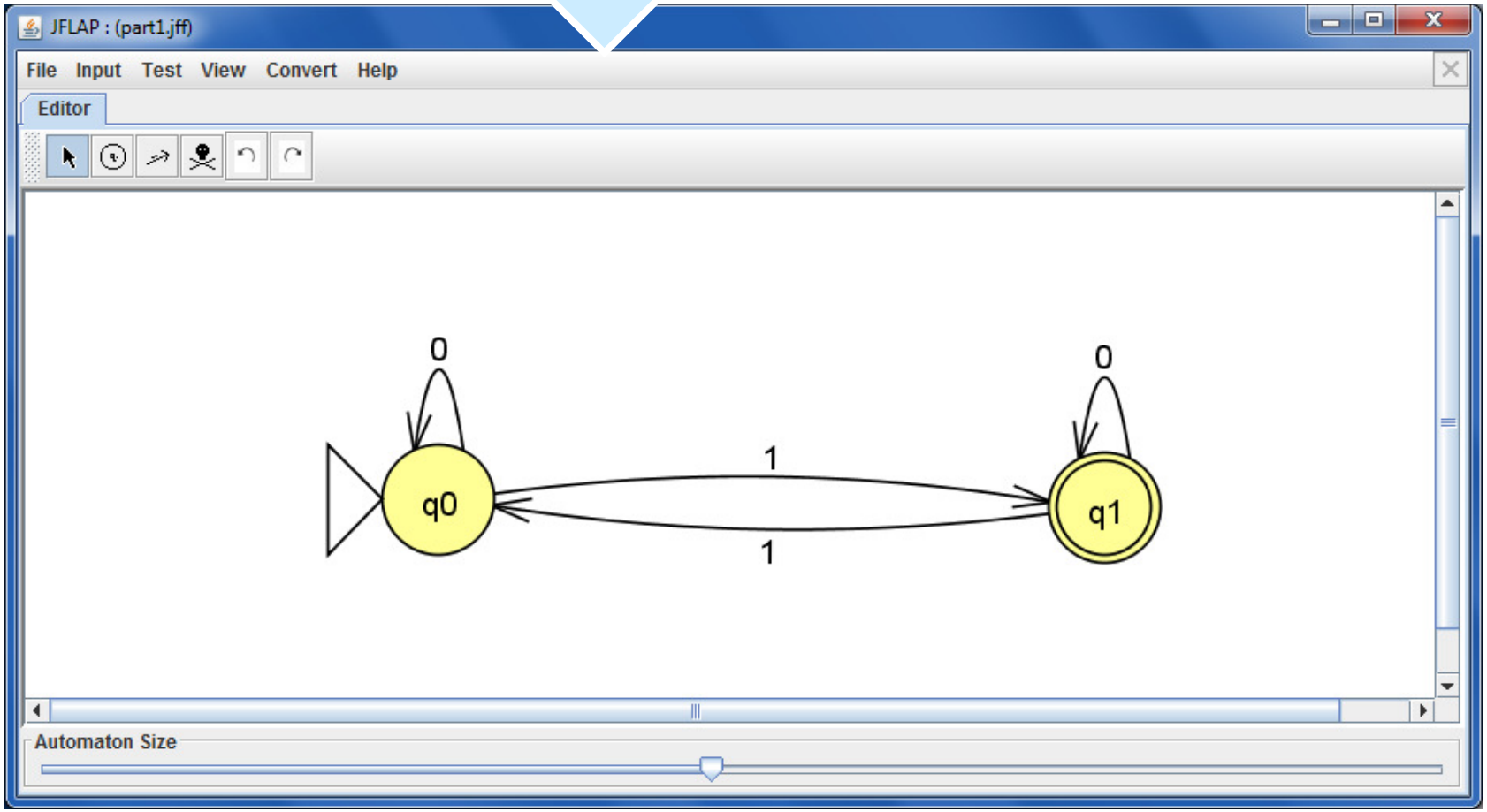
What does this FSM do overall?

FSM: *Finite state machine*

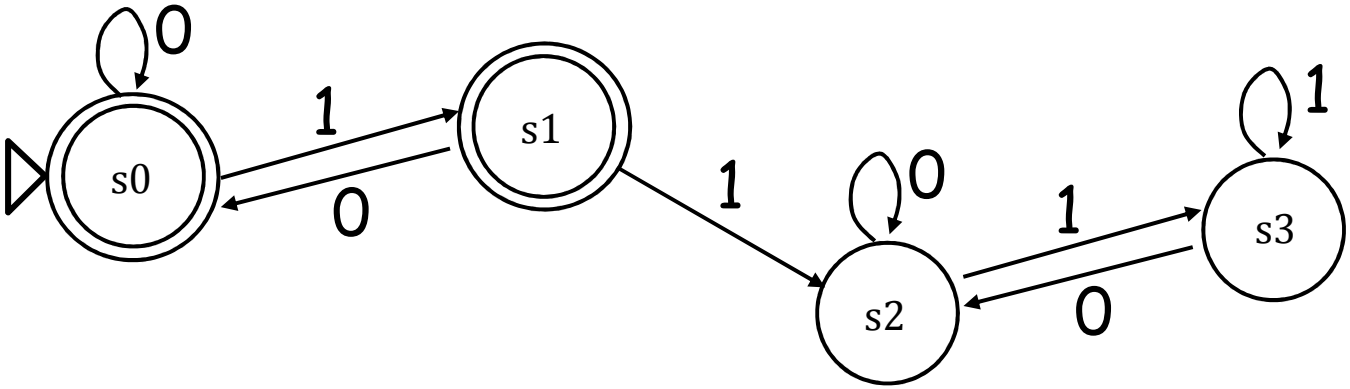


JFLAP !

graphical state-machine builder for hw12



Try it!



List three different inputs that this FSM ***accepts***:

List three different inputs that this FSM ***rejects***:

In general, what English phrase describes the ***accepted inputs***?

This machine accepts strings that…

What does each state say about the ***current state*** of the input?!?

s0 means…

s1 means…

s2 means…

s3 means…

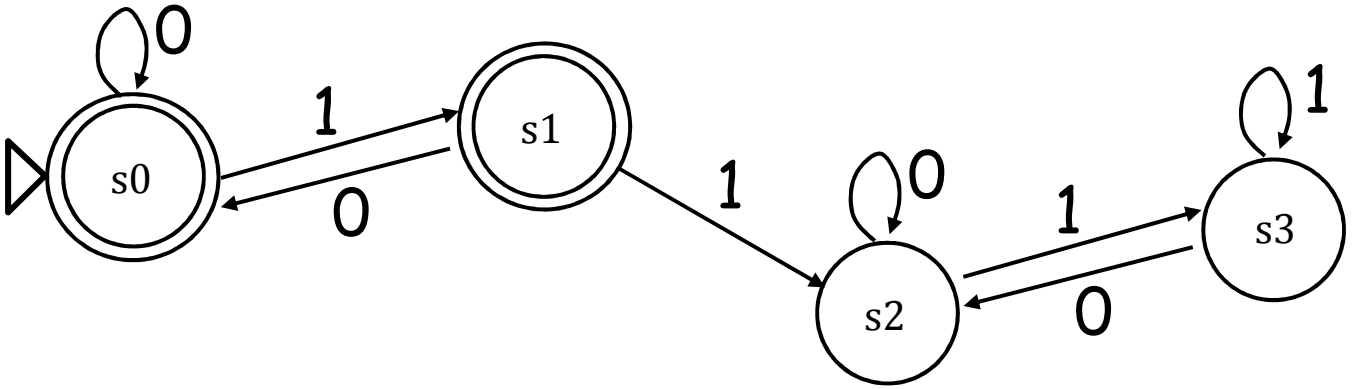
Extra!

Could you get the same behavior with *fewer* states?
What's the ***minimum*** # possible? How do you know?

Hint: find a string that ***has*** to be in each state!

Quiz!

Try this on the back page first…



List three different inputs that this FSM ***accepts***:

List three different inputs that this FSM ***rejects***:

In general, what English phrase describes the ***accepted inputs***?

This machine accepts strings that…

What does each state say about the ***current state*** of the input?!?

s0 means…

s1 means…

s2 means…

s3 means…

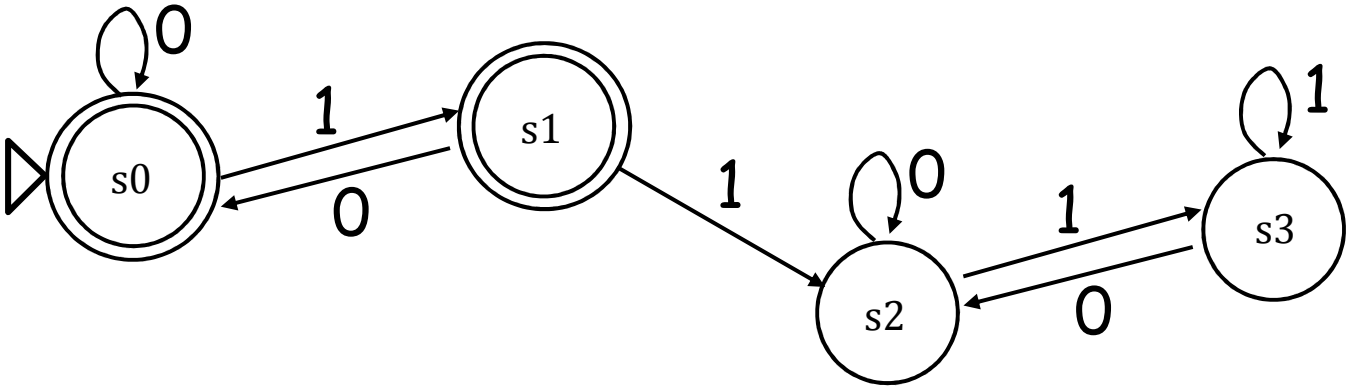
Extra!

Could you get the same behavior with *fewer* states?
What's the ***minimum*** # possible? How do you know?

Hint: find a string that ***has*** to be in each state!

State ~ fate

Strings with *different possible fates* ***must be in <u>different states</u>.***



Can we find three strings – ***all with different possible fates***? (4?)

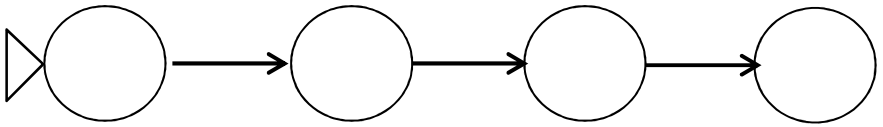
If so, then three states (or 4) are necessary! If not, fewer will be OK.

Build-your-own FSMs

Draw a FSM that accepts strings that ***don't*** contain the pattern **110** anywhere.

Accepted: 1010001 **Rejected:** 10100**110**0

Hint: modify this starter FSM:



Requires at least 4 states

Draw a FSM accepting strings in which the number of zeros (**0**s) is a multiple of 3, so there are 0, 3, 6, … zeros. **1**s don't matter.

Accepted: 110101110, 11, 0000010
Rejected: 101, 0000, 111011101111

Hint: modify *the first FSM we considered*!

Minimum possible # of states?

Draw a FSM accepting strings in which the third digit (from the left) is a **1**.

Accepted: 10**1**0001 **Rejected:** 11**0**00100 and 11

Minimum possible # of states?

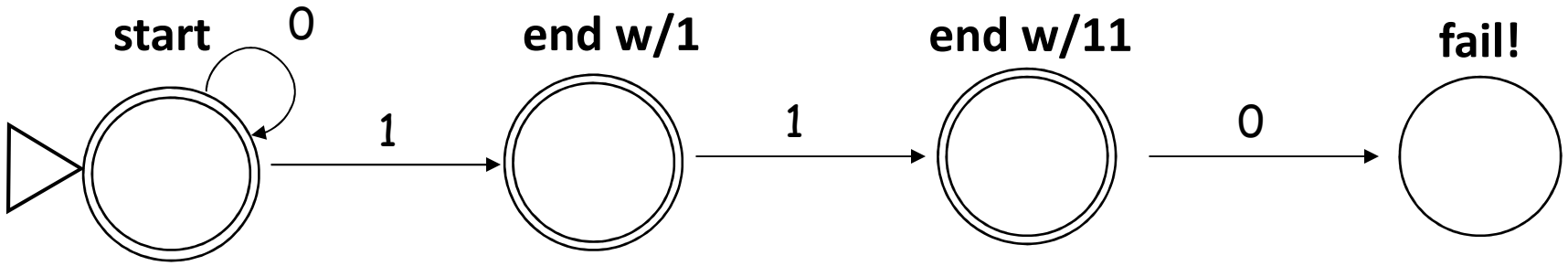
Draw a FSM accepting strings whose third-to-last digit (from the right) is a **1**.

Accepted: 0**1**00 and 01**1**01 **Rejected:** 101**0**01 and 11

Minimum possible # of states?

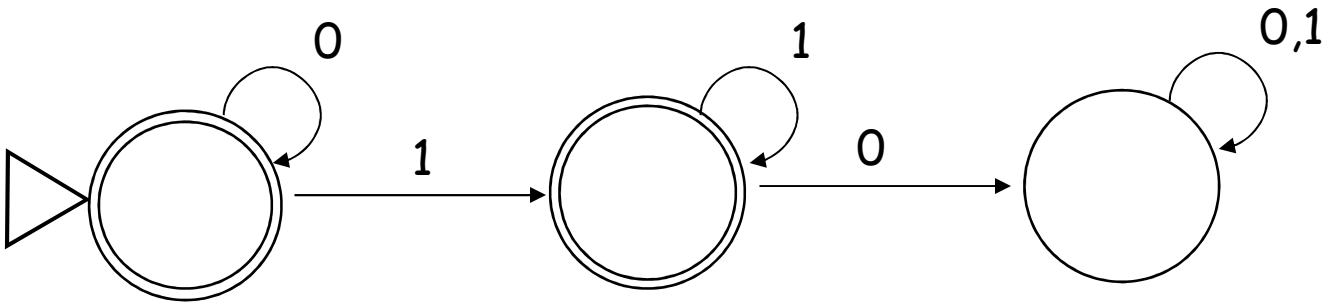
No occurrences of `110`

Draw a FSM accepting strings that do
NOT anywhere contain the pattern `110`



Minimum number of states?

No occurrences of `110`?



Why ***doesn't*** this machine work?

Could we prove 4 states are ***required***?

Number of **0**s is div. by 3

Draw a FSM accepting strings in which the number of zeros (**0**s) is a multiple of 3, so there are 0, 3, 6, … zeros. **1**s don't matter.

Accepted: 110101110, 11, 0000010
Rejected: 101, 0000, 111011101111

Minimum number of states?

Third character is a **1**

Draw a FSM accepting strings in which the third digit (from the left) is a **1**.

Accepted: 10**1**0001
Rejected: 11**0**00100 and 11

Minimum number of states?

Third-to-last character is a **1**?

Draw a FSM accepting strings whose third-to-last digit (from the right) is a **1**.

Accepted: 0**1**00 and 01**1**01
Rejected: 101**0**01 and 11

Minimum number of states?

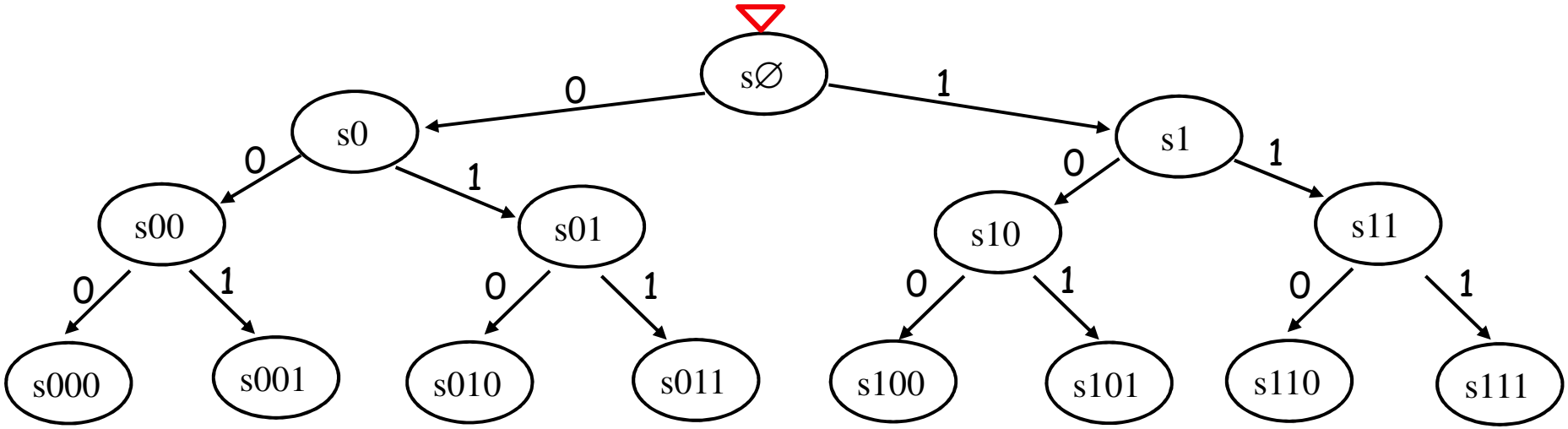
Third-to-last character is a **1**?

Draw a FSM accepting strings whose third-to-last digit (from the right) is a **1**.

Accepted: 0**1**00 and 01**1**01
Rejected: 101**0**01 and 11

Minimum number of states?

Third-to-last character is a **1**

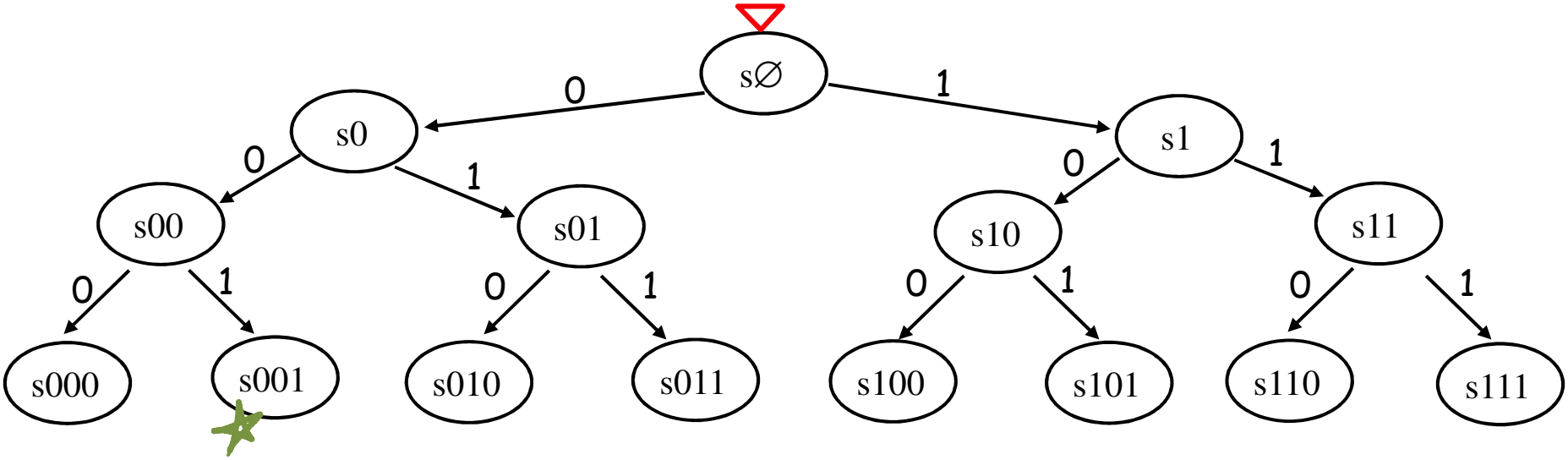


Something seems missing here…

Proof that we ***need*** 15 states?

Third-to-last character is a **1**

Draw a FSM accepting strings whose third-to-last digit (from the right) is a **1**.



Something seems missing…

I don't ***accept*** this solution!

The *minimum possible* number of states?

Third-to-last character is a **1**

8 states?

How could we prove that 8 states are <u>required</u>?

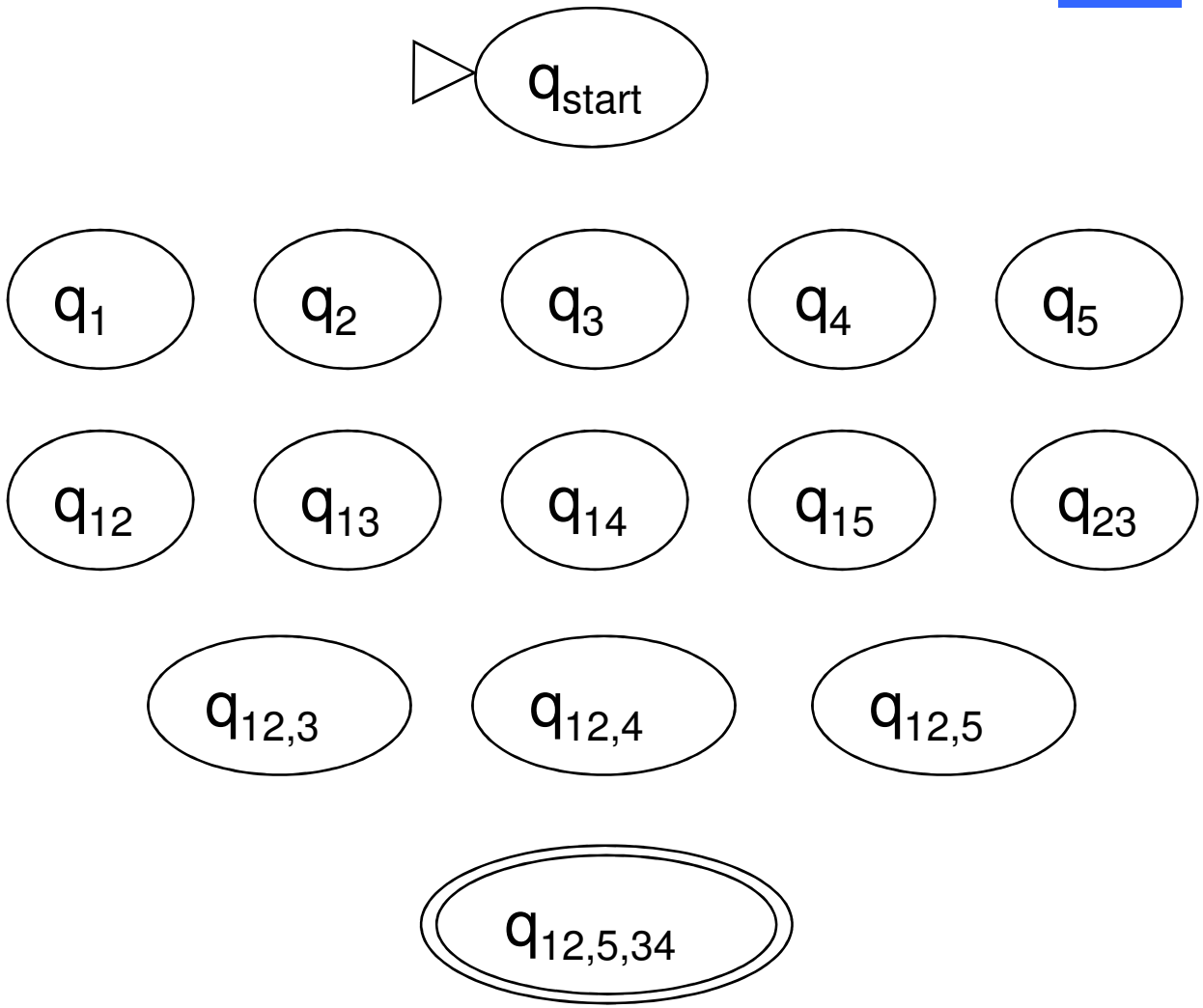
FSMs are everywhere!

12

5

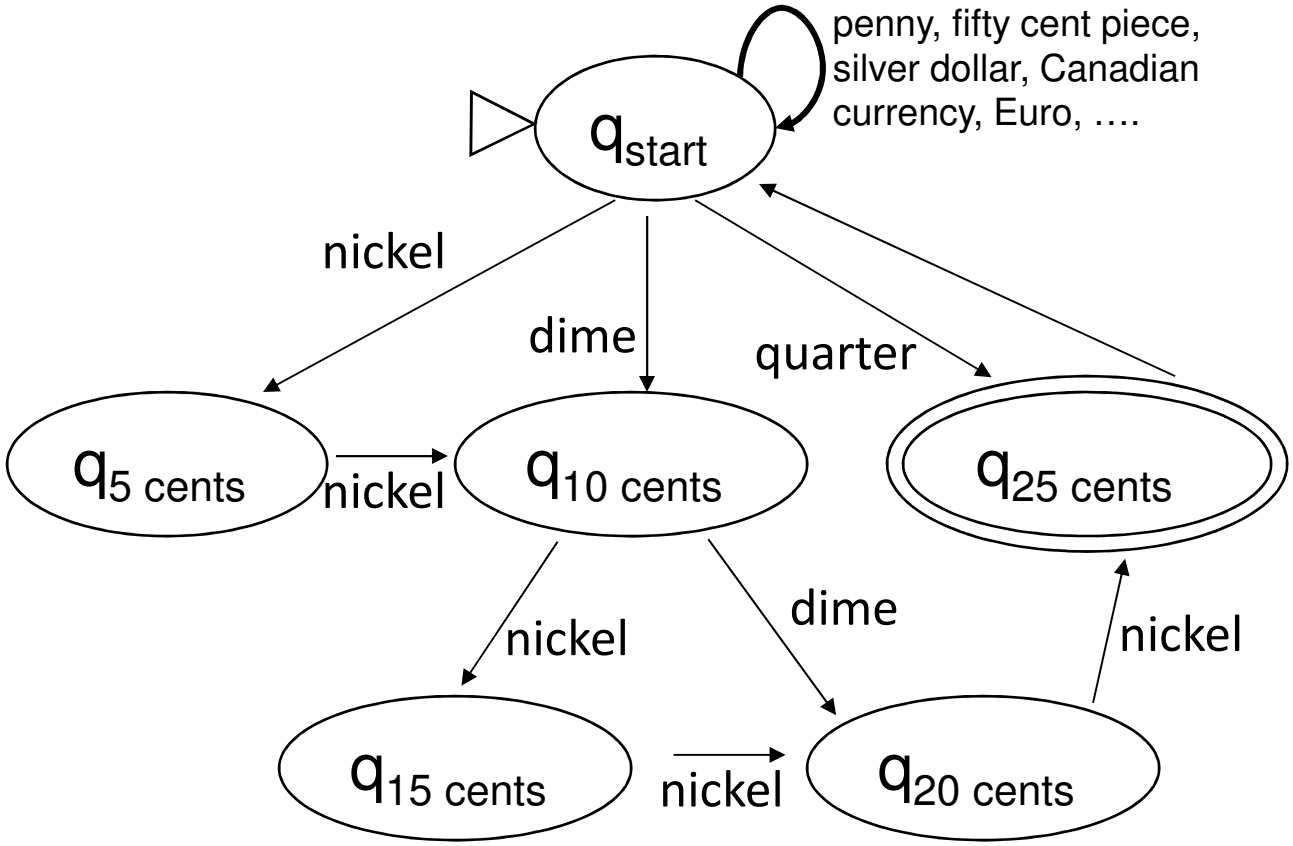
34

Locks



FSMs are everywhere!

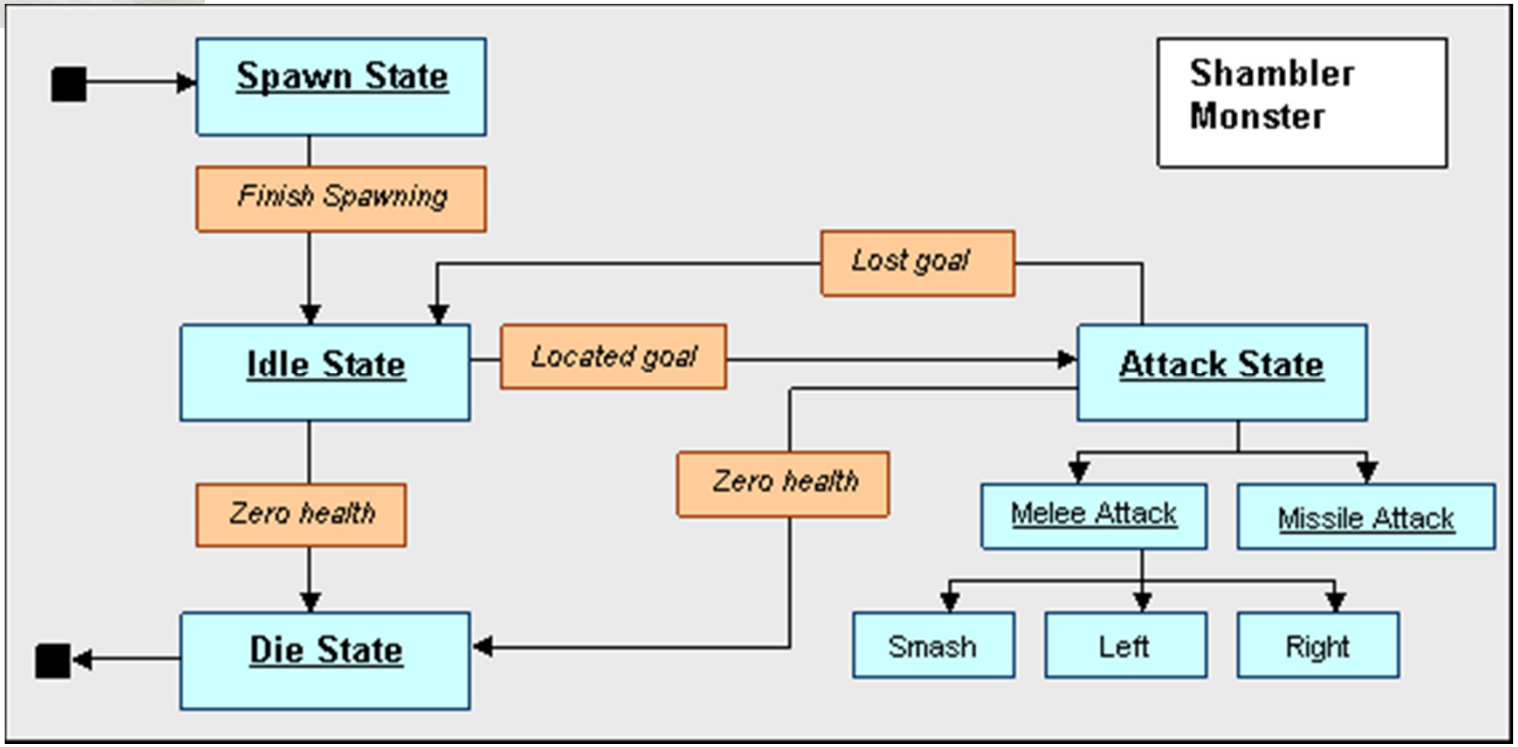
mechanical vending machine



(some transitions not shown)

FSM ~ *Game AI*

The state-machine that controls Quake's ***Shambler*** monsters…



I'm ***Quaking*** in my AstroBoots

Towel-folding states!

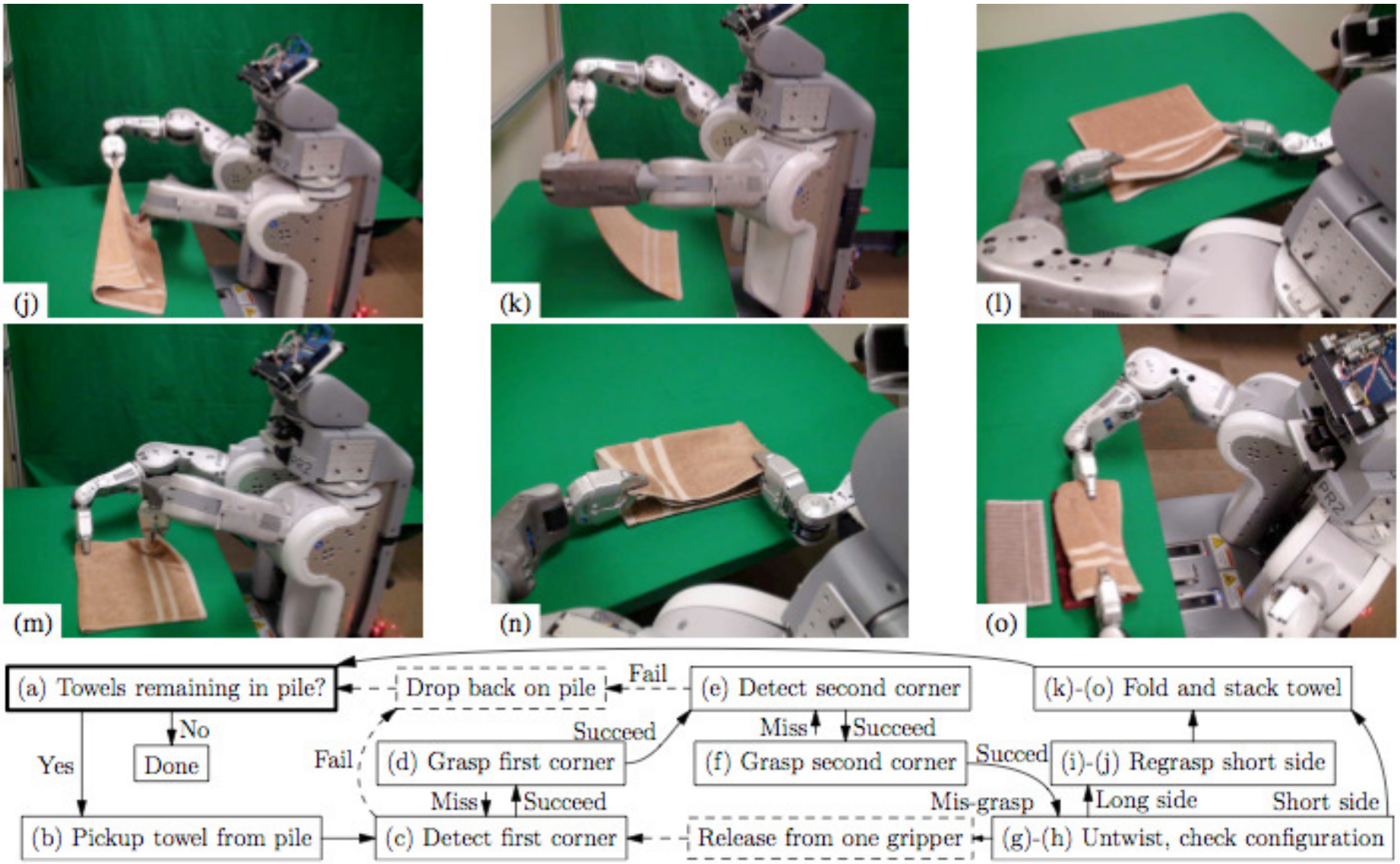


Fig. 2. The state machine model of the procedure: dashed lines indicate failure recovery cases. The images show an actual run.

All robots use FSM control

50x

ieee spectrum INSIDE TECHNOLOGY

MAGAZINE MULTIM

AEROSPACE BIOMEDICAL COMPUTING CONSUMER ELECTRONICS ENERGY

AUTOMATON The future of robots ar

BLOGS // AUTOMATON

U.S. Senator Calls Robot Projects Wasteful. Robots Call Senator Wasteful

POSTED BY: ERICO GUIZZO / TUE, JUNE 14, 2011

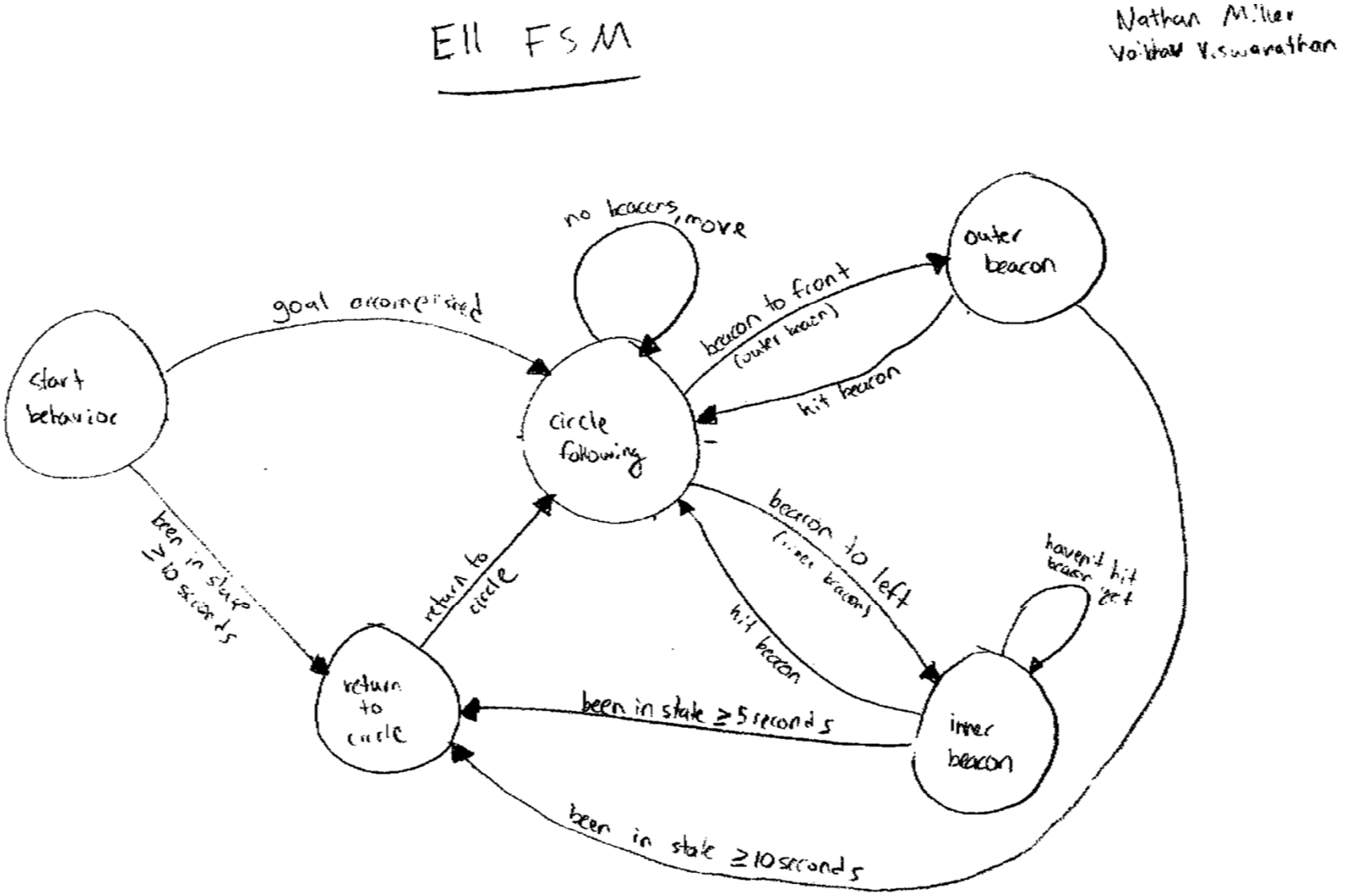
Email Print Share

Tom Coburn, a senator from Oklahoma, and PR2, a robot from California.

Towel-folding?

singled out as a questionable use of dollars...

All robots use FSM control



… send me your FSM so that I can show it off in 2015!

An autonomous vehicle's FSM

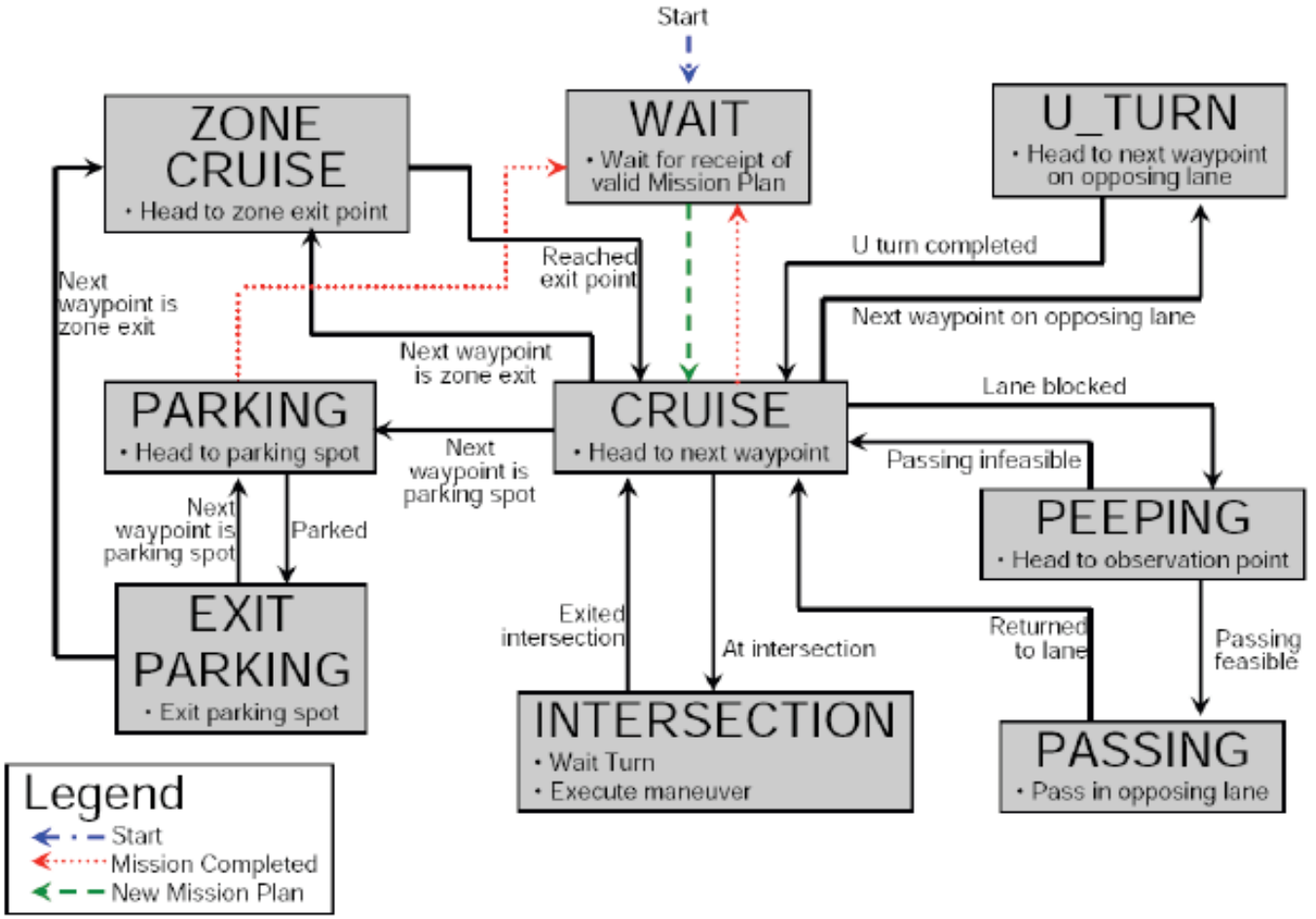


Fig. 9. Situational Interpreter State Transition Diagram. All modes are sub-modes of the system RUN mode (Fig 4(b)).

FSMs driving robots...

MIT's car, Talos

FSMs driving robots...

MIT's car, Talos - ***and its sensor suite***

State-machine *limits*?

Are there limits to what FSMs can do?

they can't necessarily drive safely...

But are there any binary-string problems
that FSMs can't solve?

State-machine *limits*?

accepted

011001

0110

10

λ

this last string is *empty*

rejected

01100

01110

0100

000

Let's build a FSM that accepts bit strings with the **SAME NUMBER of 0s as 1s**

State-machines are limited.

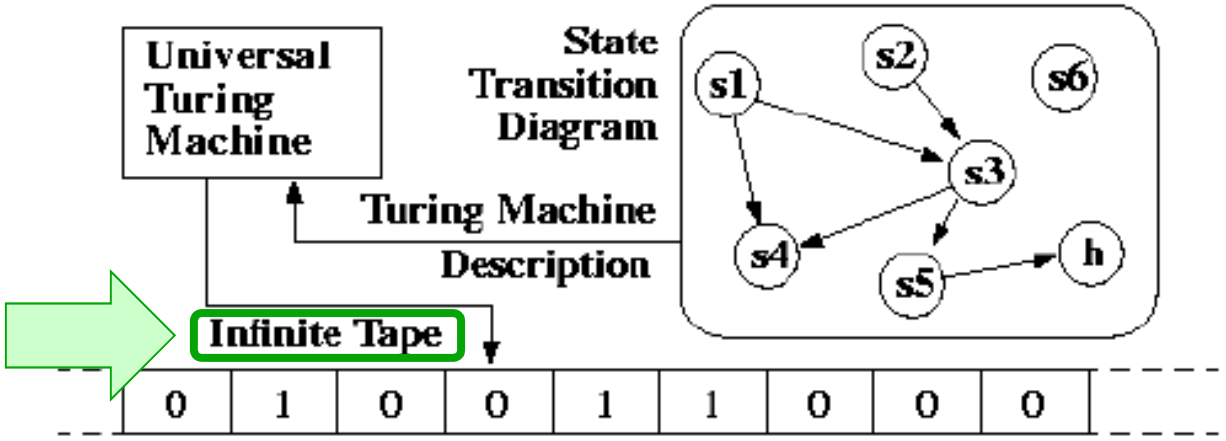
FSMs can't count

at least not arbitrarily high...

We need a **more powerful model** than FSMs...

What do we need to add? 

Next time: *Turing Machines*



*state machines + **memory**!*

Lab! Work on FSMs or final projects…