Moore Machine

Finite-state machines with state-based output

From DFA minimization to finite-state transducers

Lecture focus: definition • representation • design • tracing

Learning outcomes

01 Define
State the formal components of a Moore machine.

02 Represent
Read state diagrams and transition/output tables.

03 Trace
Generate an output sequence for a given input string.

04 Design
Construct a Moore machine from a simple output requirement.

From DFA to Moore machine

DFA	Moore machine
Recognizes a language	Produces an output sequence
Accepting states matter	Every state has an output
Result: accept or reject	Result: symbols from output alphabet

Both use a finite set of states and deterministic transitions.

Formal definition

$$M = (Q, \Sigma, \Gamma, \delta, \lambda, q_0)$$

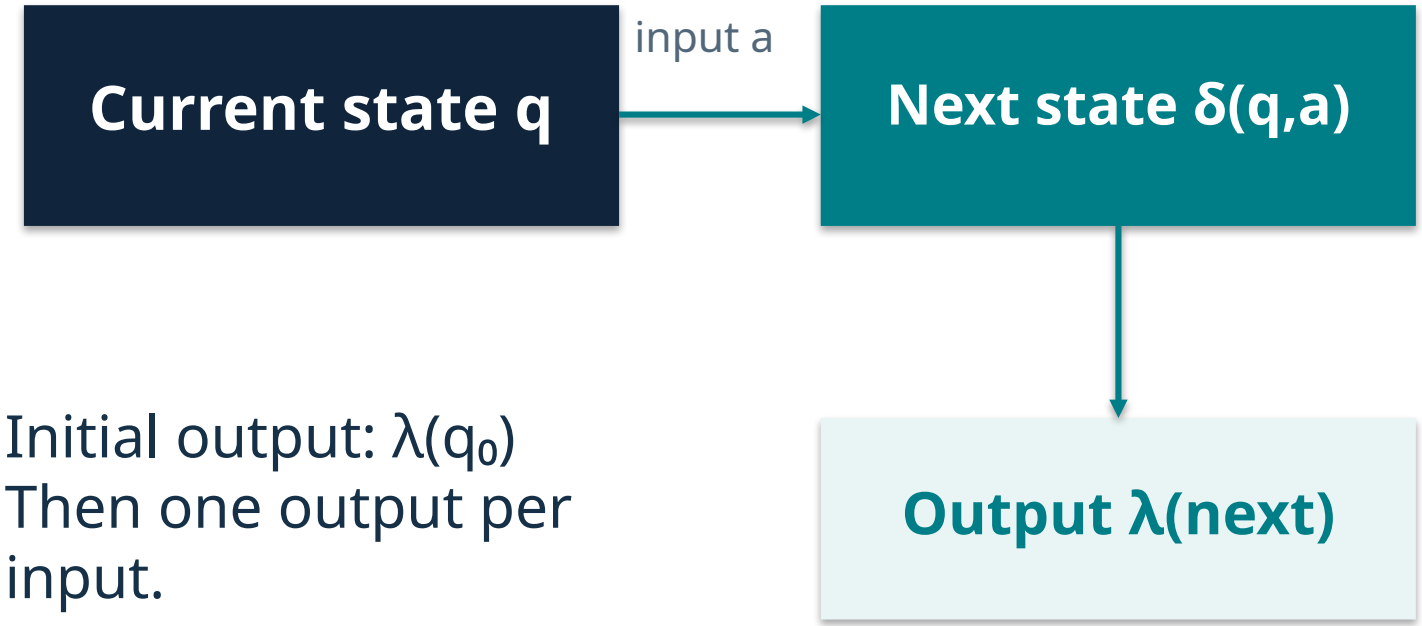
Q, Σ, Γ
Finite state set, input alphabet, output alphabet.

$\delta : Q \times \Sigma \to Q$
Transition function: current state and input determine next state.

$\lambda : Q \to \Gamma$
Output function: only the current state determines output.

$q_0 \in Q$
Initial state; its output is available before any input is read.

How output is generated



Initial output: $\lambda(q_0)$
Then one output per input.

For n input symbols, this convention gives n + 1 output symbols.

Example 1: even parity of 1s

Requirement

Output 1 if the number of 1s read so far is even; otherwise output 0.

State E / 1

Even number of 1s; initial state (zero is even).

State O / 0

Odd number of 1s.

Transition rule

Reading 0 preserves parity; reading 1 toggles parity.

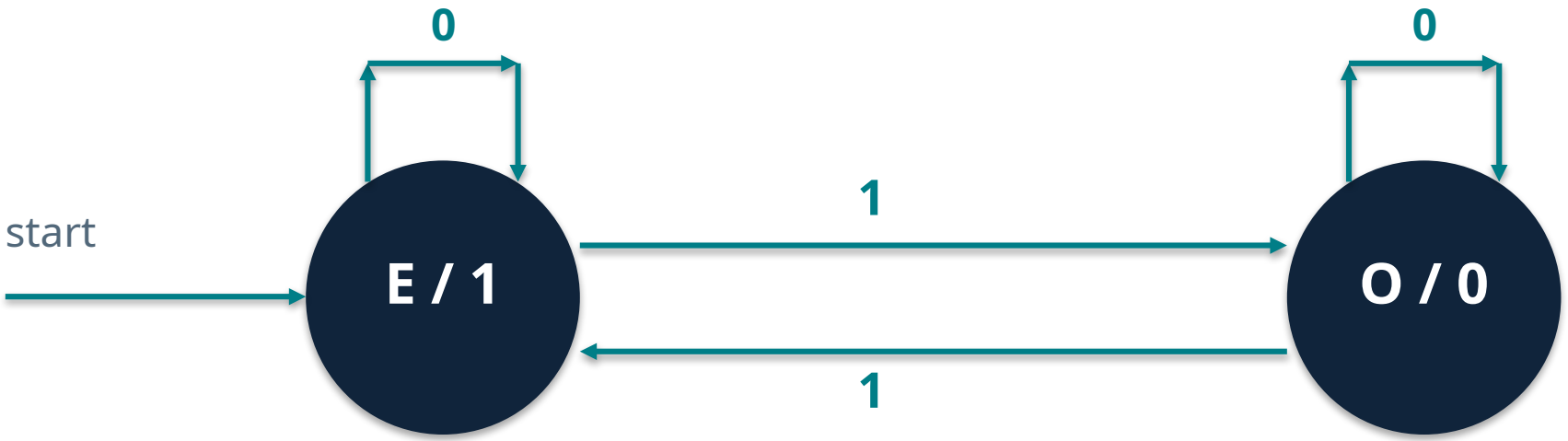
Parity machine: transition table

State	Input 0	Input 1	Output
→ E	E	O	1
O	O	E	0

The output column belongs to the state, not to an input symbol.

Parity machine: state diagram

Notation: state / output. Transition labels contain input symbols only.



Initial state E emits 1 before the first input.

Trace input 10110

Step	0	1	2	3	4	5
Input	—	1	0	1	1	0
State	E	O	O	E	O	O
Output	1	0	0	1	0	0

Output sequence = 100100 (including the initial output).

Example 2: detect suffix 01

Requirement

Output 1 exactly when the input read so far ends in 01.

A / 0

No useful suffix; initial state.

B / 0

Most recent input symbol is 0.

C / 1

The input currently ends in 01.

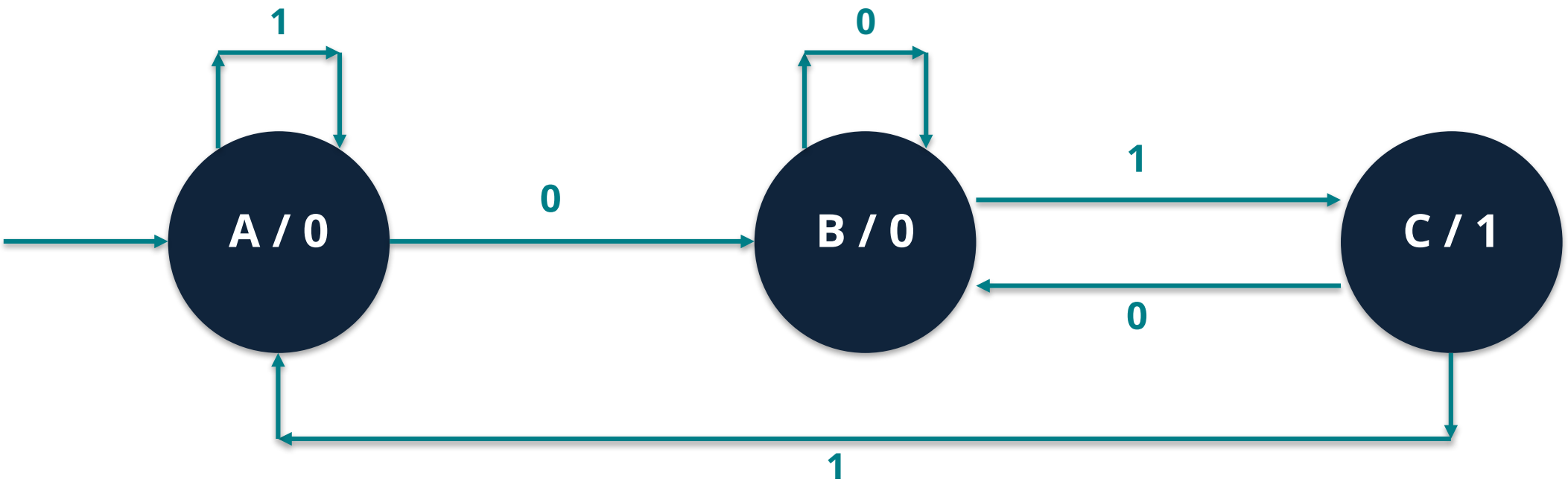
Suffix 01: transition table

State	Input 0	Input 1	Output
→ A	B	A	0
B	B	C	0
C	B	A	1

After reaching C, continue processing; detection does not stop the machine.

Suffix 01: state diagram

Output 1 occurs only in state C.



Trace input 00101

Step	0	1	2	3	4	5
Input	—	0	0	1	0	1
State	A	B	B	C	B	C
Output	0	0	0	1	0	1

Output sequence = 000101; detections at input positions 3 and 5.

Moore vs. Mealy: preview

Property	Moore	Mealy
Output function	$\lambda(q)$	$\lambda(q, a)$
Output labels	States	Transitions
Output length (n inputs)	n + 1*	n
Initial output	Defined by q_0	No transition output yet

* When the initial state output is included, as in this lecture.

Practice questions

Q1 Trace

For the parity machine, find the output for input 11001.

Q2 Explain

Why does a Moore machine produce output for an empty input?

Q3 Design

Build a Moore machine that outputs 1 iff the last input is 1.

Q4 Challenge

Build a Moore machine that outputs 1 iff the input ends in 11.

Solutions and recap

Q1

States E, O, E, E, E, O → output 101110.

Q2

The initial state already has an output $\lambda(q_0)$.

Q3

States Z/0 (start), U/1; from either state, 0 → Z and 1 → U.

Q4

A/0 (start), B/0, C/1; on 0 all go to A; on 1: A→B, B→C, C→C.

Next lecture: Mealy Machine and transition-based outputs.