Moore to Mealy Conversion

Moving outputs from states to transitions

A systematic conversion with worked examples

Lecture focus: rule • algorithm • examples • verification

Learning outcomes

01 Understand
Distinguish state outputs from transition outputs.

02 Convert
Apply the destination-state output rule to every transition.

03 Verify
Compare output traces while accounting for the initial Moore output.

04 Practice
Convert machines using both diagrams and tables.

Recall: Moore vs Mealy

Feature	Moore	Mealy
Output function	$\lambda : Q \rightarrow \Gamma$	$\lambda : Q \times \Sigma \rightarrow \Gamma$
Label	State / output	Input / output
Output length	n + 1	n
Initial output	$\lambda(q_0)$	None

Conversion preserves the outputs after each input, not the separate initial Moore output.

The conversion rule

Keep the states and initial state

Q, Σ, Γ and q_0 remain unchanged.

Keep the transition function

$\delta'(q, a) = \delta(q, a)$.

Move the destination output onto the edge

$\lambda'(q, a) = \lambda(\delta(q, a))$.

Read the rule carefully

Use the output of the destination state, not the source state.

Conversion algorithm

Step 1

Write the Moore transition table with one output per state.

Step 2

For each state q and input a, find destination p = δ(q, a).

Step 3

Look up λ(p); write p / λ(p) in the Mealy table.

Step 4

Draw the same states and edges, with input / output labels.

Example 1: original Moore machine



Even parity of 1s: E is initial. The state output is 1 for even and 0 for odd.

Example 1: original Moore table

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

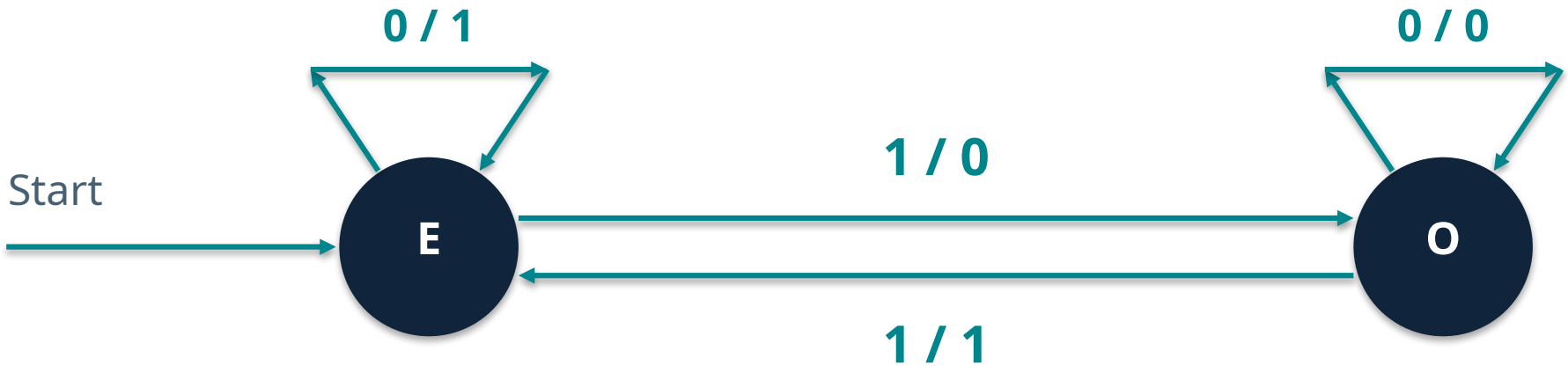
For E on input 1, the destination is O and the destination output is 0.

Example 1: convert each cell

Source	Input	Destination	Dest. output	Mealy cell
E	0	E	1	E / 1
E	1	O	0	O / 0
O	0	O	0	O / 0
O	1	E	1	E / 1

Apply $\lambda'(q, a) = \lambda(\delta(q, a))$ to all four transitions.

Example 1: converted Mealy machine



The state labels no longer contain outputs; each edge carries its destination output.

Example 1: verify input 10110

Step	Input	Destination	Moore output	Mealy output
0	—	E	1	—
1	1	O	0	0
2	0	O	0	0
3	1	E	1	1
4	1	O	0	0
5	0	O	0	0

Moore: 100100; Mealy: 00100. Remove the first Moore output to obtain the Mealy sequence.

Example 2: suffix 01 — Moore table

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

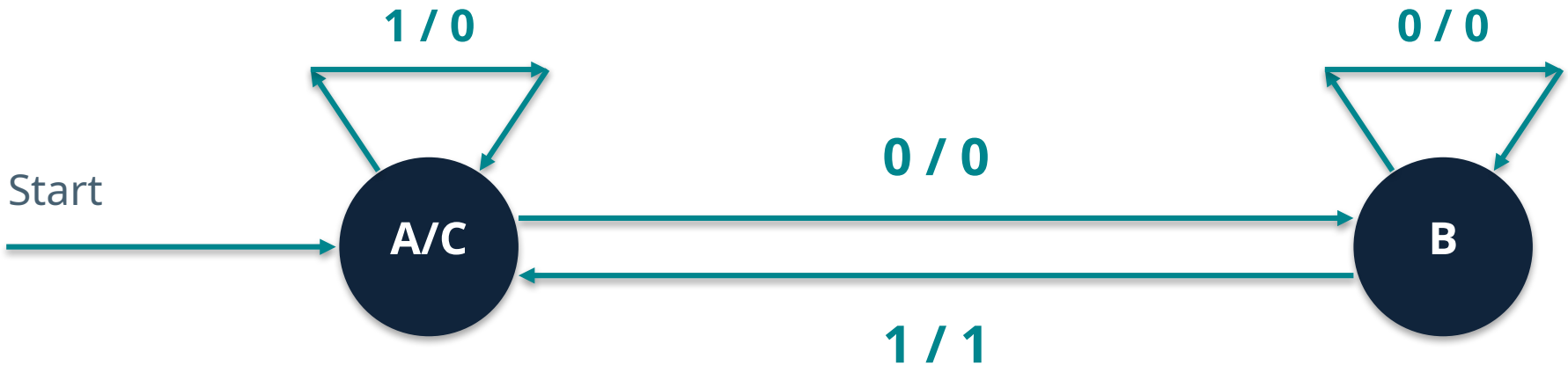
A: no trailing 0 or 01; B: trailing 0; C: trailing 01. Only C outputs 1.

Example 2: converted Mealy table

State	Input 0: next/output	Input 1: next/output
→ A	B / 0	A / 0
B	B / 0	C / 1
C	B / 0	A / 0

The edge B on 1 enters C and emits 1. A and C now have identical outgoing behavior.

Example 2: optional state merging



A and C can be merged because both have identical next states and outputs for every input.

Why the conversion works

Same state sequence

Both machines begin in q_0 and use the same transition function.

Same output after each input

At step i, Mealy emits $\lambda(q_i)$, exactly the Moore output at q_i.

Initial output exception

The Moore machine emits $\lambda(q_0)$ before input; standard Mealy does not.

State count

Direct conversion preserves state count; optional minimization may reduce it.

Practice questions

Q1

Convert a Moore transition P → R on 0, where λ(P)=1 and λ(R)=0.

Q2

Trace both parity machines for input 11001.

Q3

Trace both suffix-01 machines for input 01001.

Q4

Explain why A and C in Example 2 can be merged after conversion.

Solutions and recap

A1 — 0 / 0

The destination R has output 0, so label the edge 0 / 0.

A2 — Moore 101110; Mealy 01110

The initial parity output is 1; the five subsequent outputs match.

A3 — Moore 001001; Mealy 01001

The initial Moore output is 0; suffix 01 is detected twice.

A4 — Equivalent outgoing behavior

A and C give identical outputs and destinations on each input.