CC Lectures 7-8-9

Compiled for: 7th Sem, CE, DDU

Compiled by: Niyati J. Buch

Ref. : Automata Theory and Formal Languages: By Shyamalendu Kandar

Topics Covered

- Moore Machine
  - Count the occurrences of “01”
  - Count the occurrences of “aab”
  - Find 1’s complement of a binary number
  - Count the occurrences of 1101 (non-overlapping)
  - Count the occurrences of 1101 (overlapping)
  - Replace first 1 with 0 for every substring starting with 1

- Mealy Machine
  - Count the occurrences of “01”
  - Find 1’s complement of a binary number
  - Count the occurrences of “aab”
  - Count the occurrences of “01*0”

Topics Covered

- Convert the given Moore machine to Mealy machine
  - Ex 1
- Convert the given Mealy machine to Moore machine
  - Ex 1
  - Ex 2
- Construct Moore machine and Mealy machine
  - Ex 1
  - Ex 2

Mealy Machine and Moore Machine

- A **Mealy machine** is a finite-state machine whose output values are determined both by its current state and the current inputs.
- A **Mealy machine** is a deterministic finite-state transducer: for each state and input, at most one transition is possible.

- A **Moore machine** is a finite-state machine whose current output values are determined only by its current state.
- Like other finite state machines, in **Moore machines**, the input typically influences the next state.
- Thus the input may indirectly influence subsequent outputs, but not the current or immediate output.

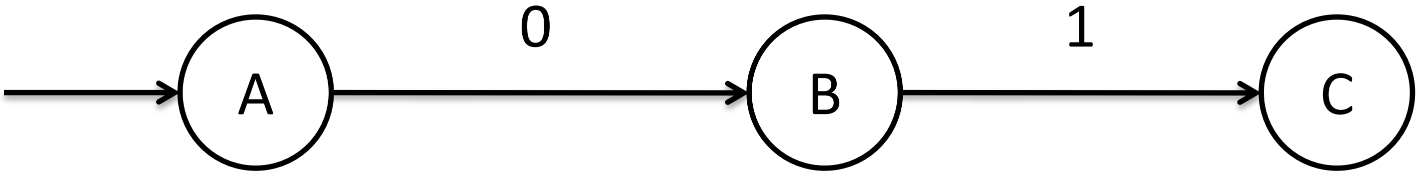
Moore Machine

- Basically a Moore machine is just a FA with two extras.
  1. It has 2alphabets- an input and output alphabet.
  2. It has an output letter associated with each state.

     The machine writes the appropriate output letter as it enters each state.

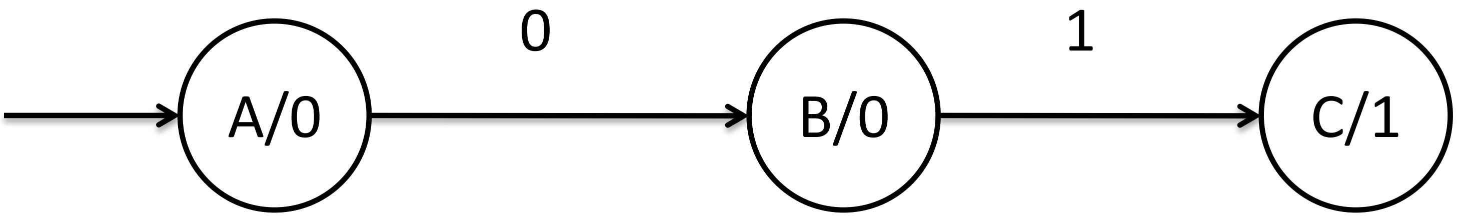
Construct a Moore Machine that counts the occurrences of the sequence “01” in any input strings over {0,1}

Construct a Moore Machine that counts the occurrences of the sequence "01" in any input strings over {0,1}



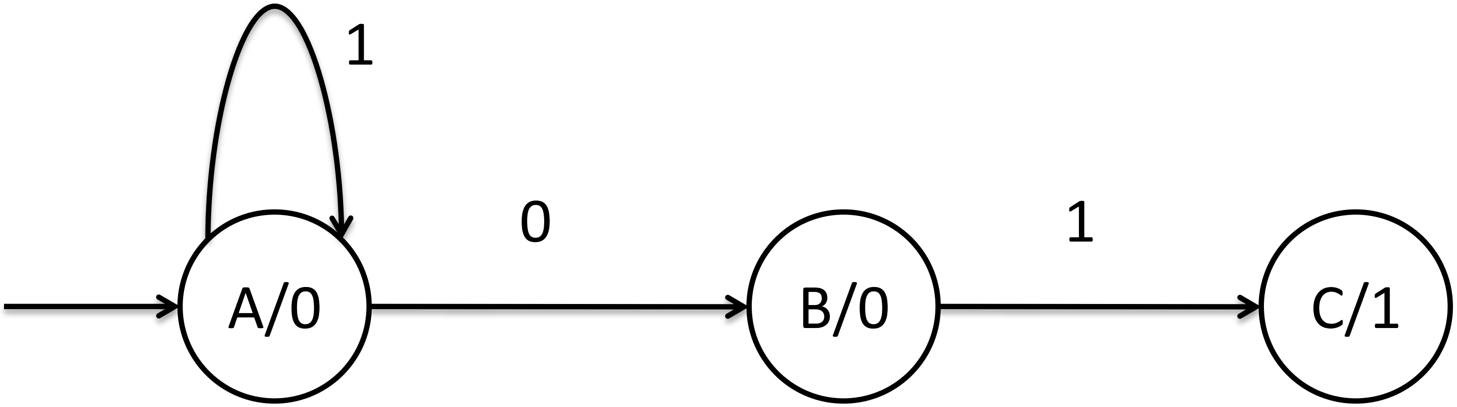
3 states are required for the smallest input 01

Construct a Moore Machine that counts the occurrences of the sequence “01” in any input strings over {0,1}



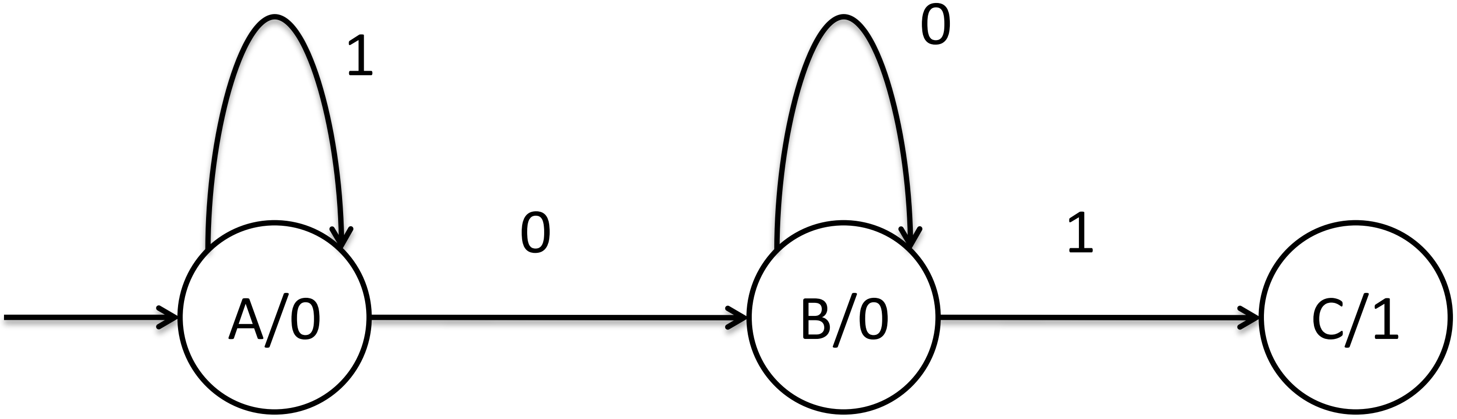
The Moore machine will print 1 as output when 01 is found.
It will print 0 for any other transition.

Construct a Moore Machine that counts the occurrences of the sequence “01” in any input strings over {0,1}



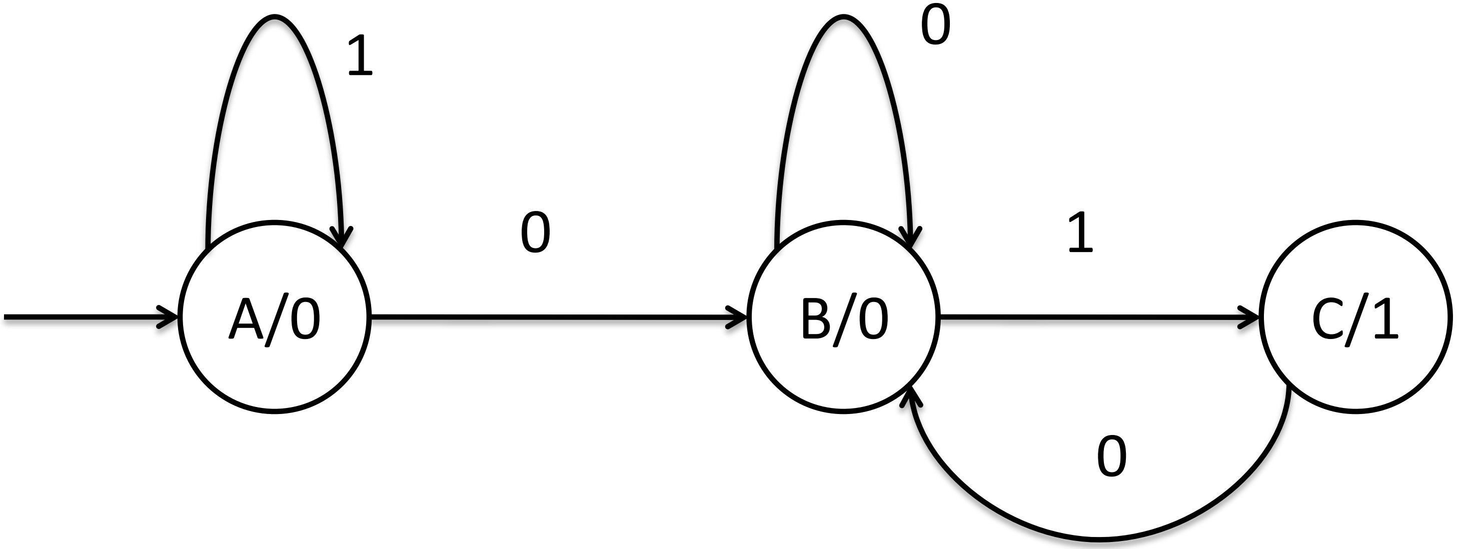
Completing other edges

Construct a Moore Machine that counts the occurrences of the sequence “01” in any input strings over {0,1}



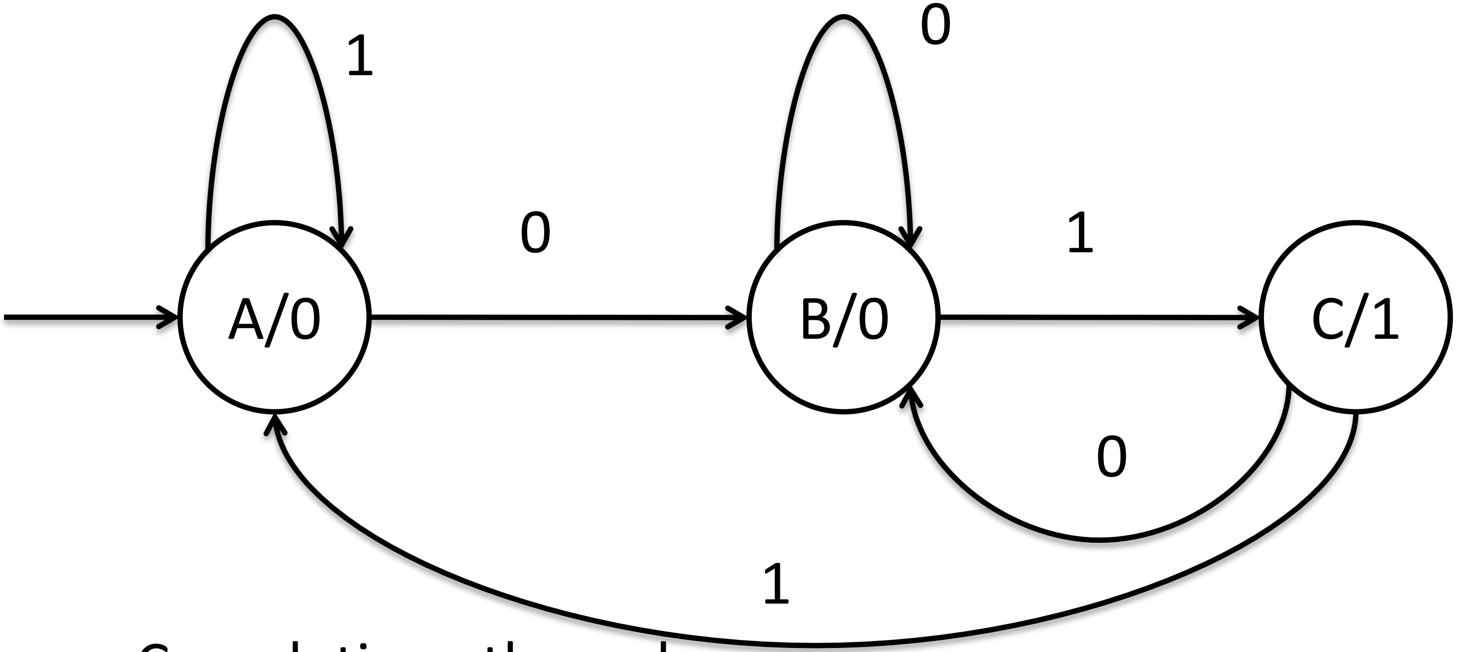
Completing other edges

Construct a Moore Machine that counts the occurrences of the sequence “01” in any input strings over {0,1}



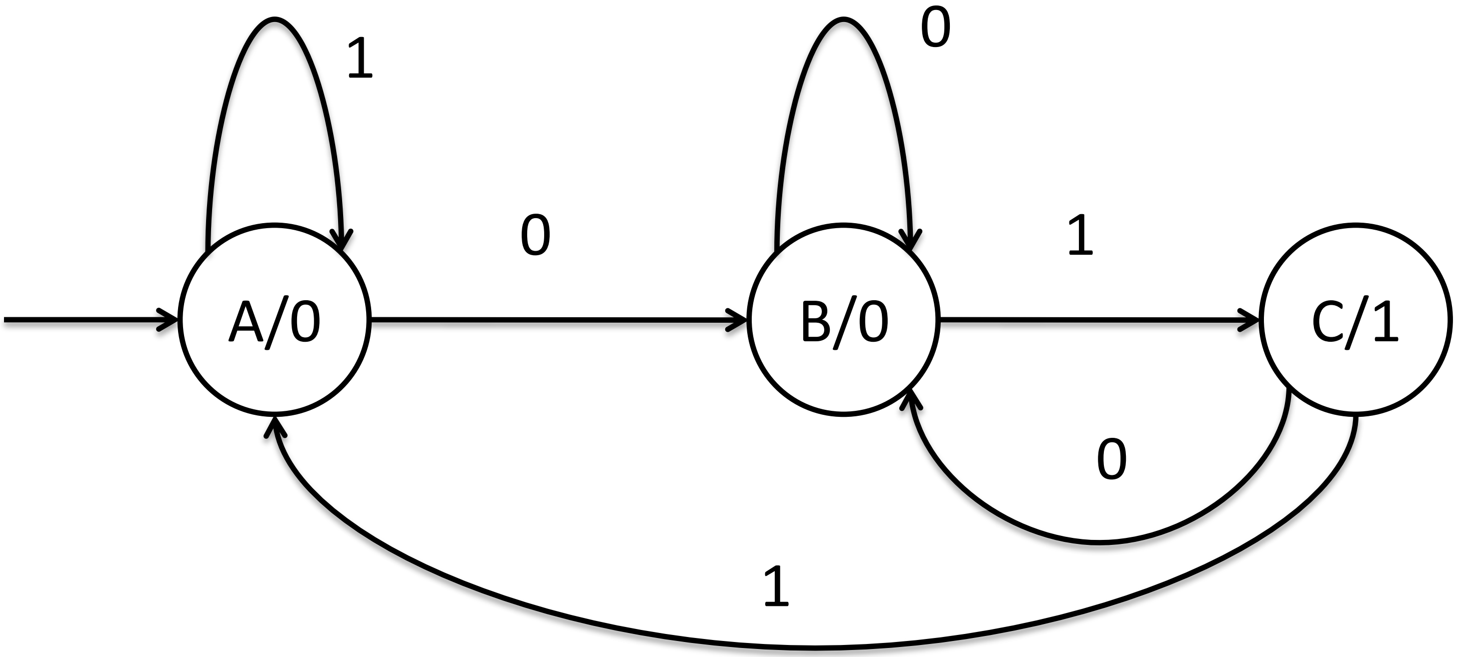
Completing other edges

Construct a Moore Machine that counts the occurrences of the sequence “01” in any input strings over {0,1}



Completing other edges

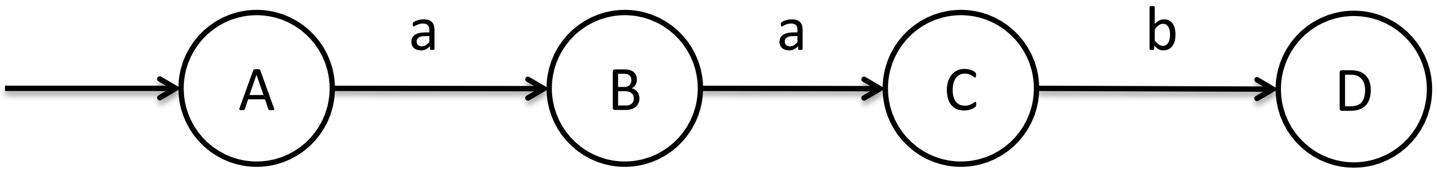
Construct a Moore Machine that counts the occurrences of the sequence “01” in any input strings over {0,1}



INPUT	STATE	OUTPUT
10101	AABCBC	000101
00010	ABBBCB	000010
110011	AAABBCA	0000010

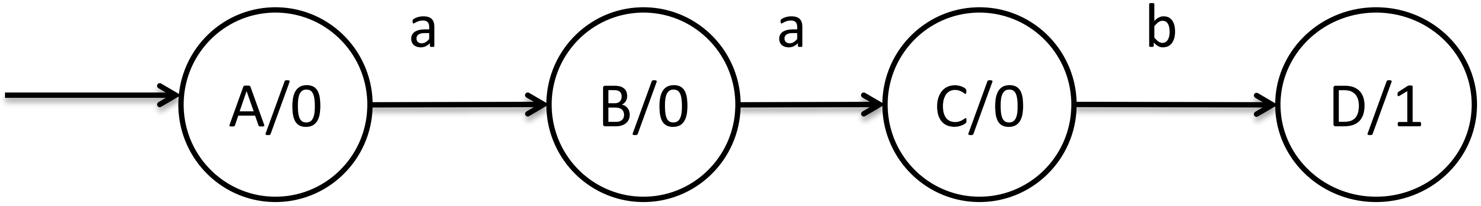
Construct a Moore Machine that counts the occurrences of the sequence “aab” in any input strings over {a, b}

Construct a Moore Machine that counts the occurrences of the sequence “aab” in any input strings over {a, b}



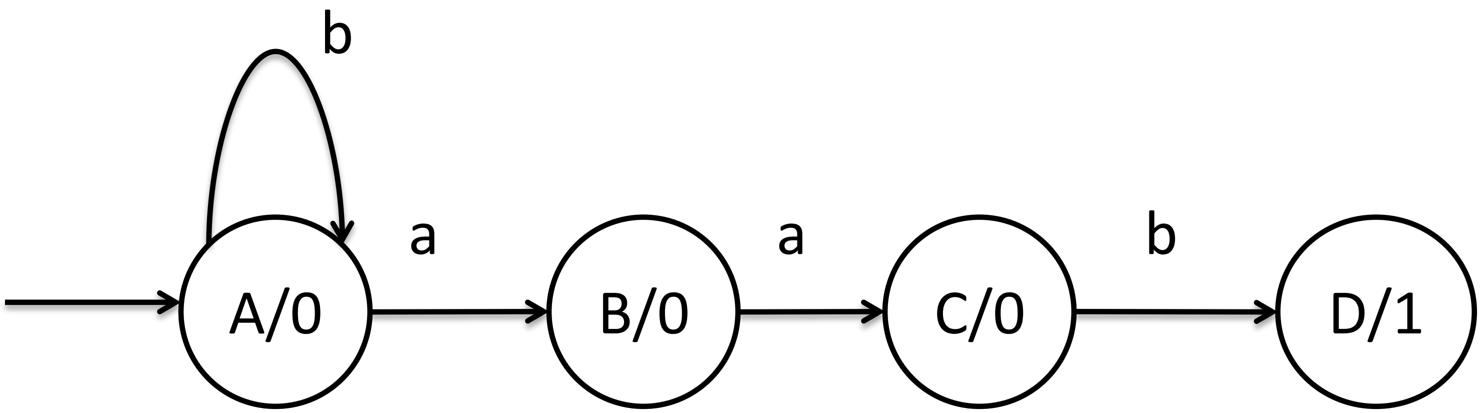
4 states are required for the smallest input aab

Construct a Moore Machine that counts the occurrences of the sequence “aab” in any input strings over {a, b}



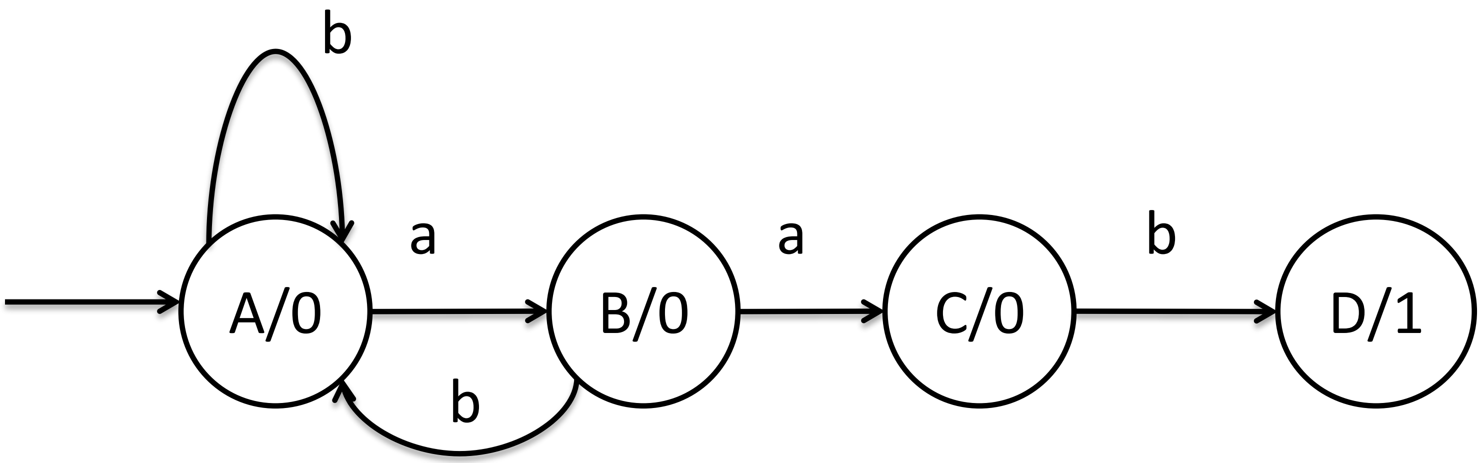
The Moore machine will print 1 as output when **aab** is found.
It will print 0 for any other transition.

Construct a Moore Machine that counts the occurrences of the sequence “aab” in any input strings over {a, b}



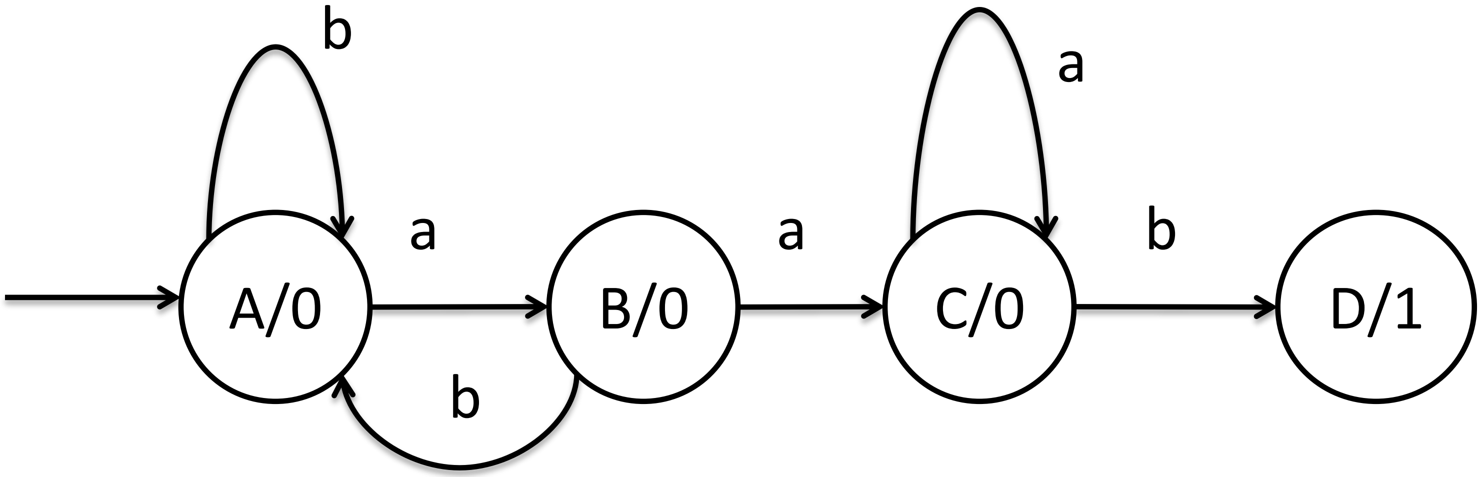
Completing other edges

Construct a Moore Machine that counts the occurrences of the sequence "aab" in any input strings over {a, b}



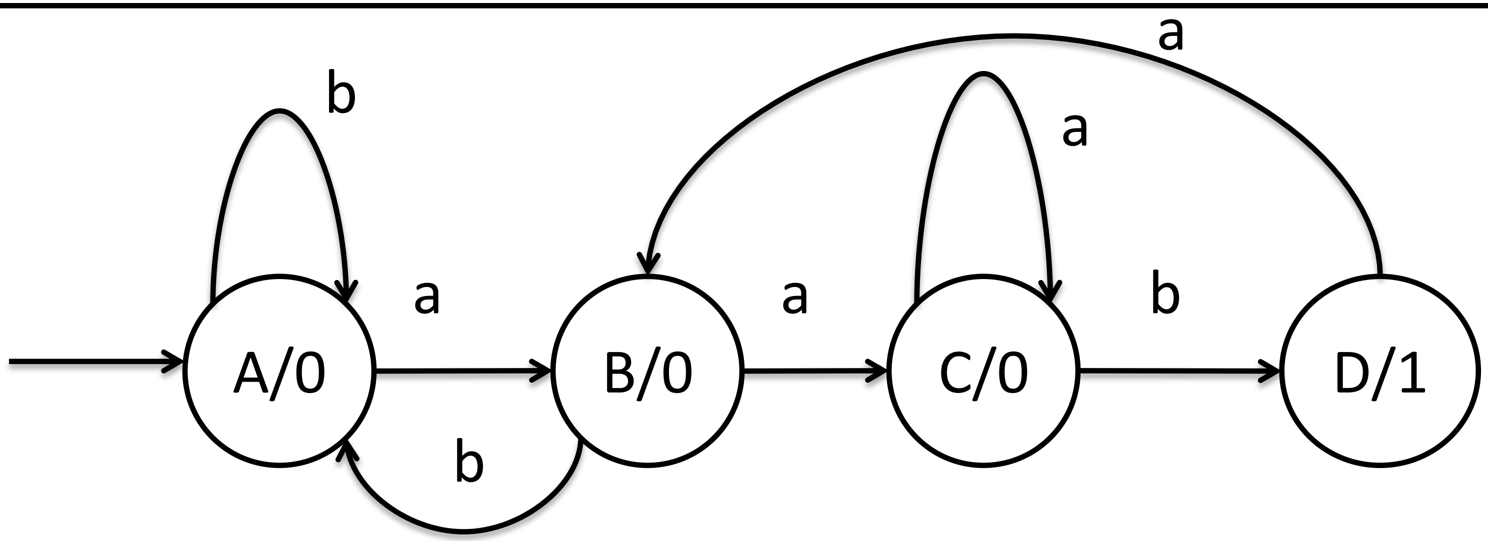
Completing other edges

Construct a Moore Machine that counts the occurrences of the sequence “aab” in any input strings over {a, b}



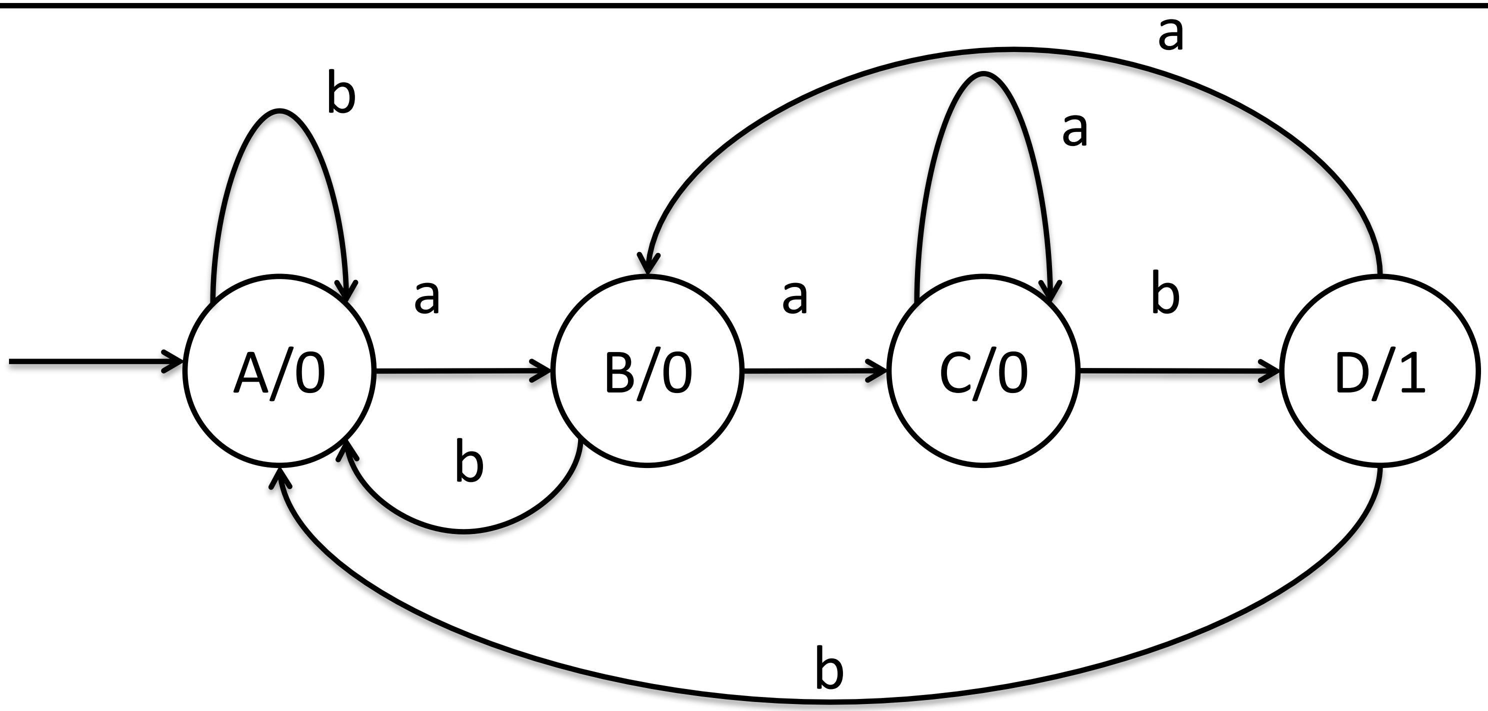
Completing other edges

Construct a Moore Machine that counts the occurrences of the sequence “aab” in any input strings over {a, b}



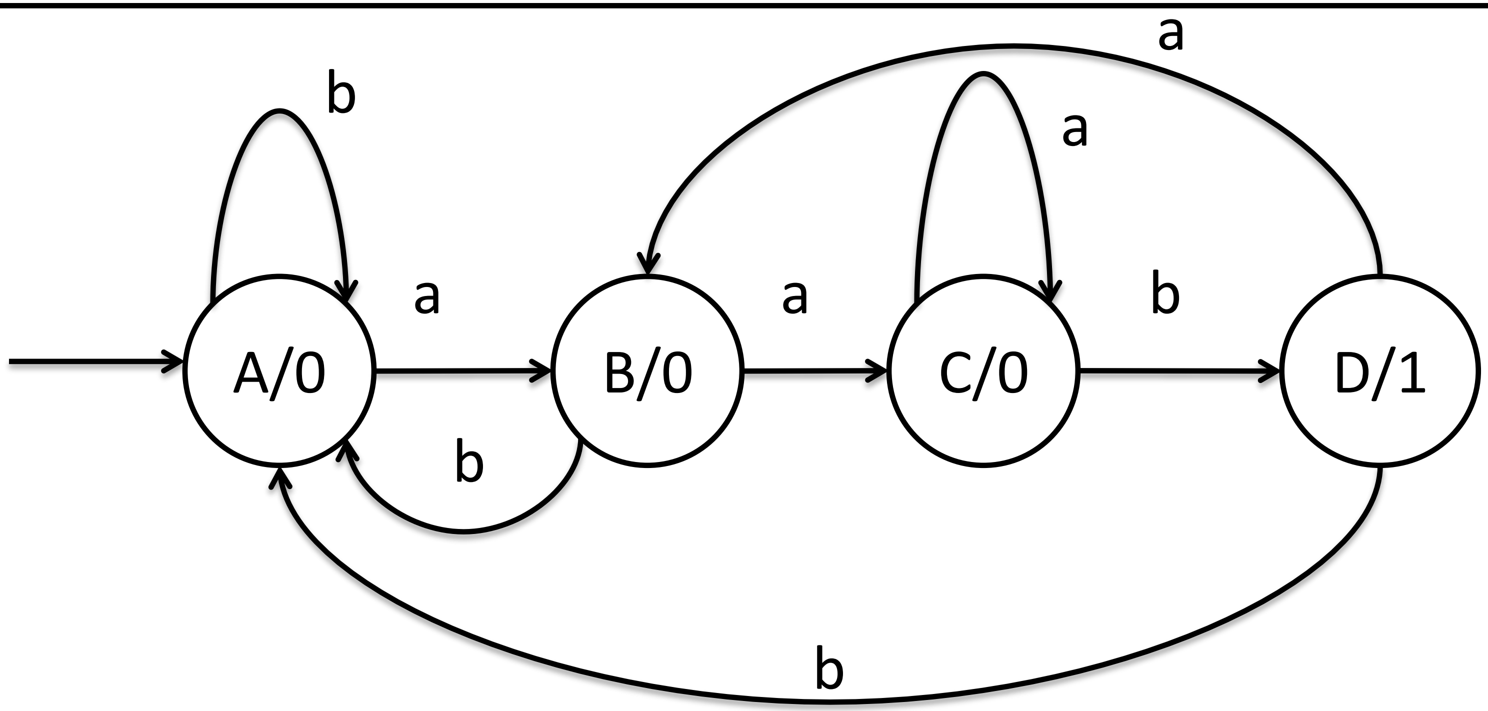
Completing other edges

Construct a Moore Machine that counts the occurrences of the sequence “aab” in any input strings over {a, b}



Completing other edges

Construct a Moore Machine that counts the occurrences of the sequence “aab” in any input strings over {a, b}

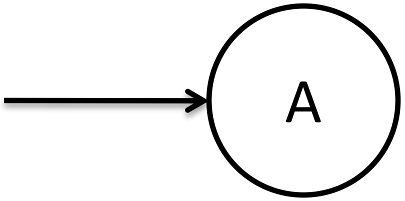


INPUT	STATE	OUTPUT
aab	ABCD	0001
abaababbb	ABABCDBAAA	0000010000
aabaabab	ABCDBCDBA	000100100

Construct a Moore Machine that gives 1’s complement of entered binary number

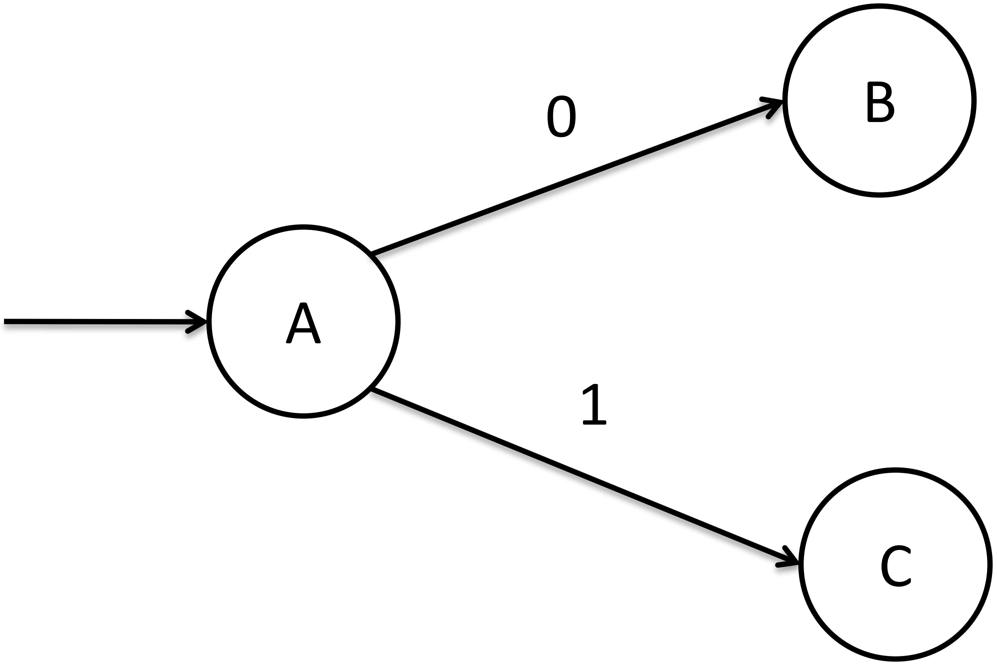
Construct a Moore Machine that gives 1’s complement of entered binary number

Start state A



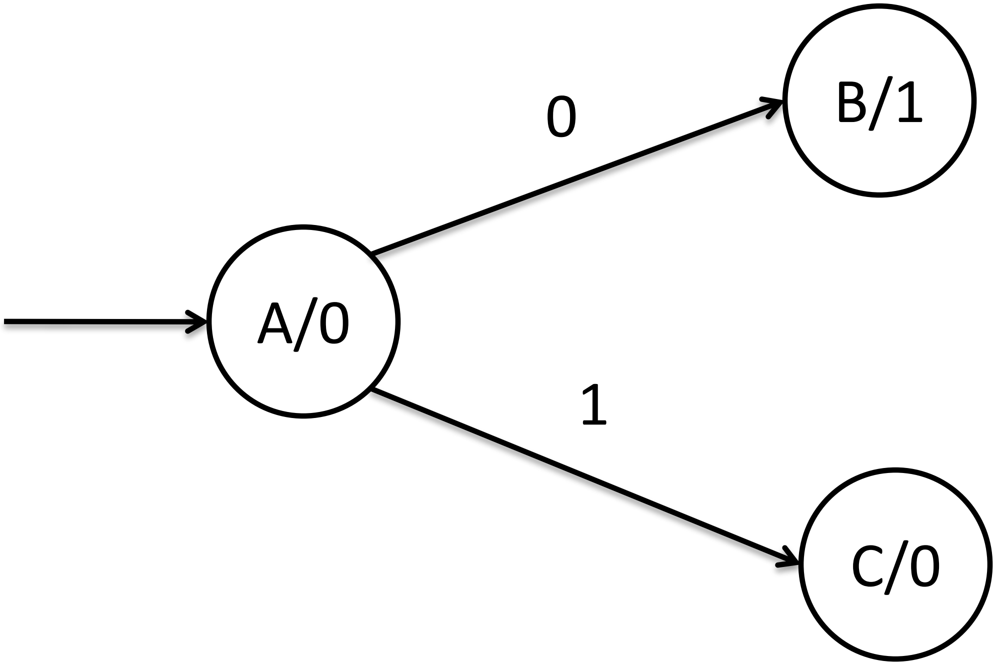
Construct a Moore Machine that gives 1’s complement of entered binary number

Two more states are required.



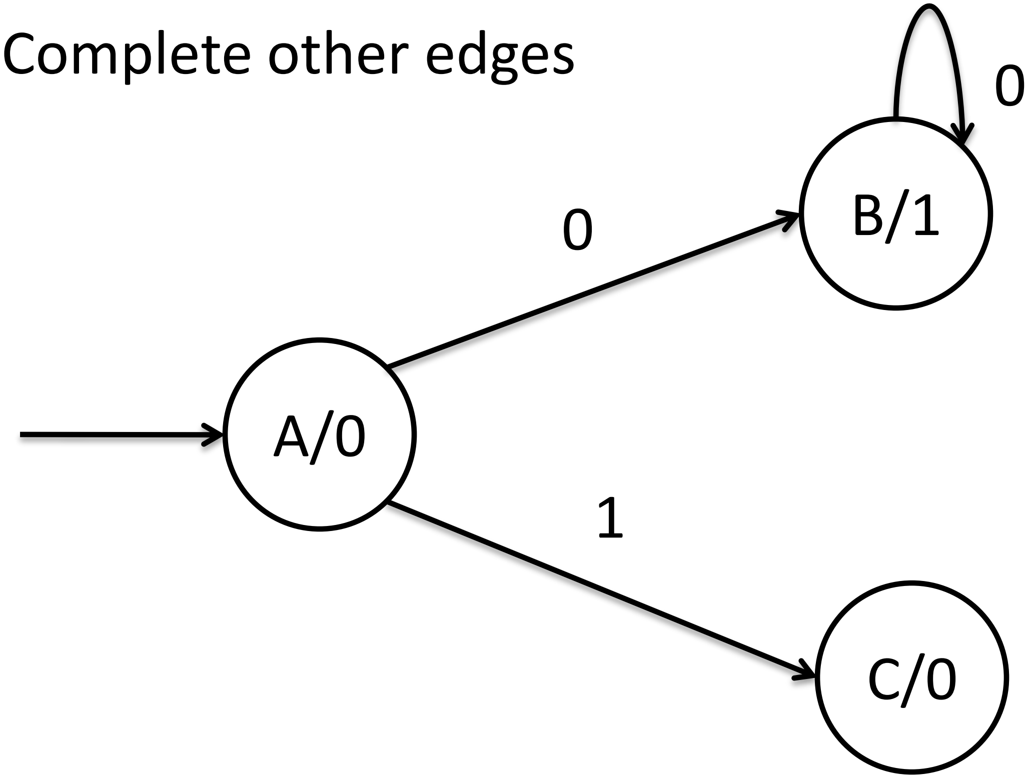
Construct a Moore Machine that gives 1's complement of entered binary number

Two more states are required.



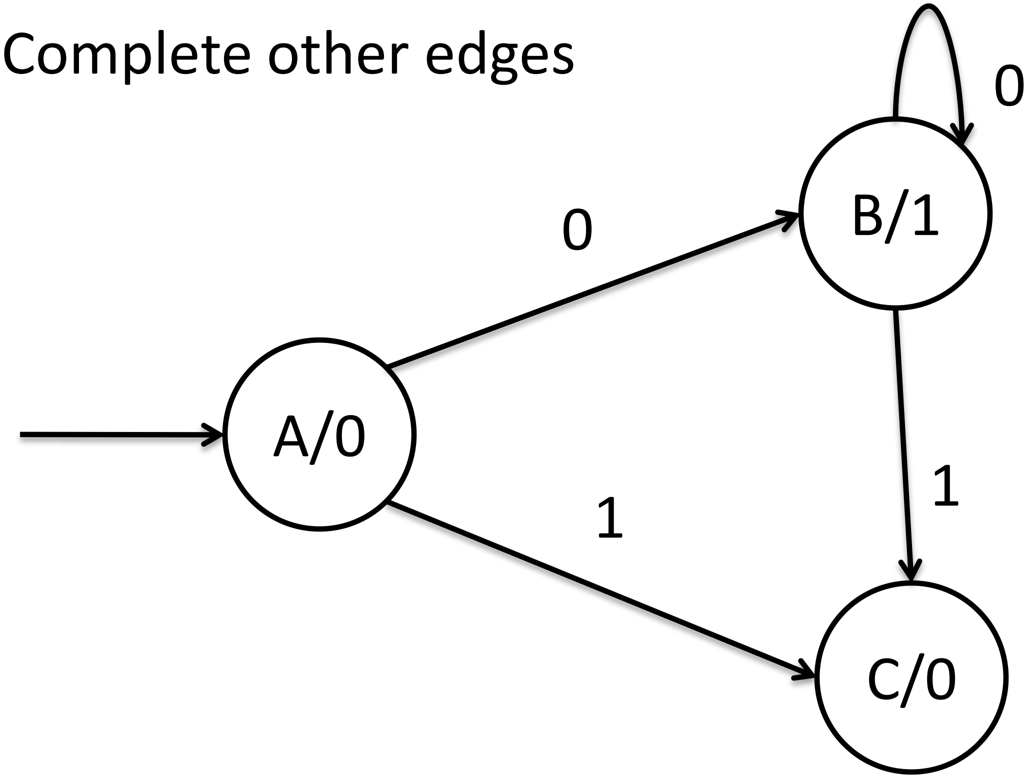
Construct a Moore Machine that gives 1's complement of entered binary number

Complete other edges



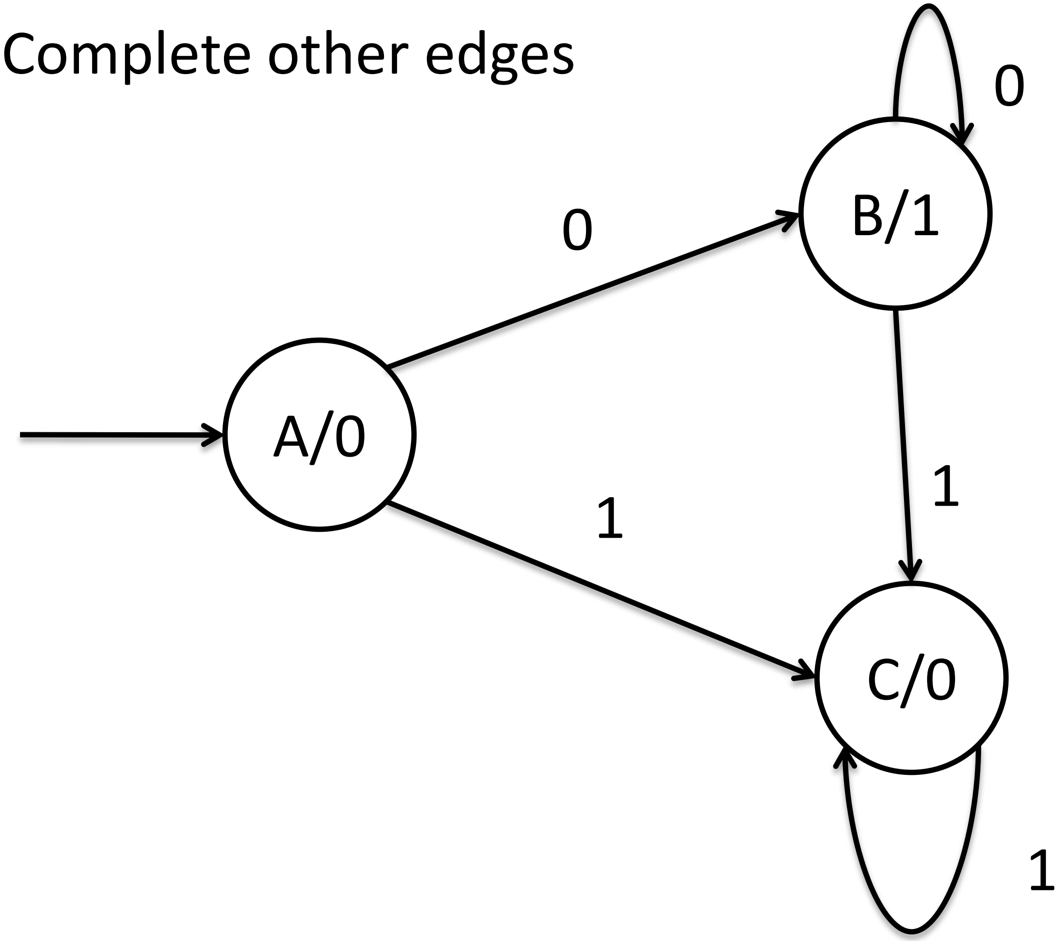
Construct a Moore Machine that gives 1's complement of entered binary number

Complete other edges



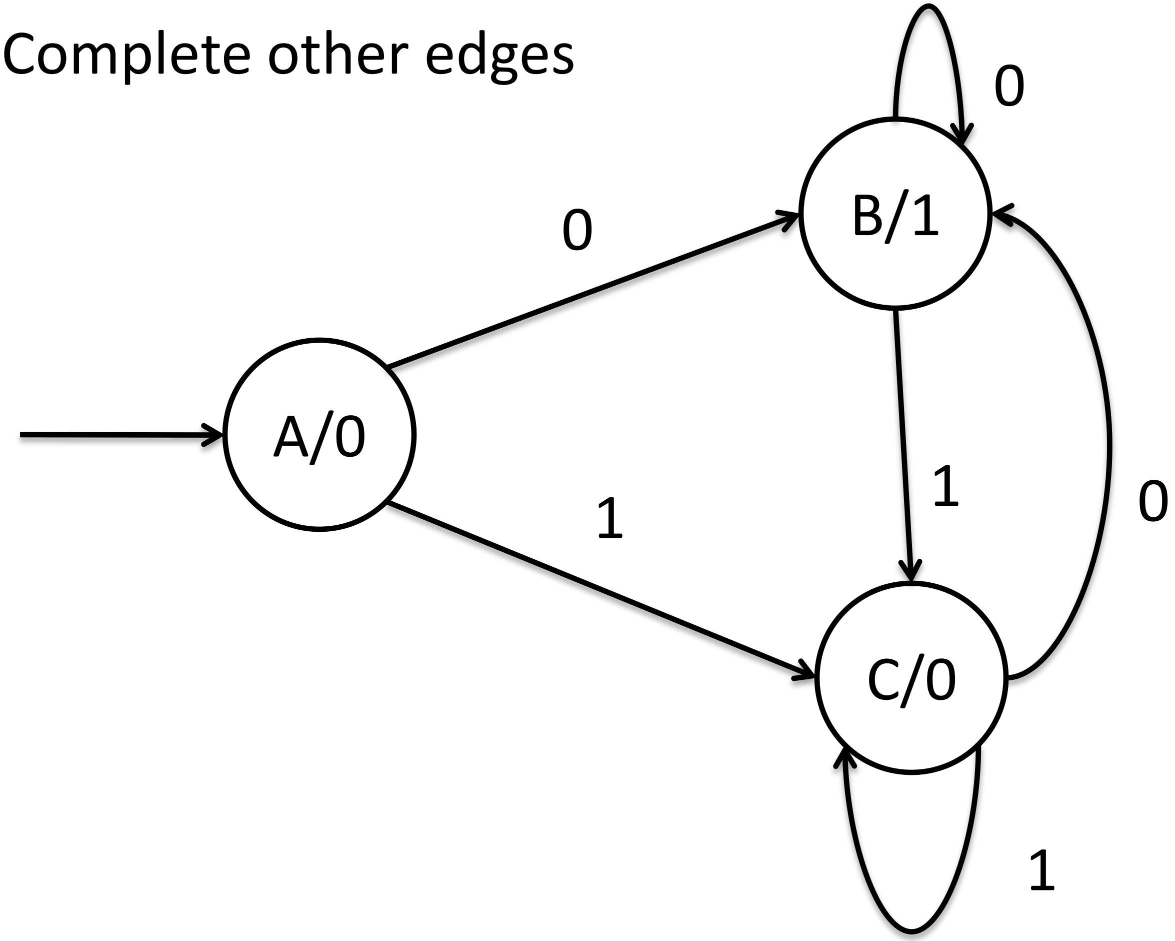
Construct a Moore Machine that gives 1's complement of entered binary number

Complete other edges

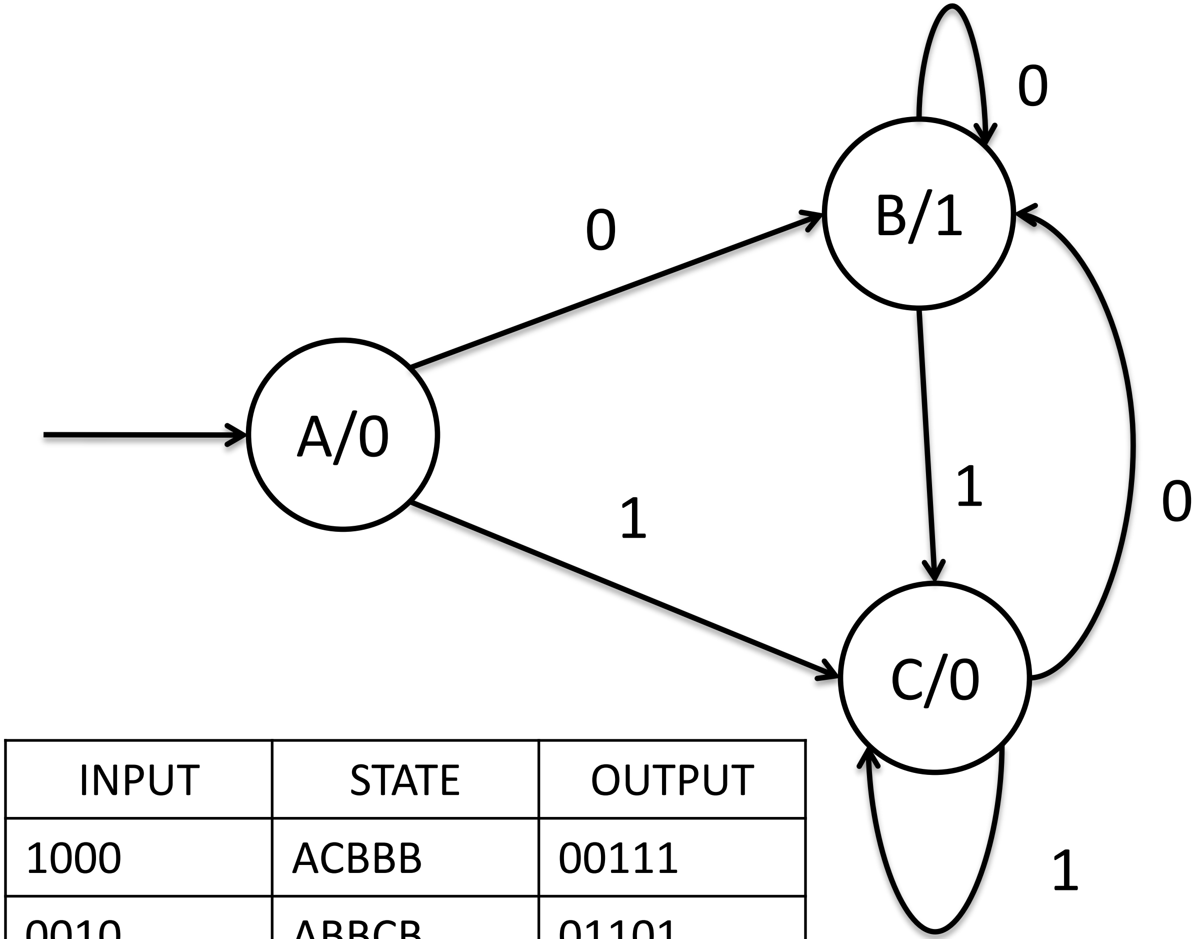


Construct a Moore Machine that gives 1's complement of entered binary number

Complete other edges



Construct a Moore Machine that gives 1’s complement of entered binary number

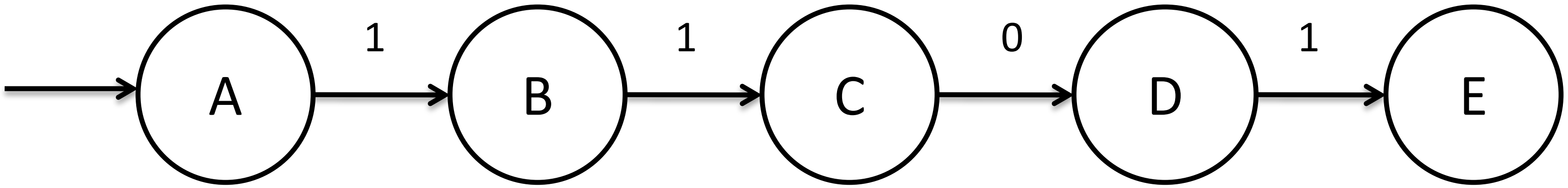


INPUT	STATE	OUTPUT
1000	ACBBB	00111
0010	ABBCB	01101
1010	ACBCB	00101

Construct a Moore Machine that counts the occurrence of 1101 (non overlapping)

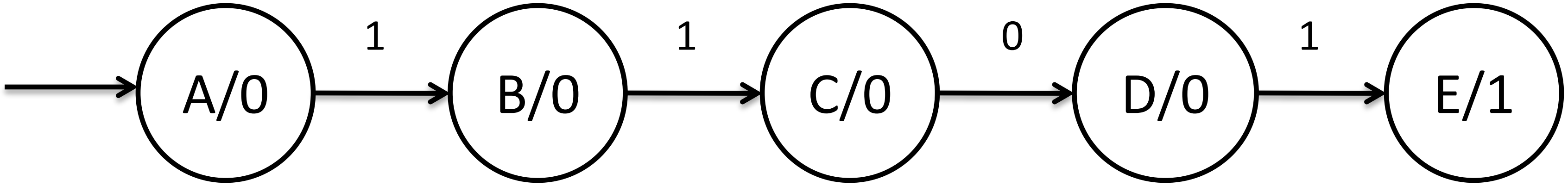
Construct a Moore Machine that counts the occurrence of 1101 (non overlapping)

For input 1101, 5 states are required.



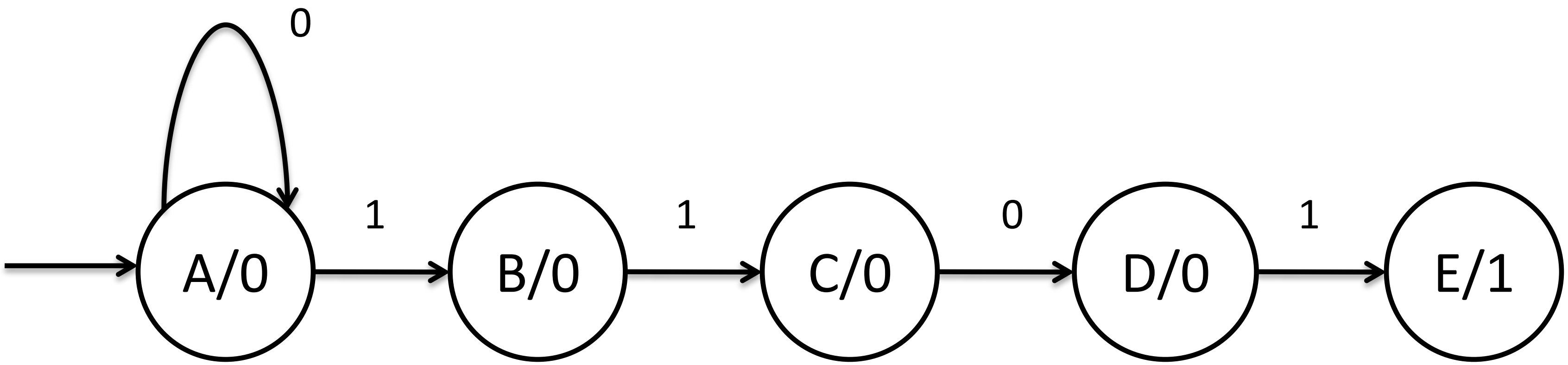
Construct a Moore Machine that counts the occurrence of 1101 (non overlapping)

For input 1101, 5 states are required.



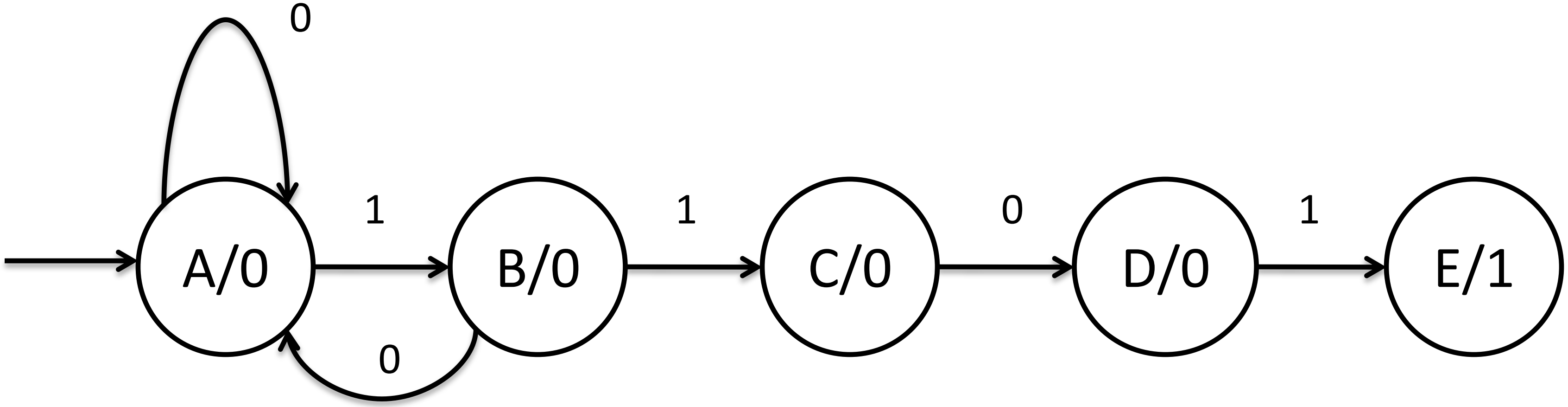
Construct a Moore Machine that counts the occurrence of 1101 (non overlapping)

Complete other edges



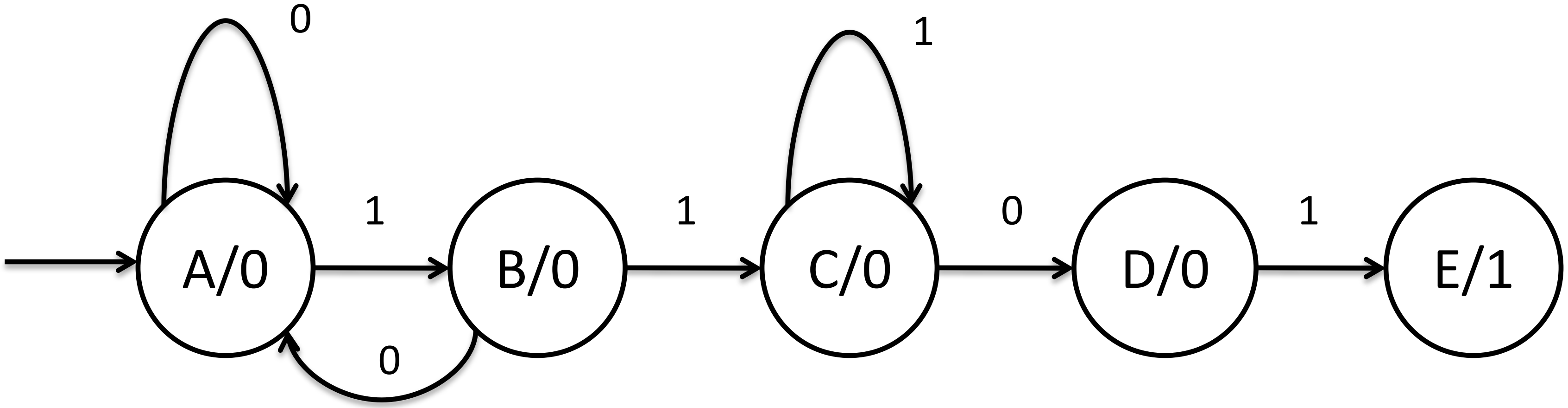
Construct a Moore Machine that counts the occurrence of 1101 (non overlapping)

Complete other edges



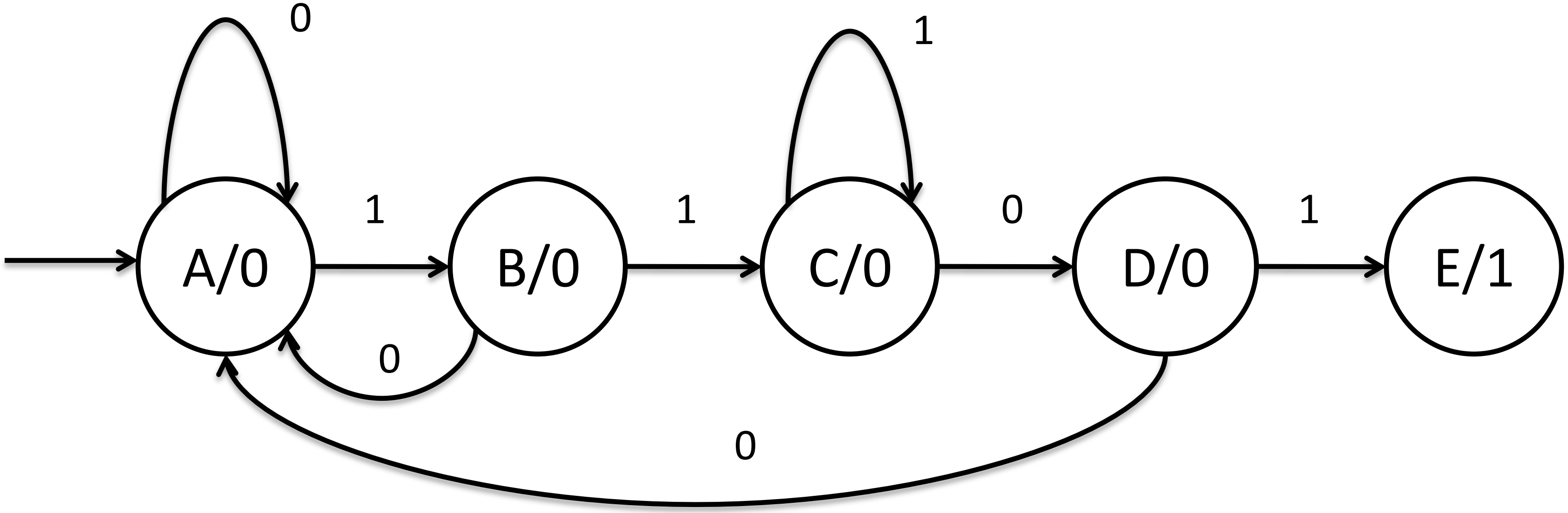
Construct a Moore Machine that counts the occurrence of 1101 (non overlapping)

Complete other edges



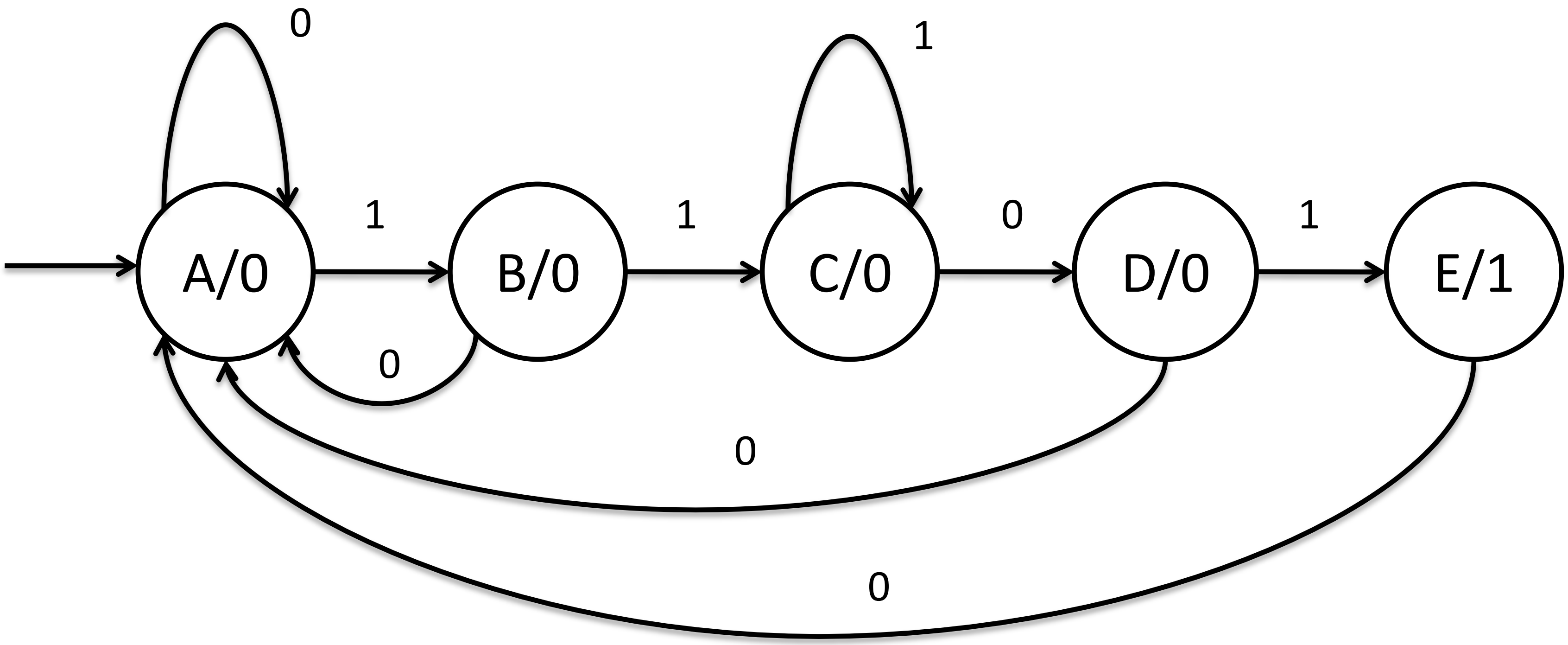
Construct a Moore Machine that counts the occurrence of 1101 (non overlapping)

Complete other edges



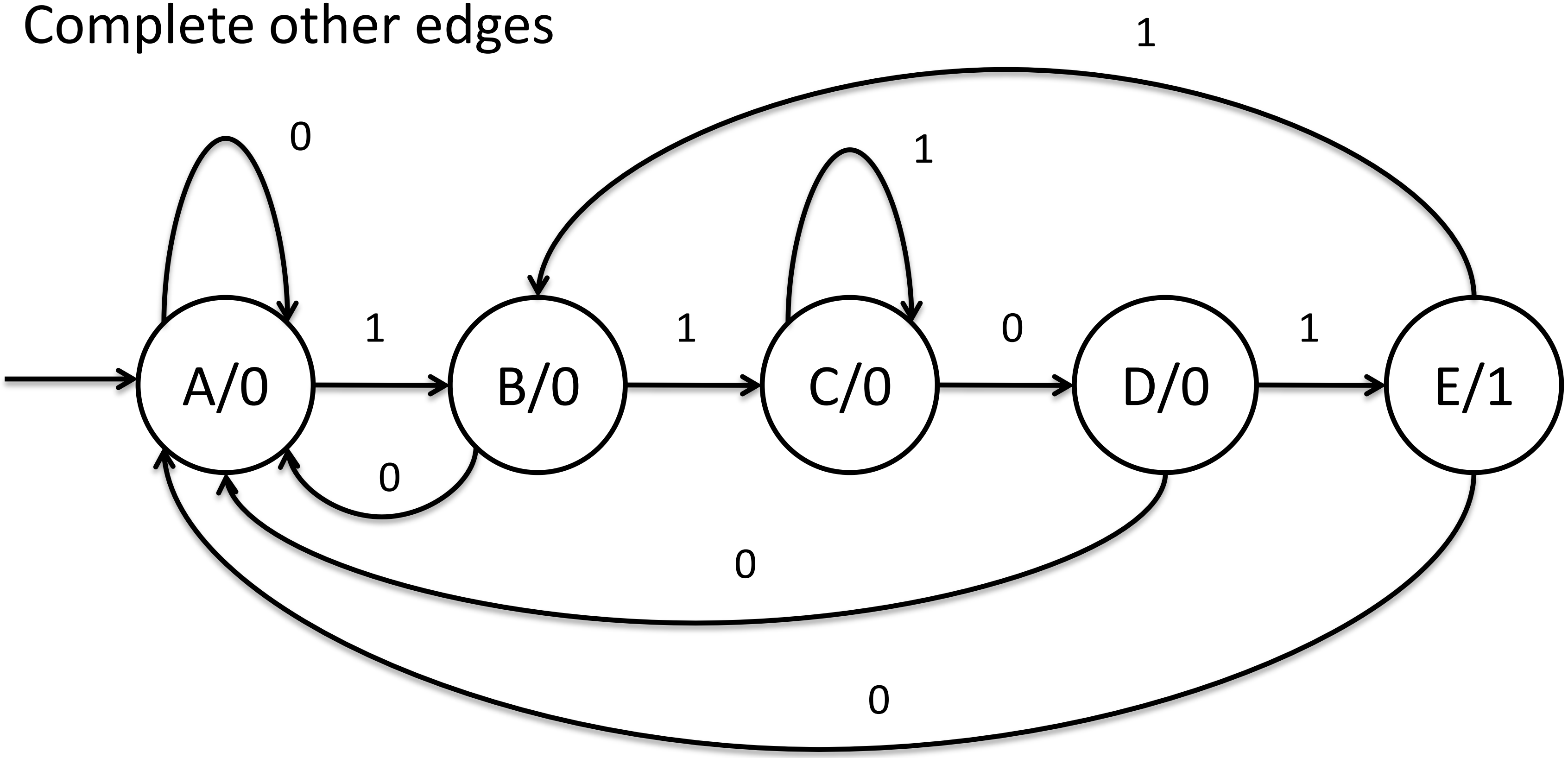
Construct a Moore Machine that counts the occurrence of 1101 (non overlapping)

Complete other edges

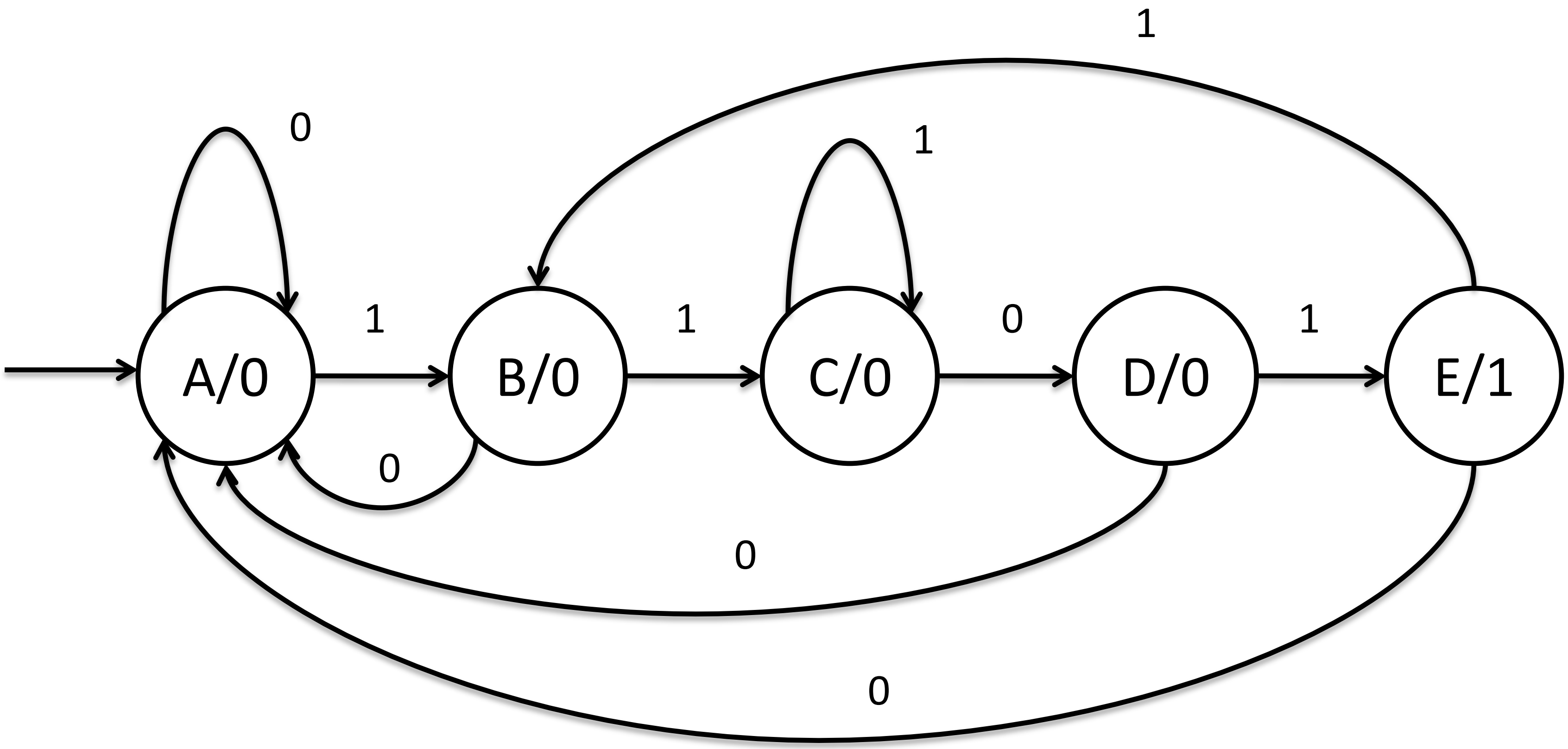


Construct a Moore Machine that counts the occurrence of 1101 (non overlapping)

Complete other edges



Construct a Moore Machine that counts the occurrence of 1101 (**non overlapping**)

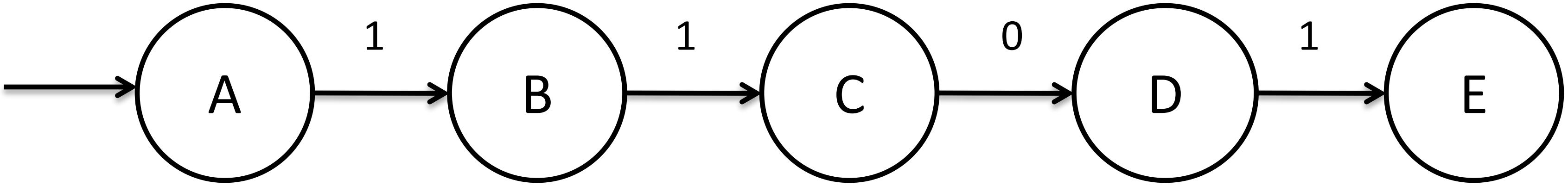


INPUT	STATE	OUTPUT
011011101	AABCDEBCDE	0000010001
01101101	AABCDEBAB	000001000

Construct a Moore Machine that counts the occurrence of 1101 (overlapping)

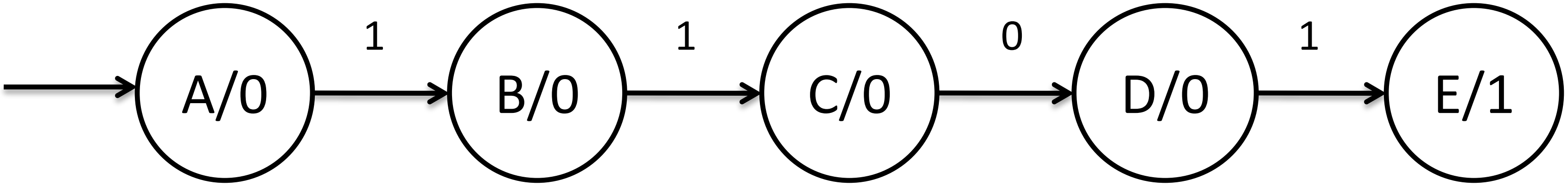
Construct a Moore Machine that counts the occurrence of 1101 (overlapping)

For input 1101, 5 states are required.



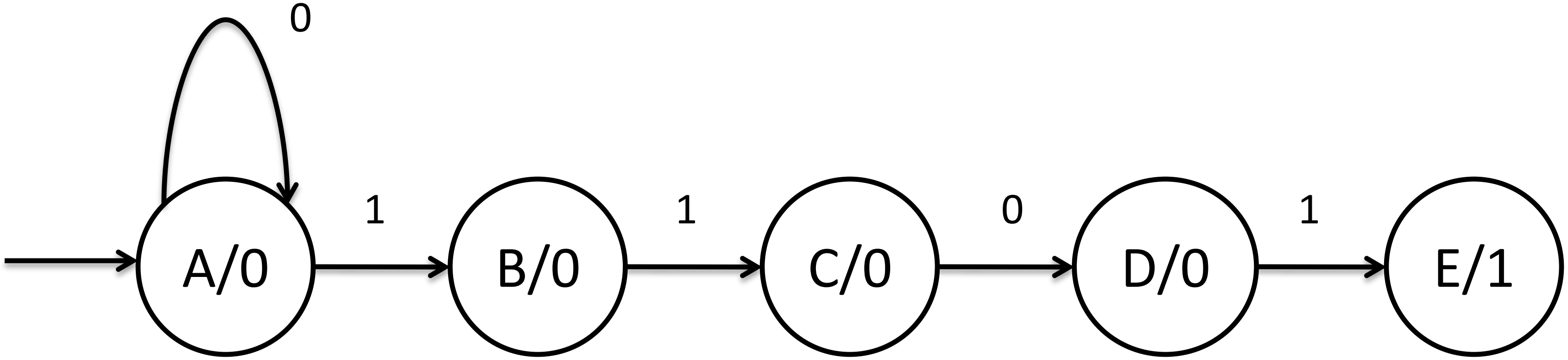
Construct a Moore Machine that counts the occurrence of 1101 (overlapping)

For input 1101, 5 states are required.



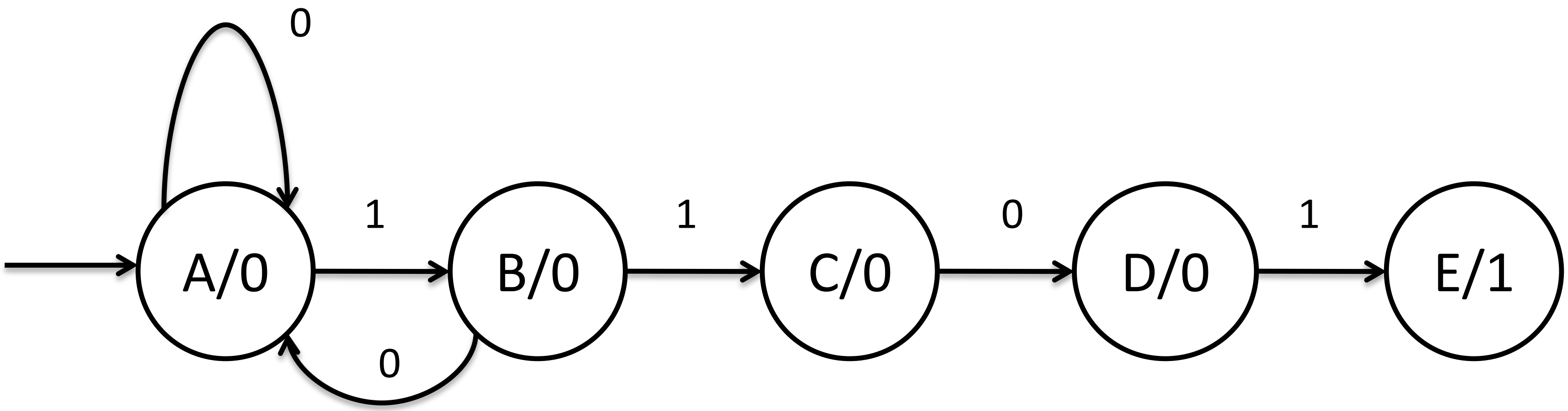
Construct a Moore Machine that counts the occurrence of 1101 (overlapping)

Complete other edges



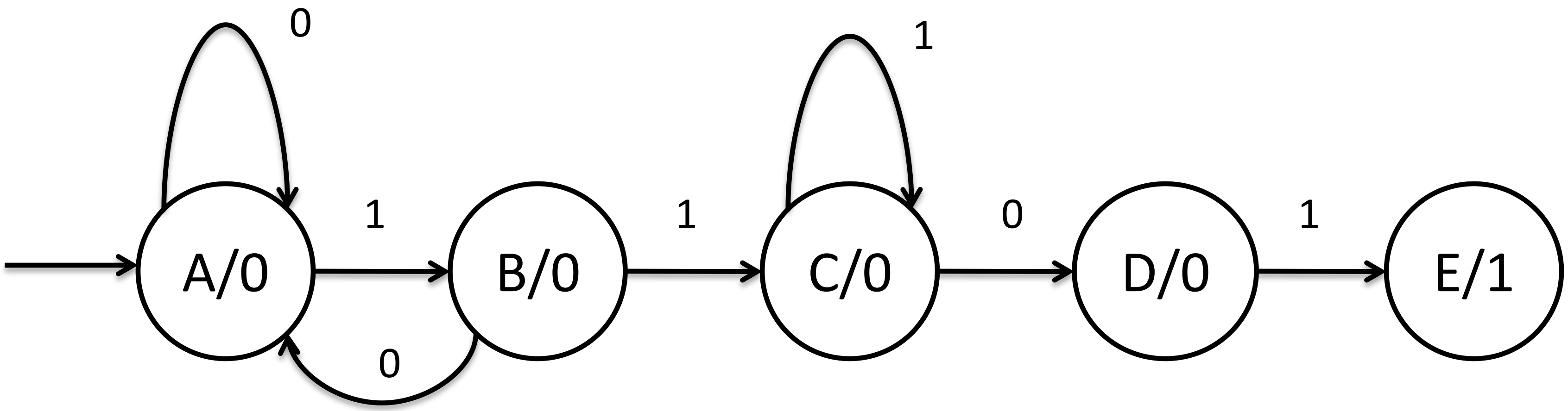
Construct a Moore Machine that counts the occurrence of 1101 (overlapping)

Complete other edges



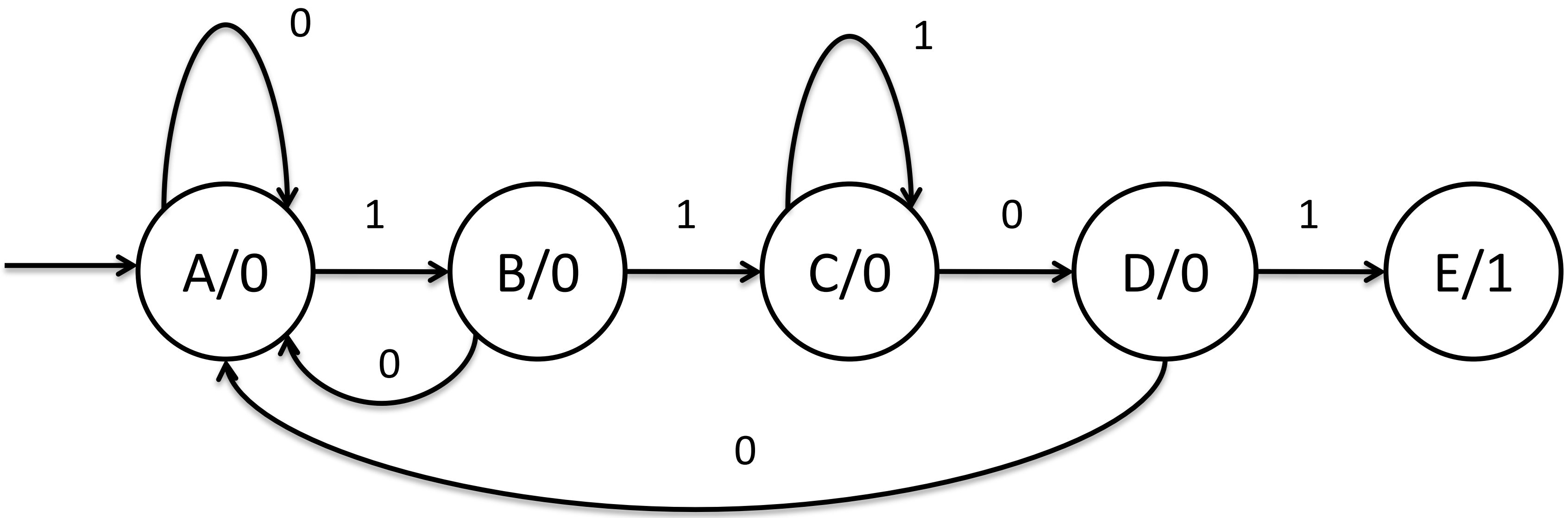
Construct a Moore Machine that counts the occurrence of 1101 (overlapping)

Complete other edges



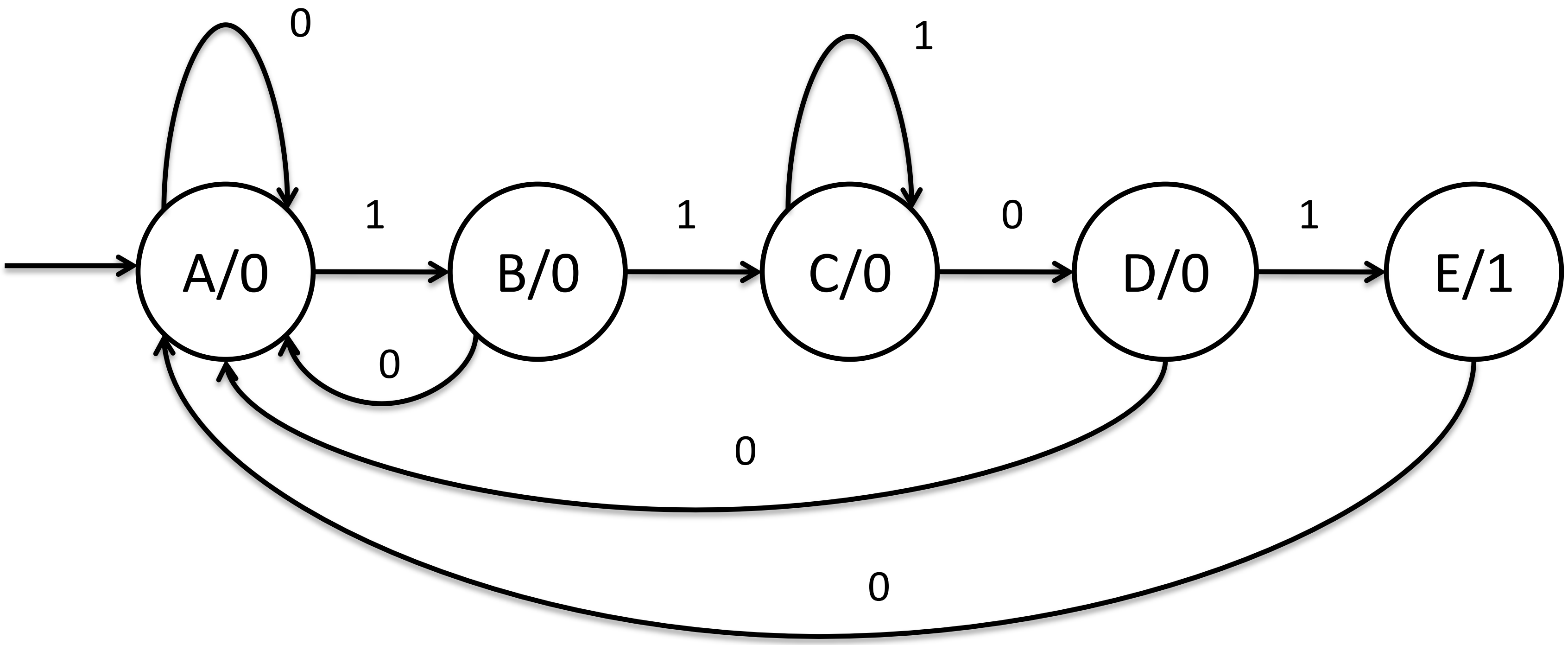
Construct a Moore Machine that counts the occurrence of 1101 (overlapping)

Complete other edges



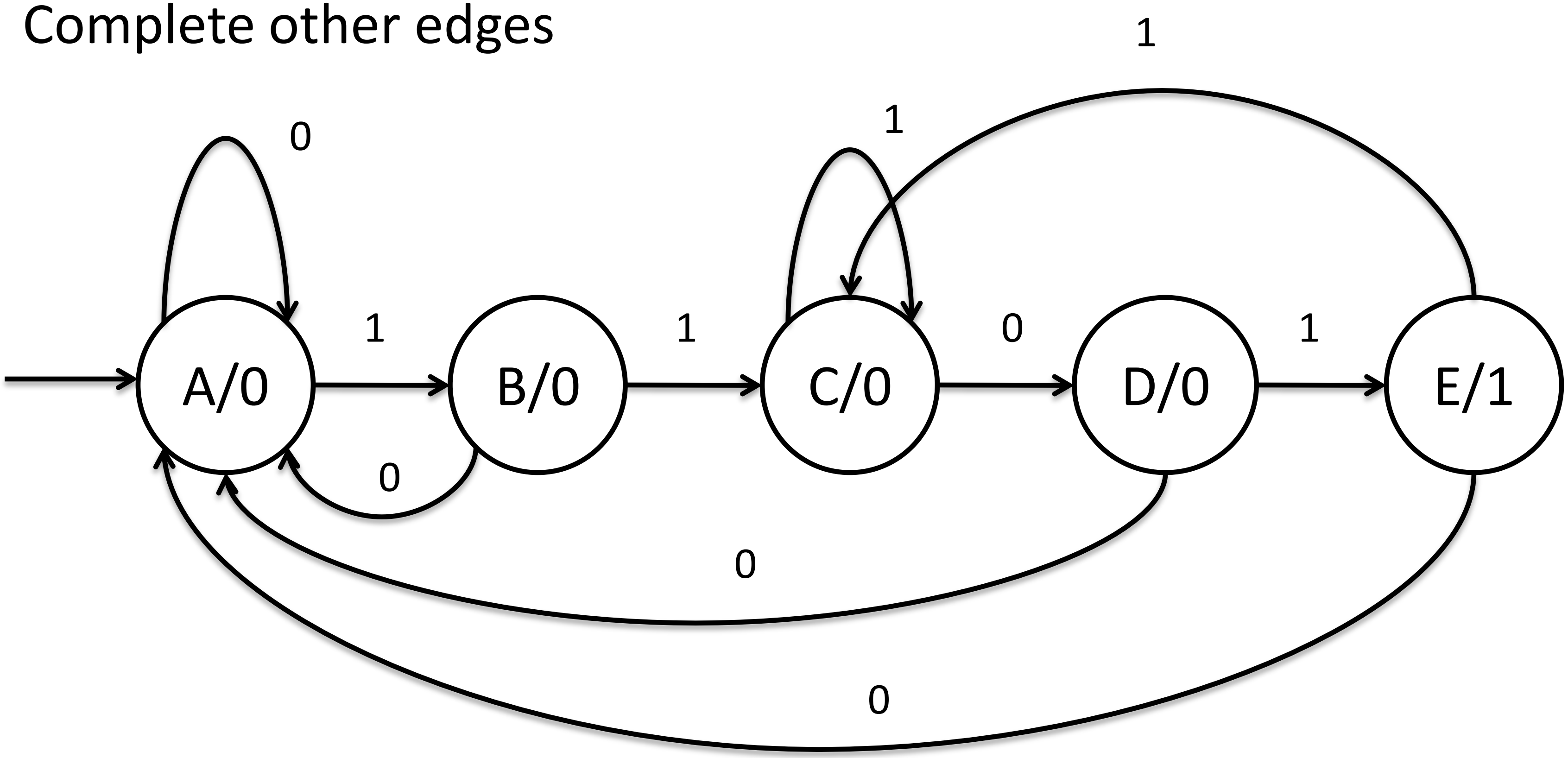
Construct a Moore Machine that counts the occurrence of 1101 (overlapping)

Complete other edges

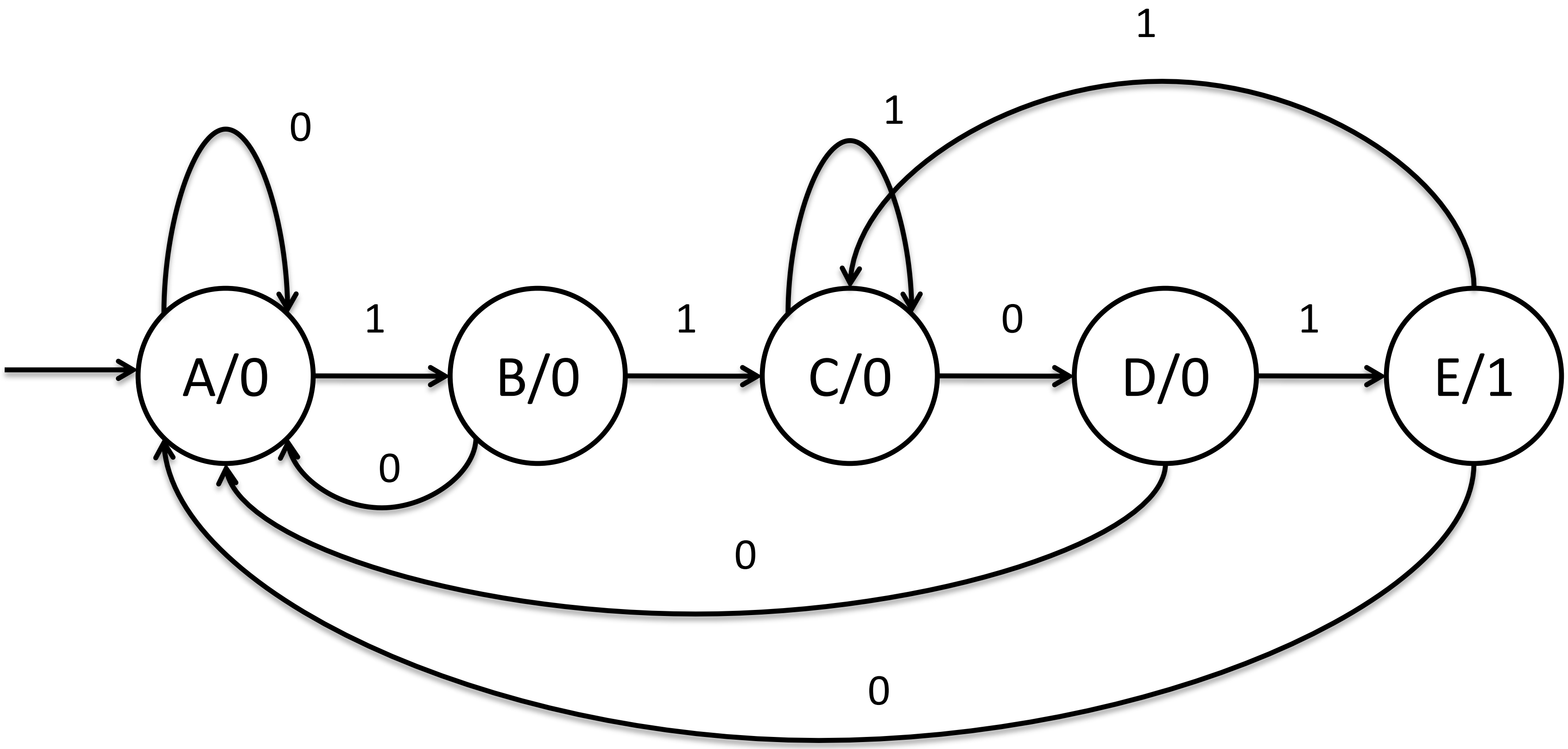


Construct a Moore Machine that counts the occurrence of 1101 (overlapping)

Complete other edges



Construct a Moore Machine that counts the occurrence of 1101 (overlapping)



INPUT	STATE	OUTPUT
011011101	AABCDECCDE	0000010001
01101101	AABCDECDE	000001001

Construct a Moore Machine replaces the first 1 with 0 from every substring starting with 1

Construct a Moore Machine replaces the first 1 with 0 from every substring(longest) starting with 1

e.g.

i/p 0111001 → 0111001

o/p 0011000

i/p 0001001110 → 0001001110

o/p 0000000110

i/p 111010110 → 111010110

o/p 011000010

Construct a Moore Machine replaces the first 1 with 0 from every substring starting with 1

So, for every 1 in input after a 1, output is 1.

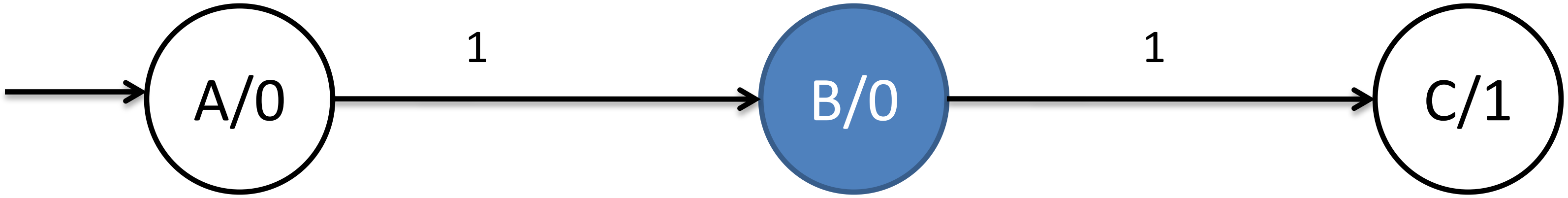
For 1 in input after a 0, output is 1.

And, for every 0 in input, output is 0.

Construct a Moore Machine replaces the first 1 with 0 from every substring starting with 1

A substring starting with 1 can be either 11 or 10.

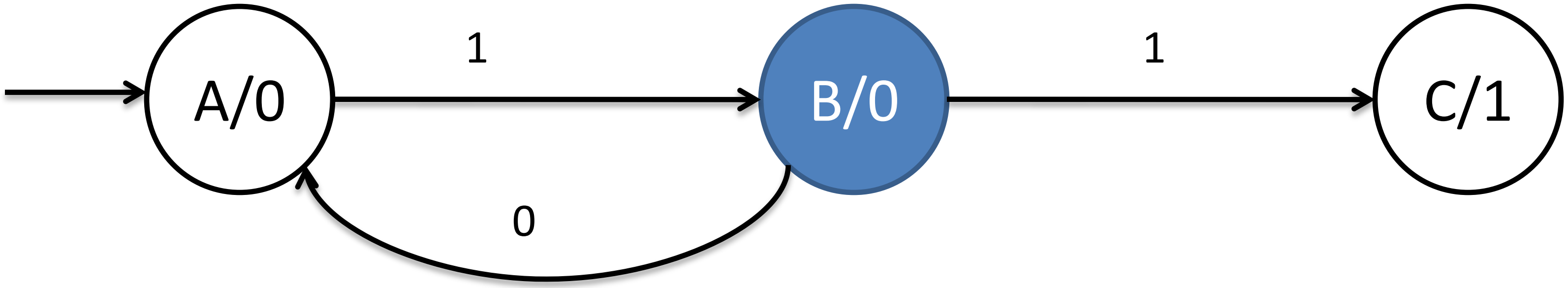
11 → 01 and 10 →00



Construct a Moore Machine replaces the first 1 with 0 from every substring starting with 1

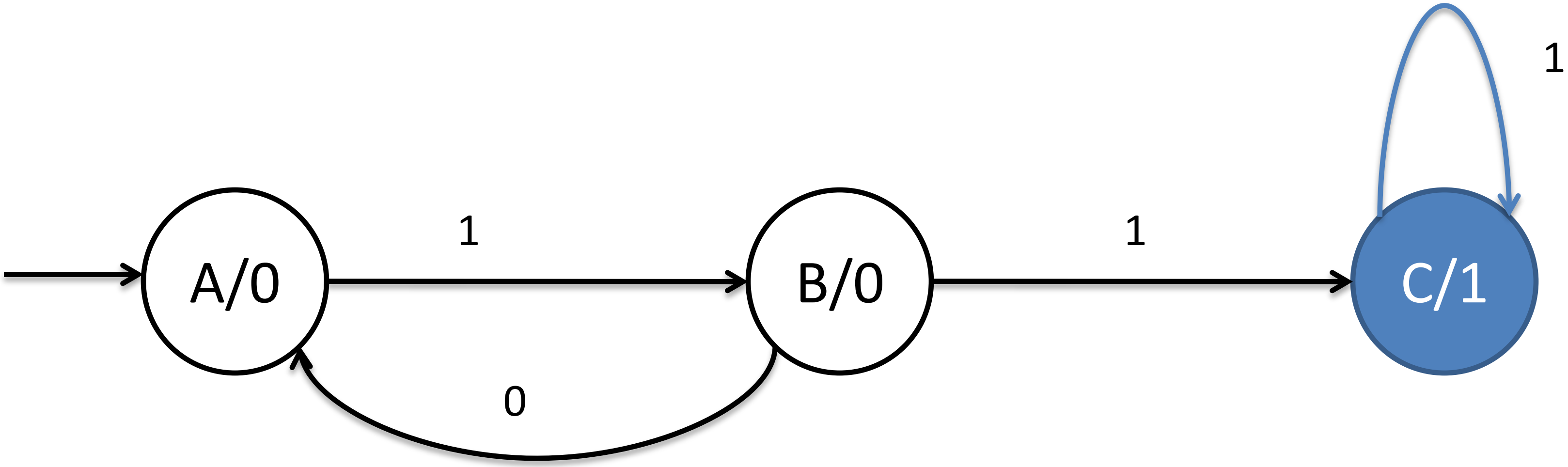
A substring starting with 1 can be either 11 or 10.

11 → 01 and 10 →00



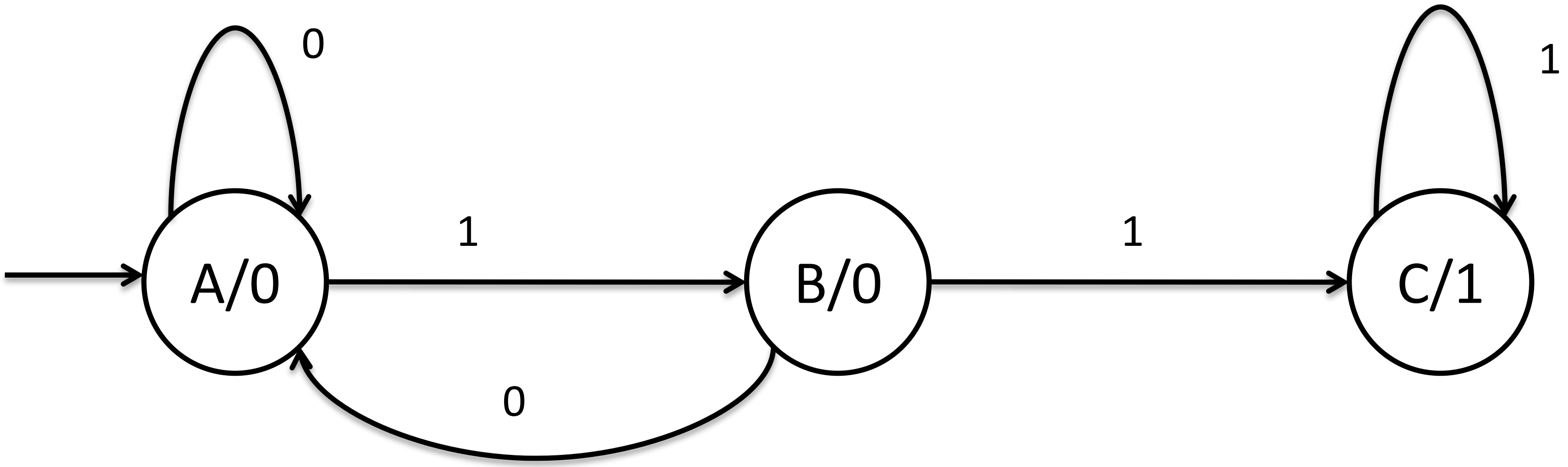
Construct a Moore Machine replaces the first 1 with 0 from every substring starting with 1

Any 1's following a 1 will remain 1.



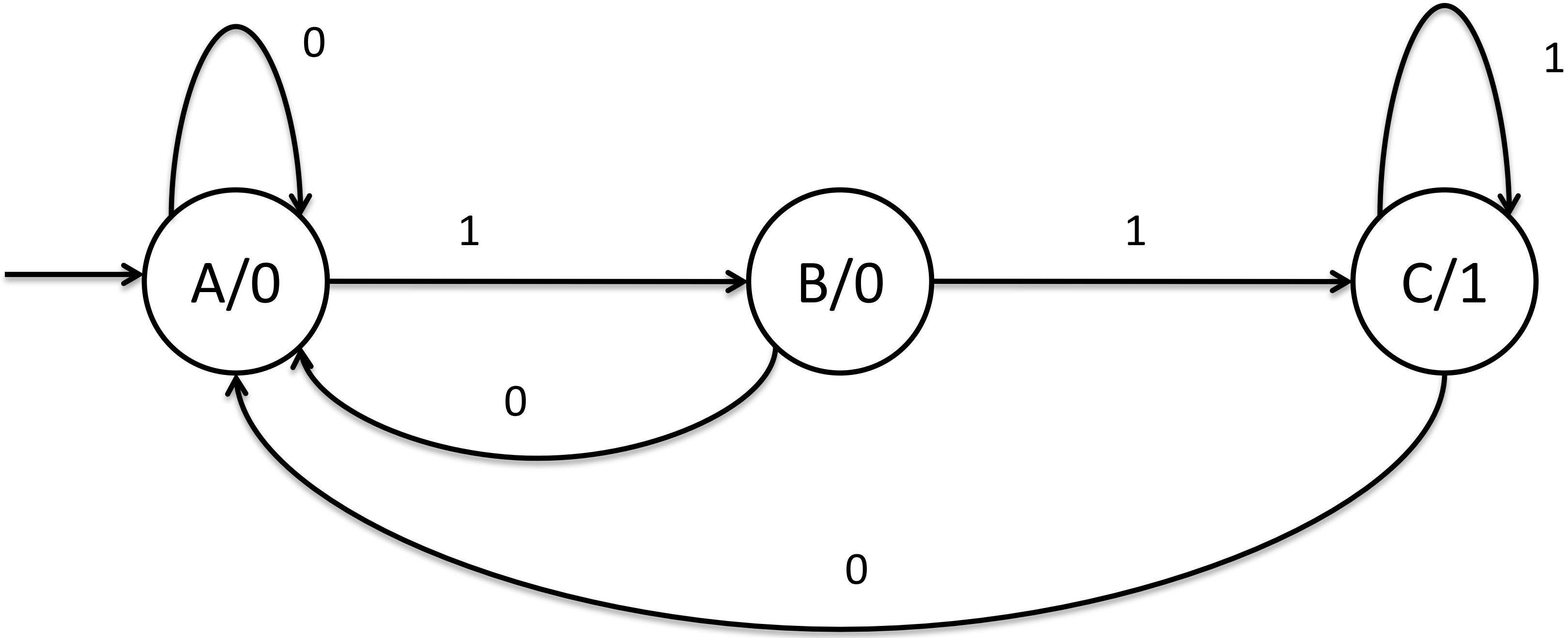
Construct a Moore Machine replaces the first 1 with 0 from every substring starting with 1

Any 0’s will remain 0.

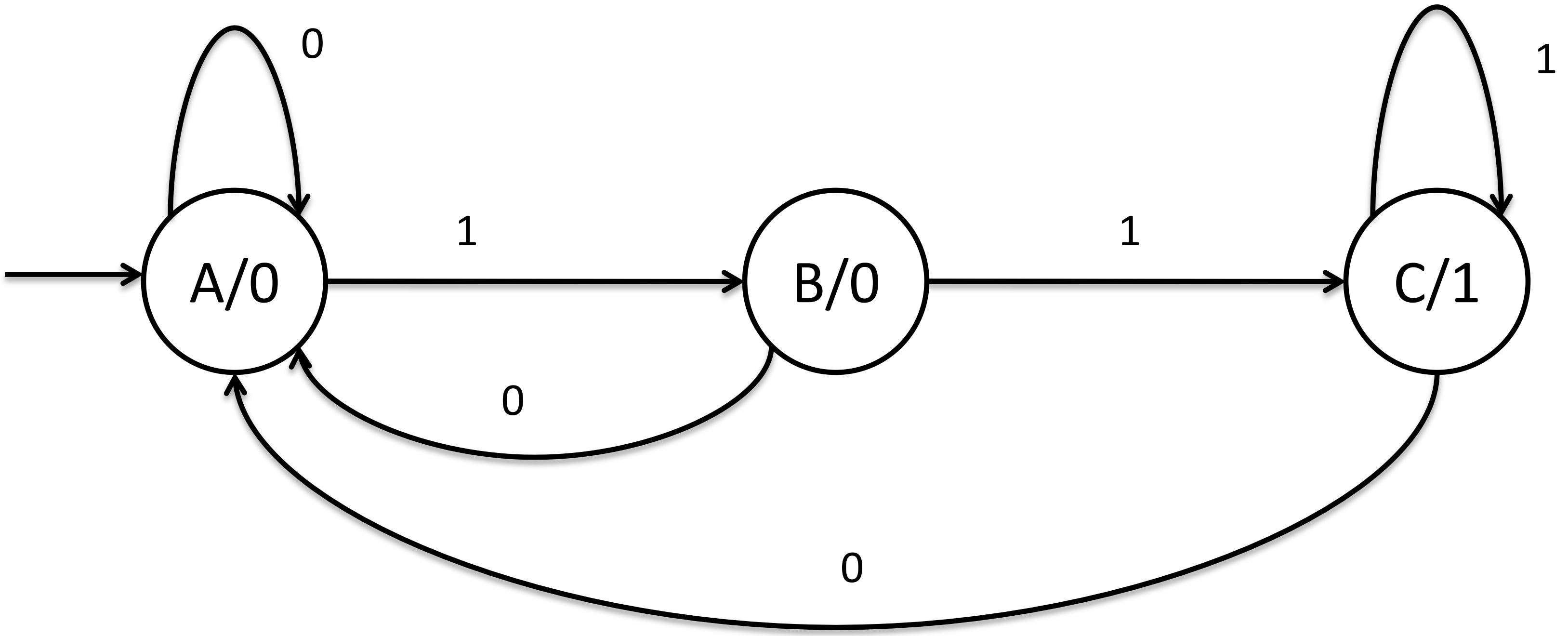


Construct a Moore Machine replaces the first 1 with 0 from every substring starting with 1

Any 0’s will remain 0.



Construct a Moore Machine replaces the first 1 with 0 from every substring starting with 1



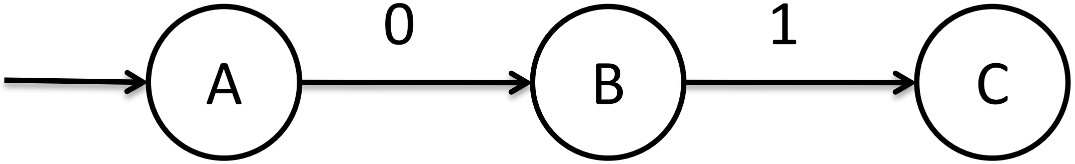
i/p	0111001	i/p	0001001110	i/p	111010110
→	0111001	→	0001001110	→	111010110
o/p	0011000	o/p	0000000110	o/p	011000010

Mealy Machine

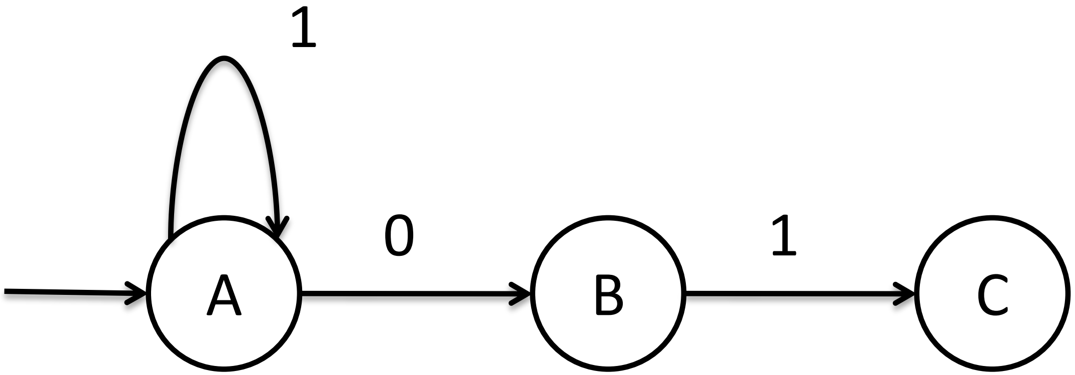
- A **Mealy machine** is a finite-state machine whose output values are determined both by its current state and the current inputs.
- The state diagram for a **Mealy machine** associates an output value with each transition edge, in contrast to the state diagram for a **Moore machine**, which associates an output value with each state.

Construct a Mealy Machine that prints ‘a’ when the sequence “01” found in any input strings over {0,1}

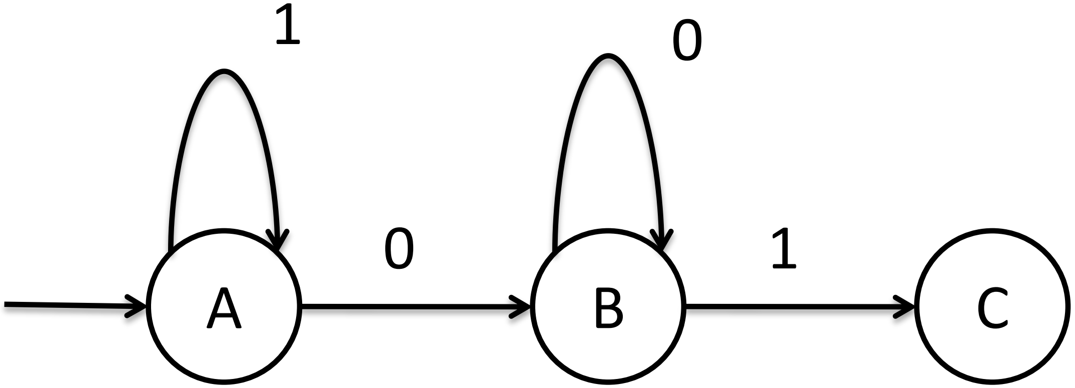
Construct a Mealy Machine that prints ‘a’ when the sequence “01” found in any input strings over {0,1}



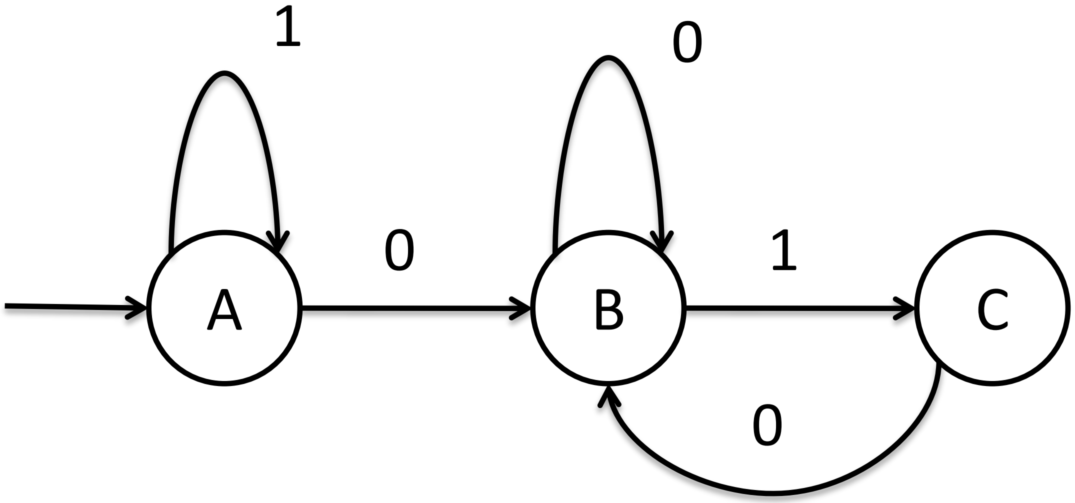
Construct a Mealy Machine that prints ‘a’ when the sequence “01” found in any input strings over {0,1}



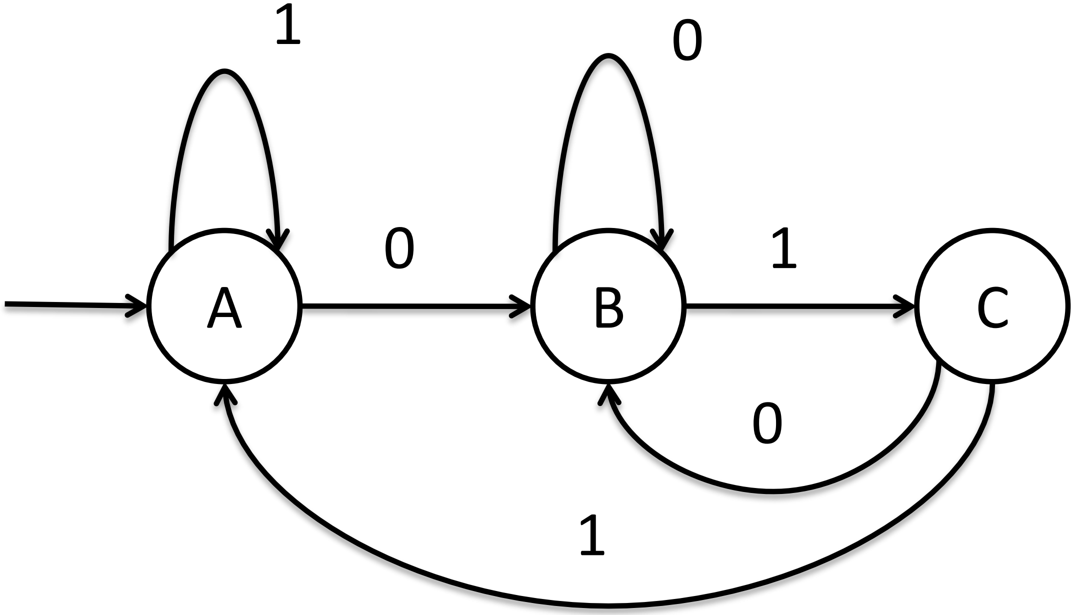
Construct a Mealy Machine that prints ‘a’ when the sequence “01” found in any input strings over {0,1}



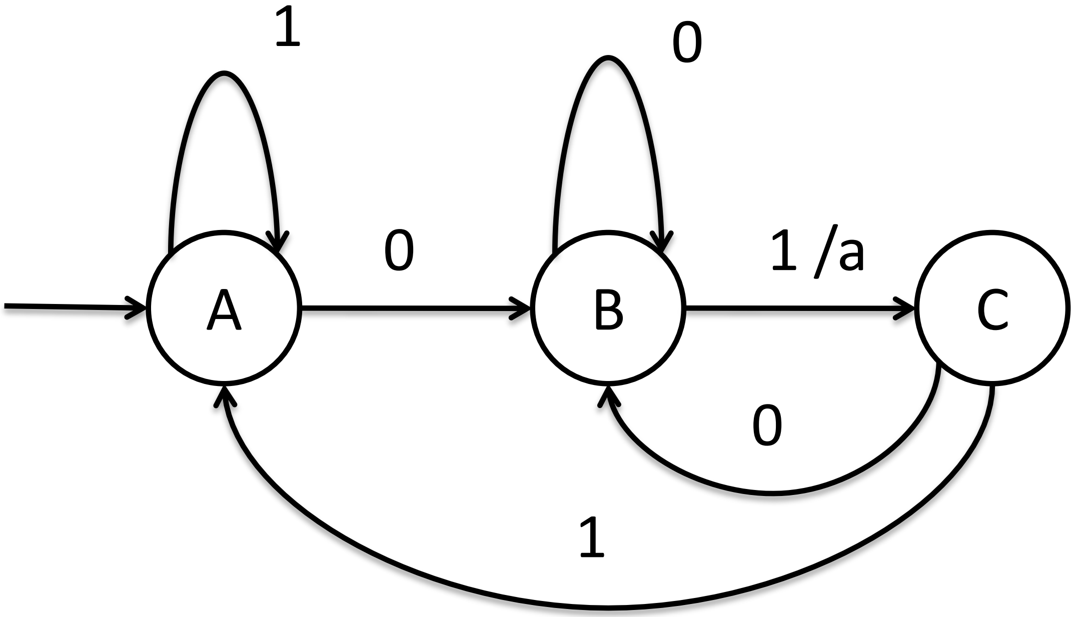
Construct a Mealy Machine that prints ‘a’ when the sequence “01” found in any input strings over {0,1}



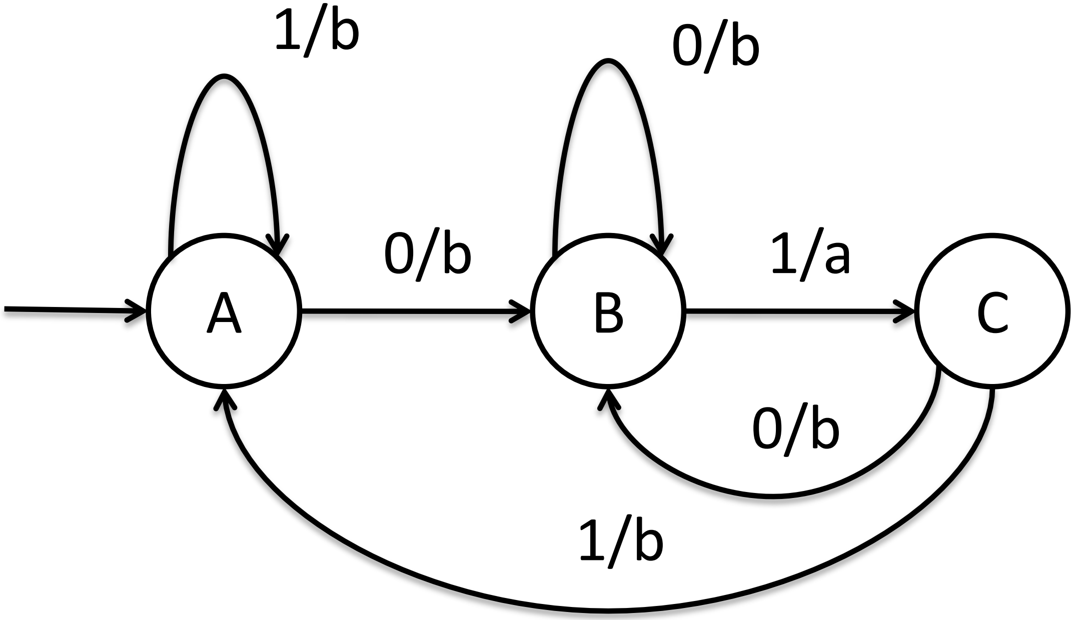
Construct a Mealy Machine that prints ‘a’ when the sequence “01” found in any input strings over {0,1}



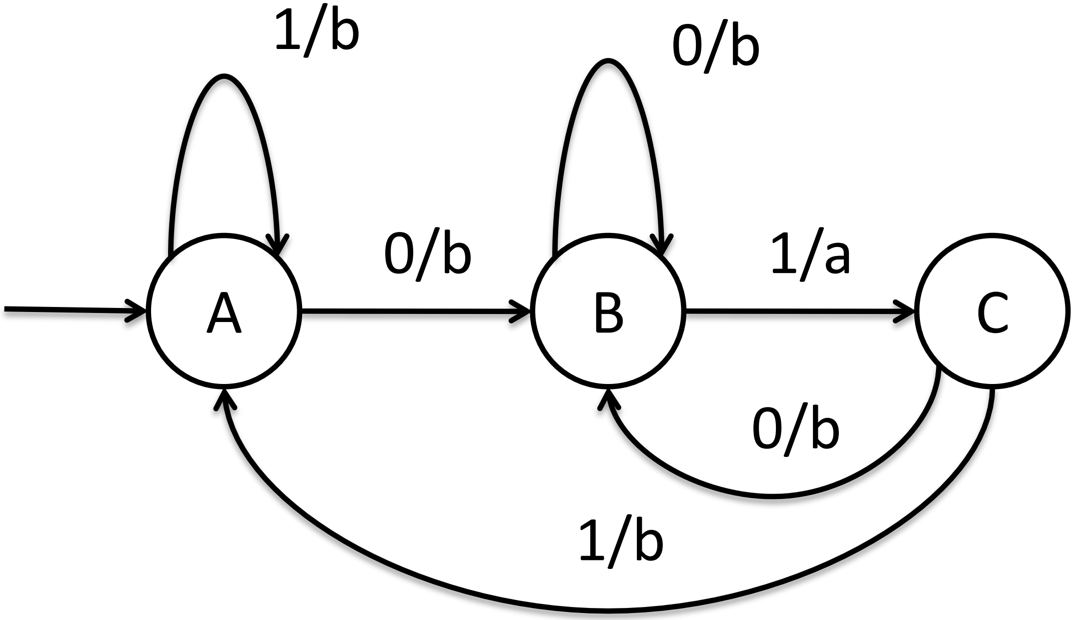
Construct a Mealy Machine that prints ‘a’ when the sequence “01” found in any input strings over {0,1}



Construct a Mealy Machine that prints ‘a’ when the sequence “01” found in any input strings over {0,1}



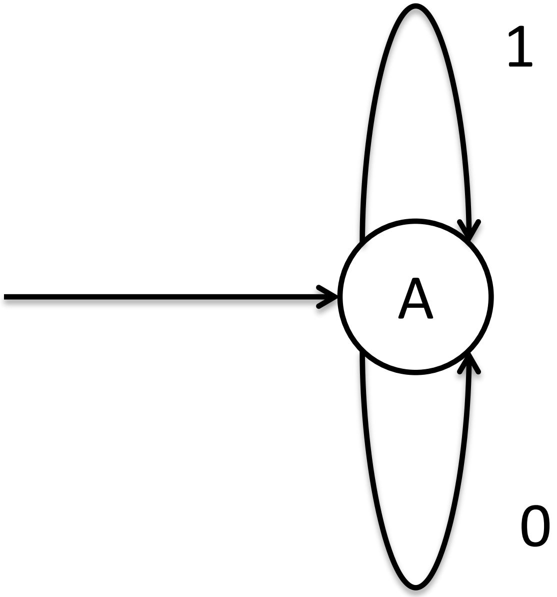
Construct a Mealy Machine that prints ‘a’ when the sequence “01” found in any input strings over {0,1}



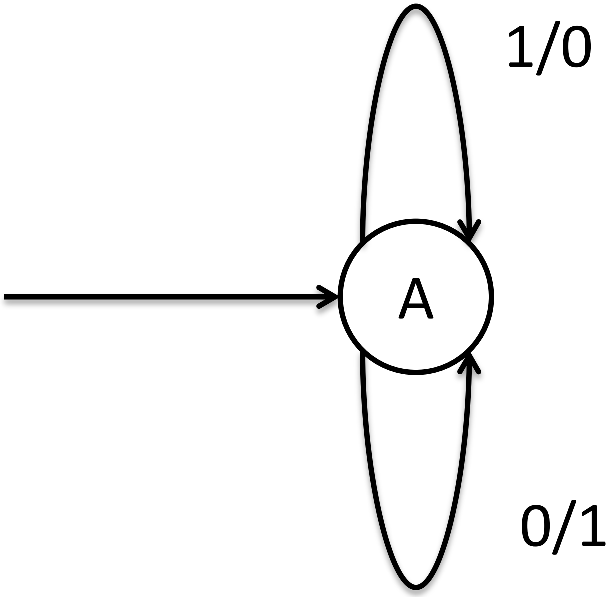
INPUT	STATE	OUTPUT
010101	ABCBCBC	bababa
001001	ABBCBBC	bbabba
1001	AABBC	bbba

Construct a Mealy Machine that produces the 1’s complement of any binary input string.

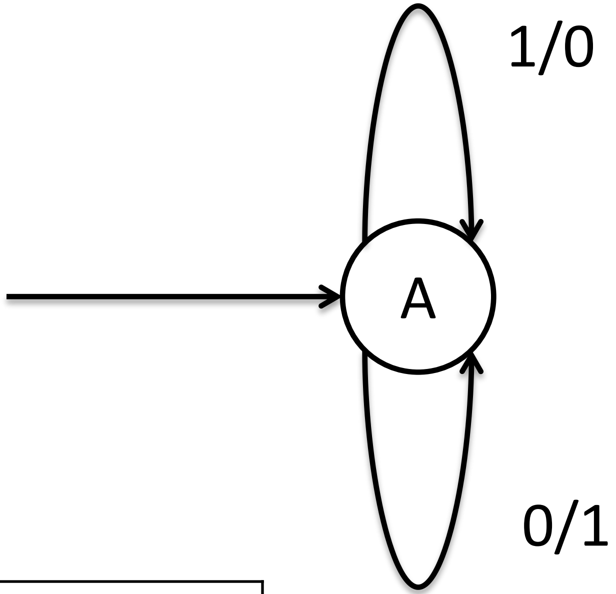
Construct a Mealy Machine that produces the 1's complement of any binary input string.



Construct a Mealy Machine that produces the 1's complement of any binary input string.



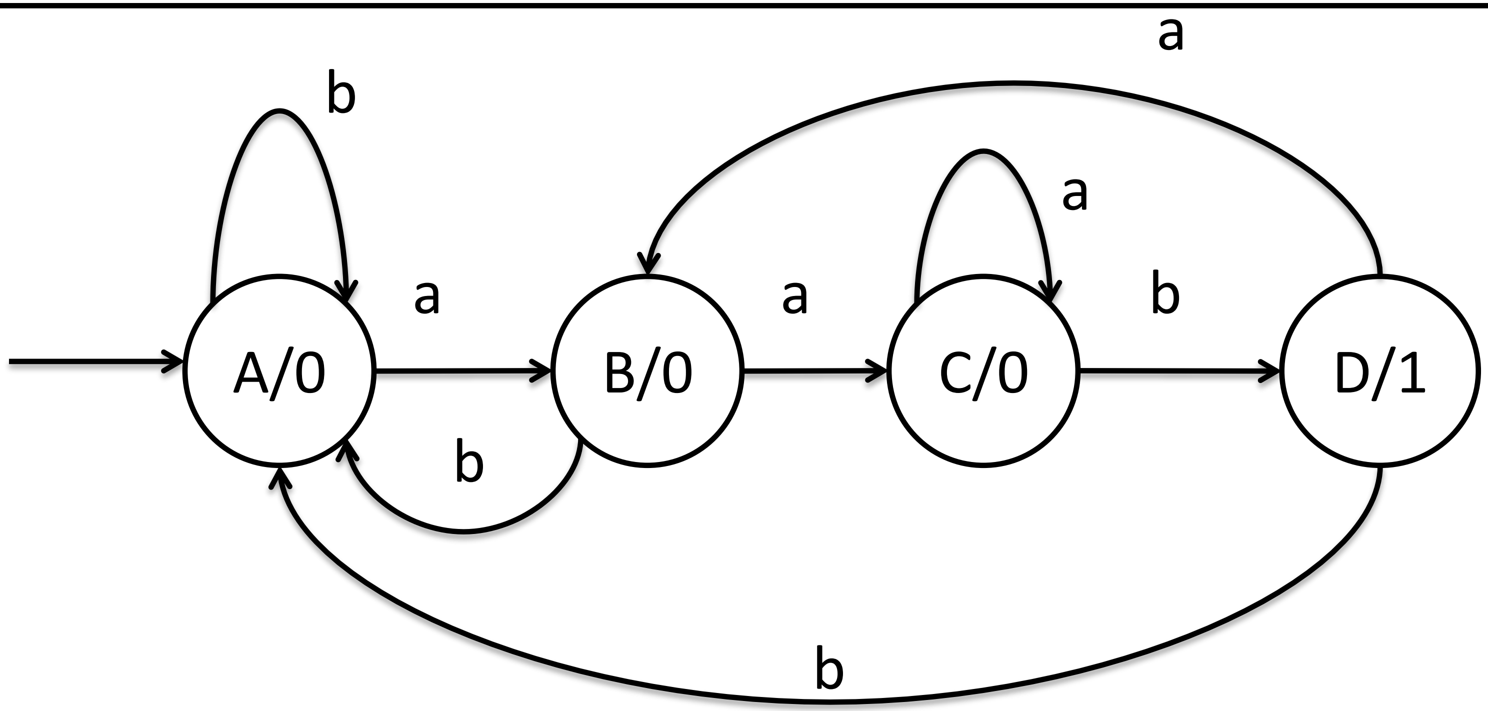
Construct a Mealy Machine that produces the 1's complement of any binary input string.



INPUT	OUTPUT
010101	101010
001001	110110
1001	0110

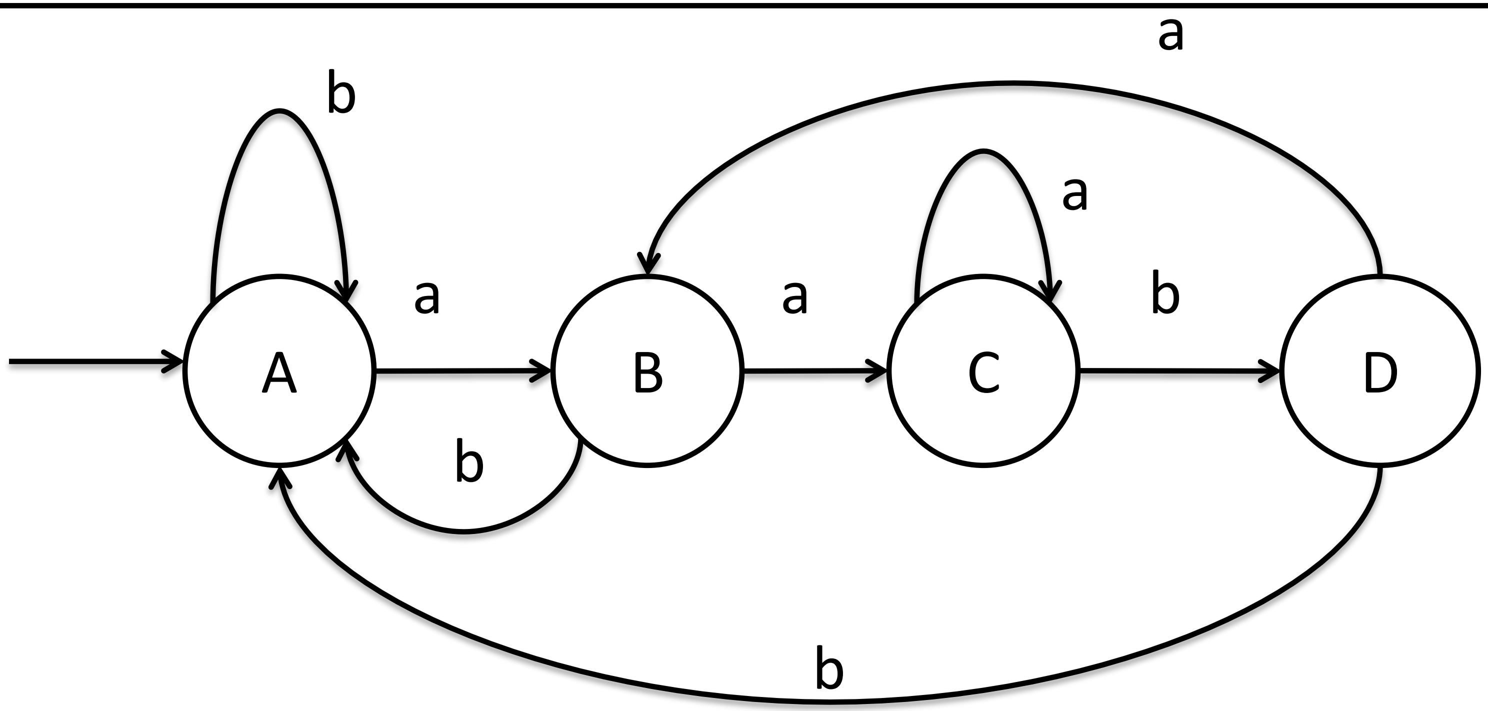
Construct a Mealy Machine that counts the occurrences of the sequence “aab” in any input strings over {a, b}

Construct a Mealy Machine that counts the occurrences of the sequence “aab” in any input strings over {a, b}



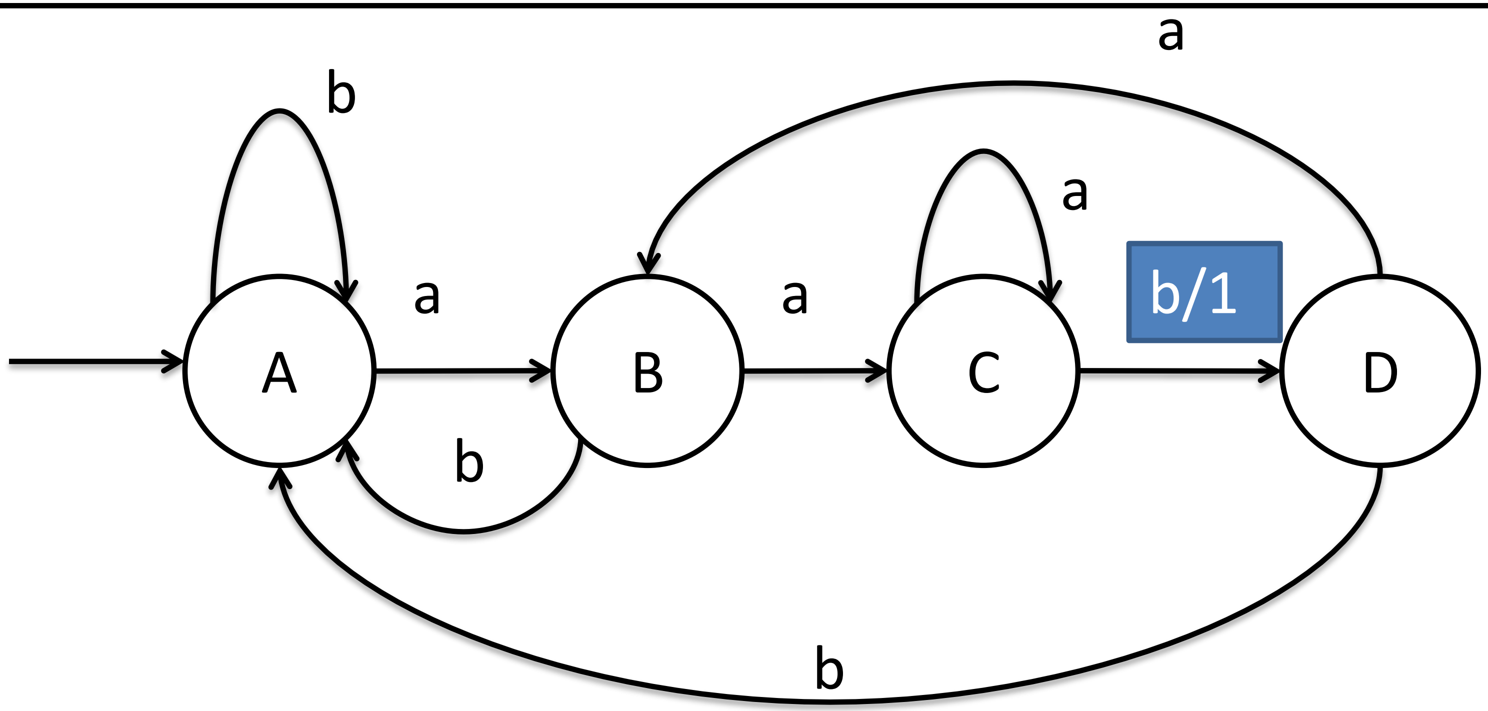
Moore Machine

Construct a Mealy Machine that counts the occurrences of the sequence “aab” in any input strings over {a, b}



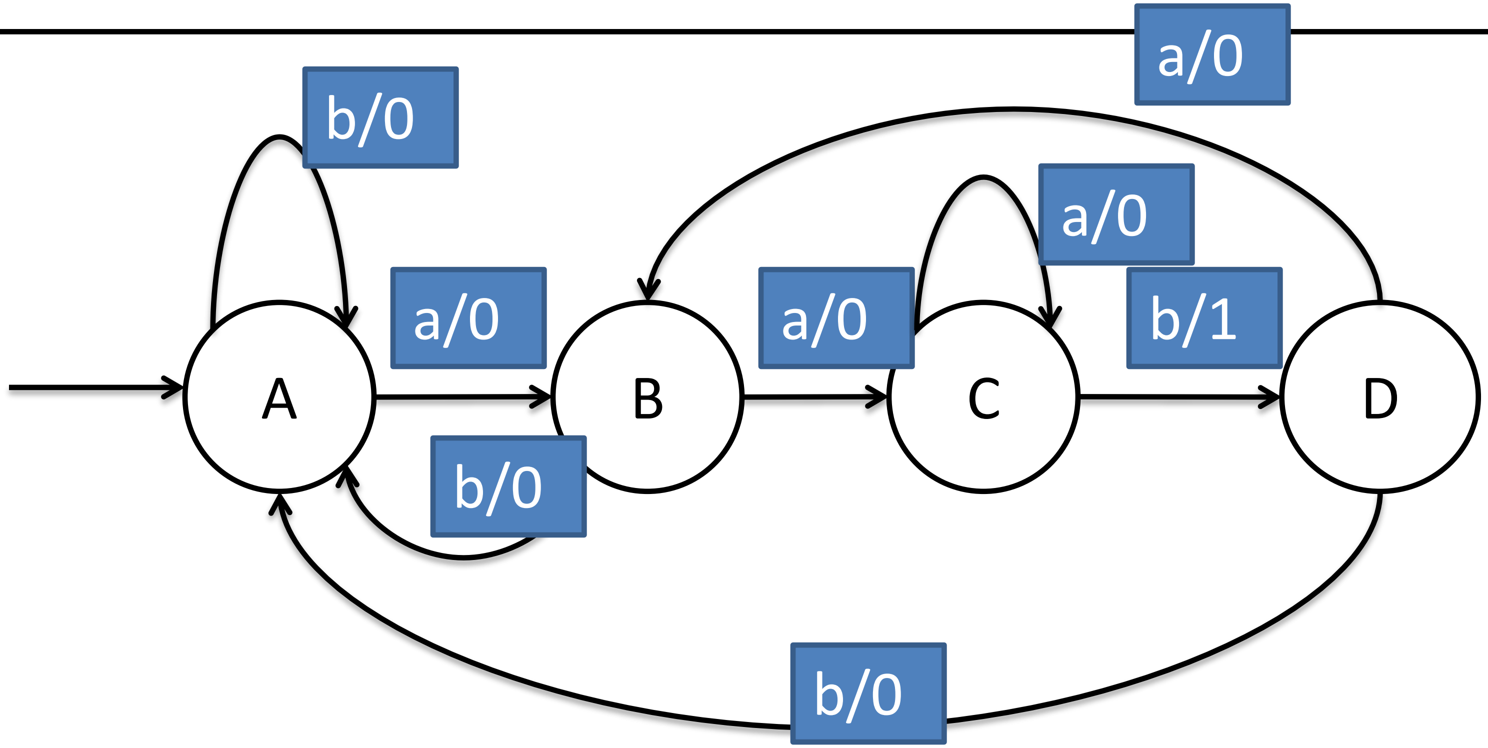
Moore Machine → Mealy Machine

Construct a Mealy Machine that counts the occurrences of the sequence “aab” in any input strings over {a, b}



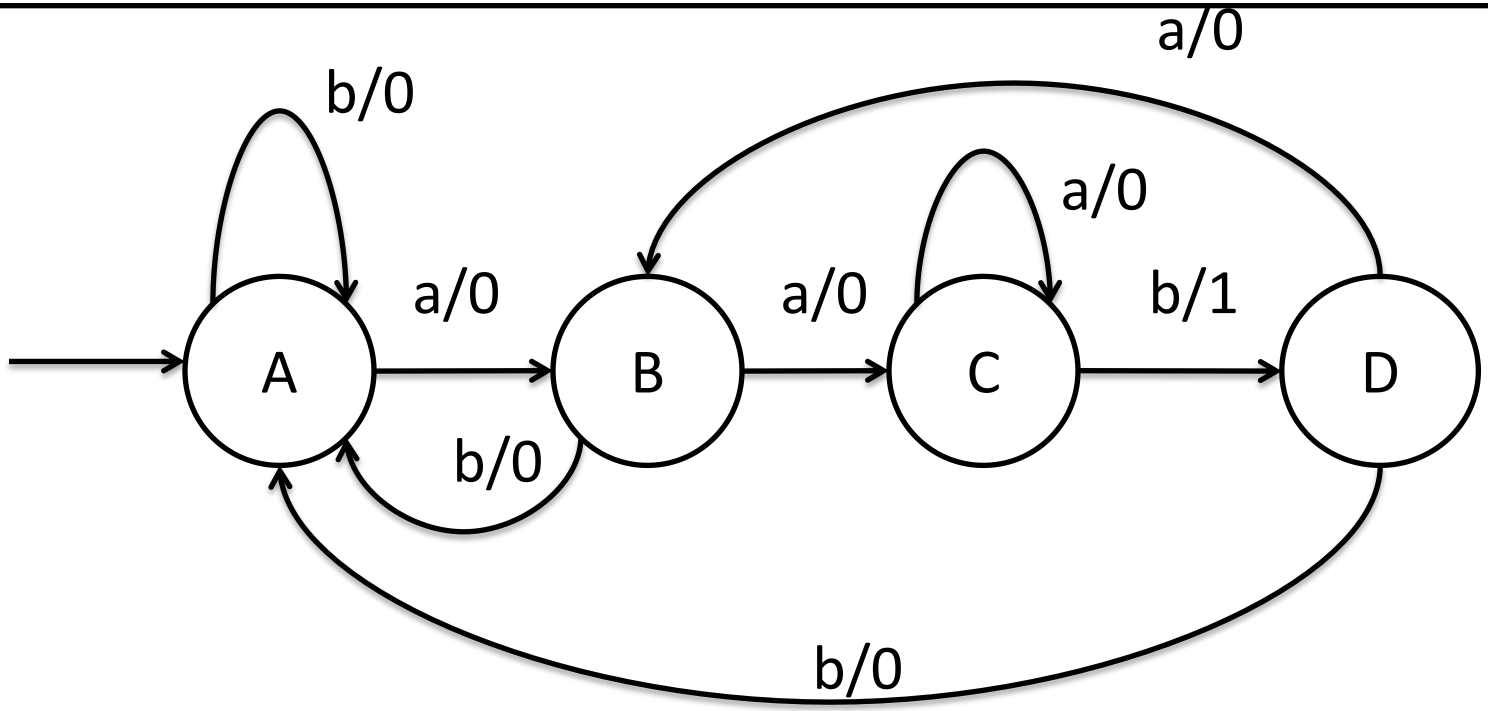
Moore Machine → Mealy Machine

Construct a Mealy Machine that counts the occurrences of the sequence “aab” in any input strings over {a, b}



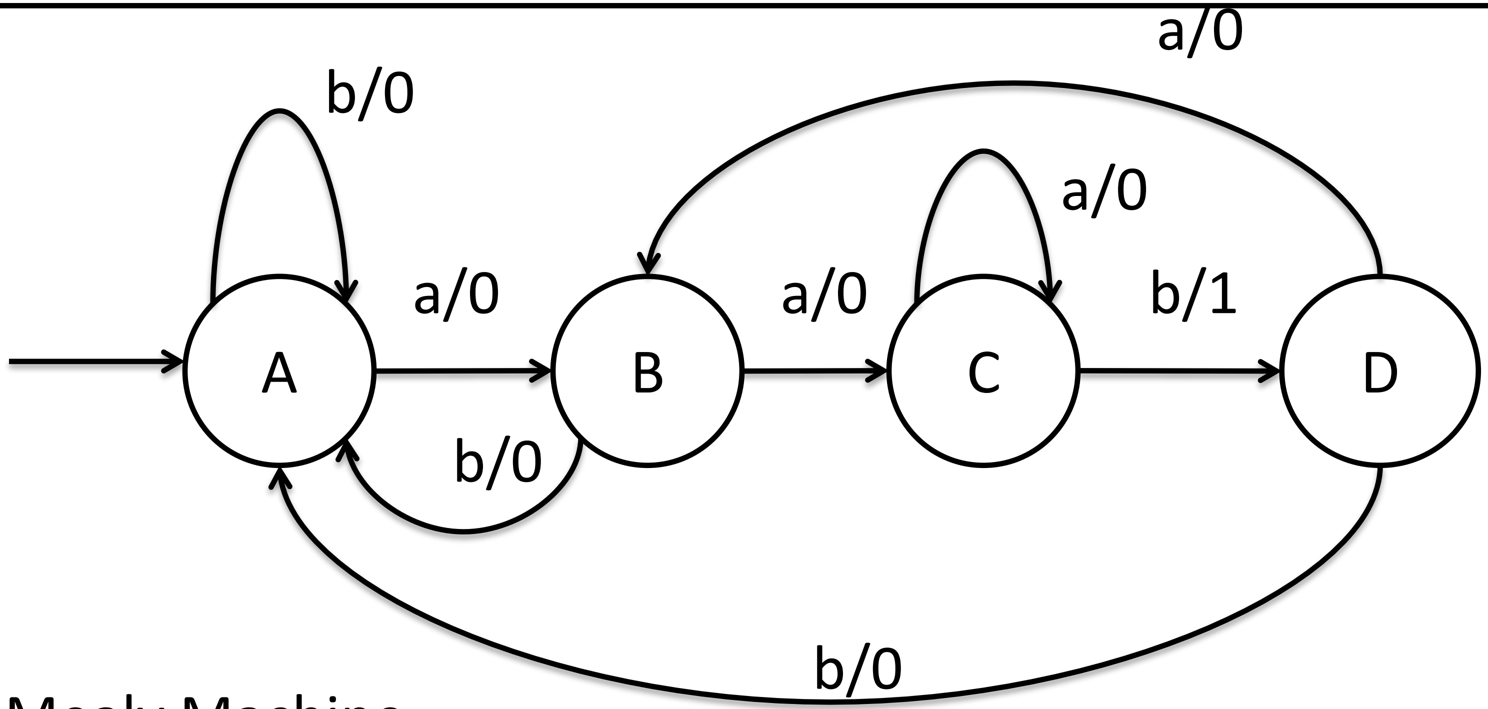
Moore Machine → Mealy Machine

Construct a Mealy Machine that counts the occurrences of the sequence “aab” in any input strings over {a, b}



Mealy Machine

Construct a Mealy Machine that counts the occurrences of the sequence "aab" in any input strings over {a, b}



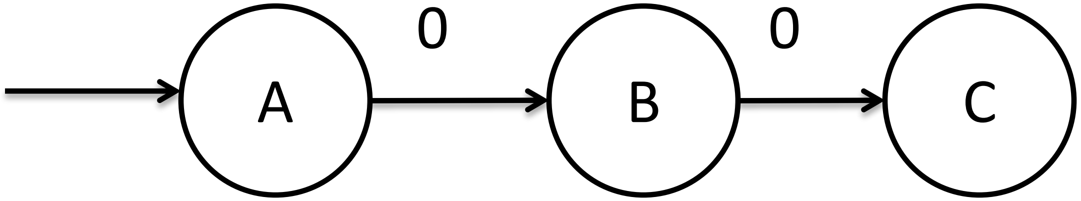
Mealy Machine

INPUT	OUTPUT
aab	001
aaab	0001
aababaab	00100001

Construct a Mealy Machine that counts the occurrences of the sequence 01*0 in any input strings over {0,1}

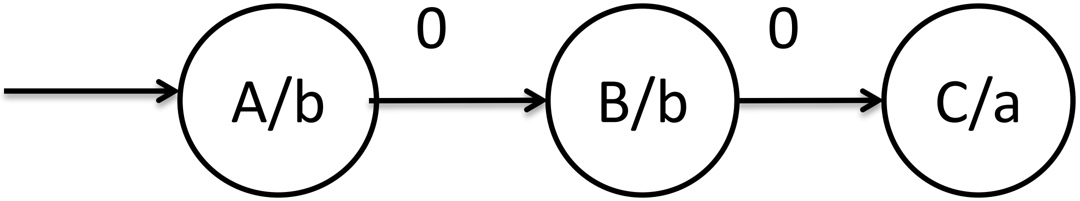
Construct a Mealy Machine that counts the occurrences of the sequence 01*0 in any input strings over {0,1}

First, constructing Moore machine. Taking 3 states for smallest input 00



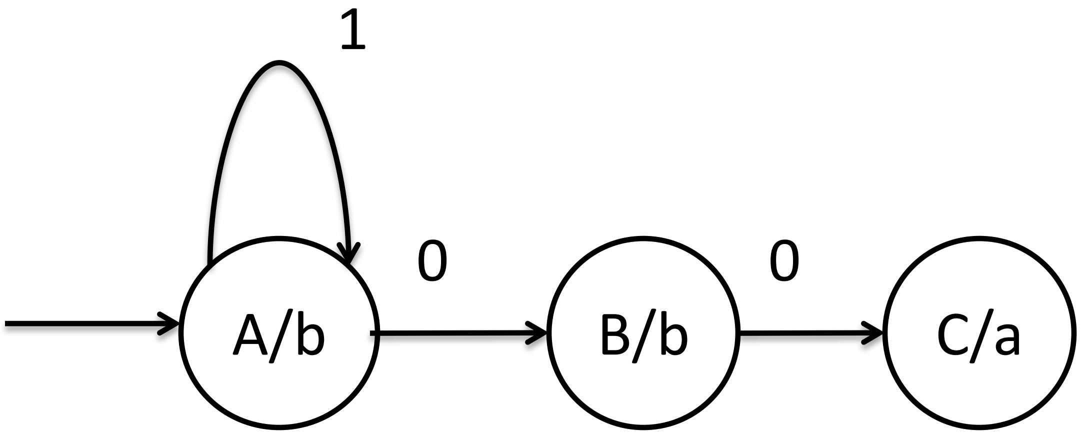
Construct a Mealy Machine that counts the occurrences of the sequence 01*0 in any input strings over {0,1}

Constructing Moore machine.



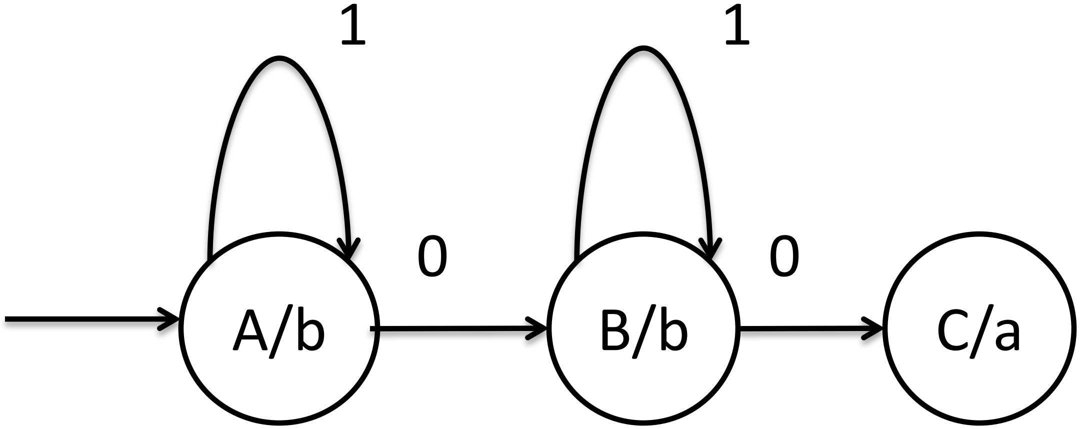
Construct a Mealy Machine that counts the occurrences of the sequence 01*0 in any input strings over {0,1}

Constructing Moore machine. Completing remaining edges.



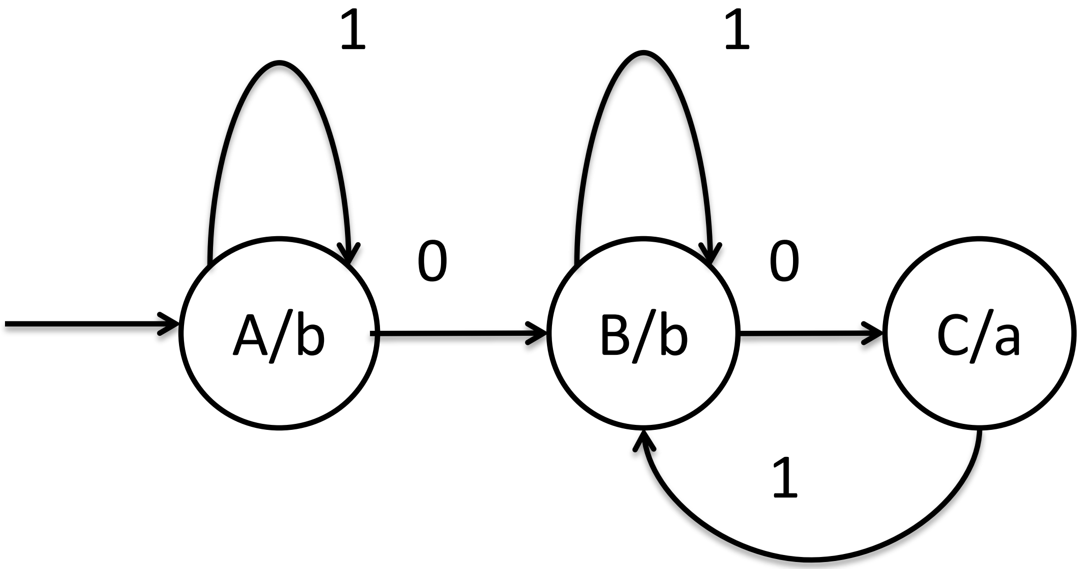
Construct a Mealy Machine that counts the occurrences of the sequence 01*0 in any input strings over {0,1}

Constructing Moore machine. Completing remaining edges.



Construct a Mealy Machine that counts the occurrences of the sequence 01*0 in any input strings over {0,1}

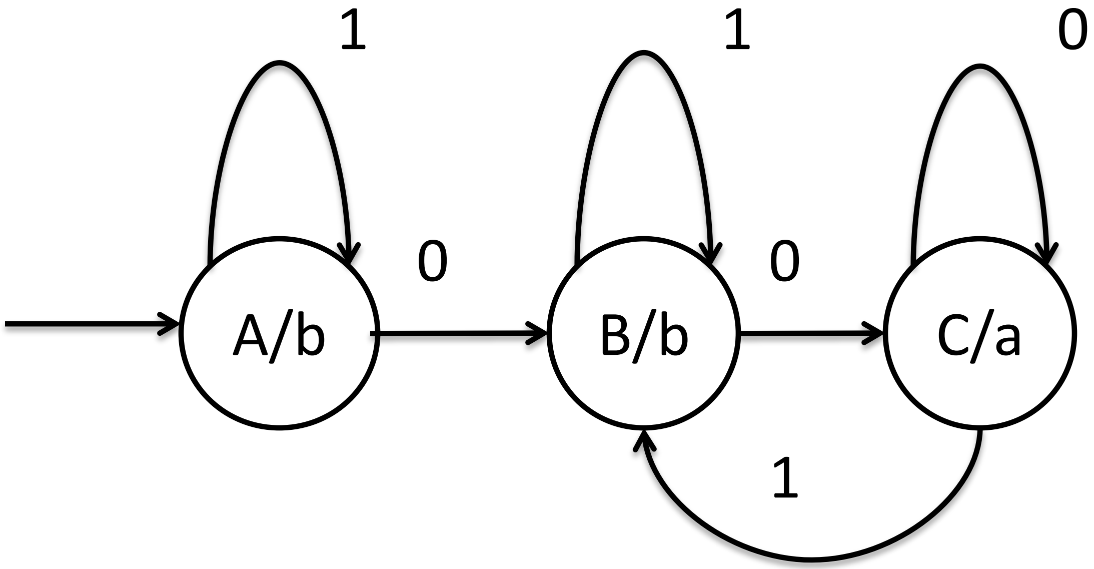
Constructing Moore machine. Completing remaining edges.



Assuming overlapping is allowed.

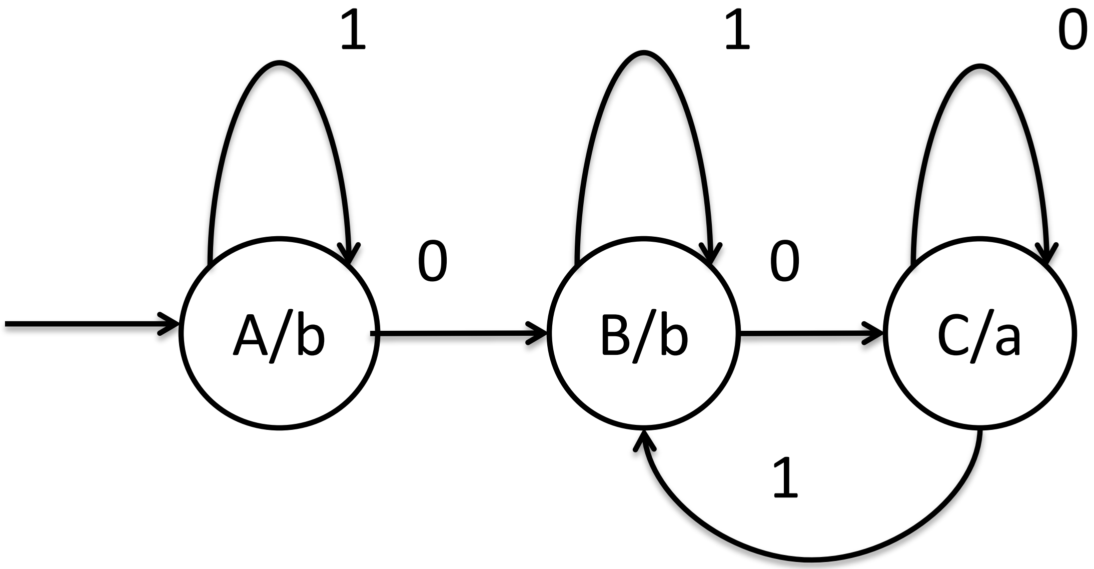
Construct a Mealy Machine that counts the occurrences of the sequence 01*0 in any input strings over {0,1}

Constructing Moore machine. Completing remaining edges.



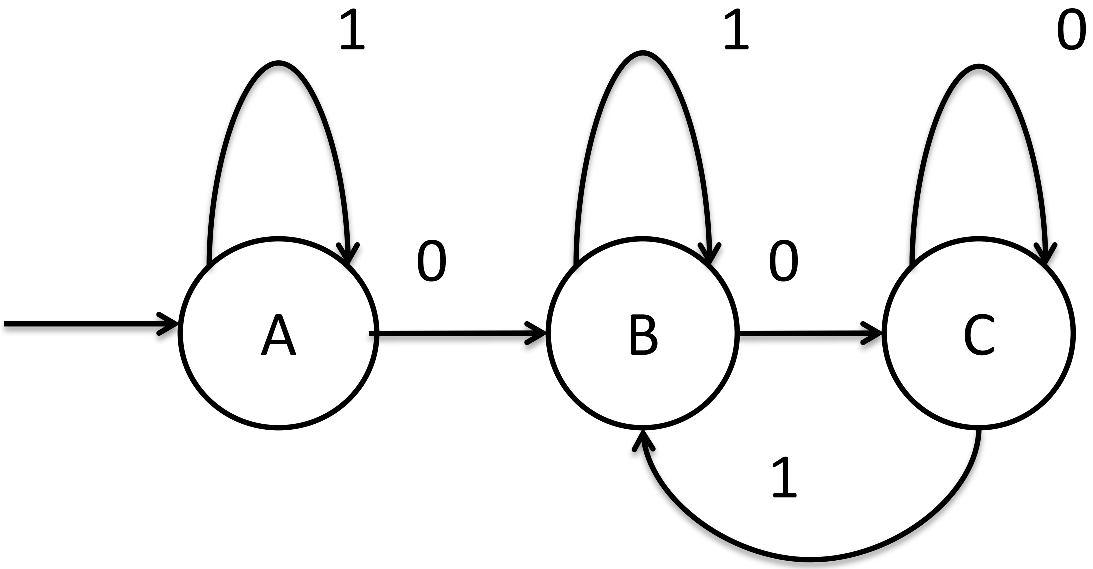
Construct a Mealy Machine that counts the occurrences of the sequence 01*0 in any input strings over {0,1}

Moore machine



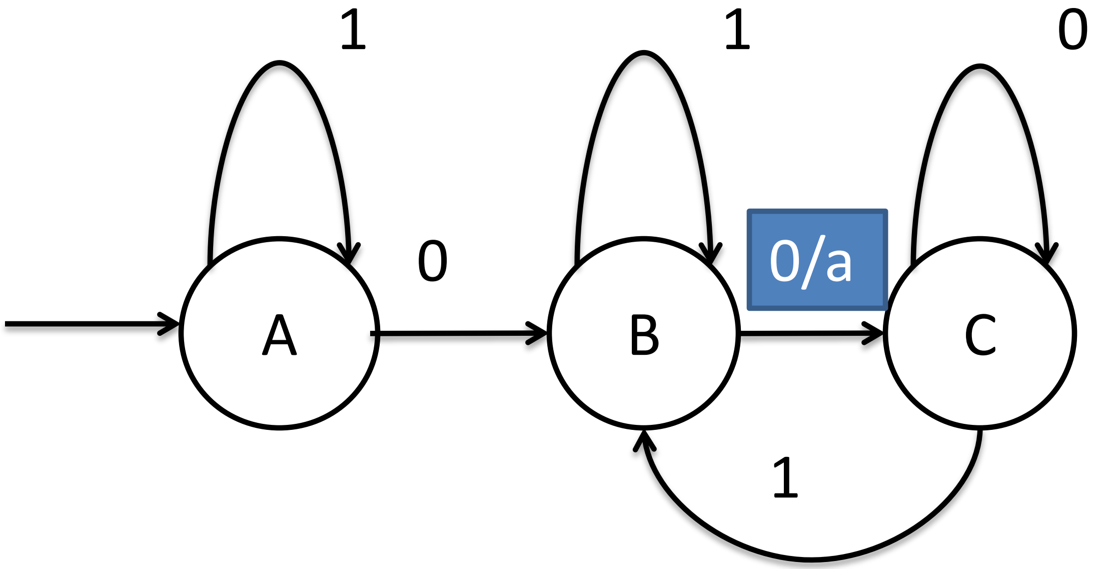
Construct a Mealy Machine that counts the occurrences of the sequence 01*0 in any input strings over {0,1}

Moore machine → Mealy Machine



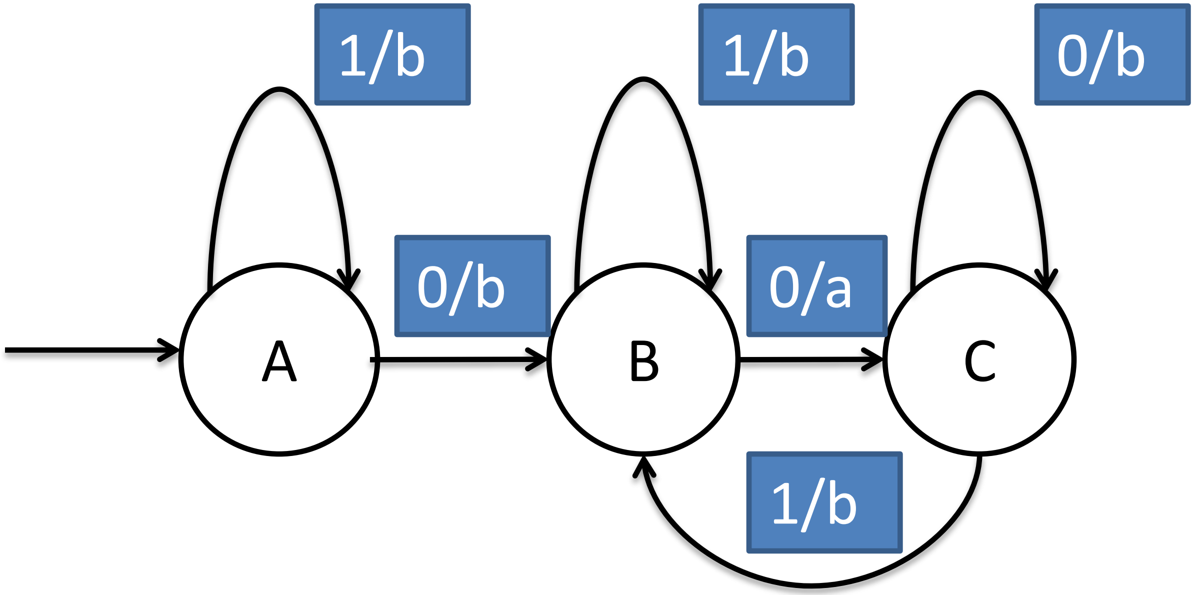
Construct a Mealy Machine that counts the occurrences of the sequence 01*0 in any input strings over {0,1}

Moore machine → Mealy Machine



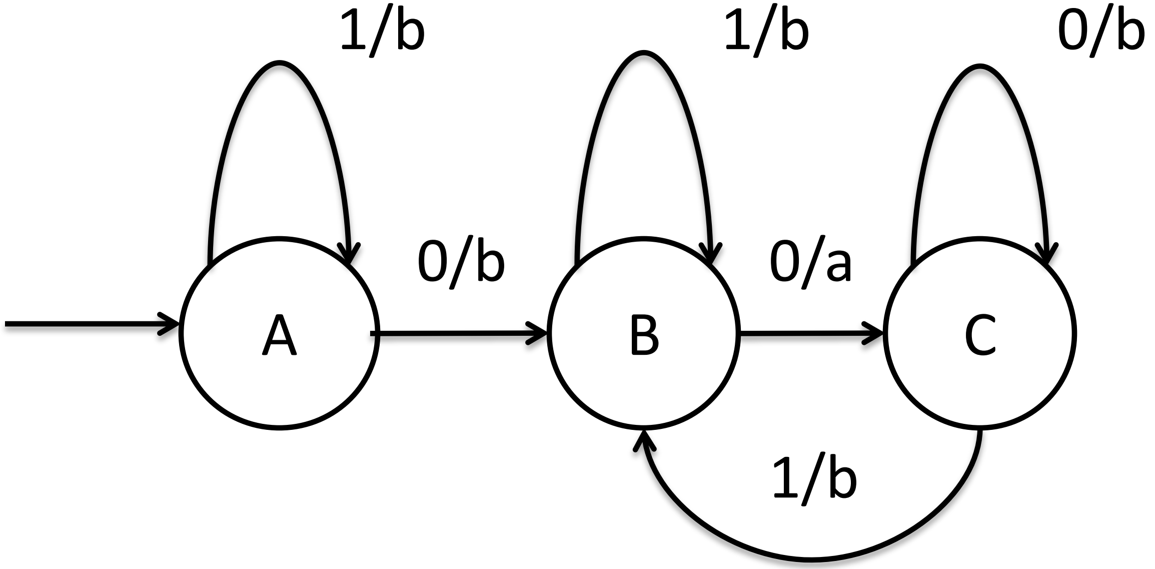
Construct a Mealy Machine that counts the occurrences of the sequence 01*0 in any input strings over {0,1}

Moore machine → Mealy Machine



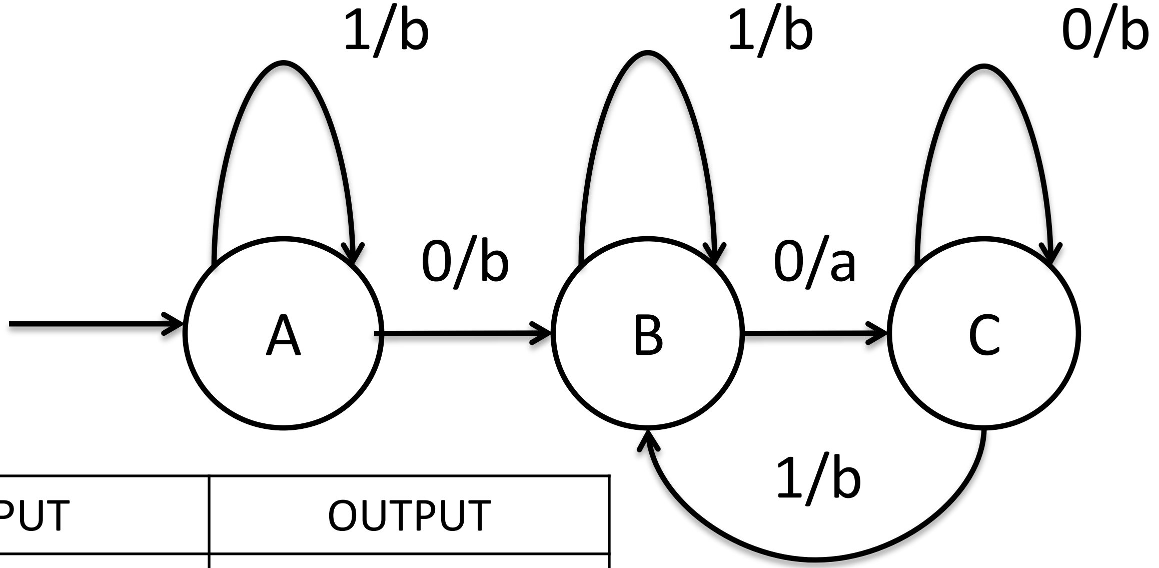
Construct a Mealy Machine that counts the occurrences of the sequence 01*0 in any input strings over {0,1}

Moore machine → Mealy Machine



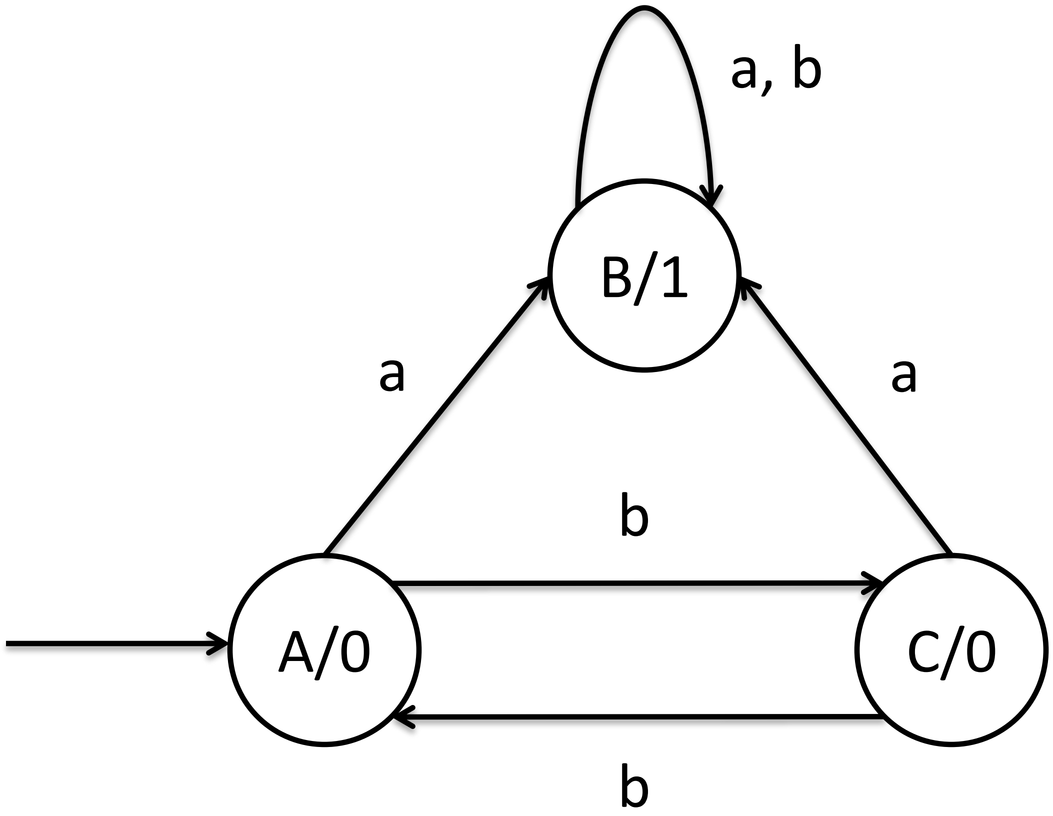
Construct a Mealy Machine that counts the occurrences of the sequence 01*0 in any input strings over {0,1}

Mealy Machine



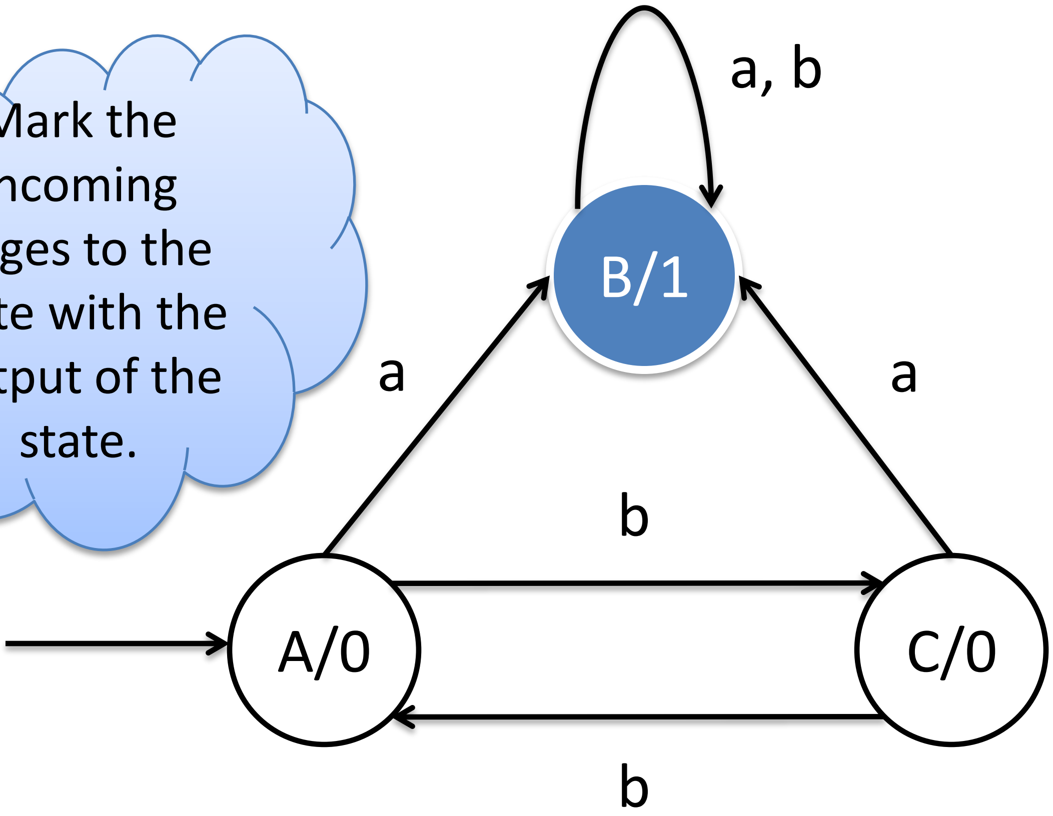
INPUT	OUTPUT
0110	bbba
001100	babbab
0011100	babbbab
0101010	bbababa
101010	bbbaba

Convert Moore Machine to Mealy Machine



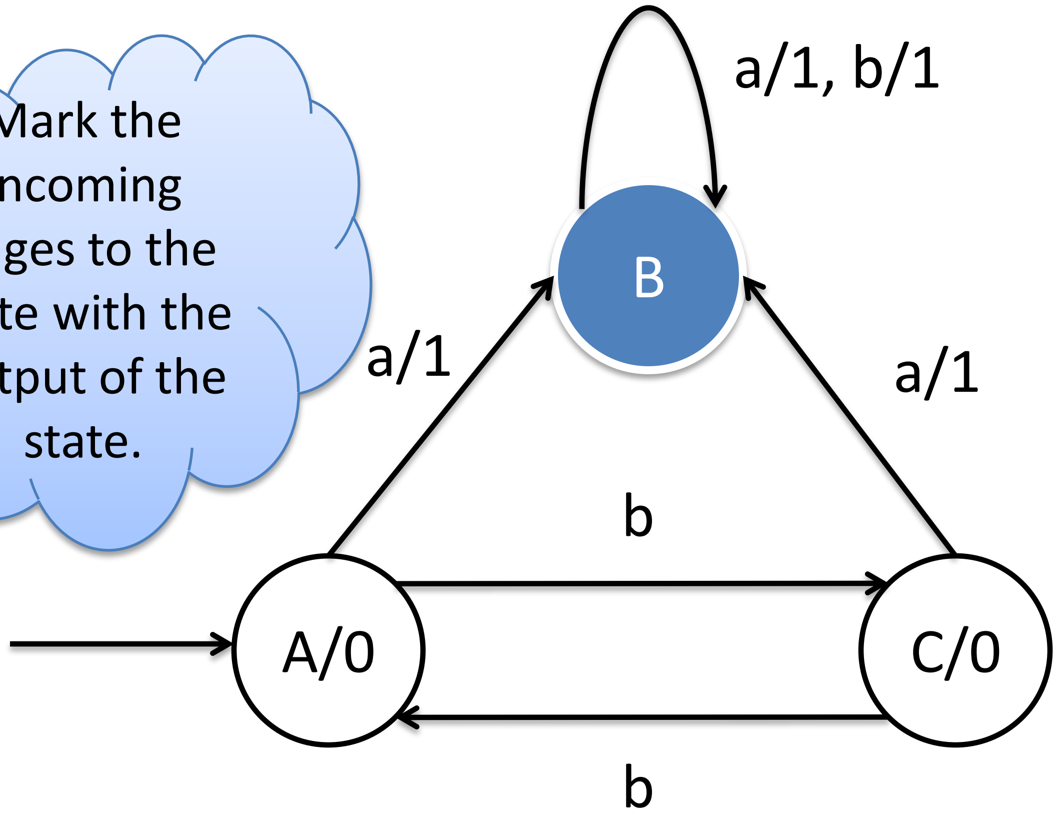
Convert Moore Machine to Mealy Machine

Mark the incoming edges to the state with the output of the state.

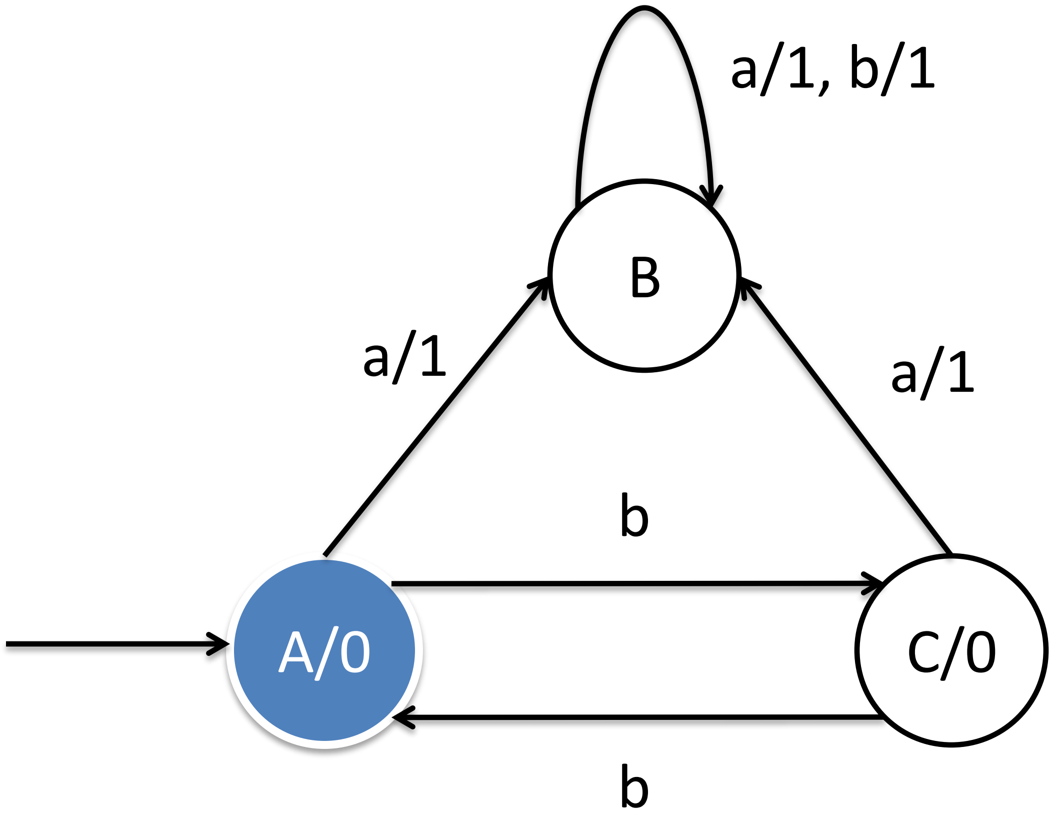


Convert Moore Machine to Mealy Machine

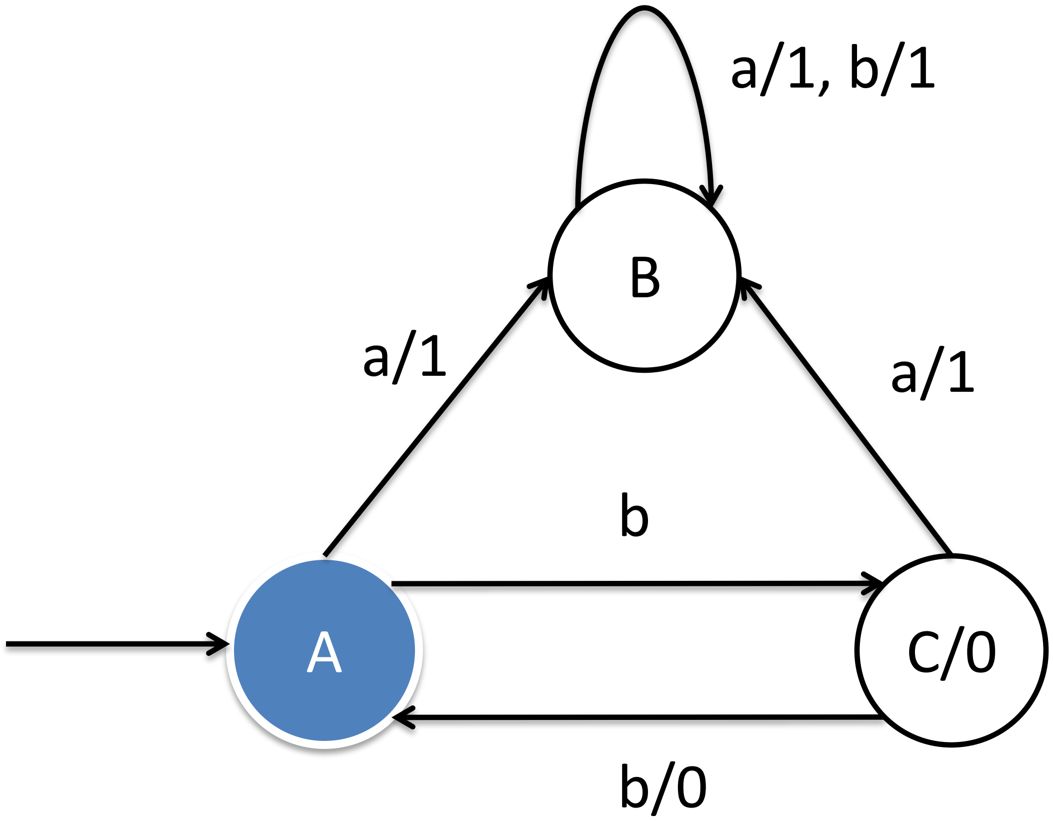
Mark the incoming edges to the state with the output of the state.



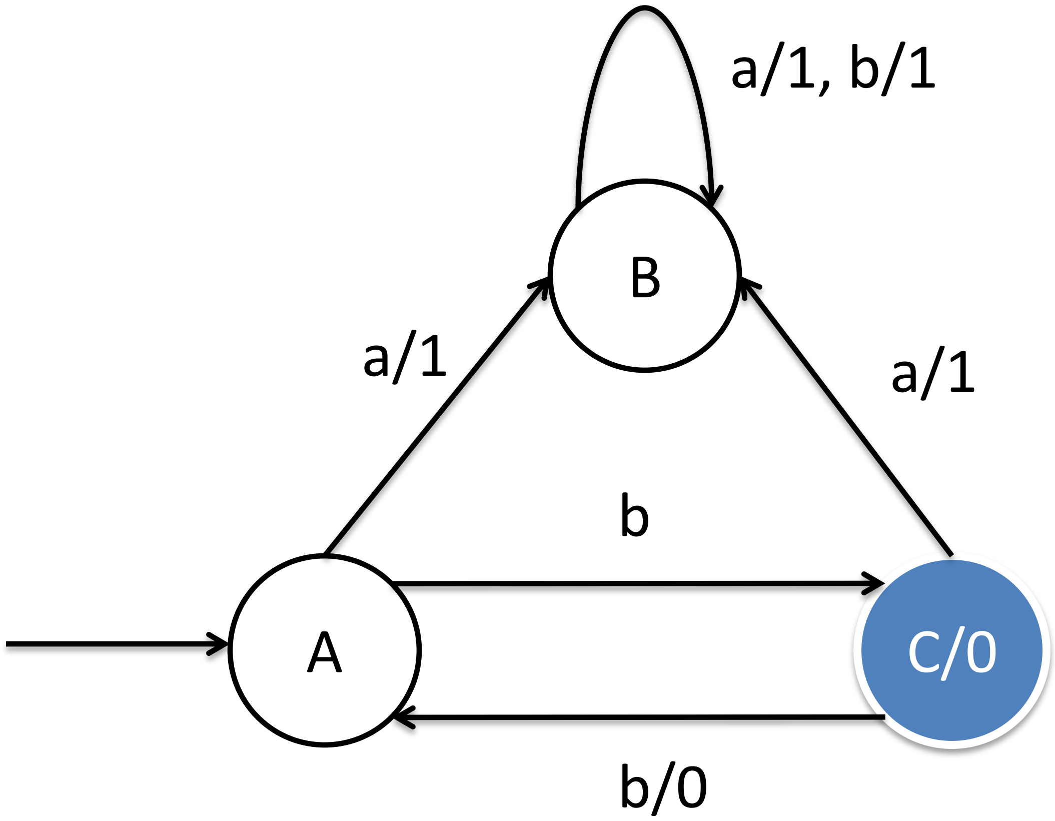
Convert Moore Machine to Mealy Machine



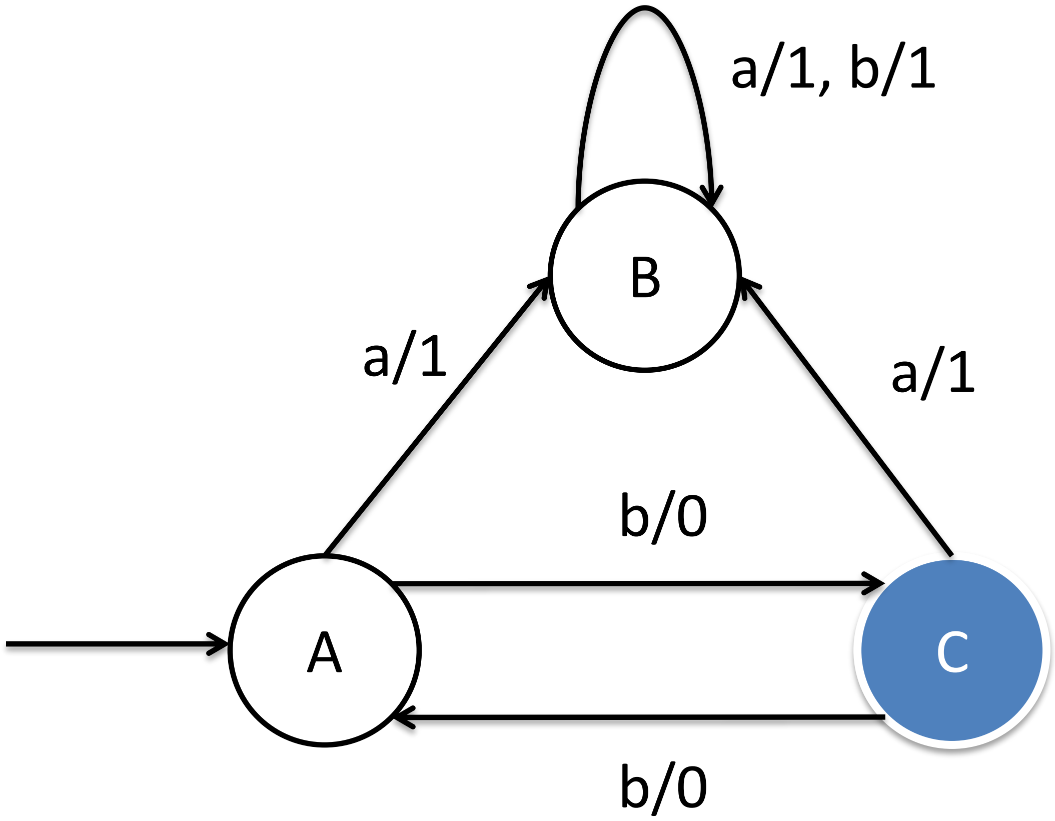
Convert Moore Machine to Mealy Machine



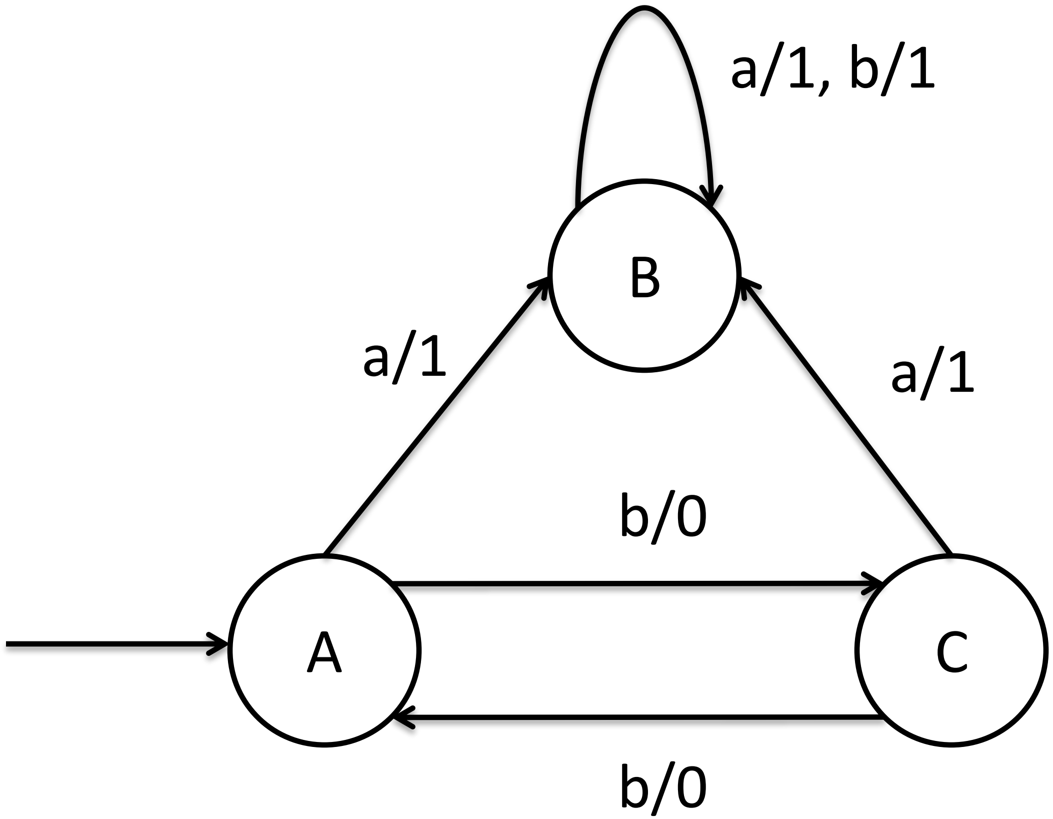
Convert Moore Machine to Mealy Machine



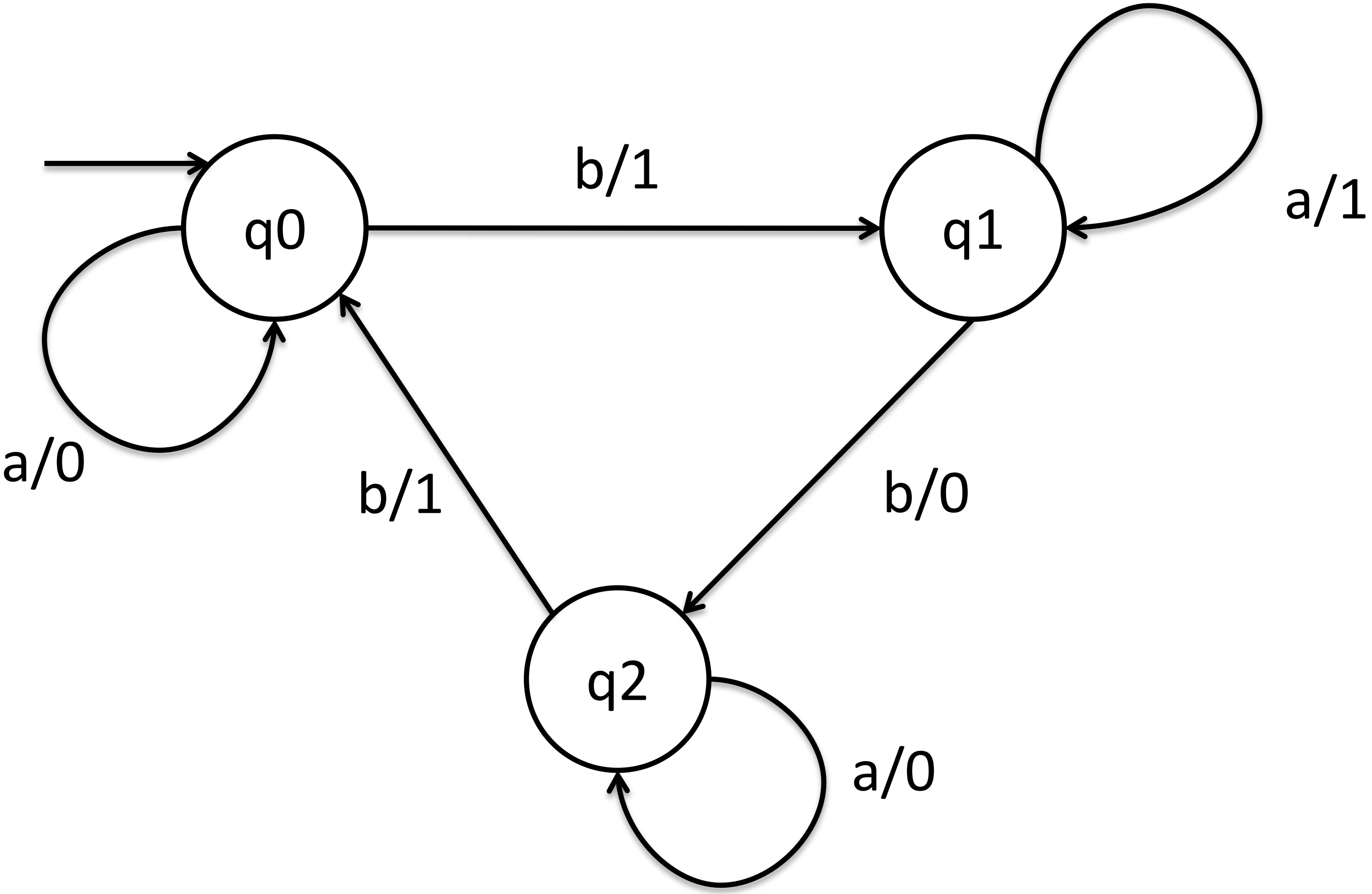
Convert Moore Machine to Mealy Machine



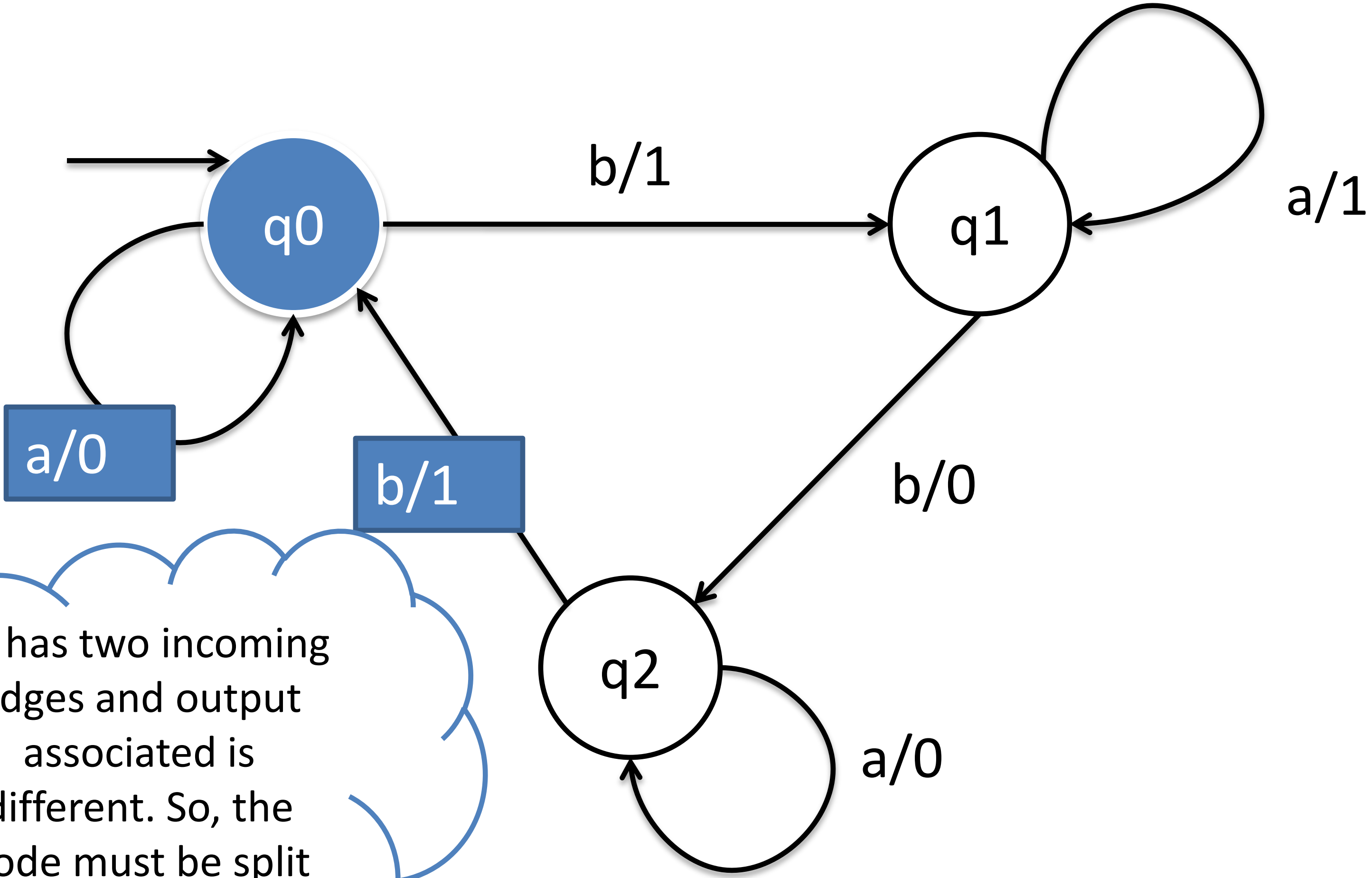
Convert Moore Machine to Mealy Machine



Convert Mealy Machine to Moore Machine

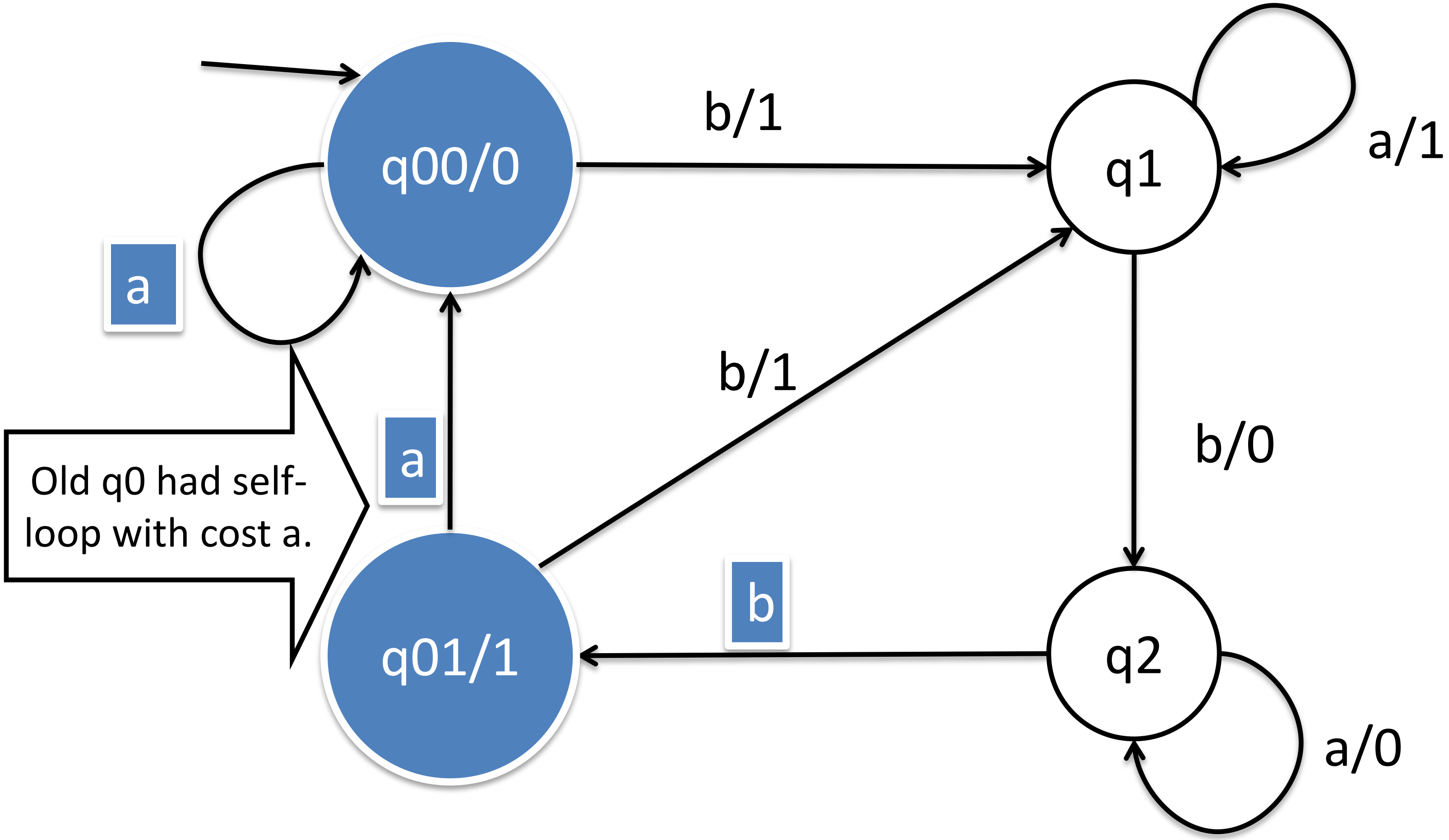


Convert Mealy Machine to Moore Machine



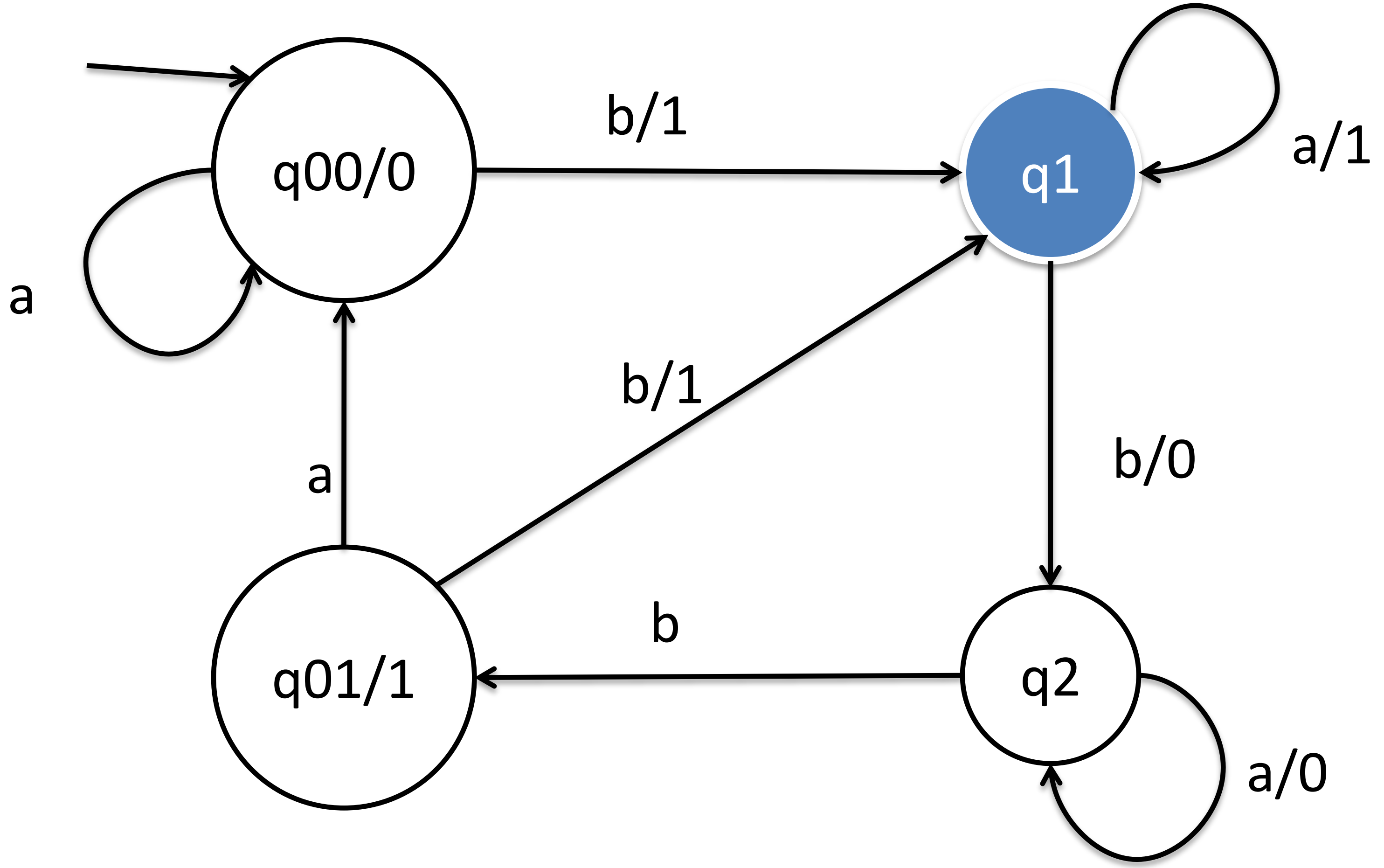
q0 has two incoming edges and output associated is different. So, the node must be split into two nodes.

Convert Mealy Machine to Moore Machine

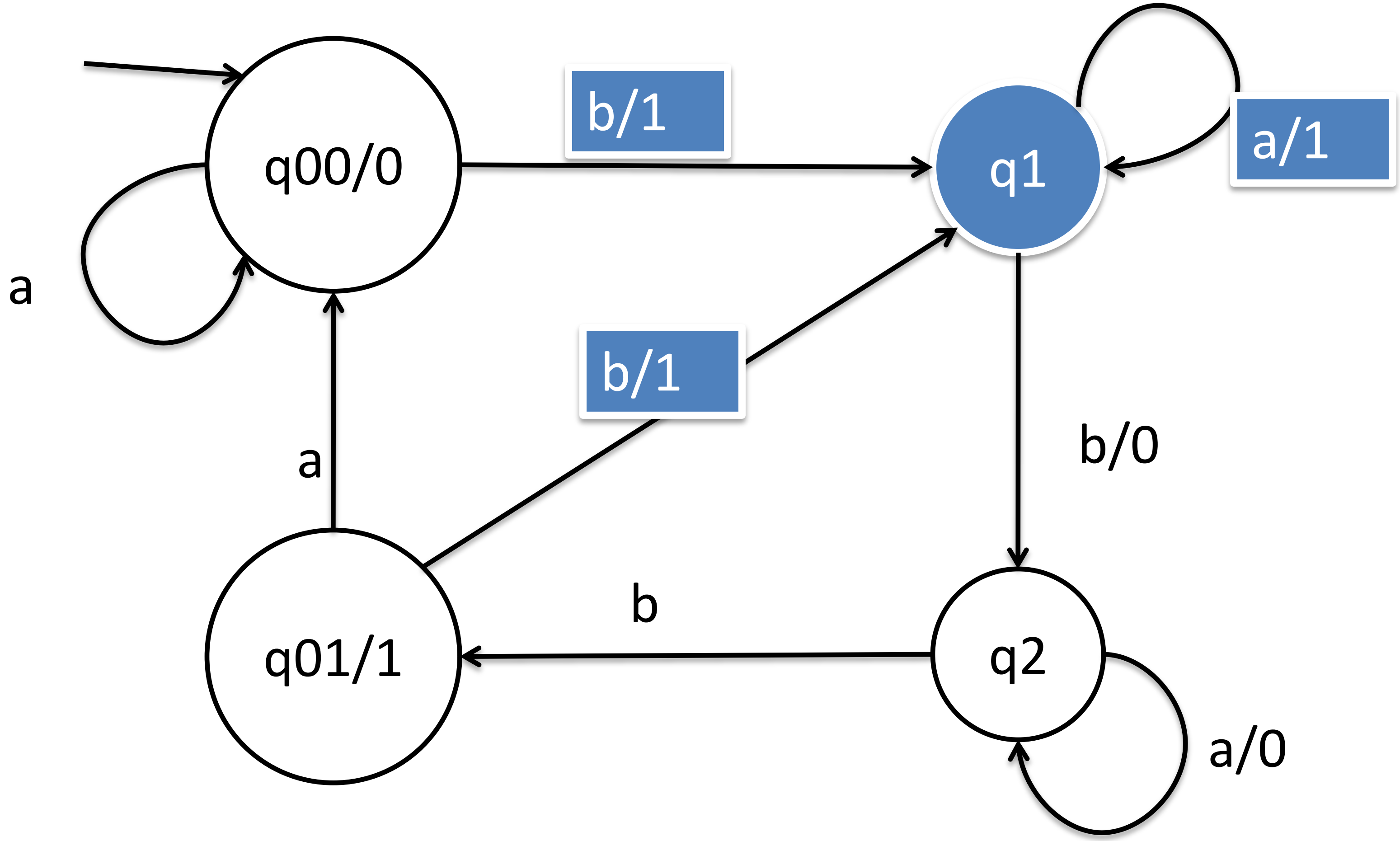


Old q0 had self-loop with cost a.

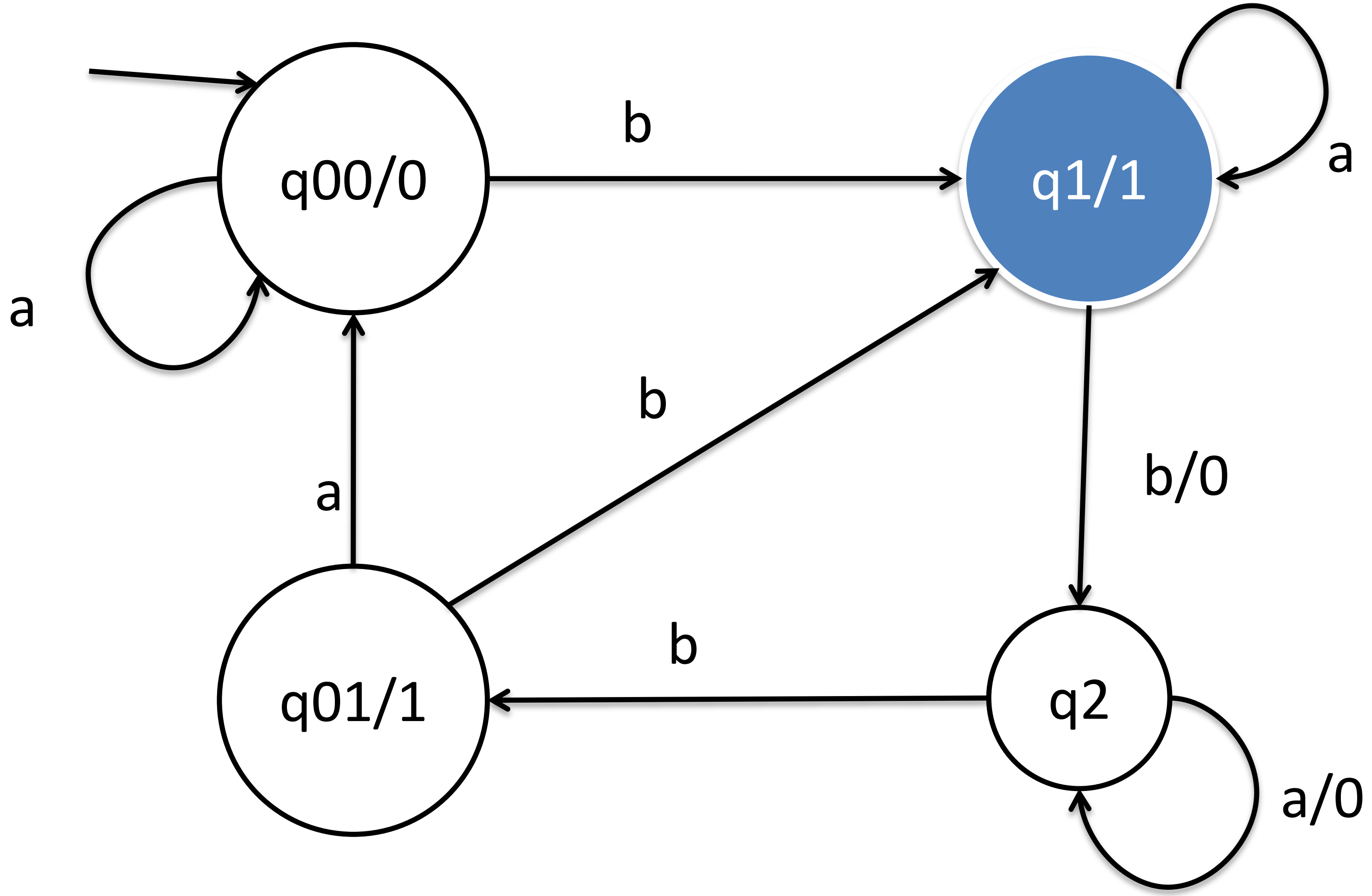
Convert Mealy Machine to Moore Machine



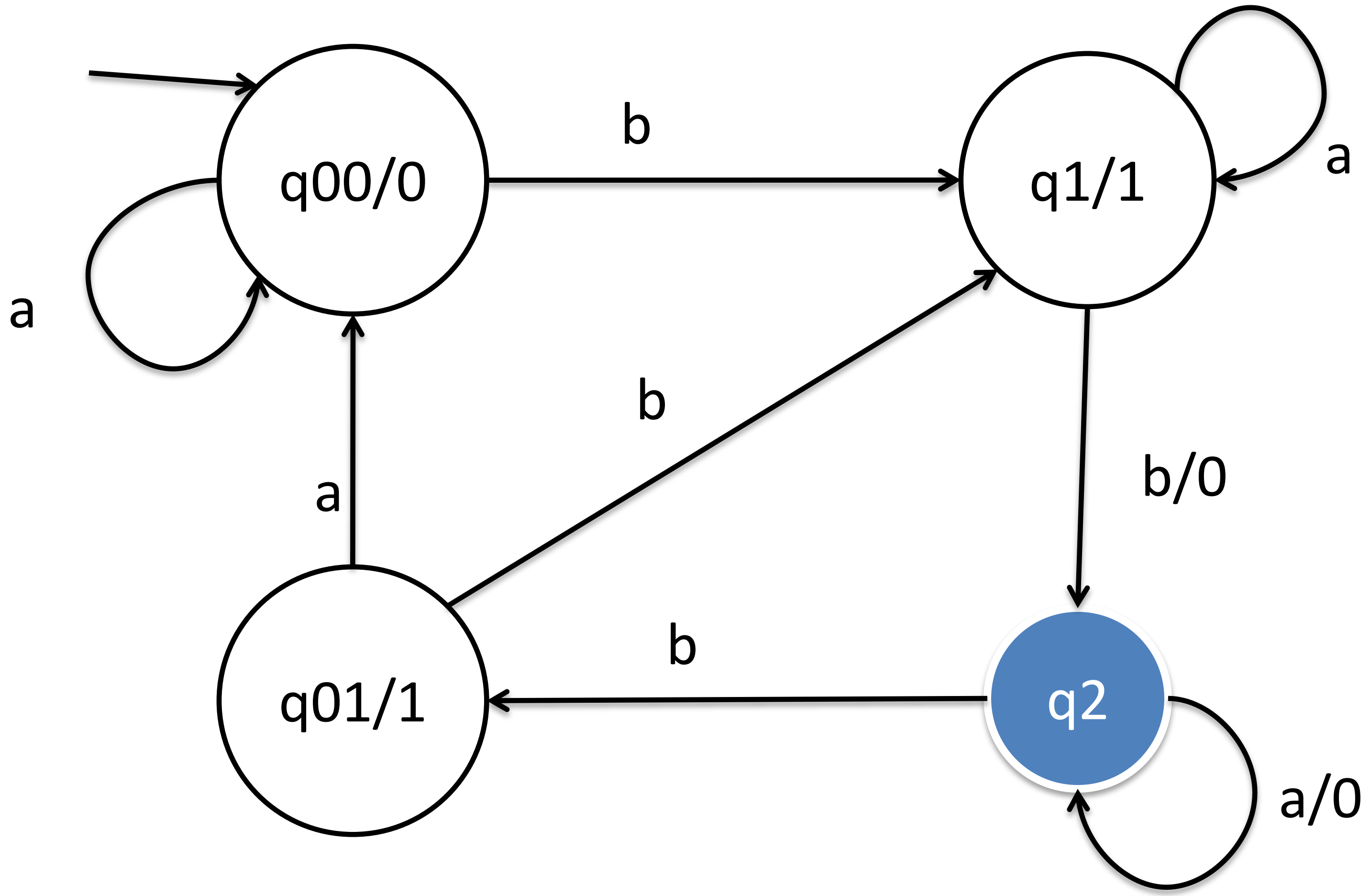
Convert Mealy Machine to Moore Machine



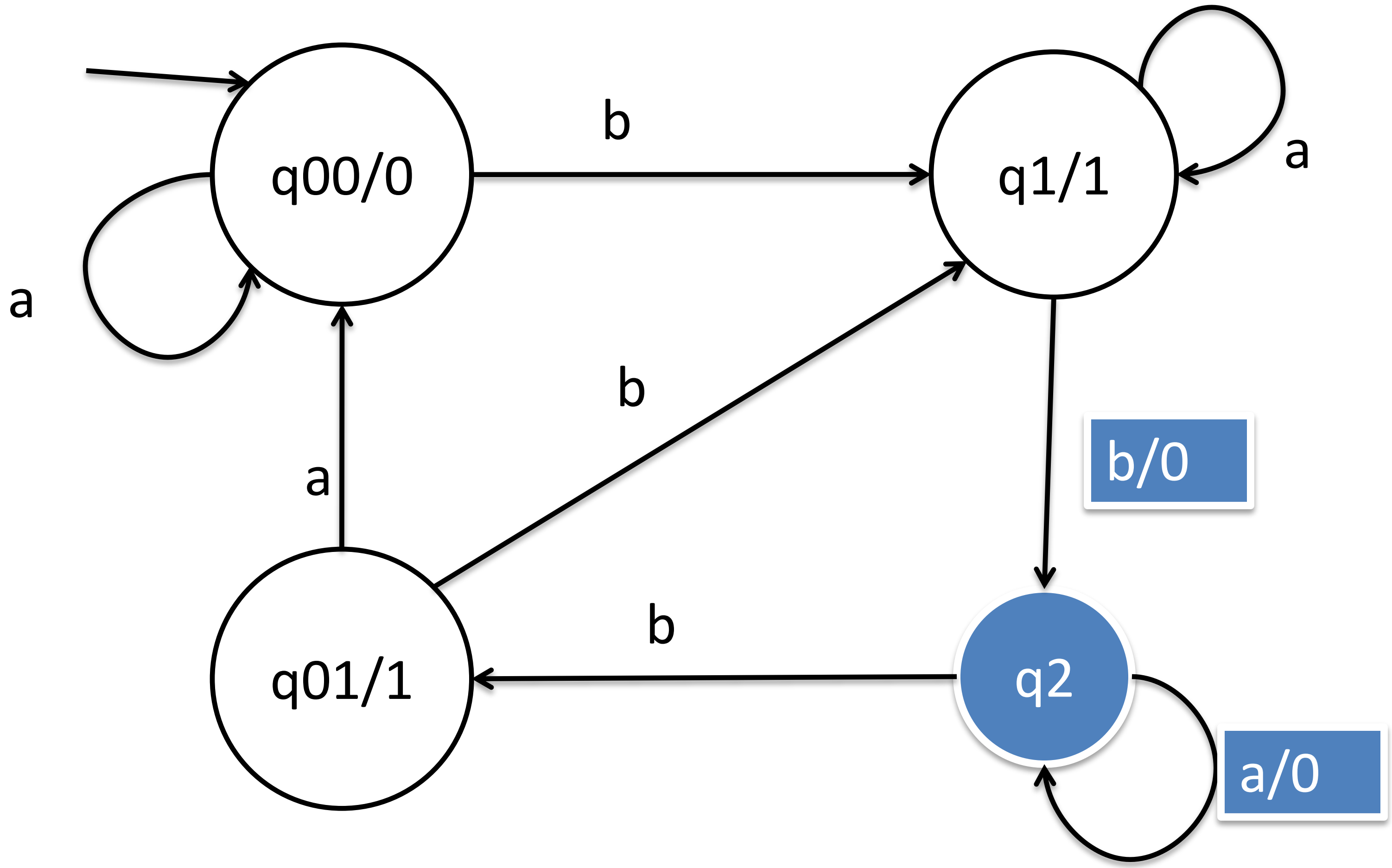
Convert Mealy Machine to Moore Machine



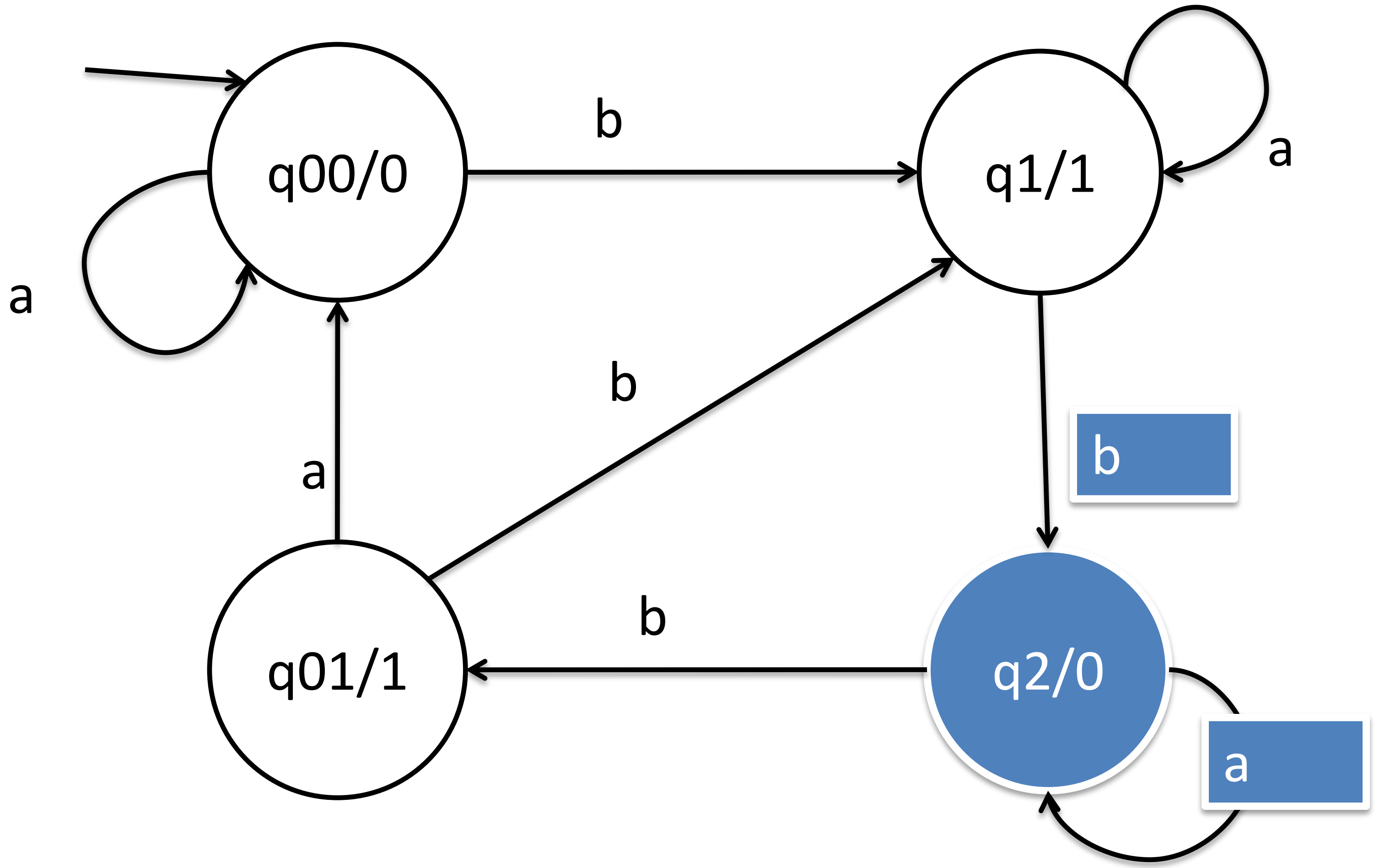
Convert Mealy Machine to Moore Machine



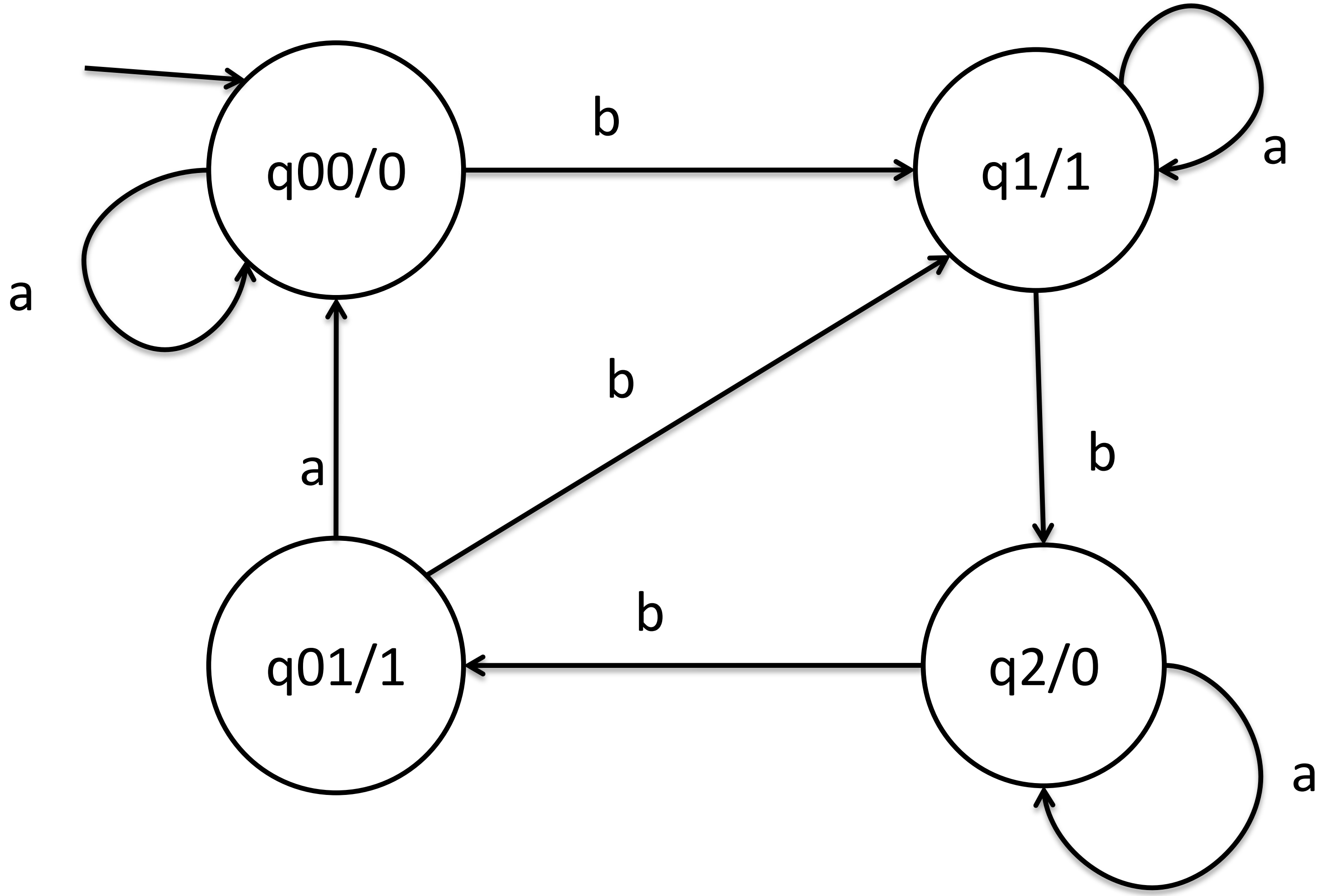
Convert Mealy Machine to Moore Machine



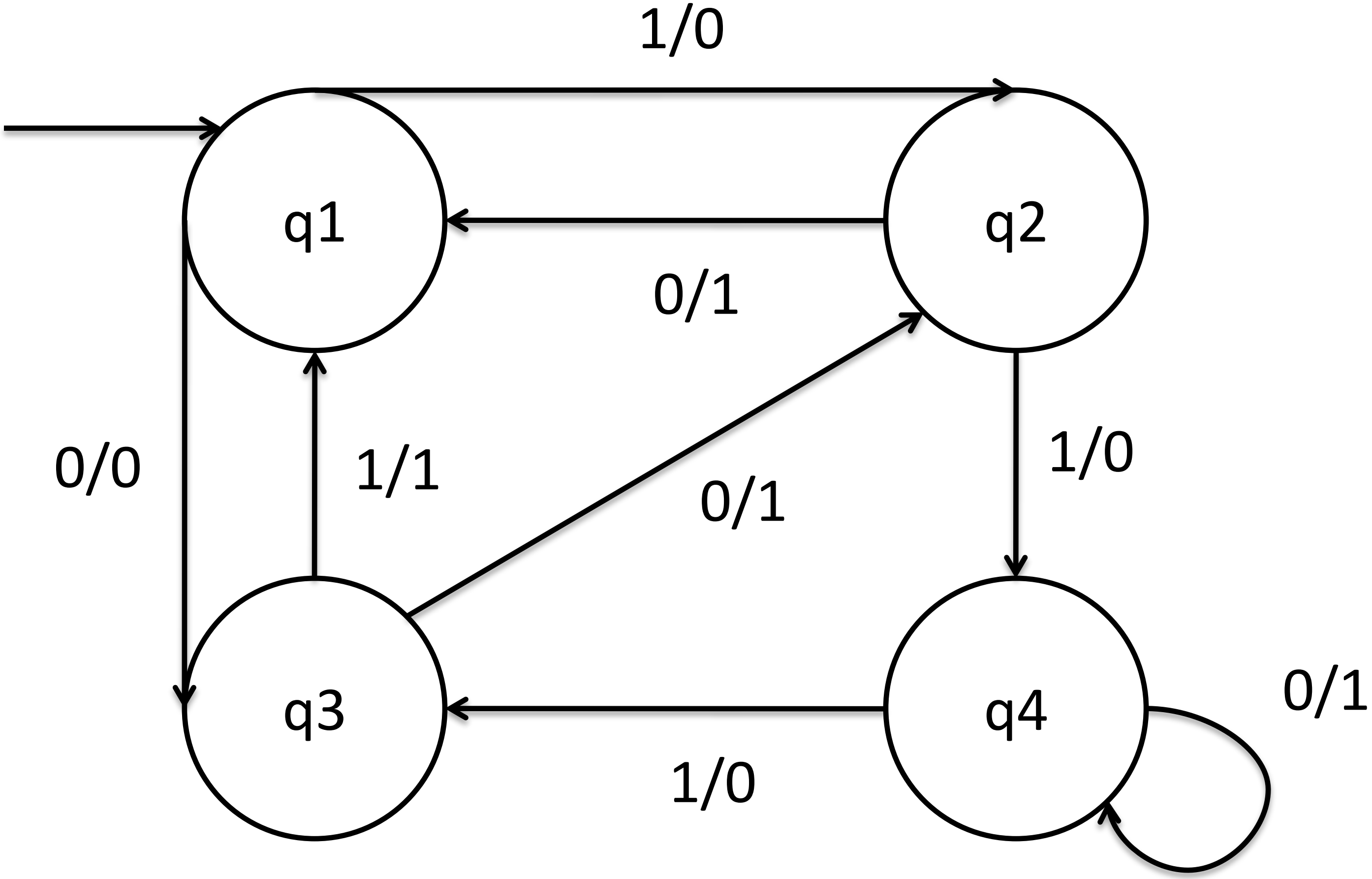
Convert Mealy Machine to Moore Machine



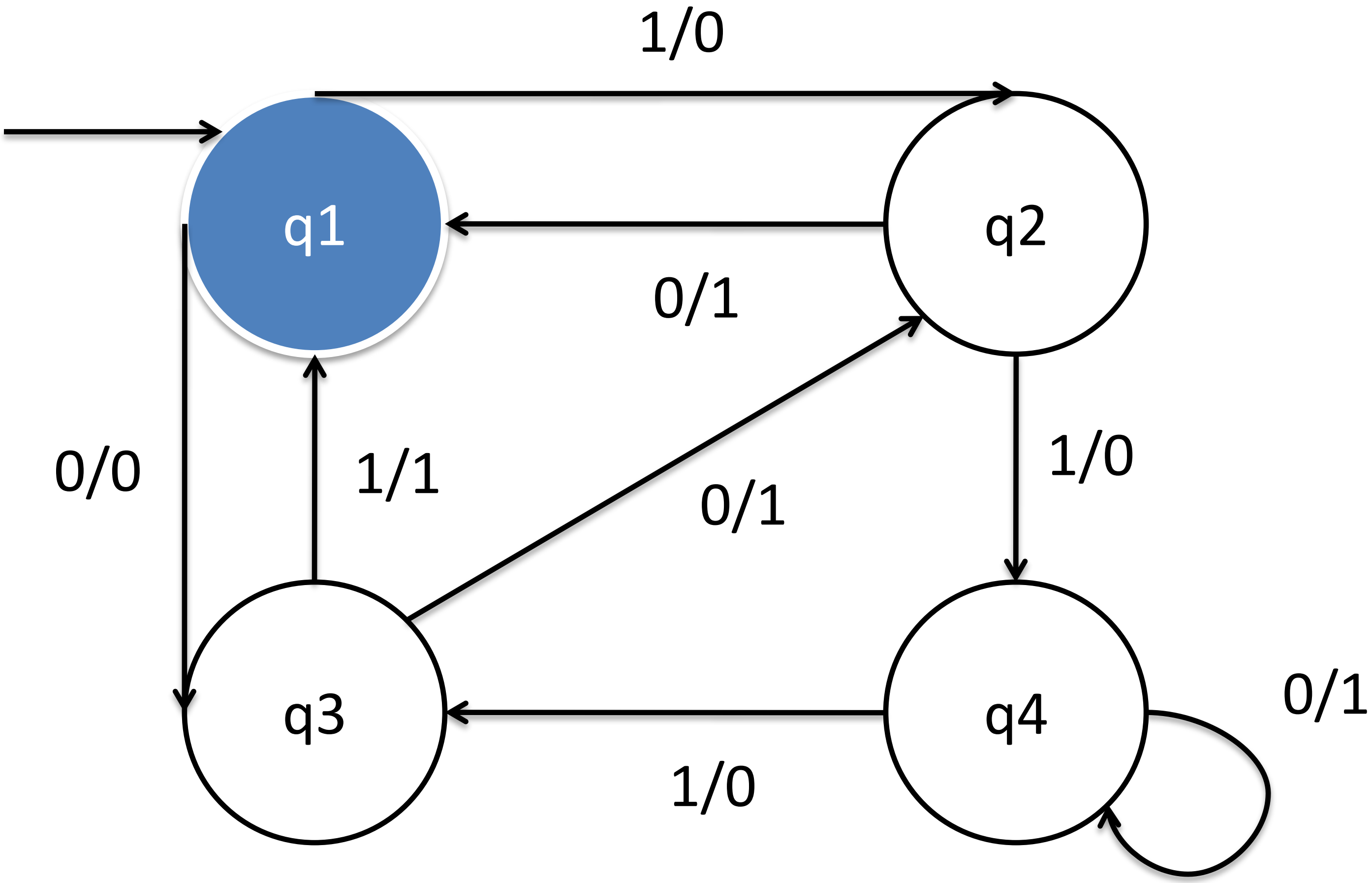
Convert Mealy Machine to **Moore Machine**



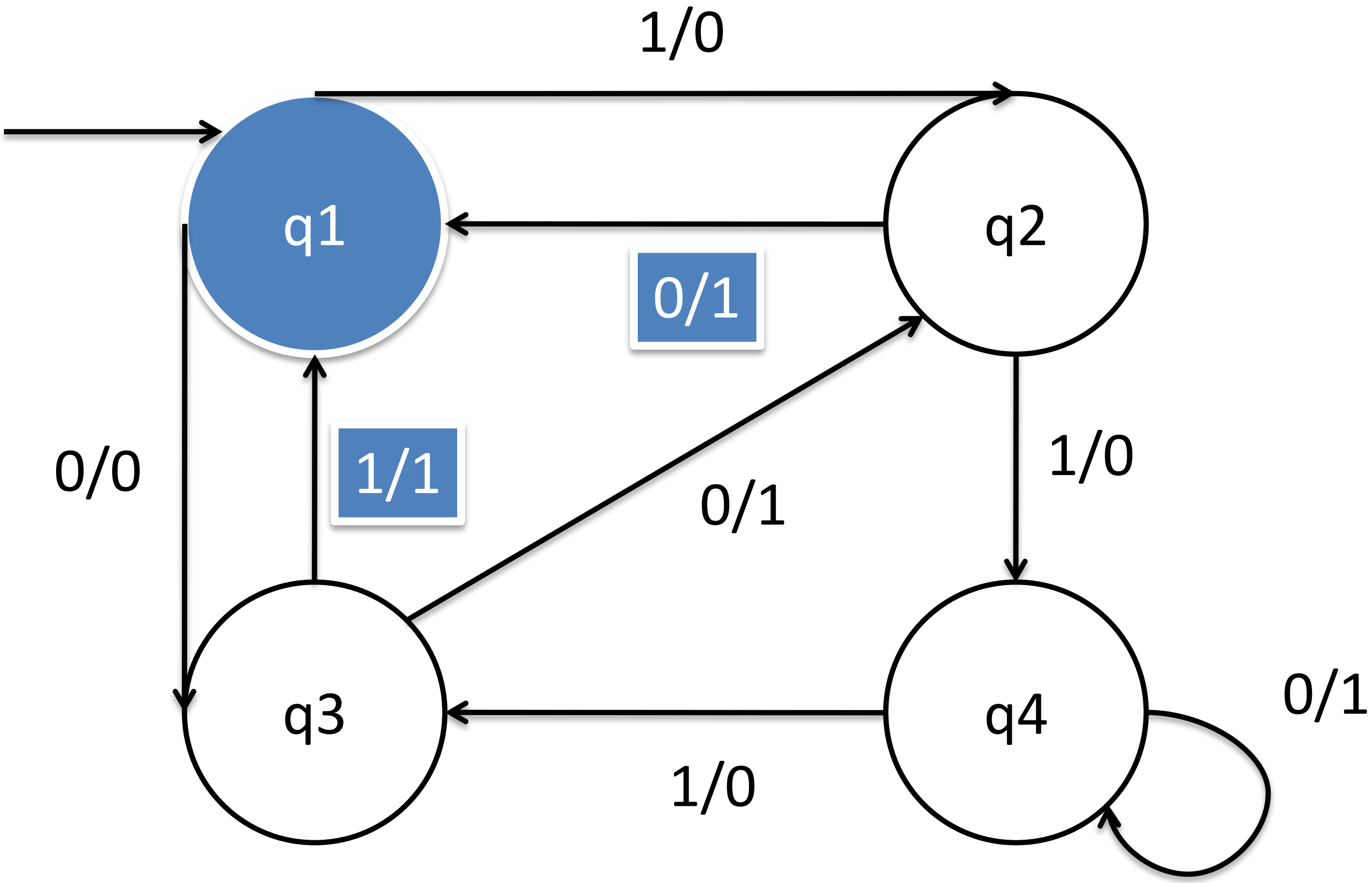
Convert Mealy Machine to Moore Machine



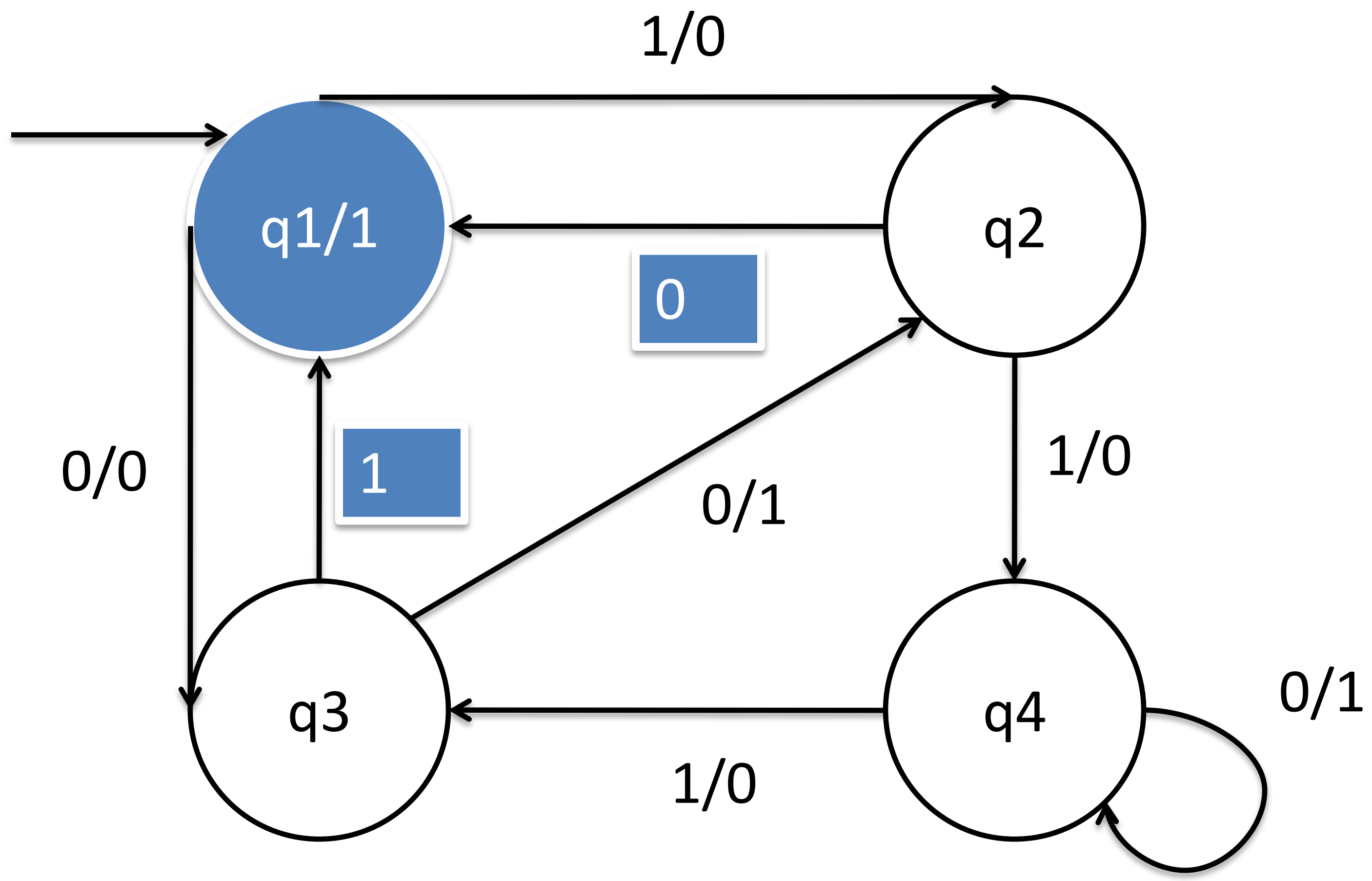
Convert Mealy Machine to Moore Machine



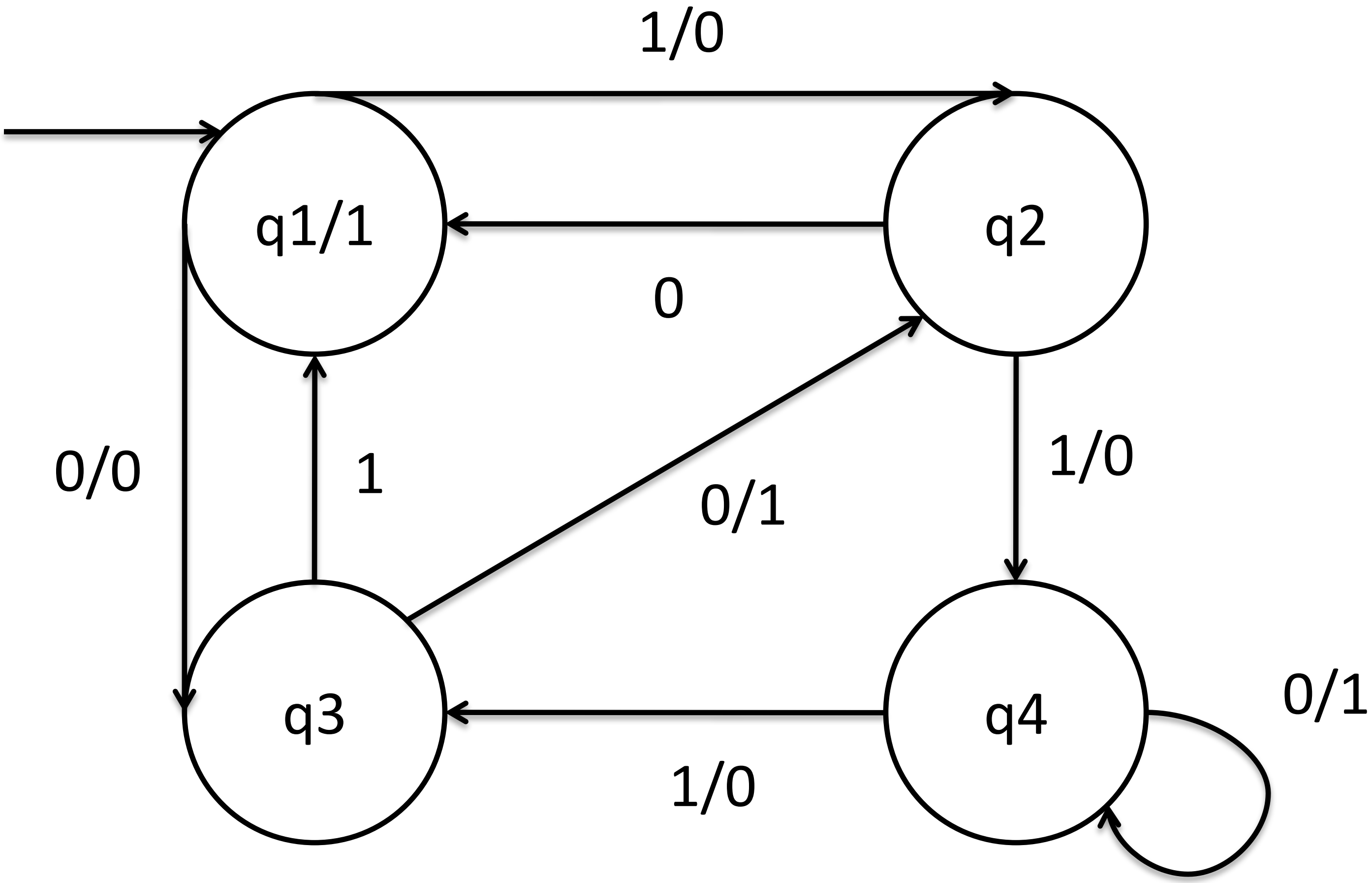
Convert Mealy Machine to Moore Machine



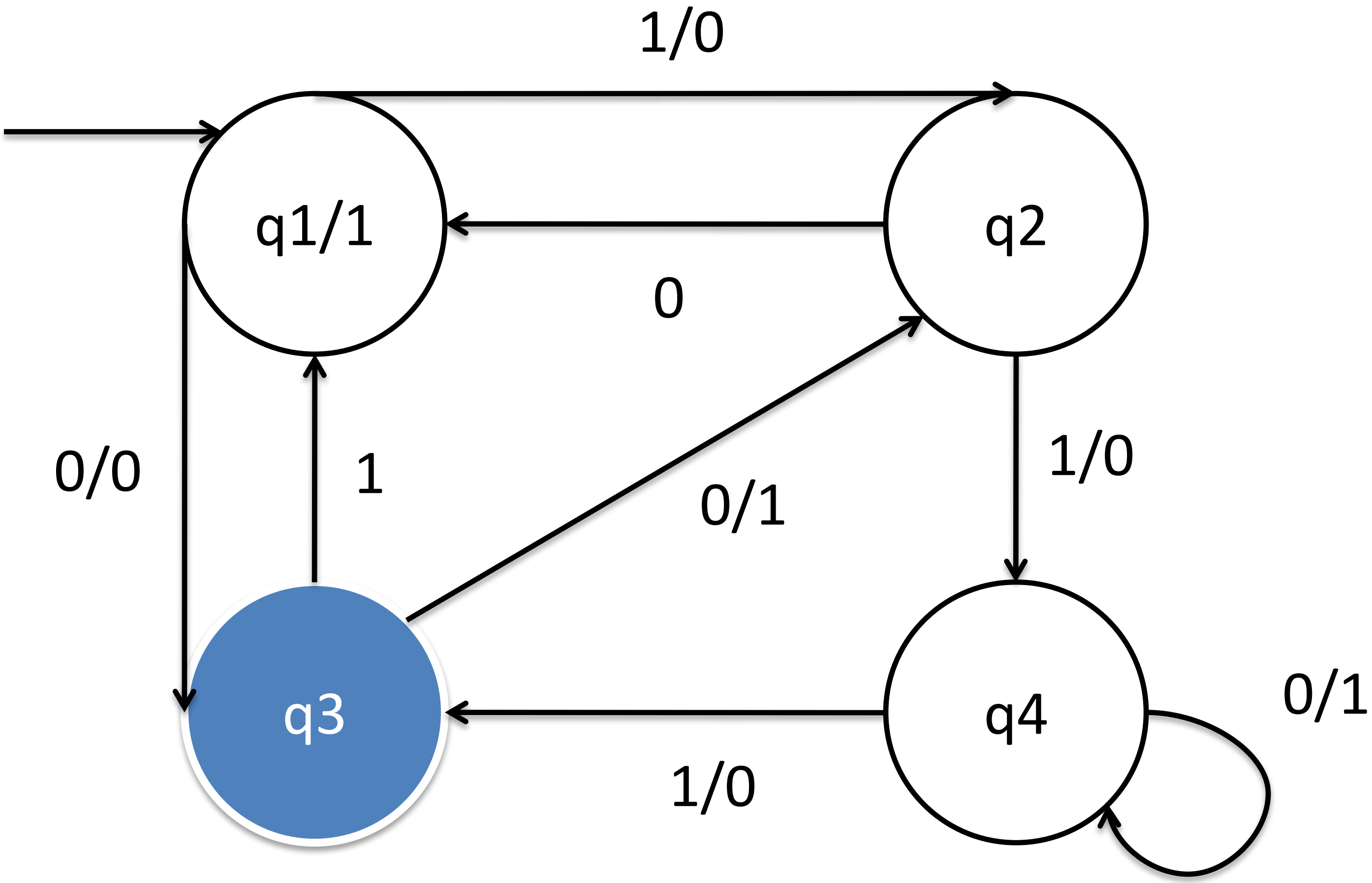
Convert Mealy Machine to Moore Machine



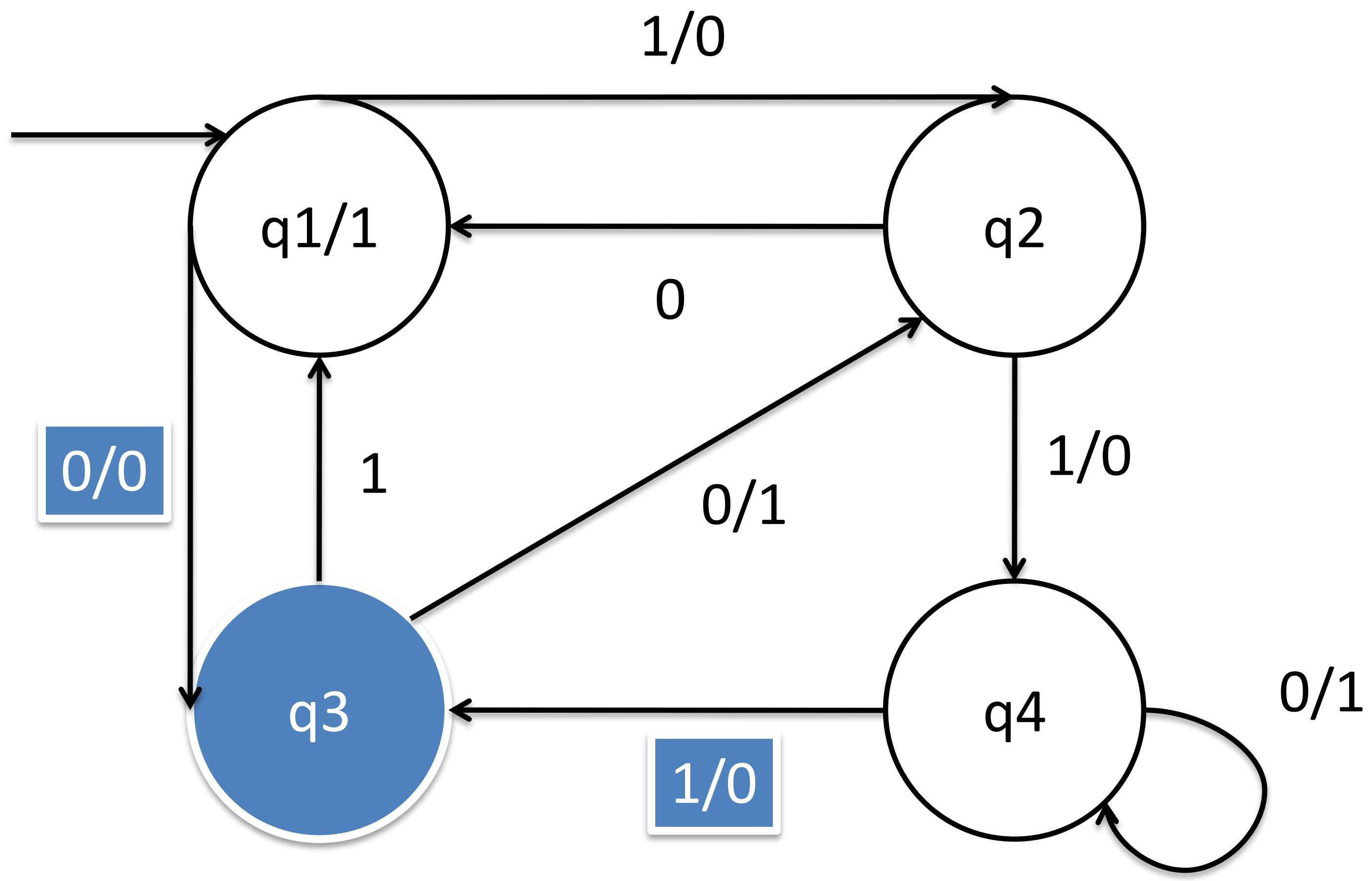
Convert Mealy Machine to Moore Machine



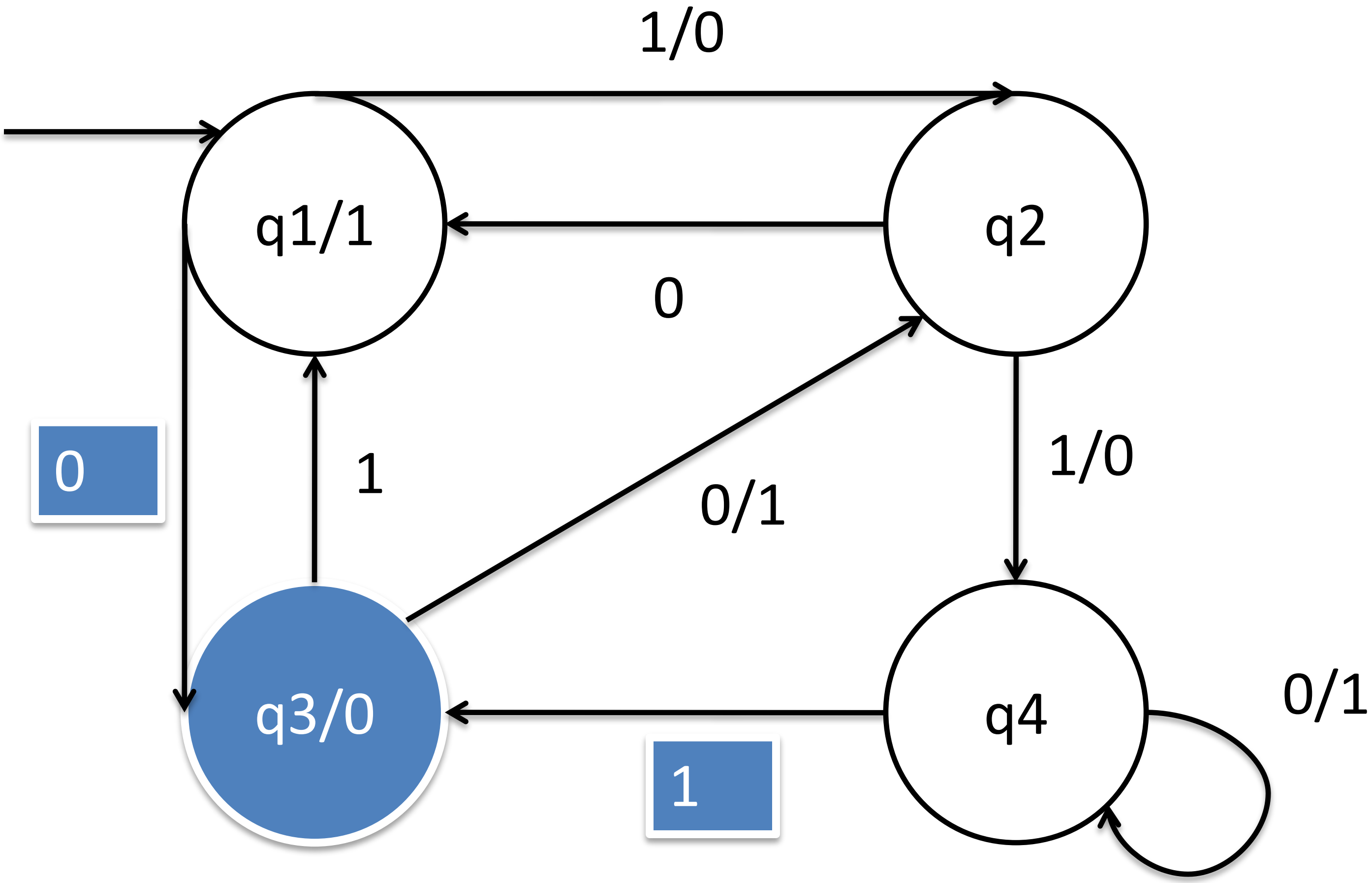
Convert Mealy Machine to Moore Machine



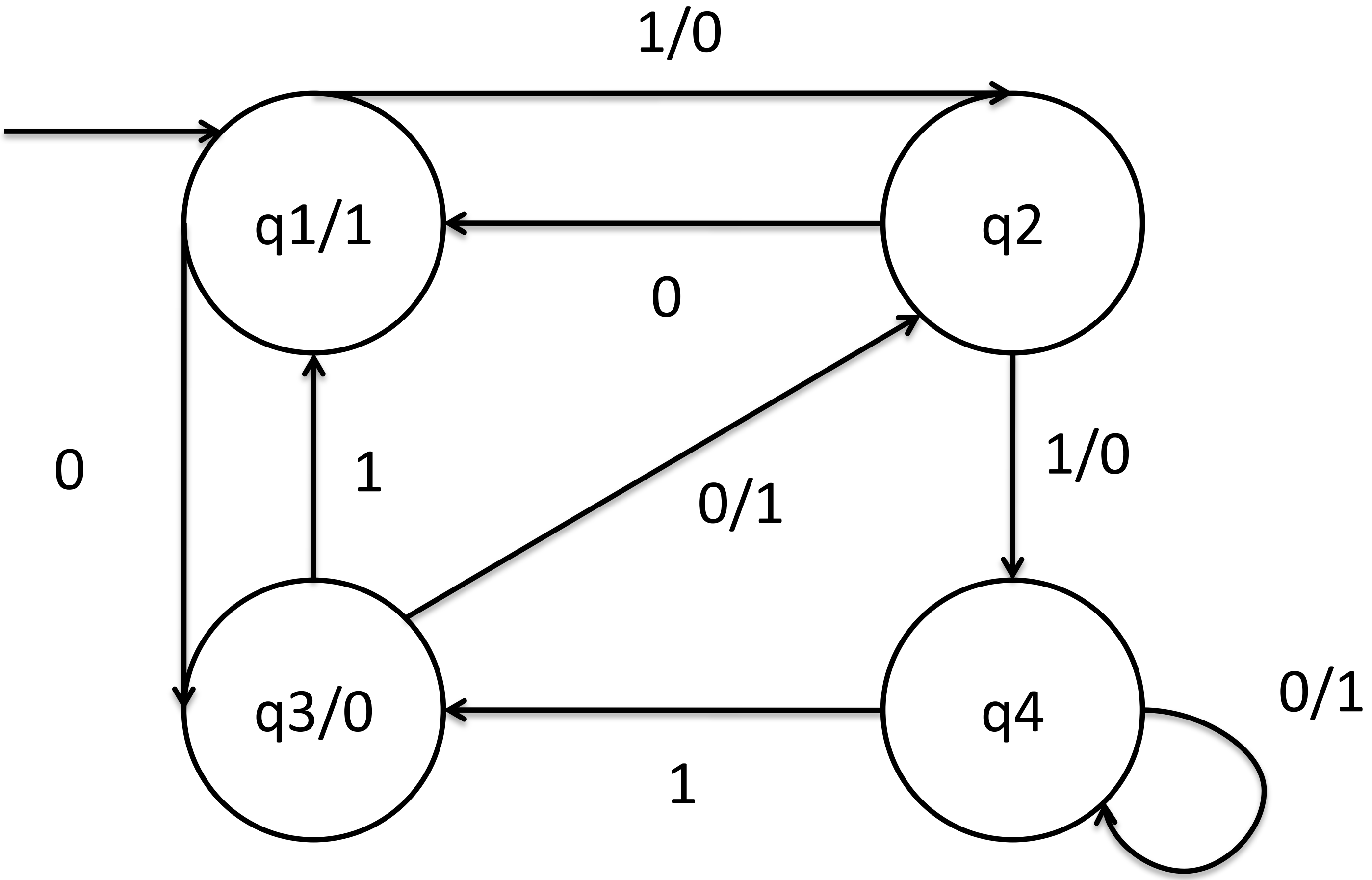
Convert Mealy Machine to Moore Machine



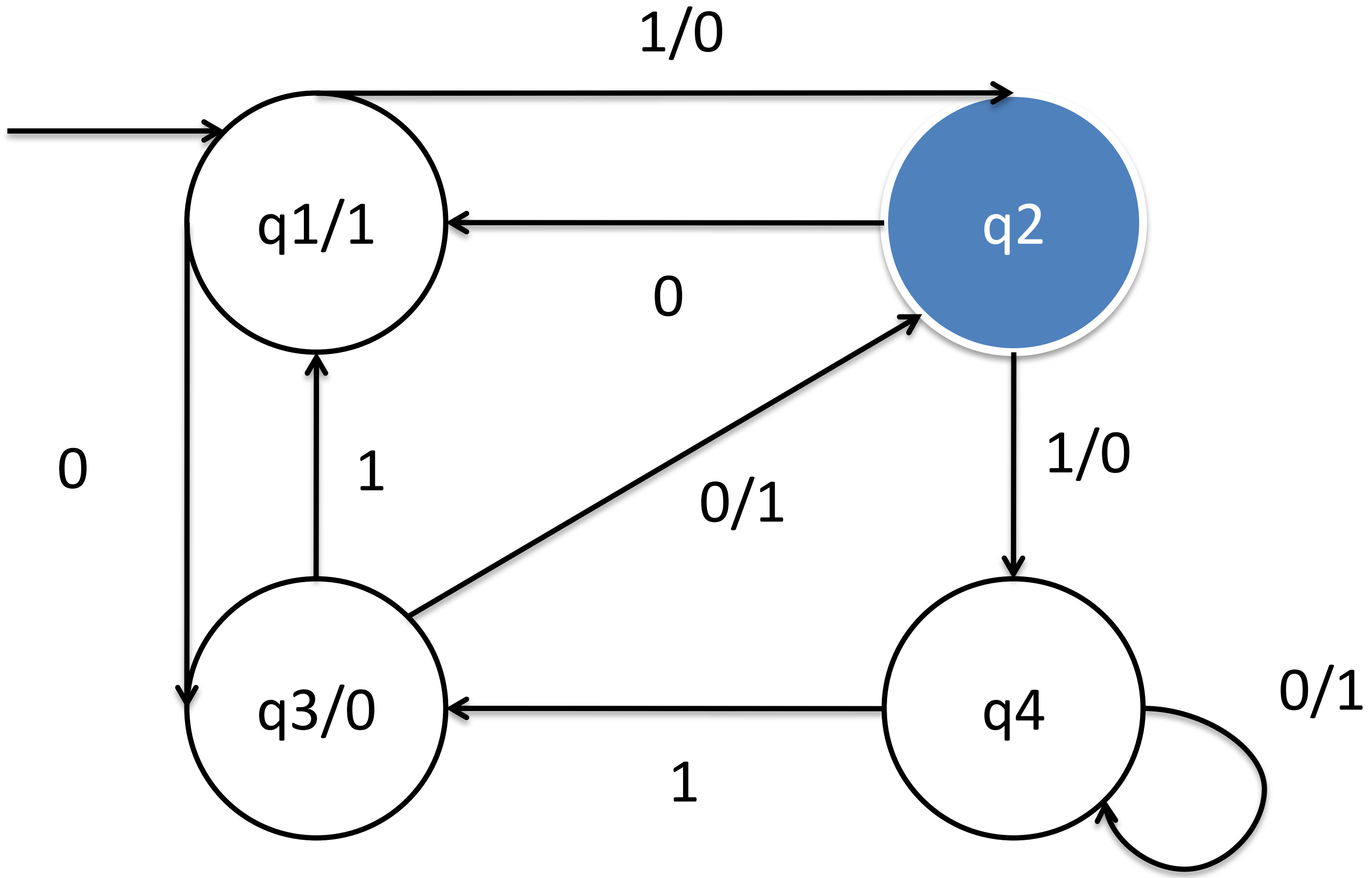
Convert Mealy Machine to Moore Machine



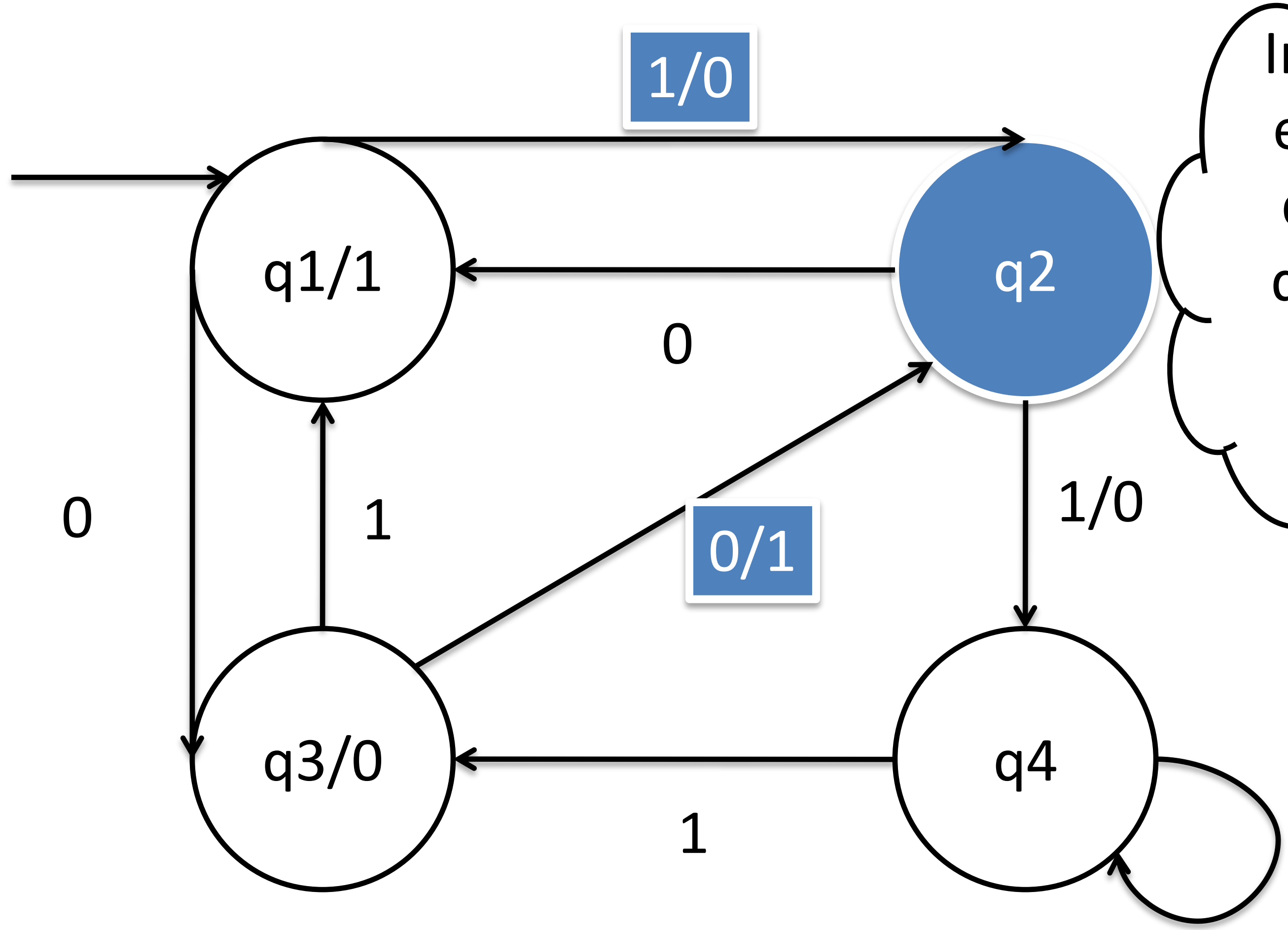
Convert Mealy Machine to Moore Machine



Convert Mealy Machine to Moore Machine

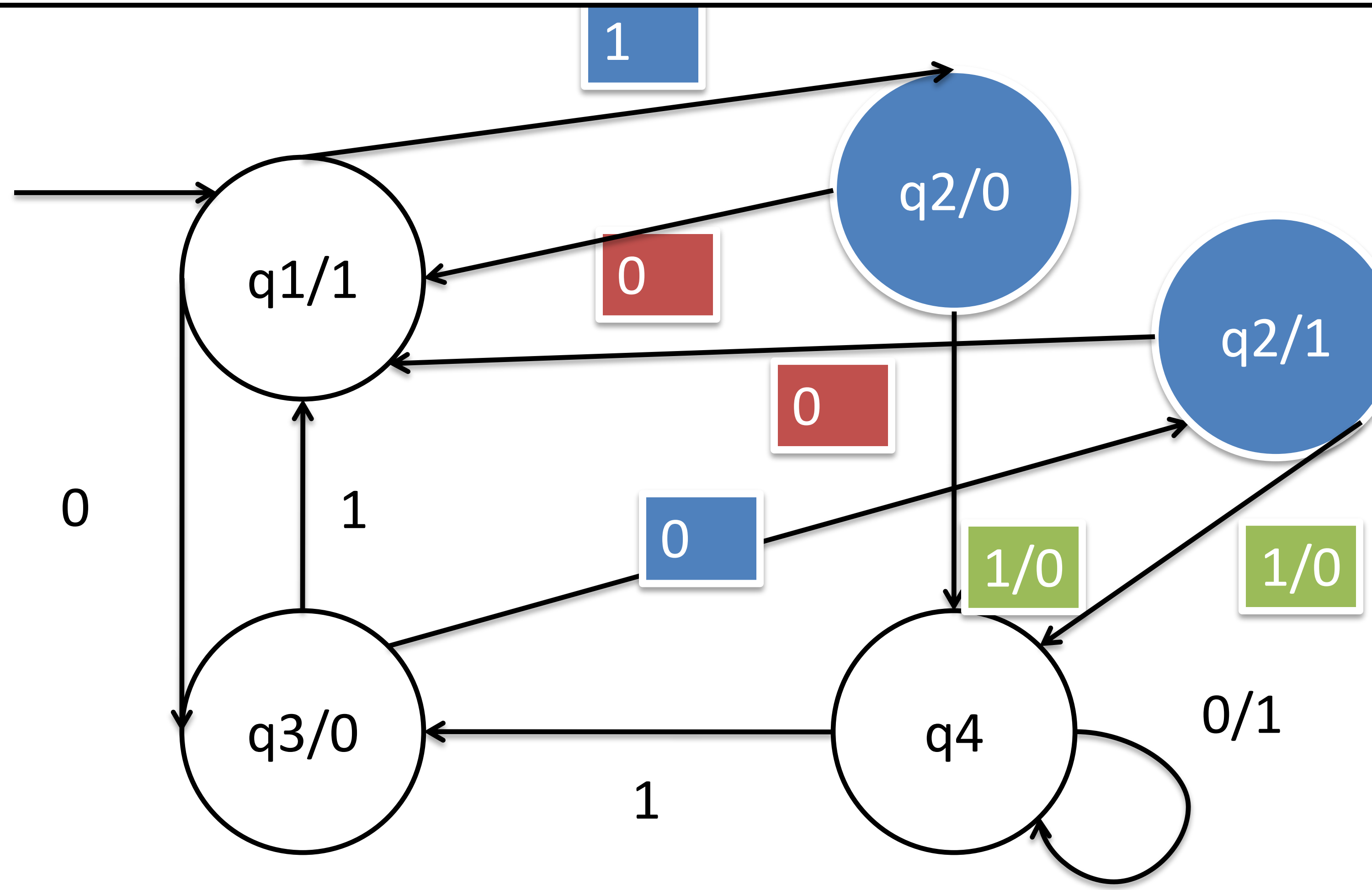


Convert Mealy Machine to Moore Machine

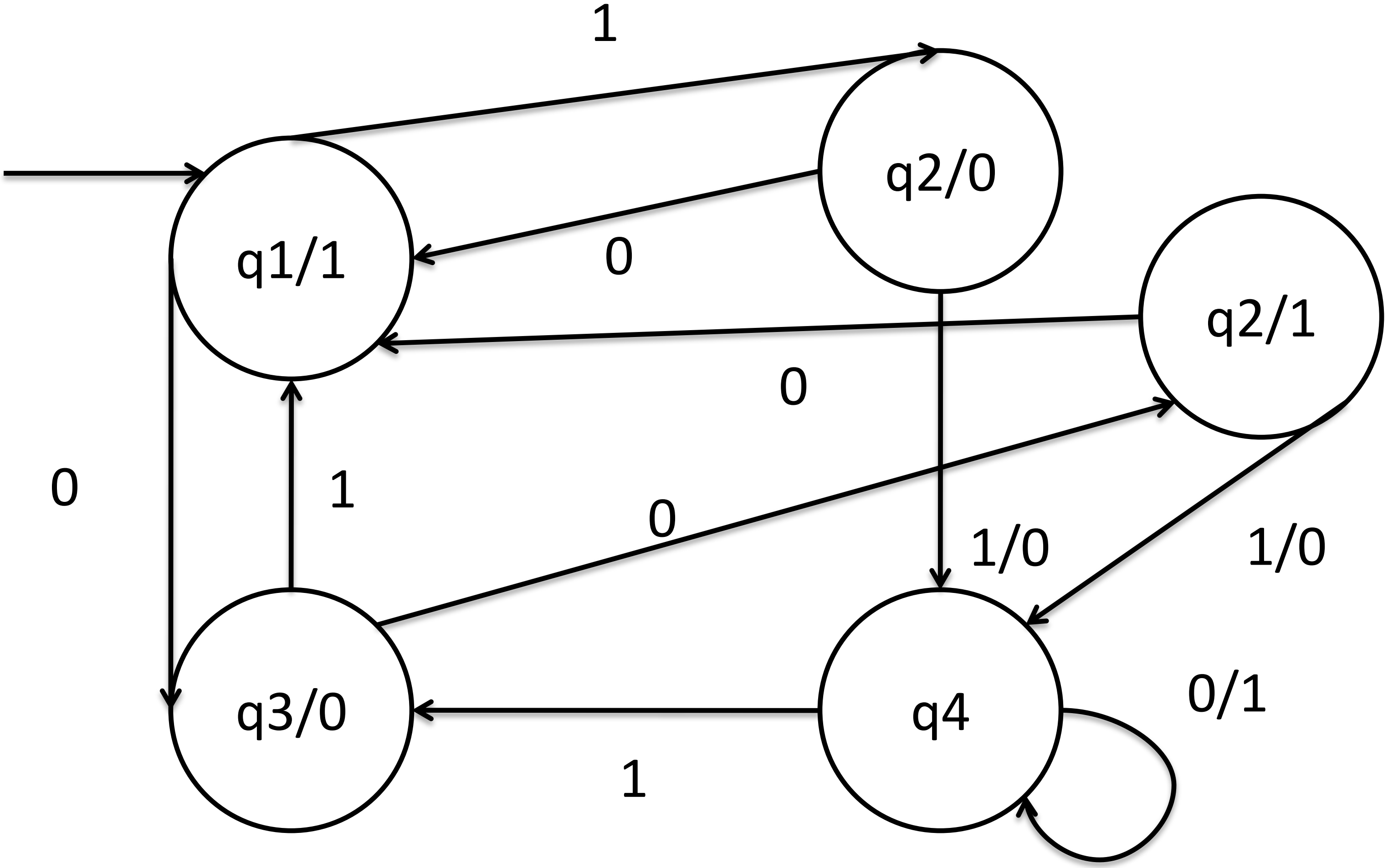


Incoming edges to q2 have different output costs.

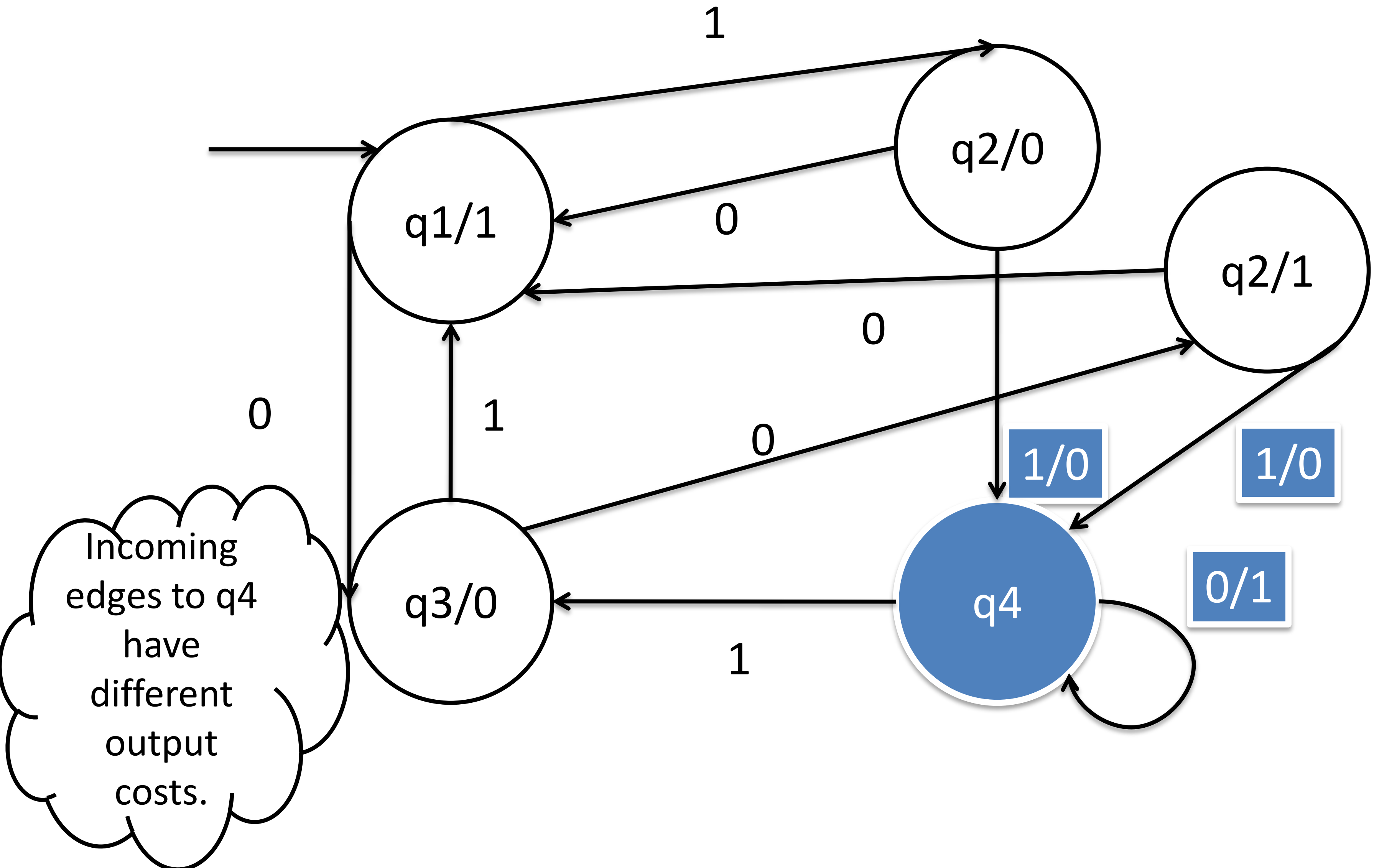
Convert Mealy Machine to Moore Machine



Convert Mealy Machine to Moore Machine

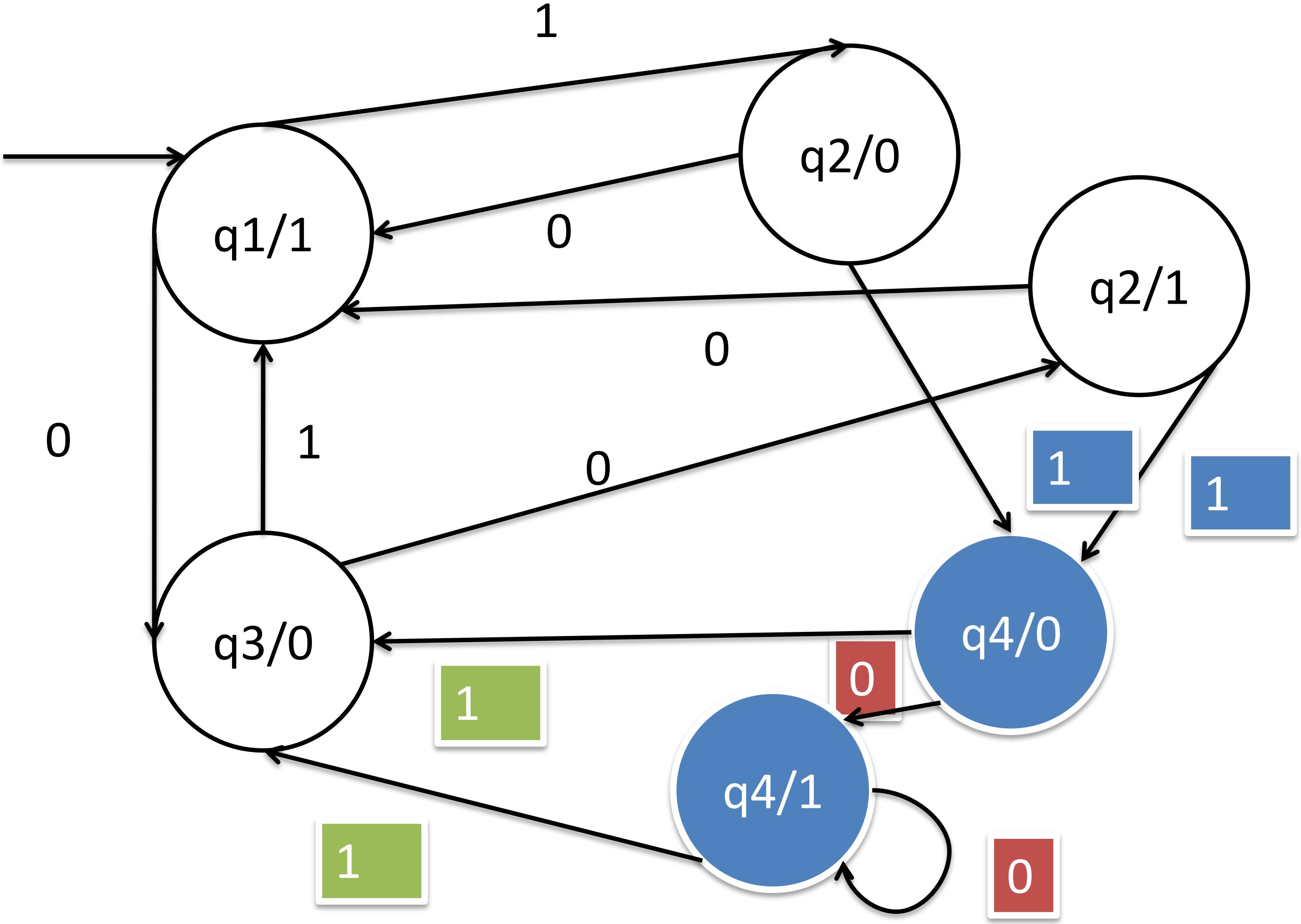


Convert Mealy Machine to Moore Machine

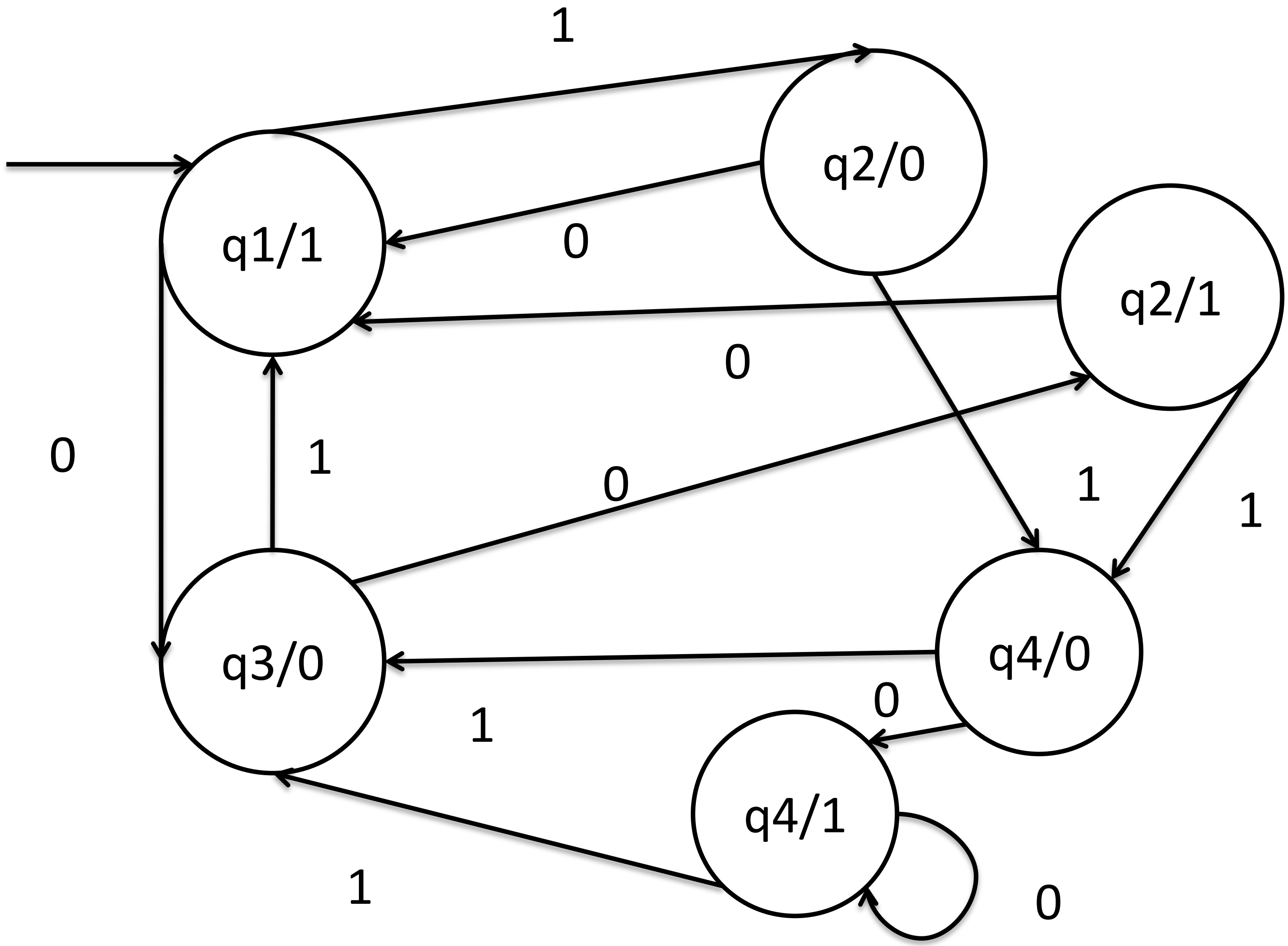


Incoming edges to q4 have different output costs.

Convert Mealy Machine to Moore Machine

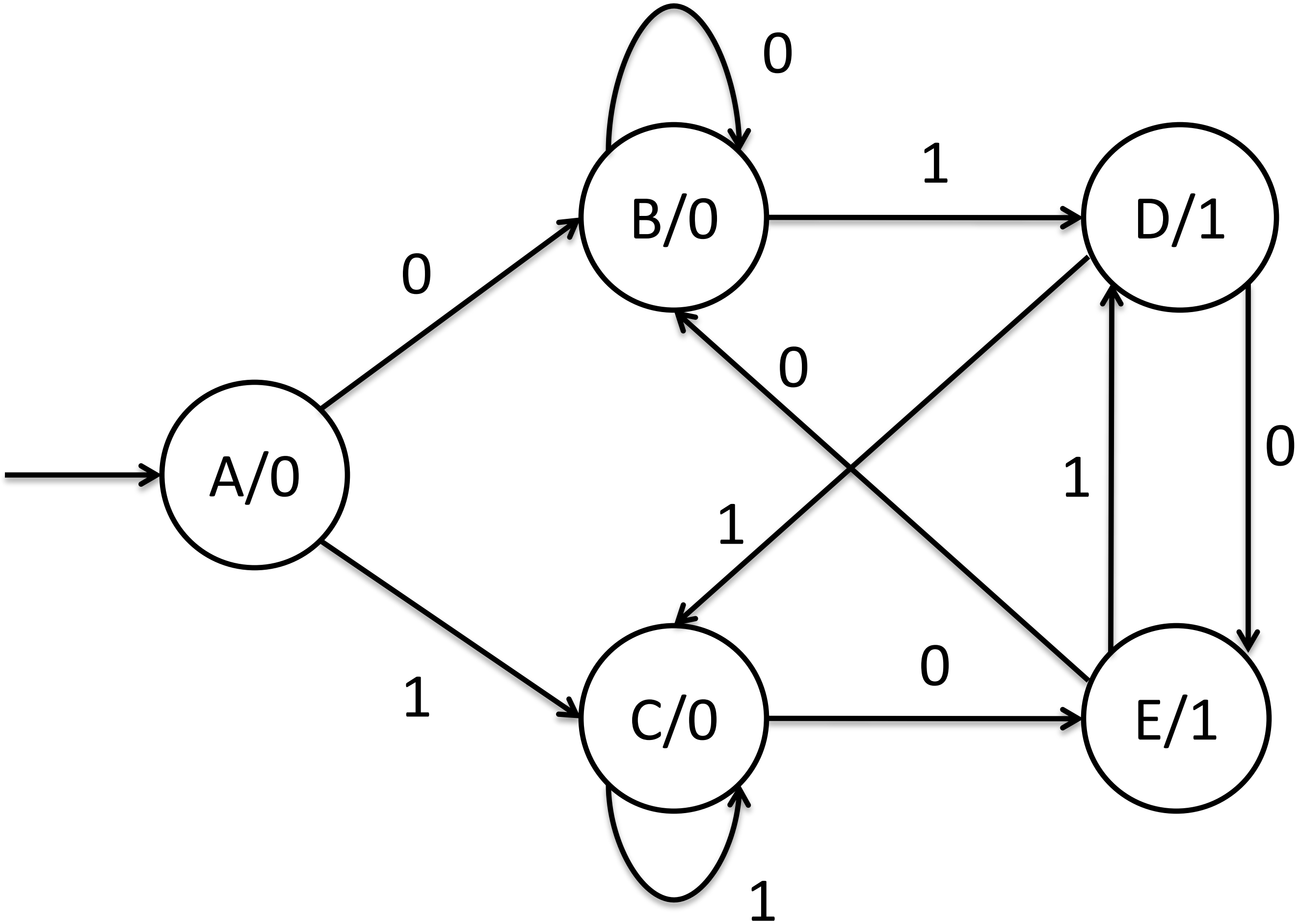


Convert Mealy Machine to Moore Machine

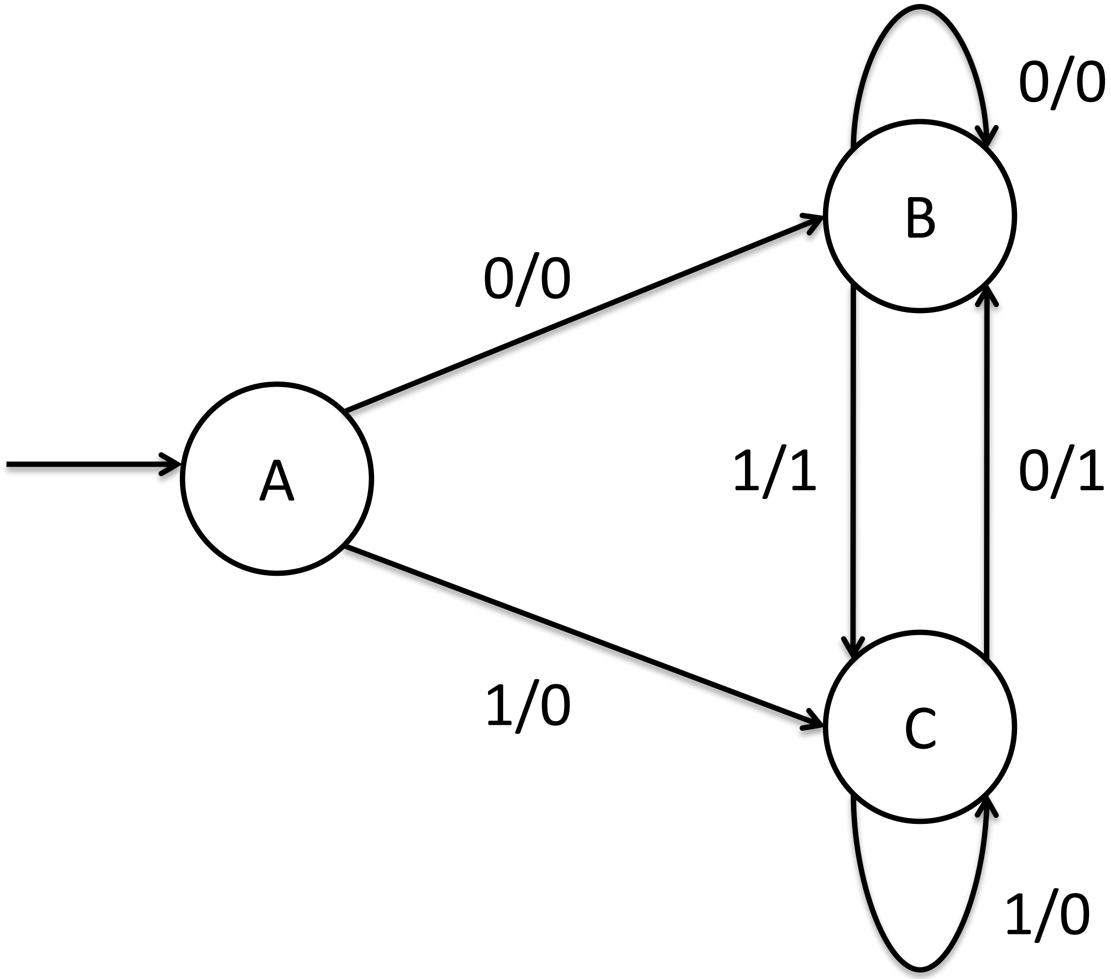


Construct a Moore Machine and a Mealy Machine that counts the occurrences of the sequence 01 or 10 in any input strings over {0,1}

Moore Machine: Count strings 01 or 10

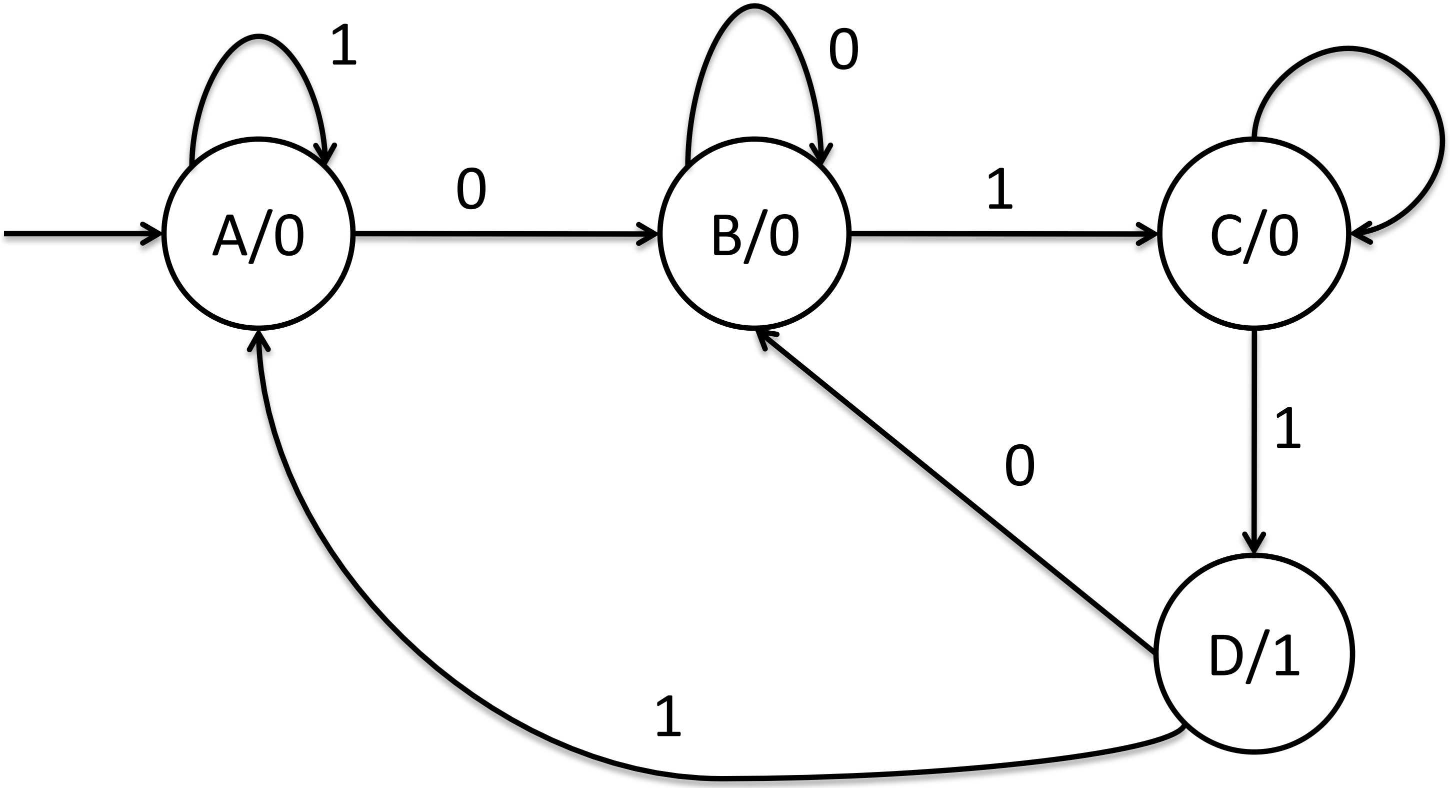


Mealy Machine: Count strings 01 or 10



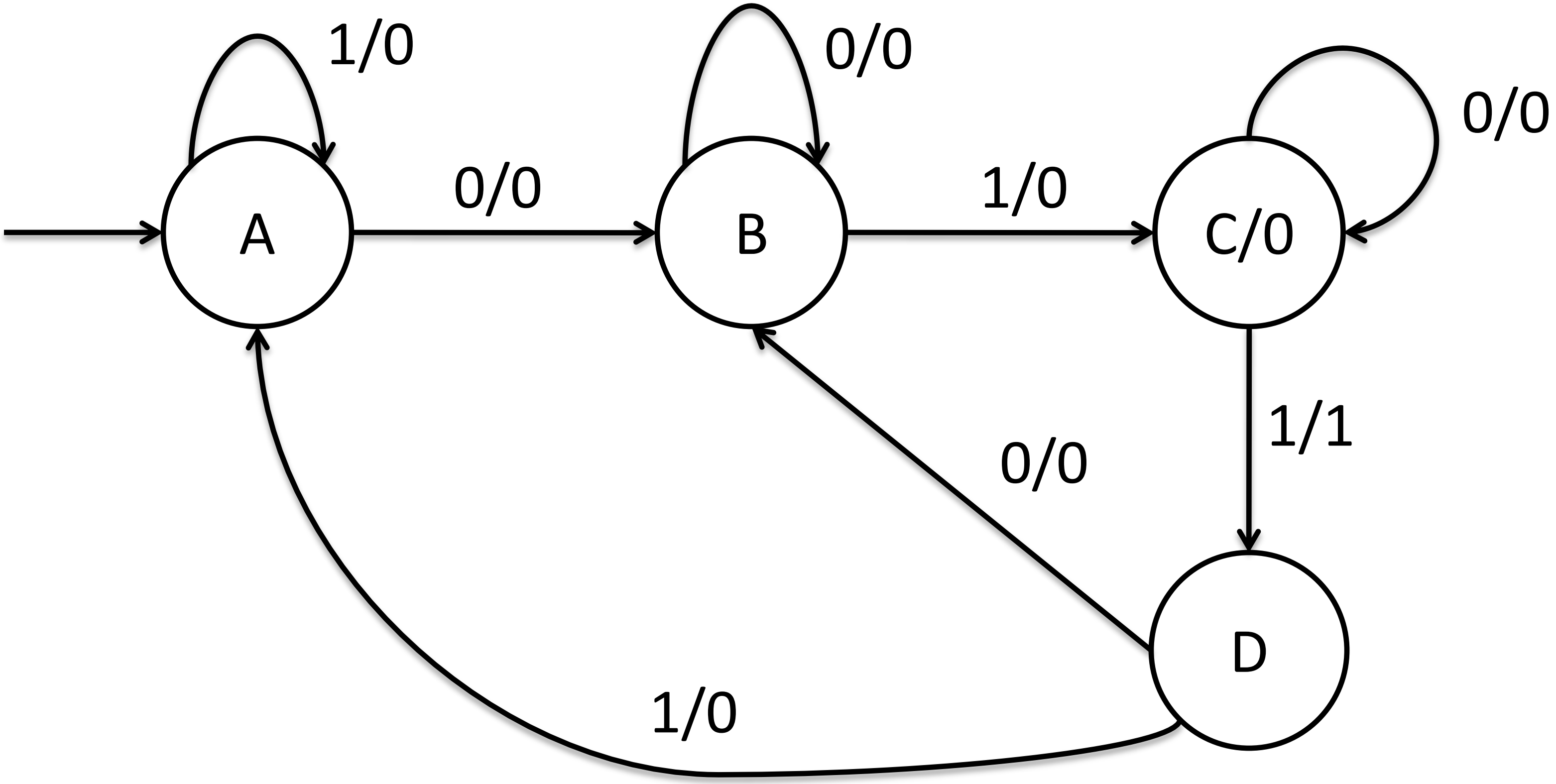
Construct a Moore Machine and a Mealy Machine that counts the occurrences of the sequence 010*1 in any input strings over {0,1}

Moore Machine: Count strings 010*1



INPUT	STATE	OUTPUT
01001	ABCCD	00001
01010101	ABCCDBCCD	000010001
0110101	ABCDBCCD	00010001

Mealy Machine: Count strings 010*1



INPUT	OUTPUT
01001	00001
01010101	00010001
0110101	0010001