① TURING Machine

- Conceptually simplest computer

Tape

0	0	0	1	1	

States

S_2 S_3 S_1

R/W Head

Limited set of Instructions

OL1

ORO

δ transitions

delta

Language

Input

Finite State Machine (FSM)

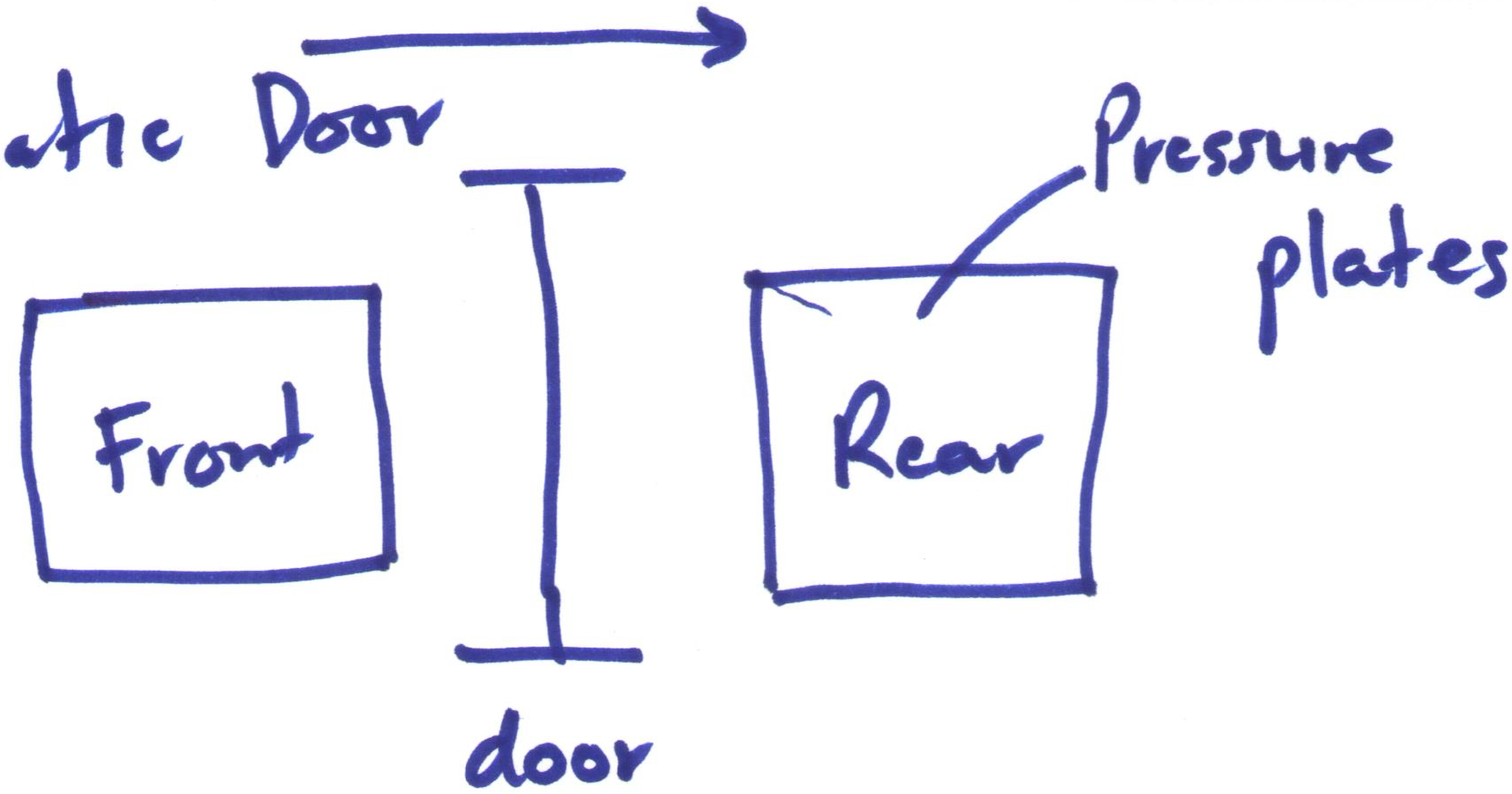
---

⑥ FSM M

Initial State

$$(S, A, \delta, s_0, S_{yes}\})$$

② Automatic Door



2 states OPEN O

CLOSED C

Input Signals

Approaching - Front F

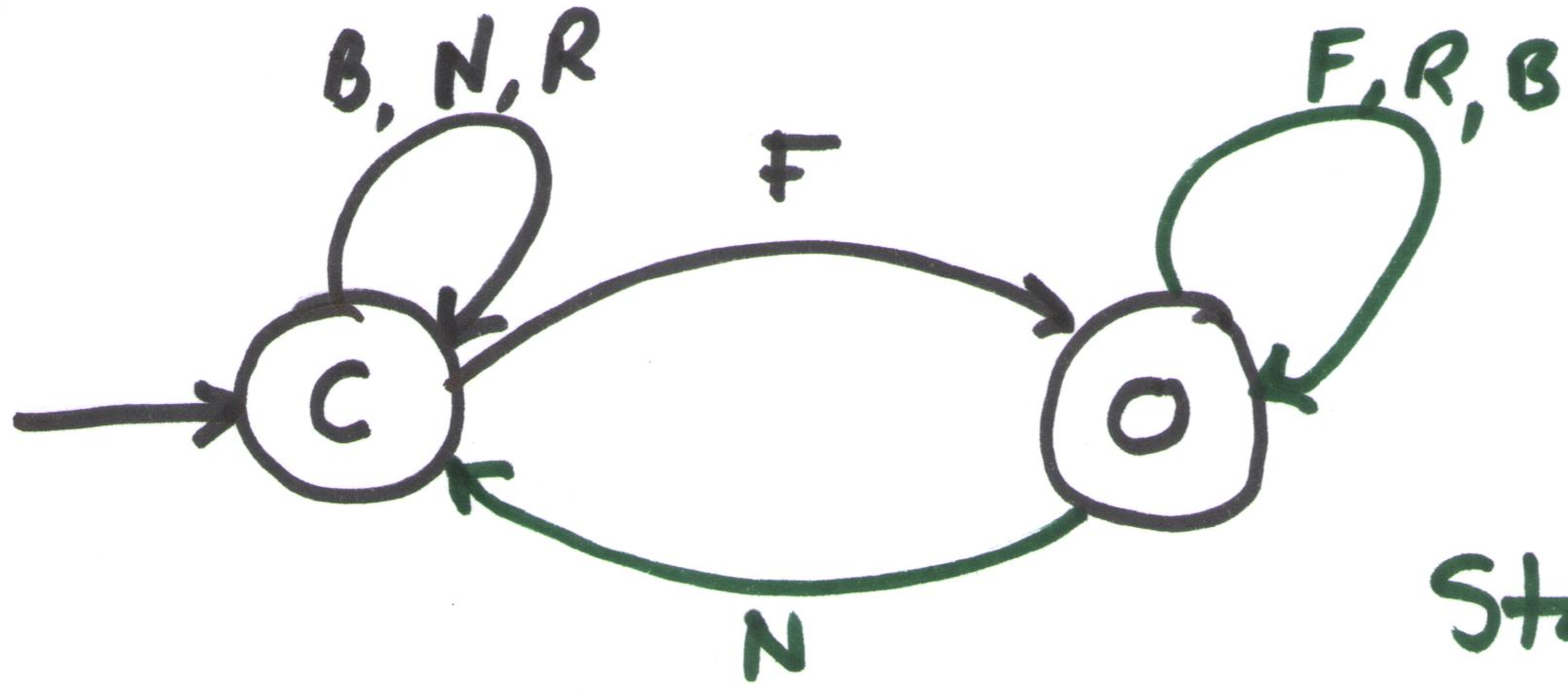
Passing - Both Front / Rear B

Gone - Neither Front / Rear N

Passed - Rear R

	N	F	R	B
O	C	O	O	O
C	C	O	C	C

3



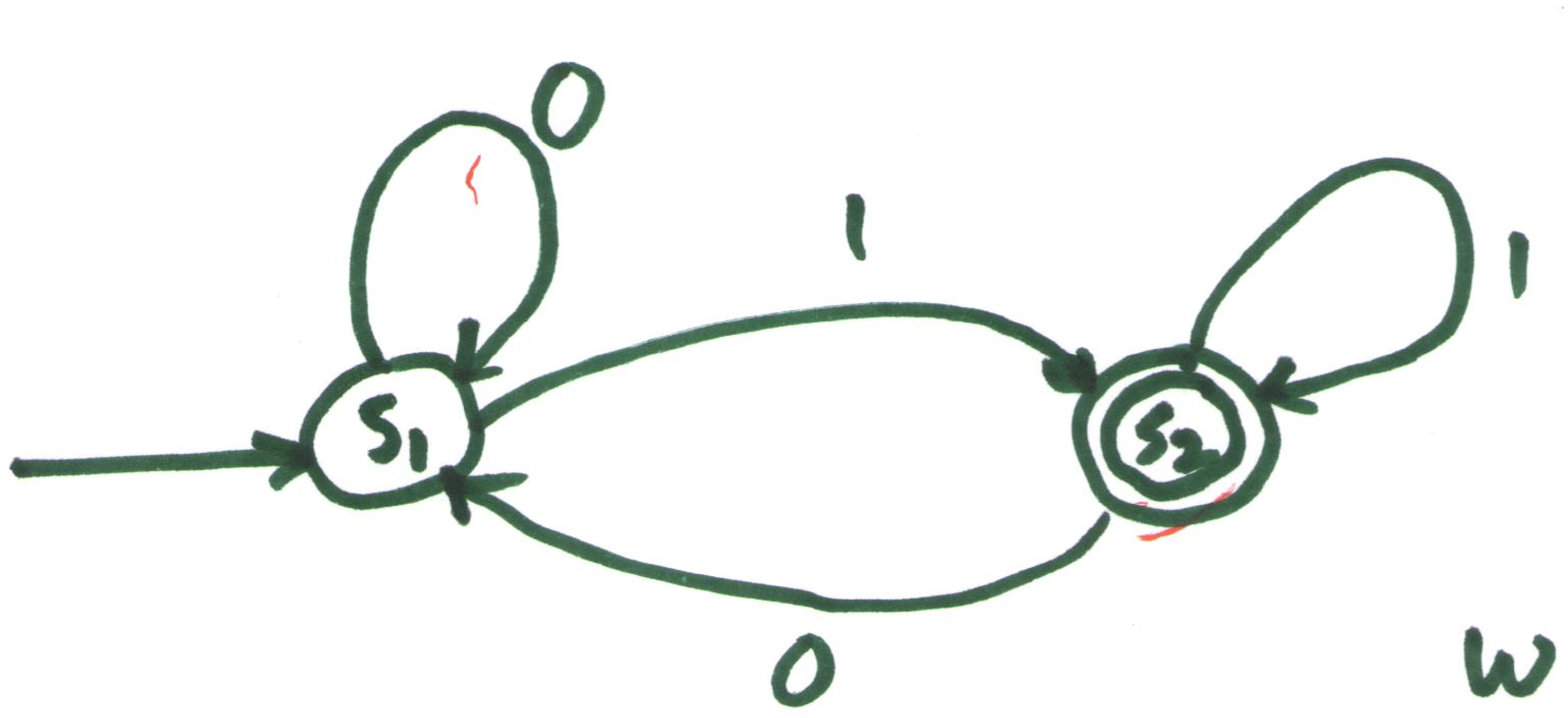
States

Transitions δ
as Edges

Symbols

Alphabets
$\{N, F, R, B\}$

4



$w = 1101$

w is a string concatenation of the alphabet A

Alphabet $A = \{0, 1\}$ States $S = \{s_1, s_2\}$

Language domain

	0	1
s_1	s_1	s_2
s_2	s_1	s_2

Transitions

$\delta(s_1, 0) = s_1$ $\delta(s_1, 1) = s_2$

$\delta(s_2, 0) = s_1$ Range $\delta(s_2, 1) = s_2$

$$\delta : S \times A \to S$$

$S_{yes} = \{s_2\}$ accept state double circle

5 FSM notation

$$\delta^*(s,w)$$ w is input string

split w recursively until we get an empty string ε

$w = ua$, u is a String
a is a character of A

$$\boxed{\delta(\delta^*(s,u), a)}$$

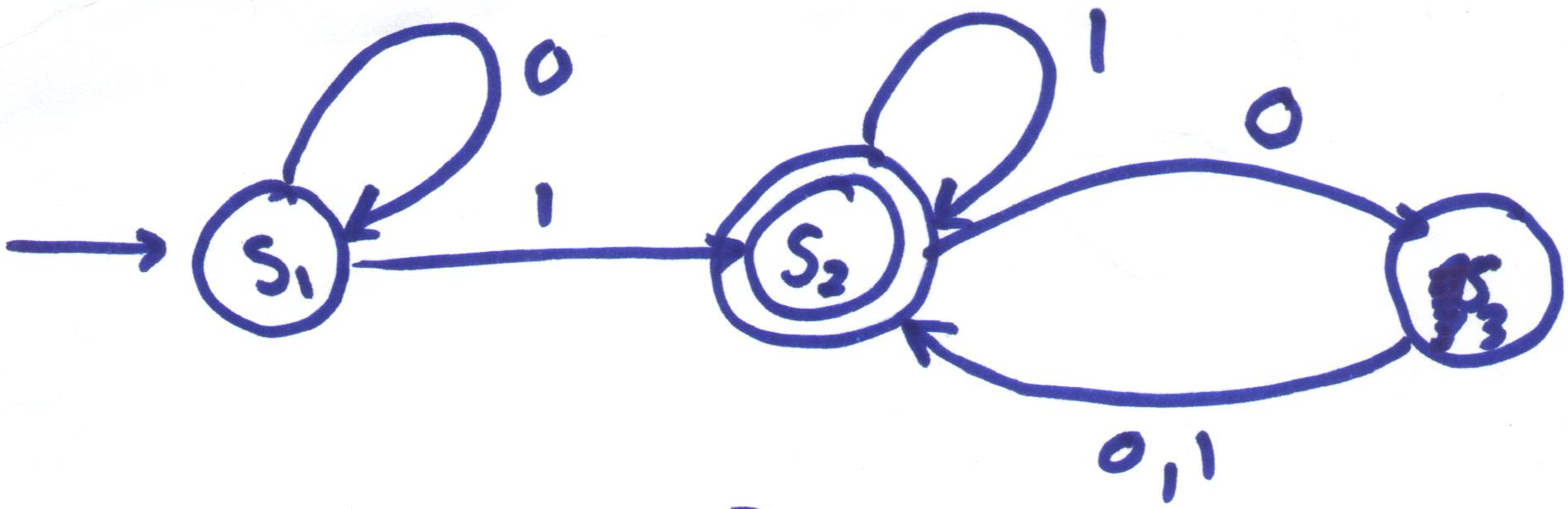
$$\delta^*(s,w) = \begin{cases} s, & \text{if } w=\varepsilon \\ \delta(\delta^*(s,u),a), & \text{when } w=ua \end{cases}$$

Language L of a FSM M

$$L(M) = \{ w \in A^* \mid \delta^*(s,w) \in S_{yes} \}$$

A^* as the set of all possible strings involving characters in A

Any L accepted by FSM is a regular Language



$S_{yes} = \{s_2\}$

$S = \{s_1, s_2, s_3\}$

$A = \{0,1\}$

δ :

	0	1
s_1	s_1	s_2
s_2	s_3	s_2
s_3	s_2	s_2

$w = 00110$

$w = 00110$

$$s_1 \xrightarrow{0} s_1 \xrightarrow{0} s_1 \xrightarrow{1} s_2 \xrightarrow{1} s_2 \xrightarrow{0} s_3$$

STOP

$s_3 \notin S_{yes}$ Reject string

$w = 001100$

$$s_1 \xrightarrow{0} s_1 \xrightarrow{0} s_1 \xrightarrow{1} s_2 \xrightarrow{1} s_2 \xrightarrow{0} s_3 \xrightarrow{0} s_2$$

$s_2 \in S_{yes}$

Accept

Summary Strings must ~~contain~~ END in an even number of zeros and at least one 1 character

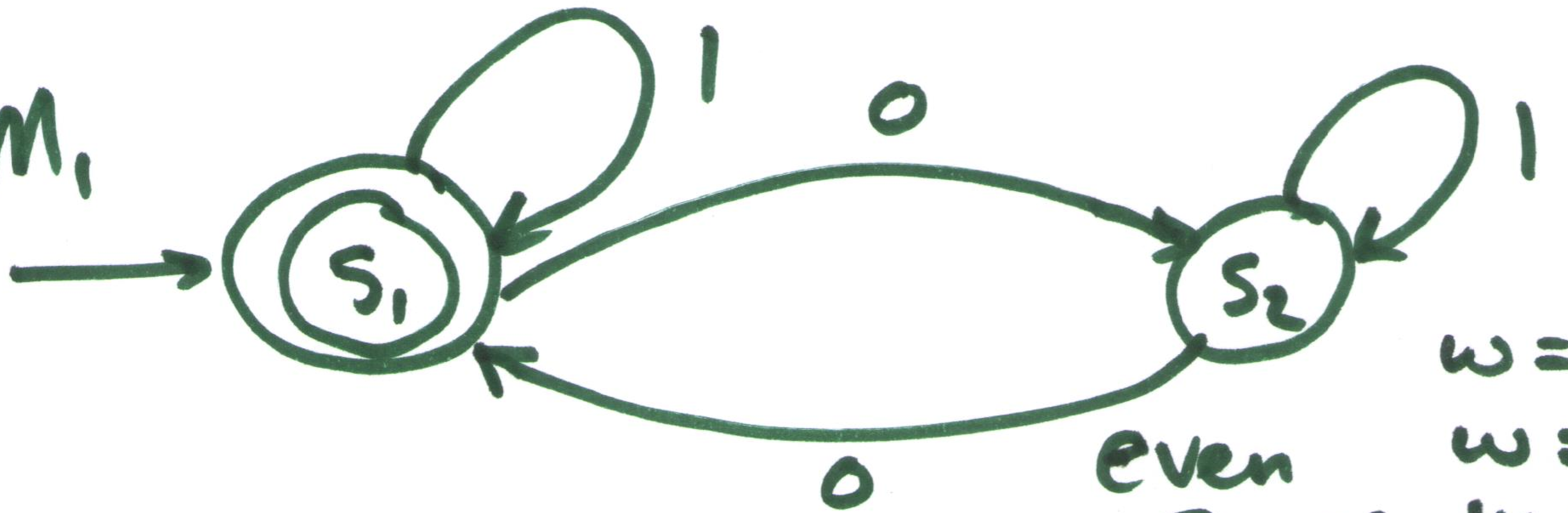
Regular Languages A & B

Regular Operations → preserve regularity

Is $A \cup B$ regular?

M_1 M_2

M_1



even zeros

$w = \varepsilon$
$w = 010$
$w = 00$

M_2

odd ones

$w = 01$
$= 1$

M_3 $A \cup B$
Run M_1 & M_2 Parallels

$\cancel{S_3 = S_1 \times S_2}$

$$S_3 = \{\underbrace{(s_1, r_1)}_{q_1}, \underbrace{(s_1, r_2)}_{q_2}, \underbrace{(s_2, r_1)}_{q_3}, \underbrace{(s_2, r_2)}_{q_4}\}$$

$$S_1 \times S_2 \longrightarrow S_3$$

Start : (s_1, s_2) Applies to Both M_1 & M_2

$\delta_3 : S_3 \times A \longrightarrow S_3$

$\delta_3((s_1, r_1), 0) \longrightarrow (s_2, r_1)$

$\vdots$

8 rules

$S_{yes} = \{(s_1, r_2), (s_2, r_2), (s_1, r_1)\}$

$A \cup B$ is also regular

$A \cap B$?

~~A∩~~ $A \cup B$

$S_{yes_3} = \{(r_1, r_2) \mid r_1 \in S_{yes_1}$ **OR** $r_2 \in S_{yes_2}$

$A \cap B$

$S_{yes_3} = \{(r_1, r_2) \mid r_1 \in S_{yes_1}$ **AND** $r_2 \in S_{yes_2}$