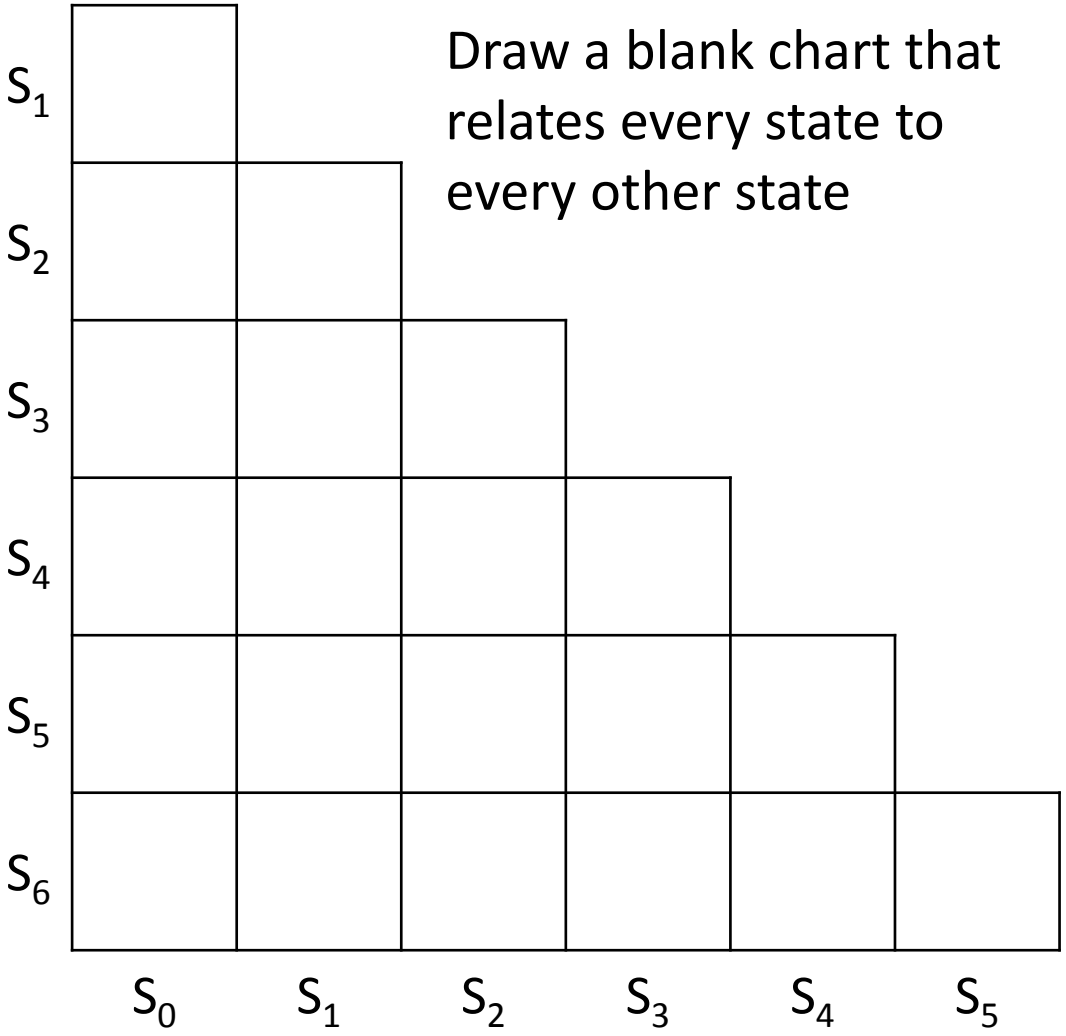
Implication Chart Method

- Another method for minimizing our state table
- Applies to Moore and Mealy implementations
- Graphical analysis, and therefore, sometimes easier to follow what's happening
- Also best suited for CAD...but of course we're going try it out by hand anyways!

Creating an Implication Chart

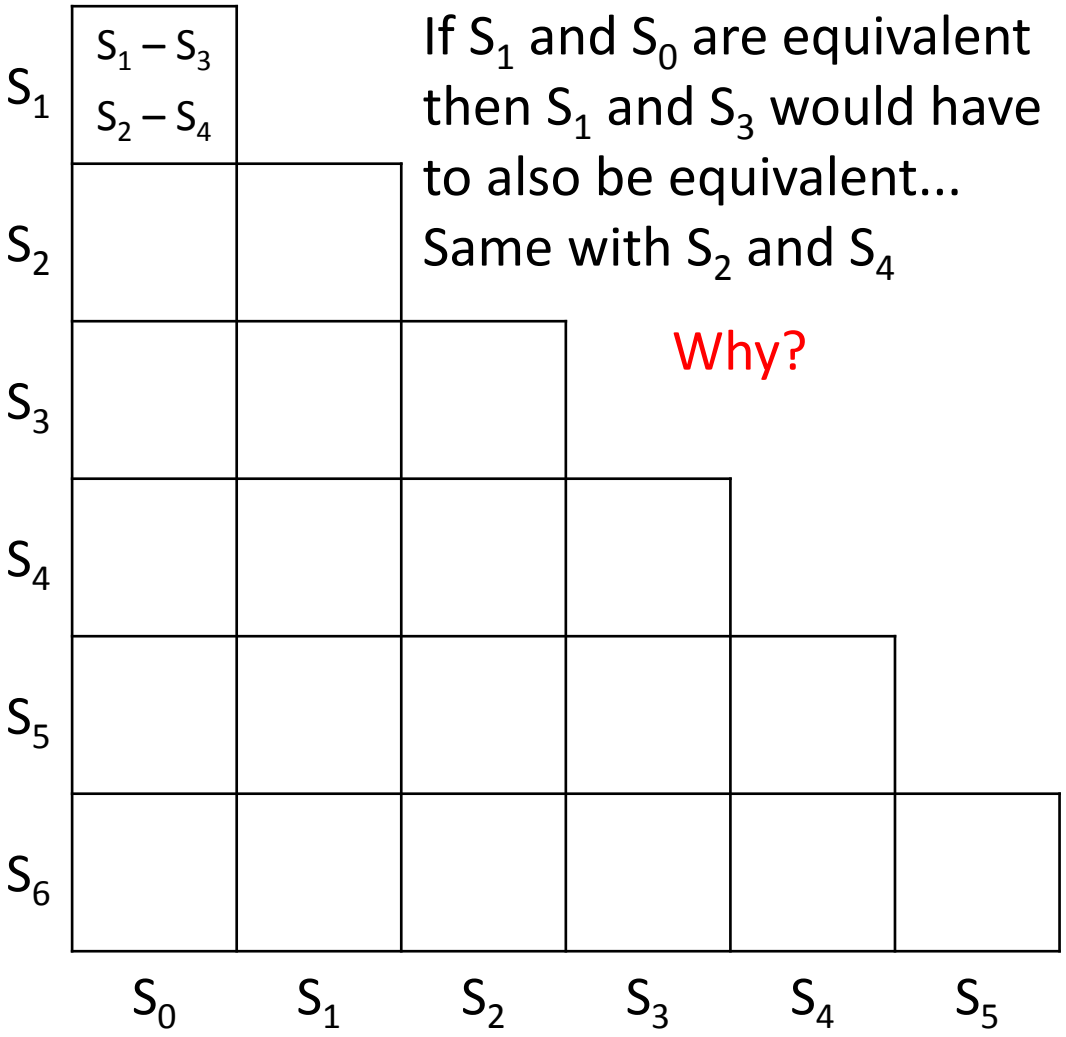


Draw a blank chart that relates every state to every other state

Present State	Next State		Output	
	w = 0	w = 1	w = 0	w = 1
S_0	S_1	S_2	0	0
S_1	S_3	S_4	0	0
S_2	S_5	S_6	0	0
S_3	S_0	S_0	0	0
S_4	S_0	S_0	1	0
S_5	S_0	S_0	0	0
S_6	S_0	S_0	1	0

Notice that the upper half (including the diagonal) is unnecessary!

Creating an Implication Chart



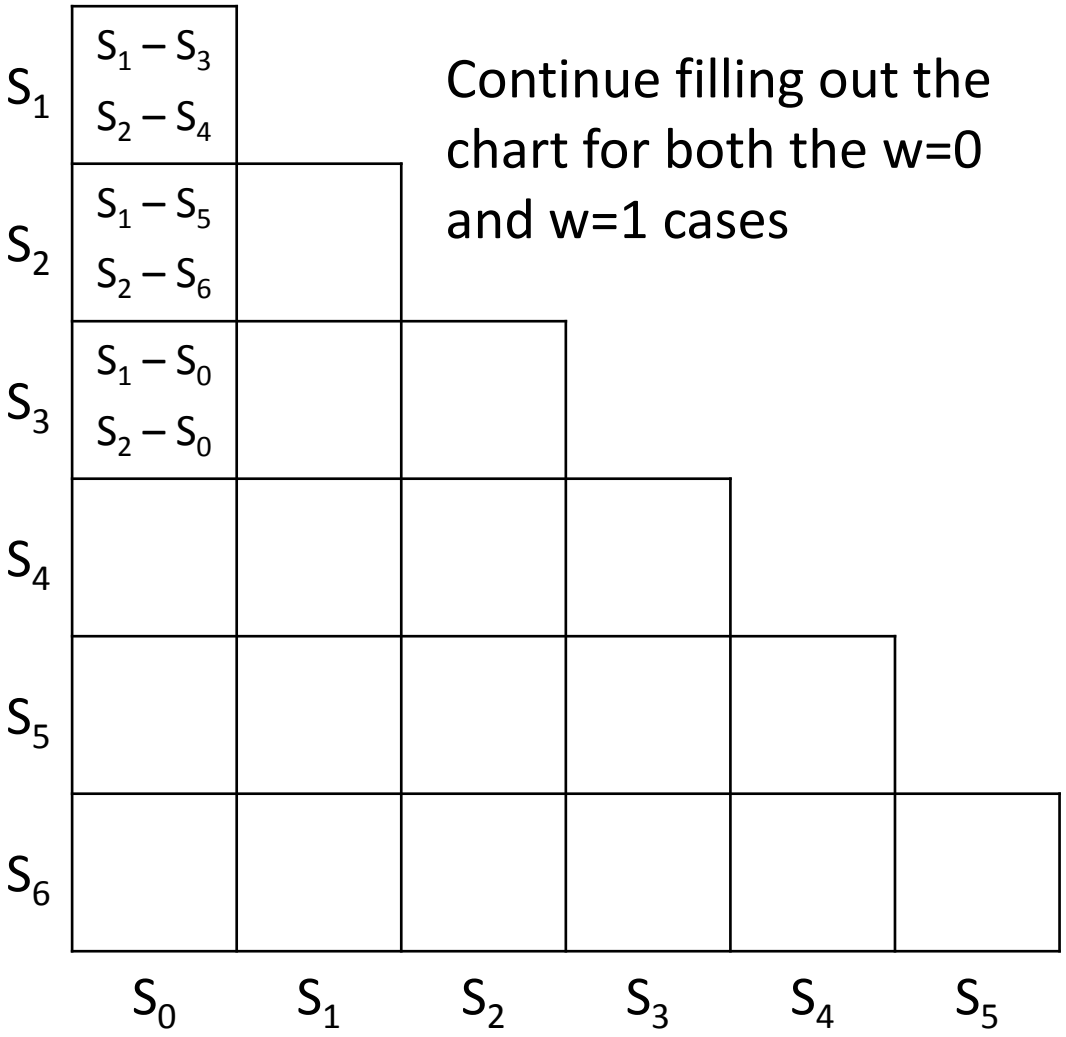
If S_1 and S_0 are equivalent then S_1 and S_3 would have to also be equivalent... Same with S_2 and S_4

Why?

Present State	Next State		Output	
	w = 0	w = 1	w = 0	w = 1
S_0	S_1	S_2	0	0
S_1	S_3	S_4	0	0
S_2	S_5	S_6	0	0
S_3	S_0	S_0	0	0
S_4	S_0	S_0	1	0
S_5	S_0	S_0	0	0
S_6	S_0	S_0	1	0

Because equivalent states must have equivalent next-state transitions

Creating an Implication Chart

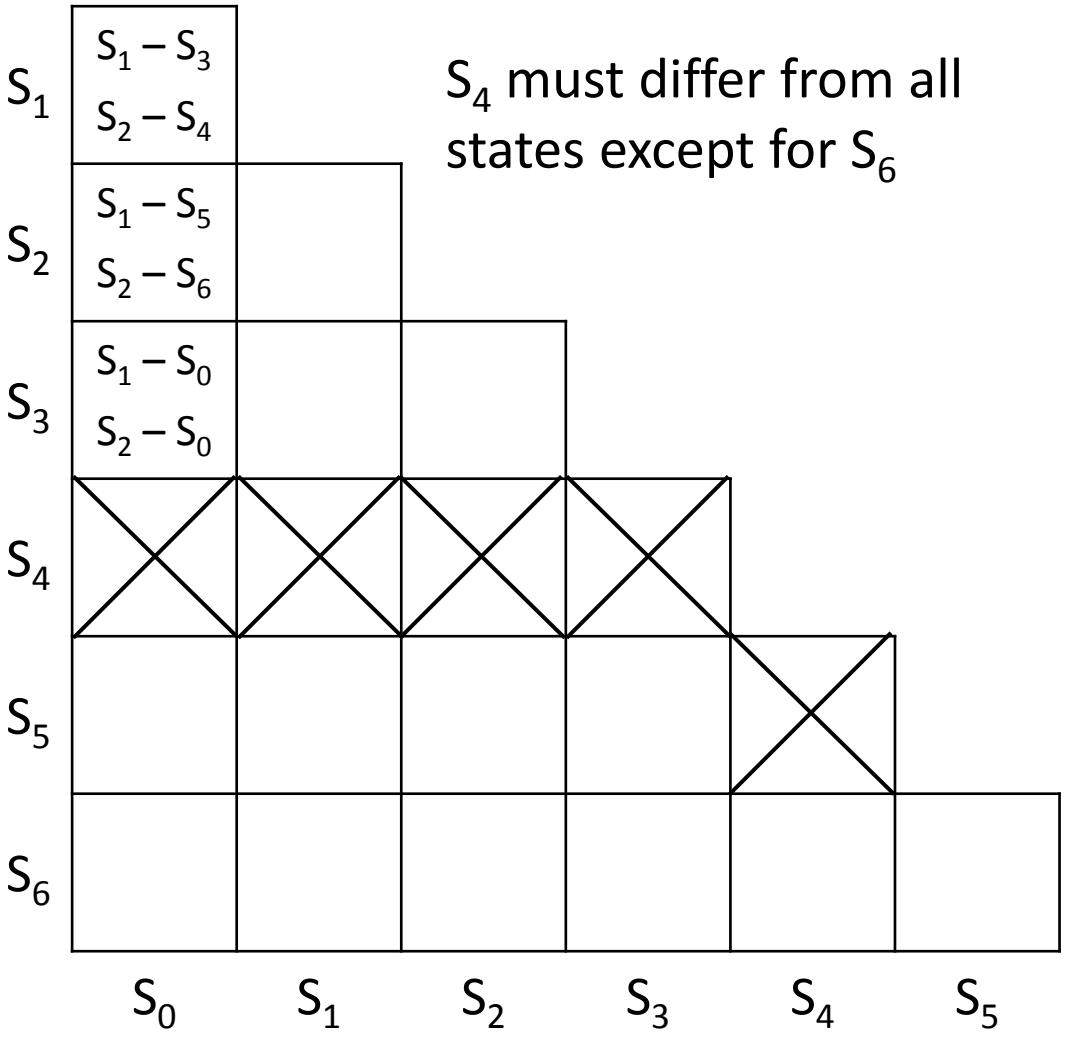


Continue filling out the chart for both the w=0 and w=1 cases

Present State	Next State		Output	
	w = 0	w = 1	w = 0	w = 1
S_0	S_1	S_2	0	0
S_1	S_3	S_4	0	0
S_2	S_5	S_6	0	0
S_3	S_0	S_0	0	0
S_4	S_0	S_0	1	0
S_5	S_0	S_0	0	0
S_6	S_0	S_0	1	0

What about S_4?
Could S_4 be equivalent to S_0, S_1, S_2, S_3, or S_5?

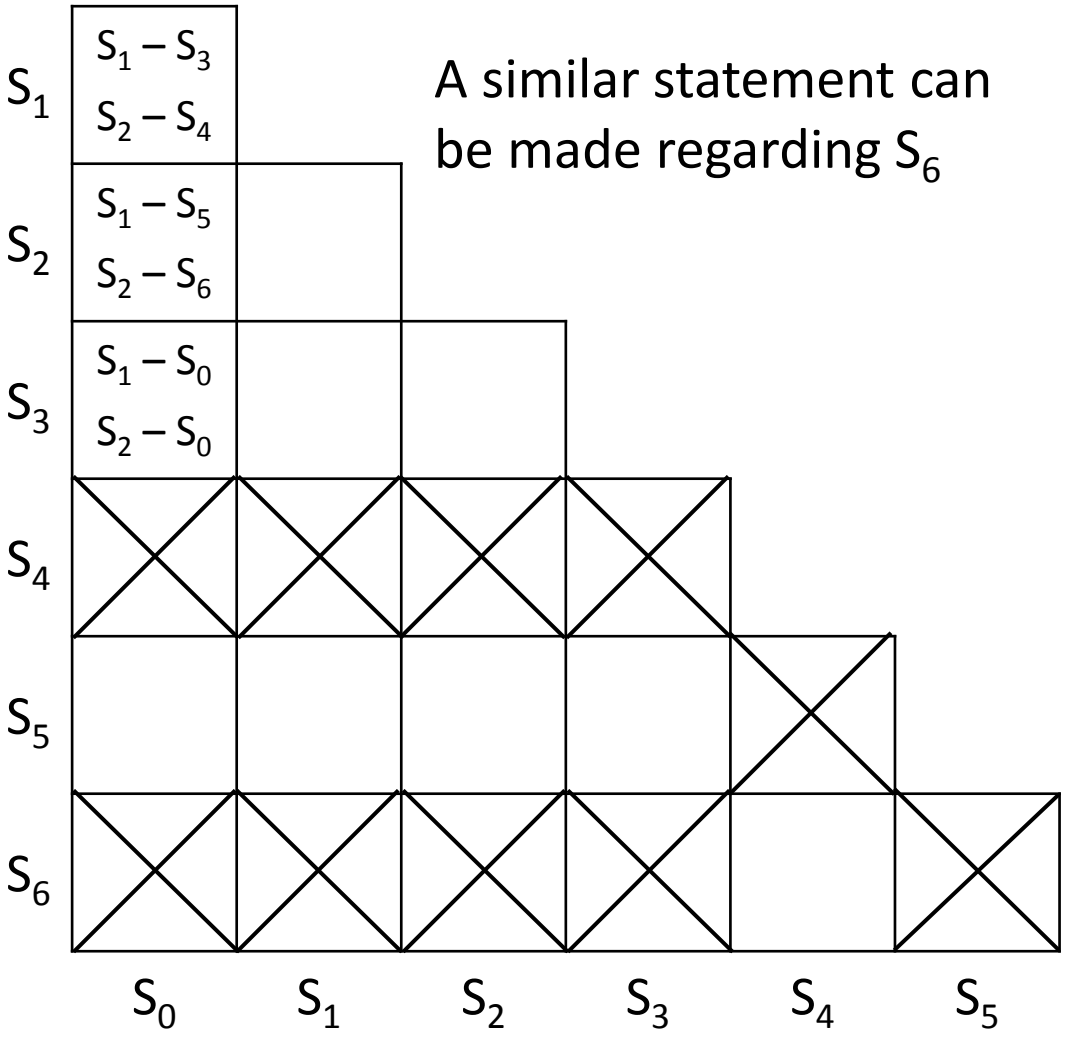
Creating an Implication Chart



S_4 must differ from all states except for S_6

Present State	Next State		Output	
	w = 0	w = 1	w = 0	w = 1
S_0	S_1	S_2	0	0
S_1	S_3	S_4	0	0
S_2	S_5	S_6	0	0
S_3	S_0	S_0	0	0
S_4	S_0	S_0	1	0
S_5	S_0	S_0	0	0
S_6	S_0	S_0	1	0

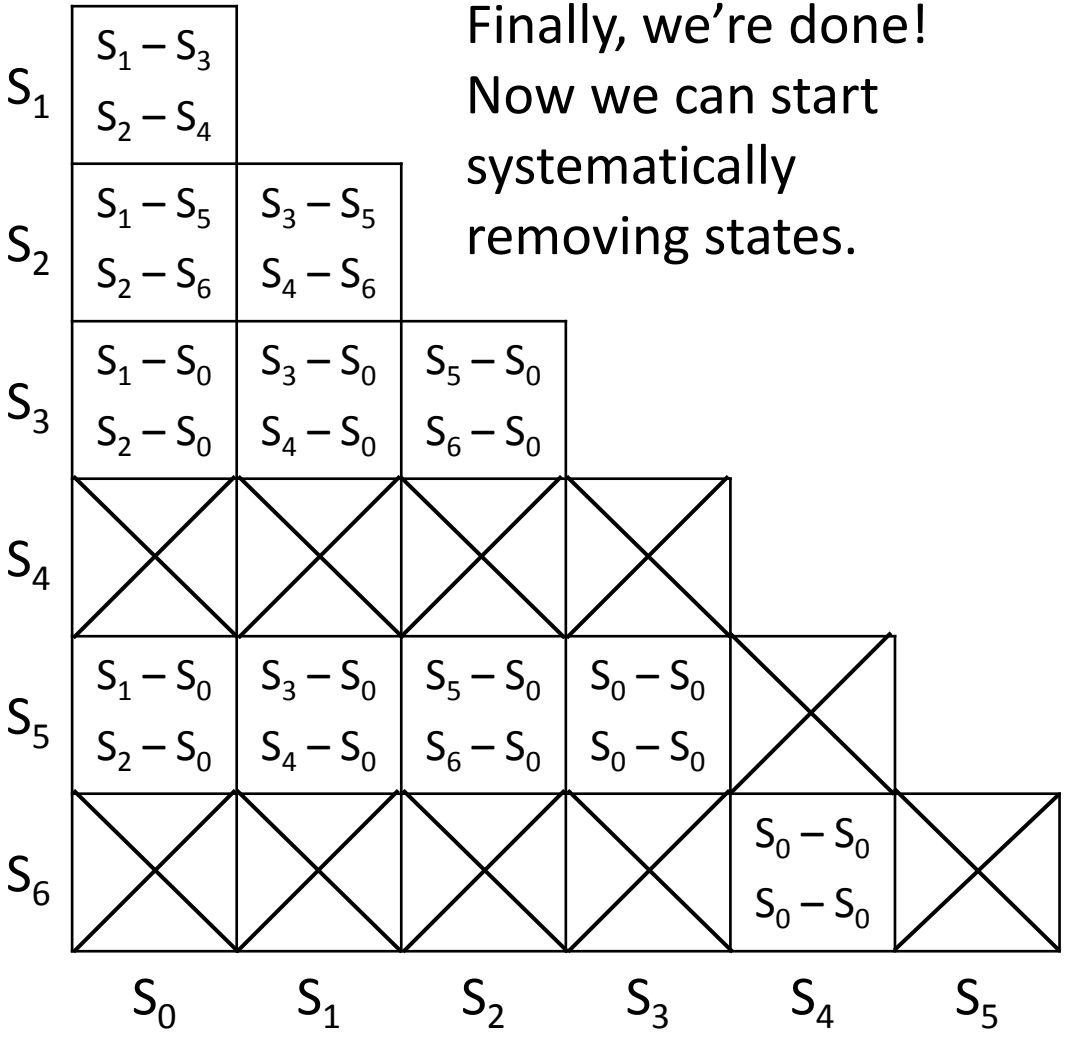
Creating an Implication Chart



A similar statement can be made regarding S_6

Present State	Next State		Output	
	w = 0	w = 1	w = 0	w = 1
S_0	S_1	S_2	0	0
S_1	S_3	S_4	0	0
S_2	S_5	S_6	0	0
S_3	S_0	S_0	0	0
S_4	S_0	S_0	1	0
S_5	S_0	S_0	0	0
S_6	S_0	S_0	1	0

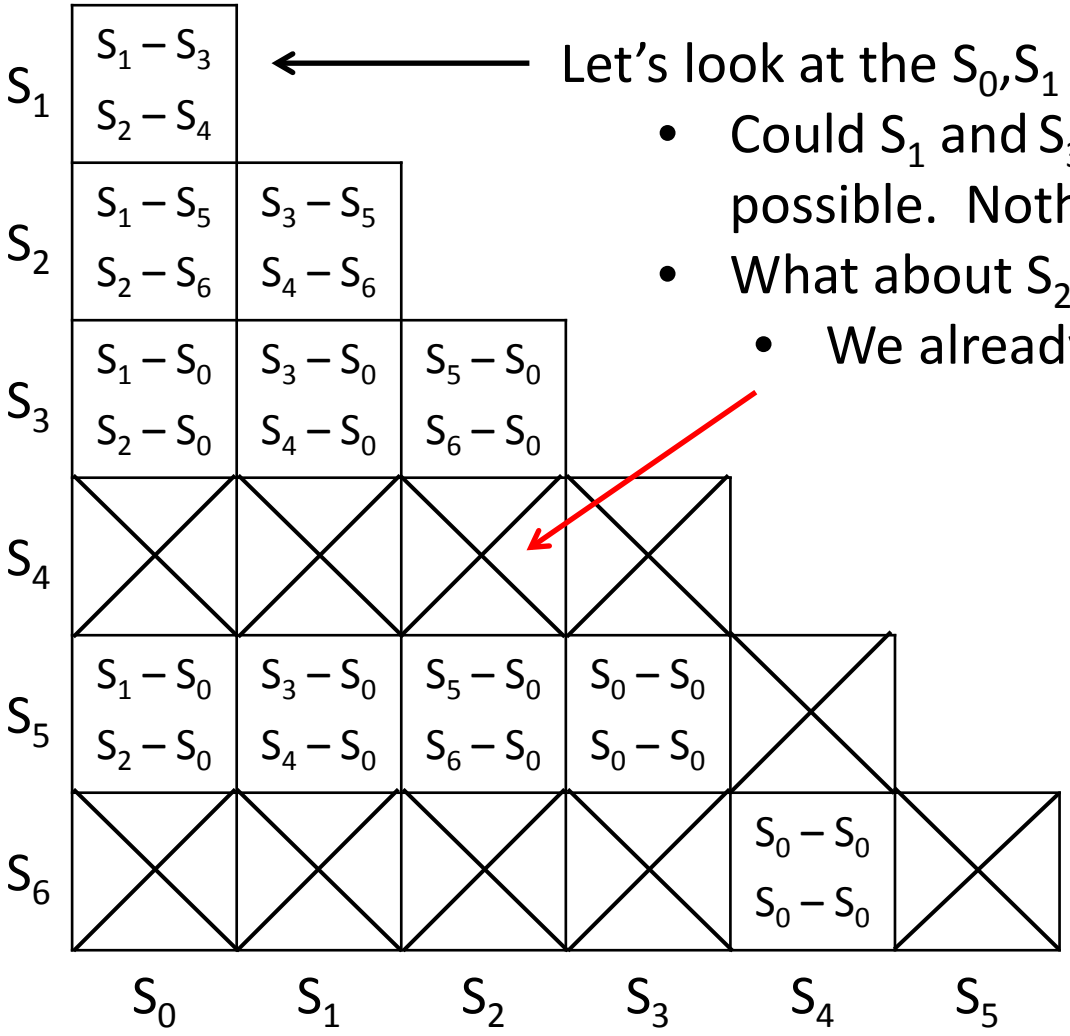
Creating an Implication Chart



Finally, we're done! Now we can start systematically removing states.

Present State	Next State		Output	
	w = 0	w = 1	w = 0	w = 1
S_0	S_1	S_2	0	0
S_1	S_3	S_4	0	0
S_2	S_5	S_6	0	0
S_3	S_0	S_0	0	0
S_4	S_0	S_0	1	0
S_5	S_0	S_0	0	0
S_6	S_0	S_0	1	0

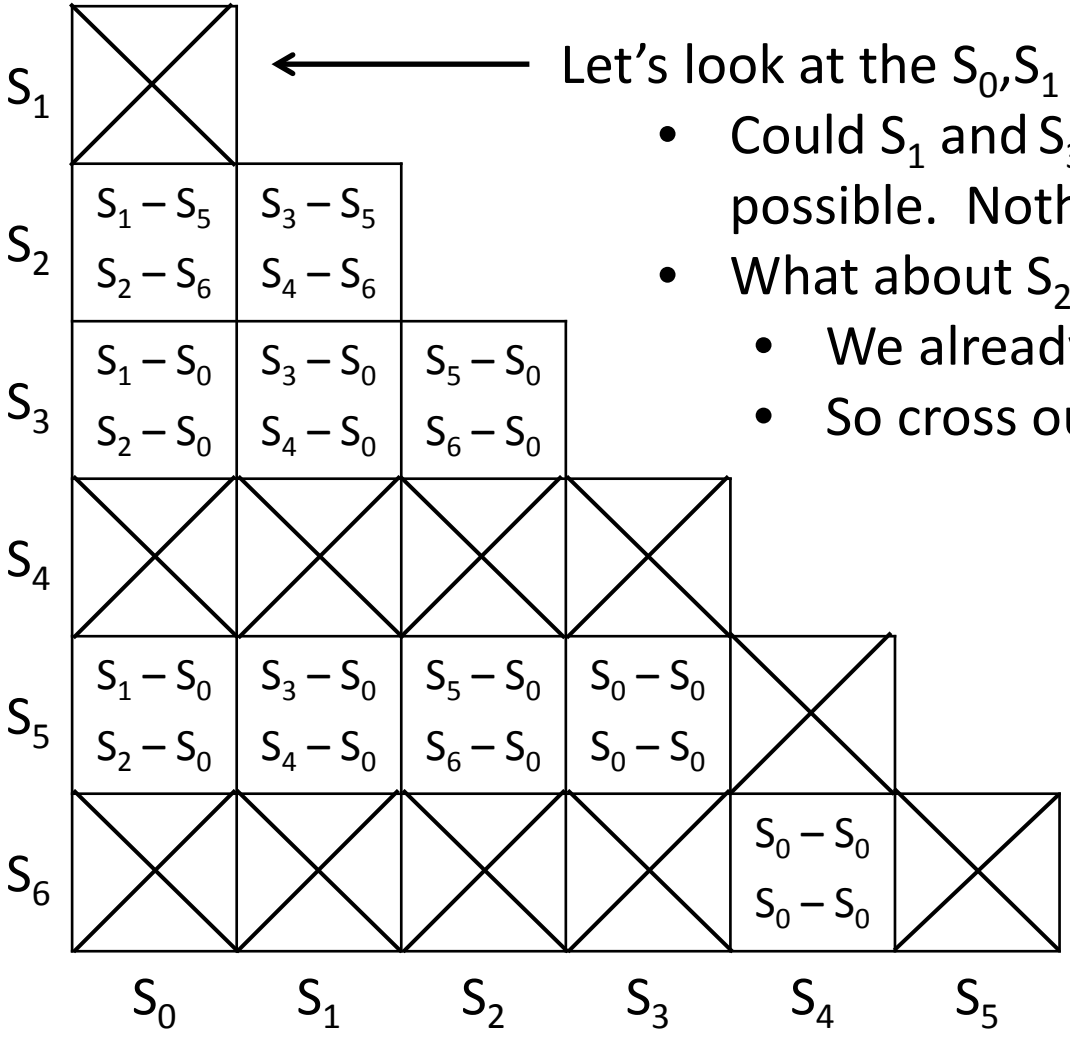
Minimization Procedure



Let's look at the S_0,S_1 entry

- Could S_1 and S_3 still be equivalent? Yes, it's possible. Nothing in the chart contradicts this.
- What about S_2 and S_4?
  - We already know these are not equivalent!

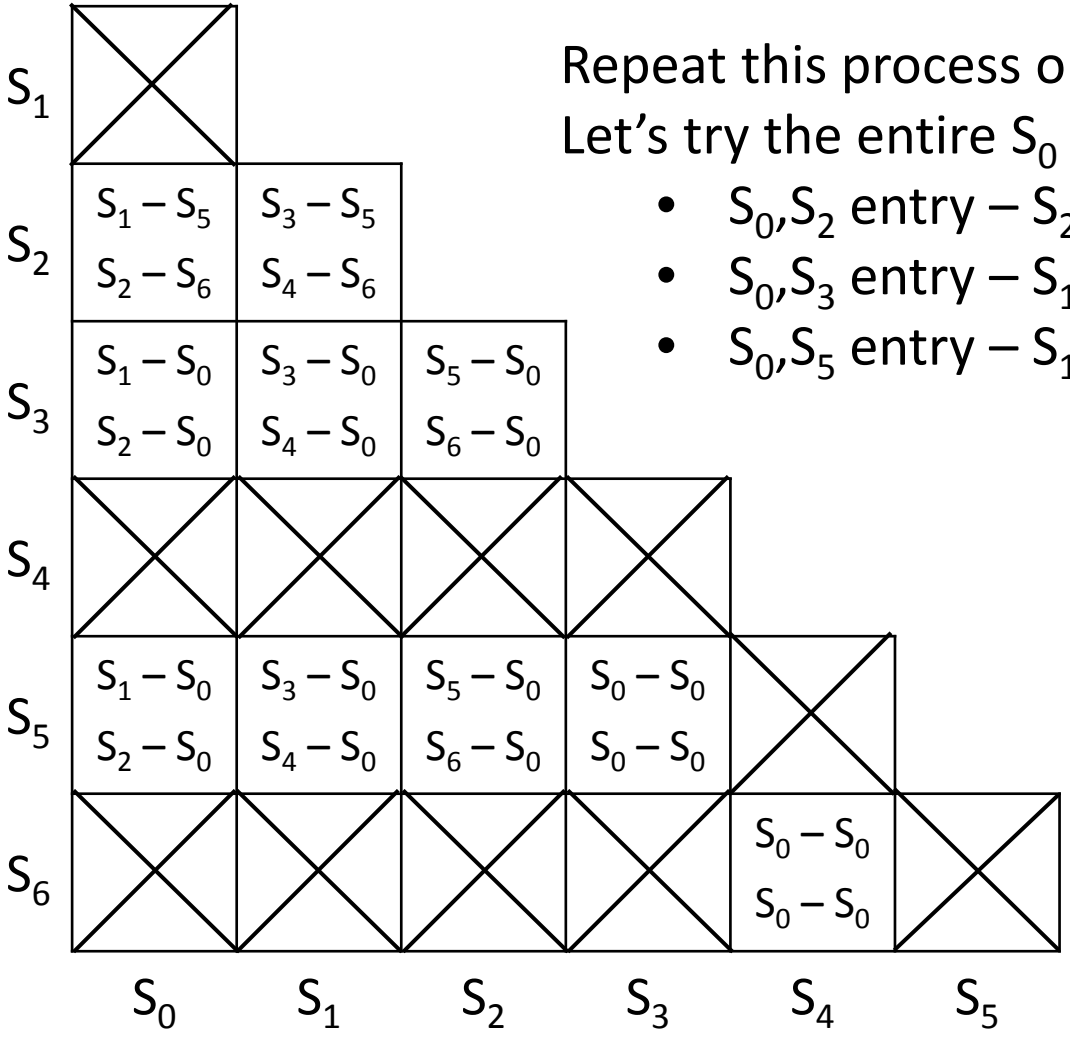
Minimization Procedure



Let's look at the S_0,S_1 entry

- Could S_1 and S_3 still be equivalent? Yes, it's possible. Nothing in the chart contradicts this.
- What about S_2 and S_4?
  - We already know these are not equivalent!
  - So cross out the S_0,S_1 entry

Minimization Procedure

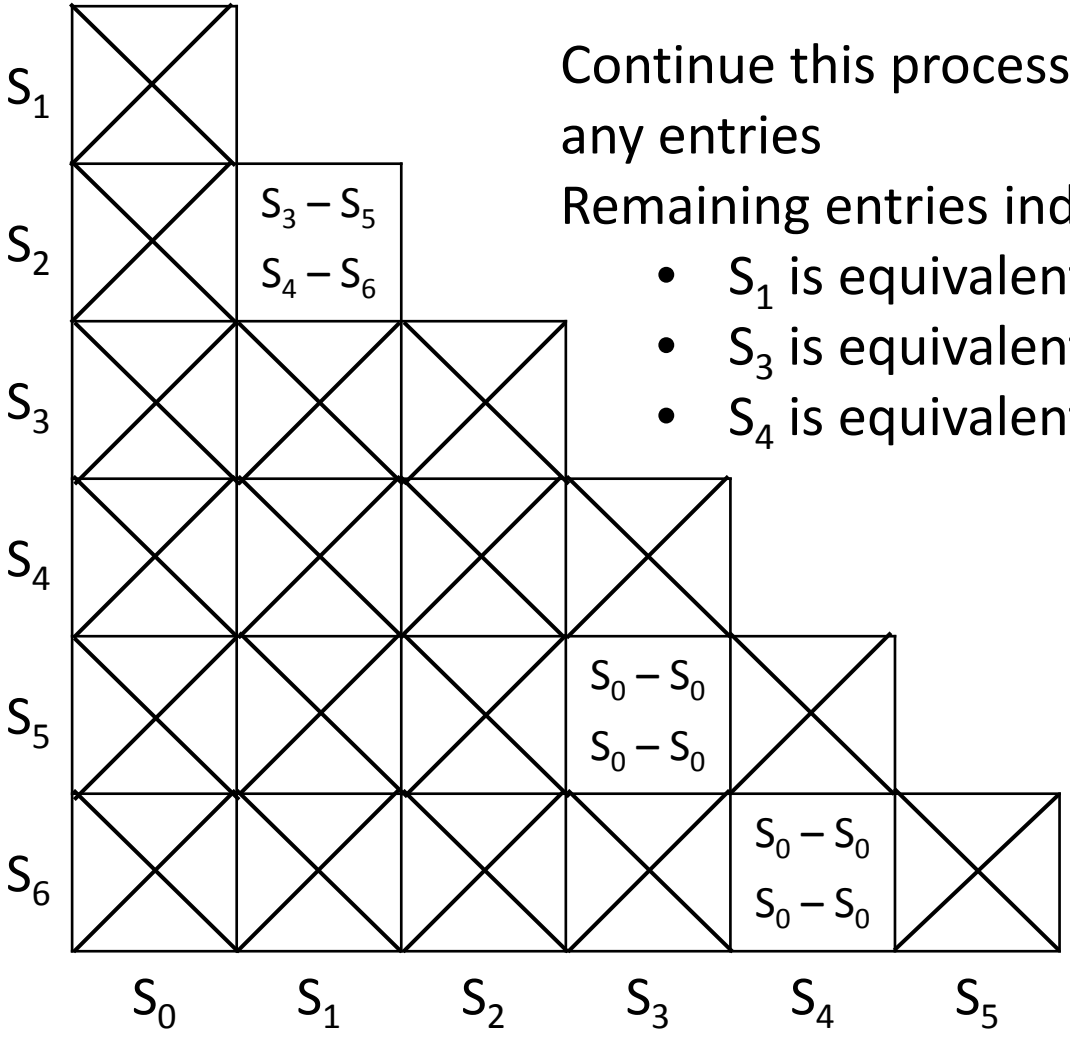


Repeat this process one column (or row) at a time

Let's try the entire S_0 column:

- S_0,S_2 entry – S_2 and S_6 cannot be the same
- S_0,S_3 entry – S_1 and S_0 cannot be the same
- S_0,S_5 entry – S_1 and S_0 cannot be the same

Minimization Procedure



Continue this process until we can no longer remove any entries

Remaining entries indicate equivalent states:

- S_1 is equivalent to S_2
- S_3 is equivalent to S_5
- S_4 is equivalent to S_6

Rewriting the state table

Using this new information, we can now rewrite our state table

- S_1 is equivalent to S_2
- S_3 is equivalent to S_5
- S_4 is equivalent to S_6

Present State	Next State		Output	
	w = 0	w = 1	w = 0	w = 1
S_0	S_1	S_2	0	0
S_1	S_3	S_4	0	0
S_2	S_5	S_6	0	0
S_3	S_0	S_0	0	0
S_4	S_0	S_0	1	0
S_5	S_0	S_0	0	0
S_6	S_0	S_0	1	0

Present State	Next State		Output	
	w = 0	w = 1	w = 0	w = 1
S_0	S_1	S_1	0	0
S_1,S_2	S_3	S_4	0	0
S_3,S_5	S_0	S_0	0	0
S_4,S_6	S_0	S_0	1	0