Let's think a bit more about the FSM abstraction.

If we see an FSM that uses K state bits, what can we say about the number of states in its state transition diagram?

Well, we know the FSM can have at most 2^K states, since that's the number of unique combinations of K bits.

Suppose we connect two FSMs in series, with the outputs of the first FSM serving as the inputs to the second.

This larger system is also an FSM — how many states does it have?

Well, if we don't know the details of either of the component FSMs, the upper bound on the number of states for the larger system is M*N.

This is because it may be possible for the first FSM to be in any of its M states while the second FSM is any of its N states.

Note that the answer doesn't depend on X or Y, the number of input signals to each of the component FSMs.

Wider inputs just mean that there are longer labels on the transitions in the state transition diagram, but doesn't tell us anything about the number of internal states.

Finally, here's a question that's a bit trickier than it seems.

I give you an FSM with two inputs labeled 0 and 1, and one output implemented as a light.

Then I ask you to discover its state transition diagram.

Can you do it?

Just to be a bit more concrete, you experiment for an hour pushing the buttons in a variety of sequences.

Each time you push the 0 button the light turns off if it was on.

And when you push the 1 button the light turns on if it was off, otherwise nothing seems to happen.

What state transition diagram could we draw based on our experiments?

Consider the following two state transition diagrams.

The top diagram describes the behavior we observed in our experiments: pushing 0 turns the light off, pushing 1 turns the light on.

The second diagram appears to do the same thing unless you happened to push the 1 button 4 times in a row!

If we don't have an upper bound on the number of states in the FSM, we can never be sure that we've explored all of its possible behaviors.

But if we do have an upper bound, say, K, on the number of states and we reset the FSM to its initial state, we can discover its behavior.

This is because in a K-state FSM every reachable state can reached in less than K transitions, starting from the initial state. .244 So if we try all the possible K-step input sequences one after the other starting each trial at the initial state, we'll be guaranteed to have visited every state in the machine.

Our answer is also complicated by the observation that FSMs with different number of states may be equivalent.

Here are two FSMs, one with 2 states, one with 5 states.

Are they different?

Well, not in any practical sense.

Since the FSMs are externally indistinguishable, we can use them interchangeably.

We say that two FSMs are equivalent if and only if every input sequence yields identical output sequences from both FSMs.

So as engineers, if we have an FSM we would like to find the the simplest (and hence the least expensive) equivalent FSM.

We'll talk about how to find smaller equivalent FSMs in the context of our next example.