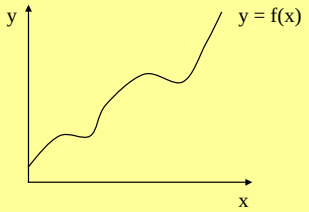
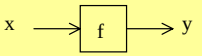
State Machine Basics

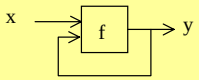
- ✘ Mathematical Origin
- ✘ A Look at System Theory
- ✘ Software Perspective
- ✘ Process Perspective (for Real time SW Development)
- ✘ Some Practices

Functions





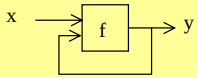
Back-coupled Systems



$y = f(x,y) = f(x,f(x,y)) = f(x,f(x,f(x,y))) = \ldots\ldots\ldots$

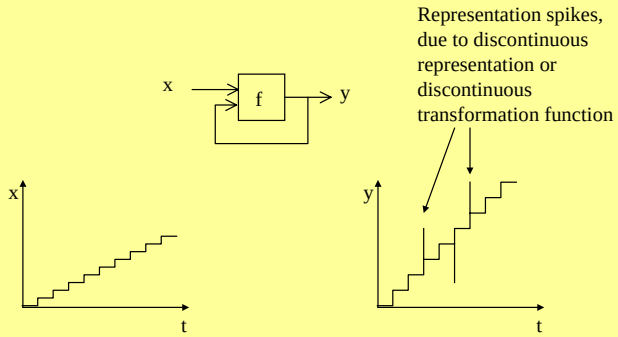
History Dependent!

Continuous Transformation, Continuous Representation



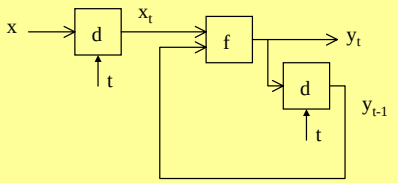
- Linear transformation
- Non-linear transformation

Discrete Representation



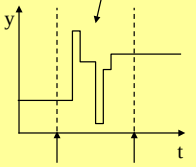
Representation spikes, due to discontinuous representation or discontinuous transformation function

Discrete Representation, Discrete Time

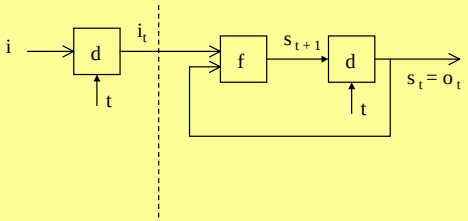


$y_t = f(x_t, y_{t-1})$

Representation spikes



Simple Automata (State Machine)



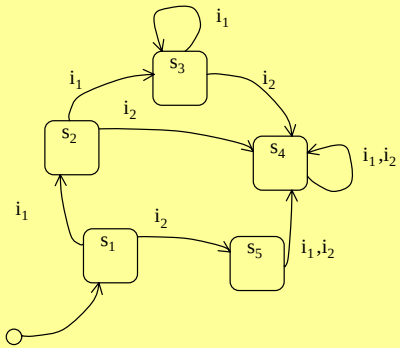
$s_{t+1} = f(i_t, s_t)$

$o_t = s_t$

Finite State Machine (FSM)

An FSM is an automata, where the set of possible values for s is finite.

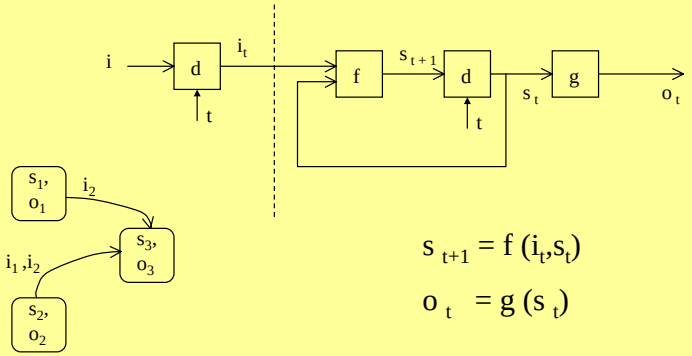
FSM State Chart



All automata:
$s_{t+1} = f(i_t, s_t)$

Simple automata:
$o_t = s_t$

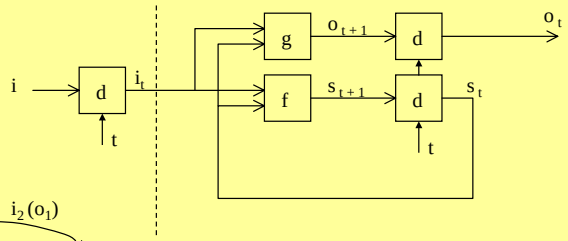
Moore Automata



$s_{t+1} = f(i_t, s_t)$

$o_t = g(s_t)$

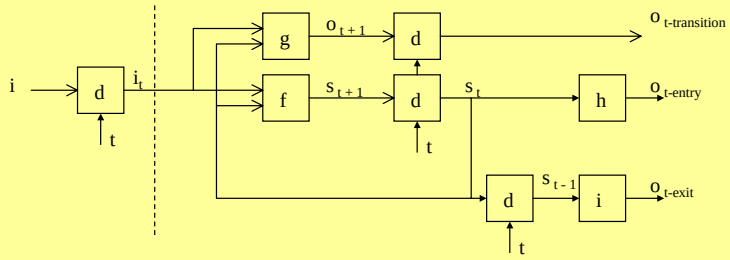
Mealy Automata



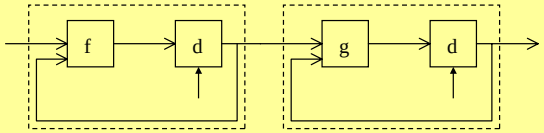
$s_{t+1} = f(i_t, s_t)$

$o_t = g(i_t, s_t, s_{t-1})$

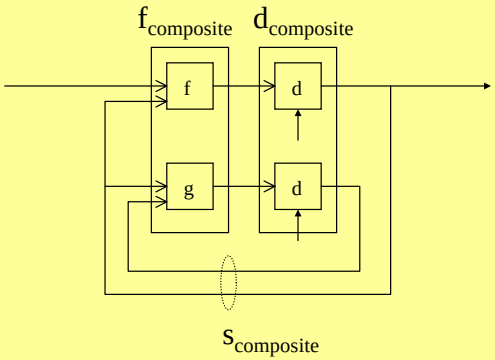
Moore/Mealy Hybrid



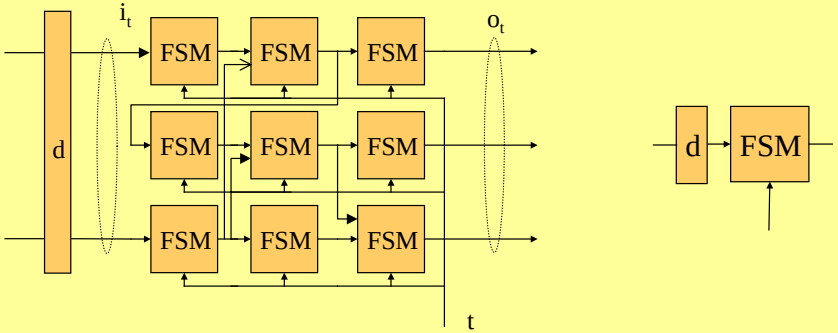
Cascadability



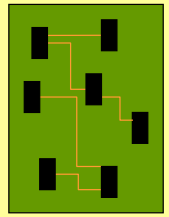
Composability

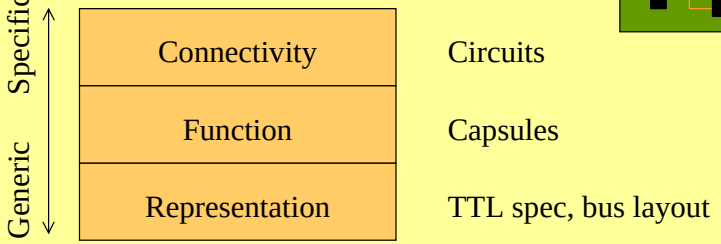


Time Synchronous Systems



"PCB" Metaphor



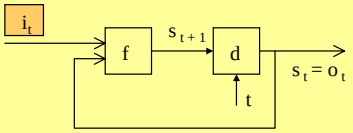


Realizing State Machines in Software:

Event Driven State Machines

epltos@epl.ericsson.se

Input
Something happened



- ✘ Point of time
- ✘ Name of event
- ✘ Data
- ✘ Boolean aspect of data

temp > *tempLimit*

Influence:

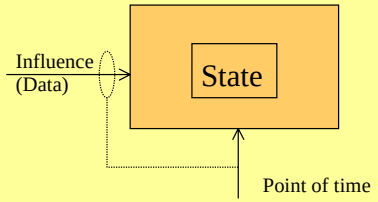
External boolean information that is input to a system in a non divisible* operation

```
tempChanged ( bool sendAlarm )
{
   temp = getTemp();
   if ( temp > tempLimit ) ….
   ….
}
```

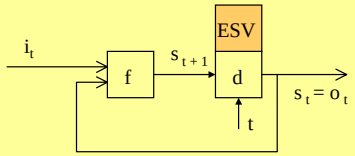
*) The operation is non divisible if the state of the system does not change in between different information items are received by the system.

Input (cont'd)

- ✘ Point of time
- ✘ Influence
- ✘ Data



Extended State Variables

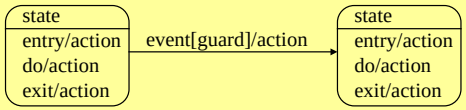


Arbitrary data types with a boolean aspect

if ((**queue.***isFull()*)) ….

if (*isElvisPresley*(**person**)) ….

UML State Diagrams (Harel State Charts)



Questions:

How do events relate to input?
How do actions relate to output?
What about do/?