



STATECHARTS: A Visual Formalism for Complex Systems

Alessandra Nardi

Fan Mo

EE249 Discussion Section

September 14, 1999



STATECHARTS - 1

- Visual formalism for the specification of reactive systems
 - not a support, but the way itself
- Extension of State Transition Diagram of Finite State Machines
 - State/Event Description



STATECHARTS - 2

- Flat FSM - too complex
- Mechanisms to enhance the descriptive power:
 - Hierarchy
 - Orthogonality

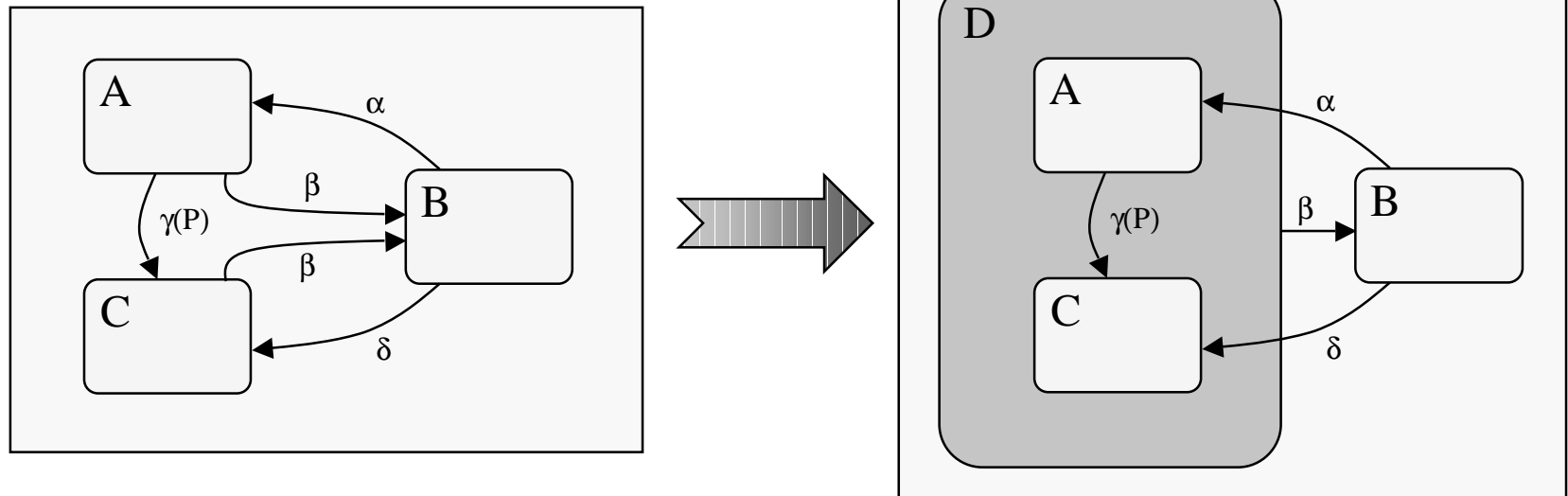


HIERARCHY - 1

- Boxes denotes states
- Encapsulation express hierarchy
 - different state levels
- Arrows can originate and terminate at any level
- Arrows labeled with events
 - optionally: parenthesized conditions

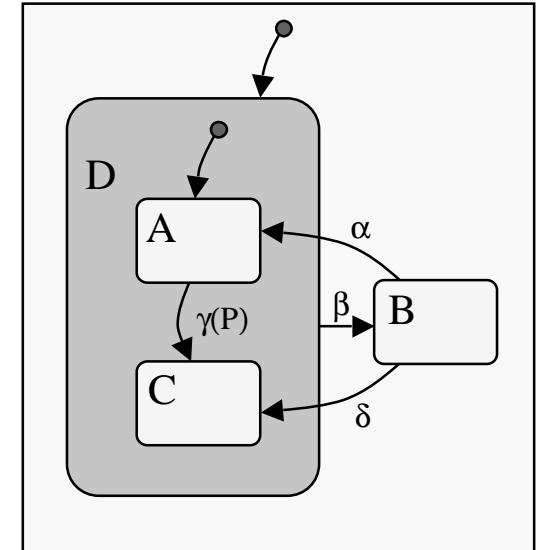
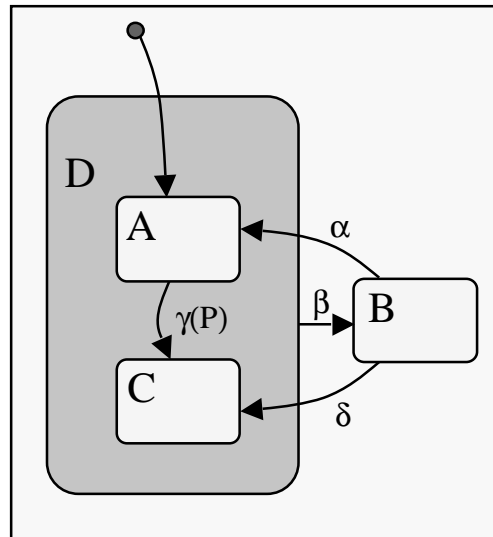
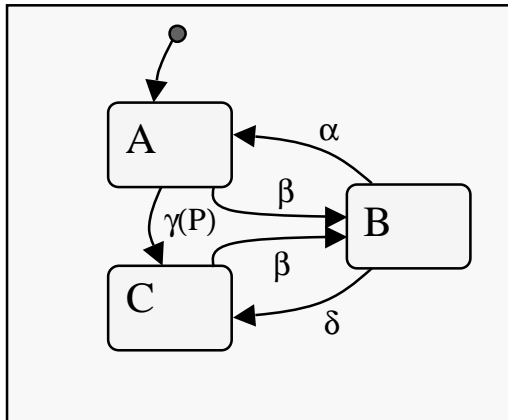
HIERARCHY - 2

- XOR Decomposition
- Way to economize arrows



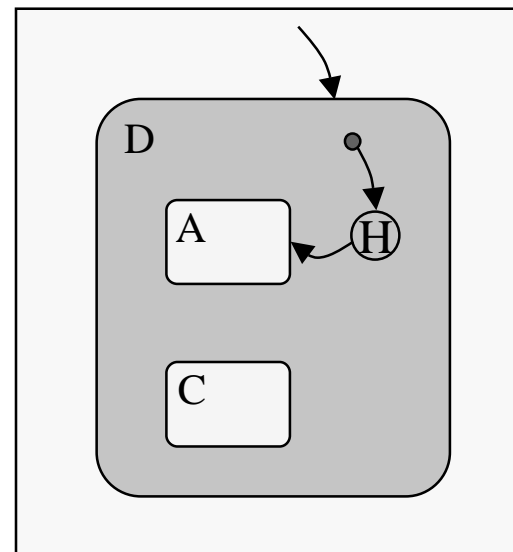
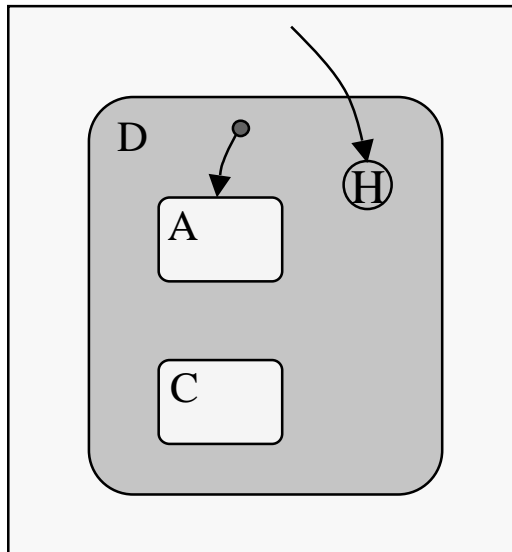
HIERARCHY - 3

■ Default states



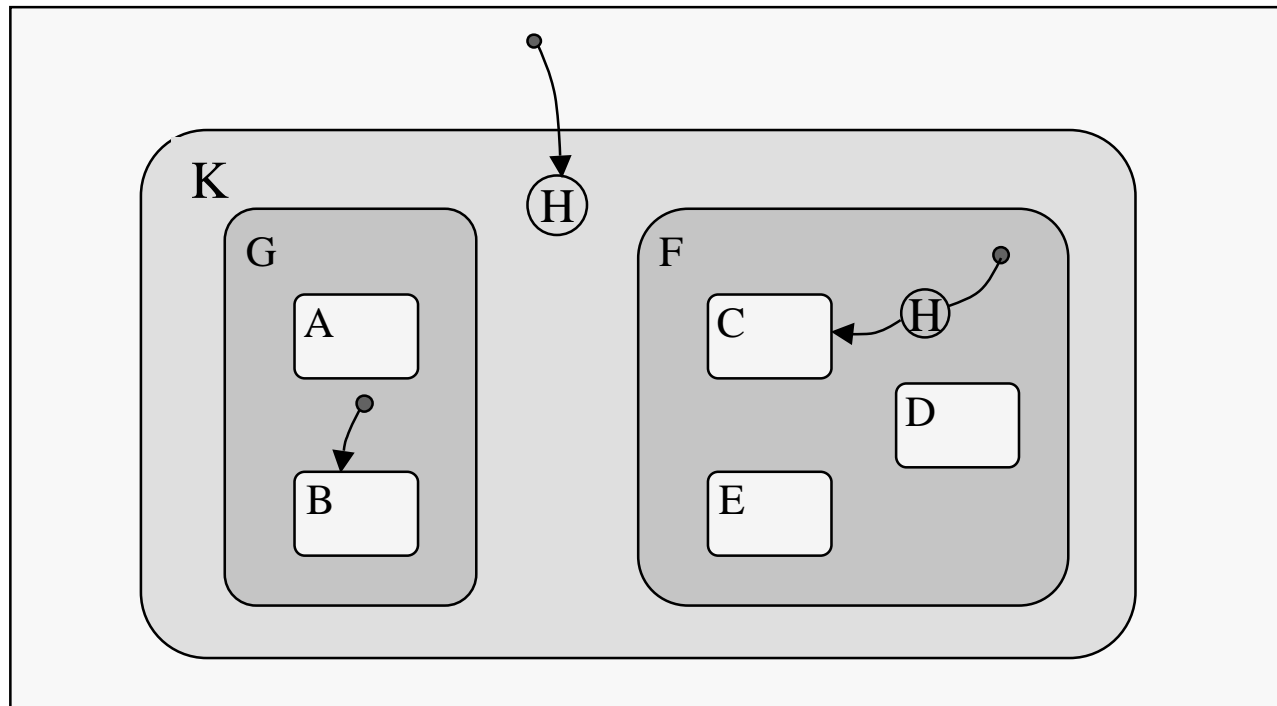
HIERARCHY - 4

- Entering a group of states - History
 - State most recently visited



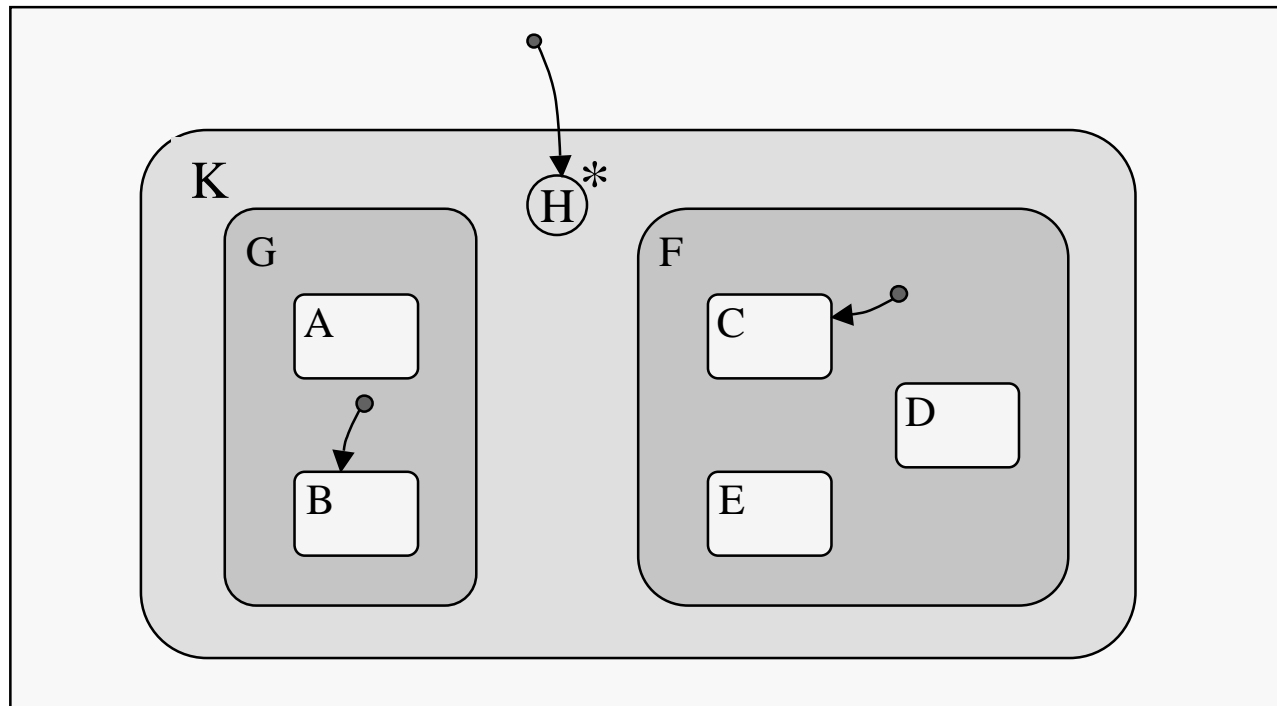
HIERARCHY - 5

- Entering a group of states - History
 - Only at one level



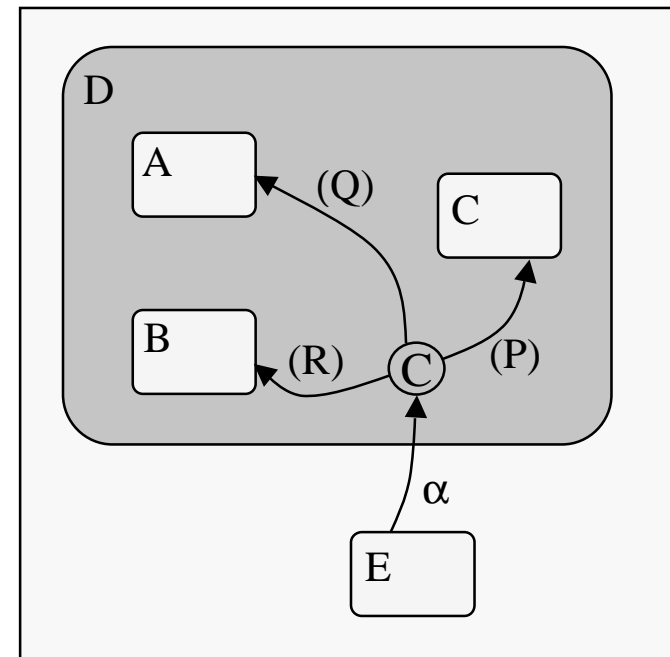
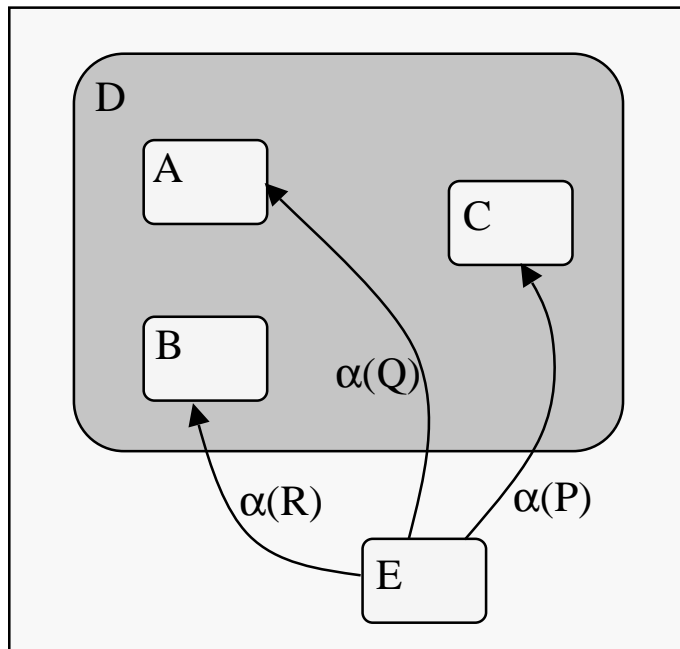
HIERARCHY - 6

- Entering a group of states - History*
 - through the hierarchy



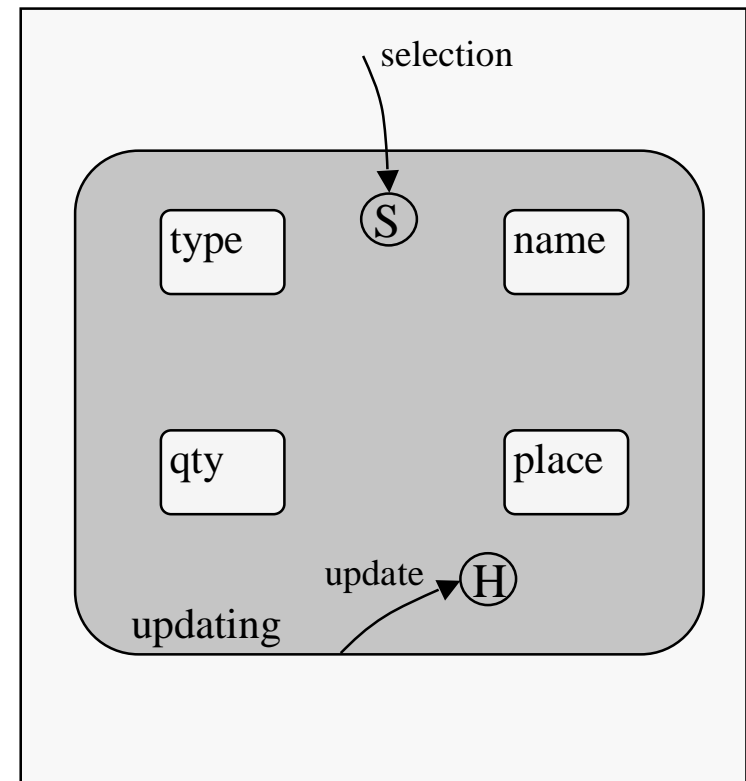
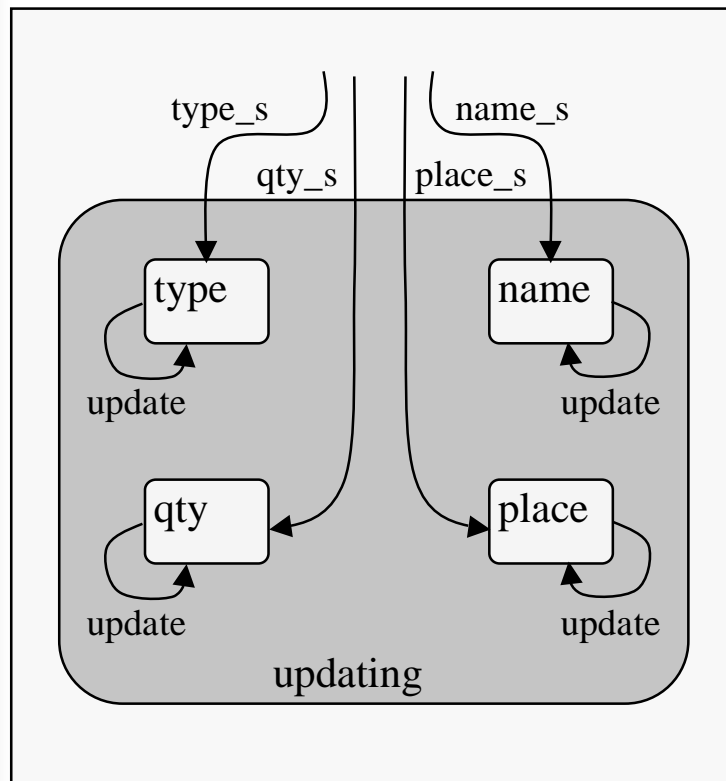
HIERARCHY - 7

■ Entering a group of states - Condition



HIERARCHY - 8

■ Entering a group of states - Selection



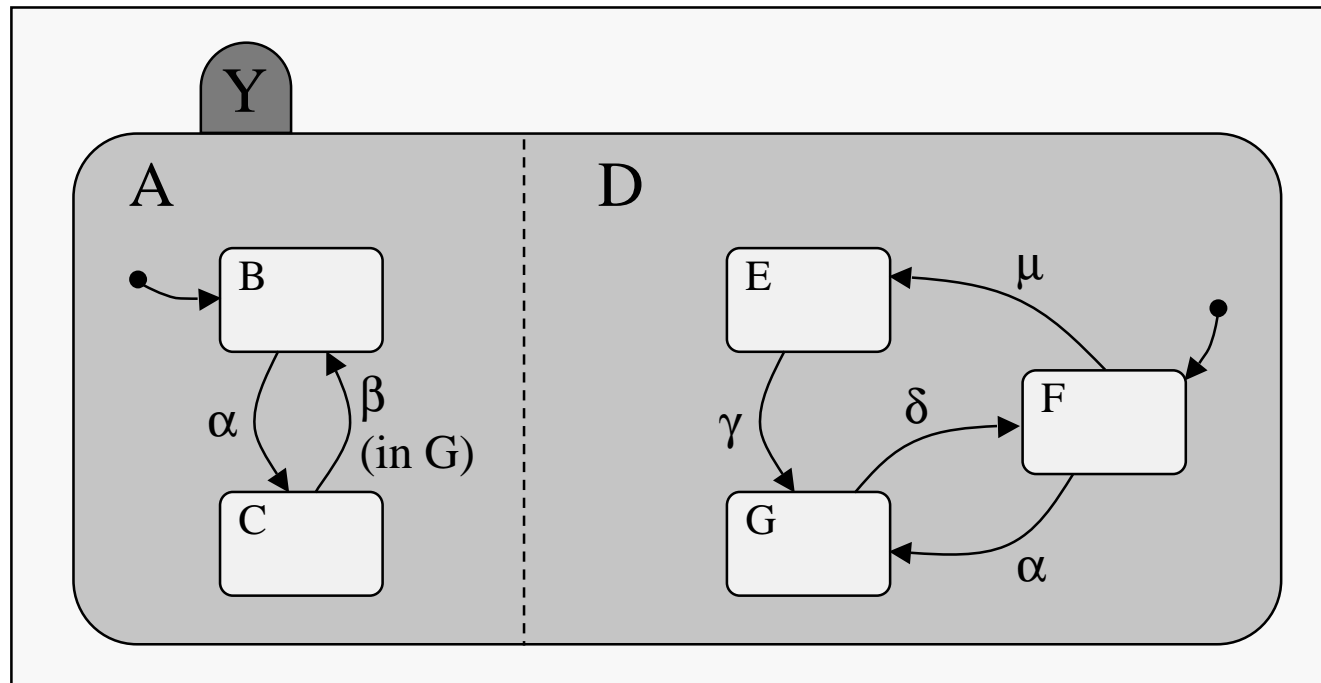


HIERARCHY - Summary

- XOR Decomposition
- Default states
- Entering a group of states
 - History
 - History*
 - Condition
 - Selection

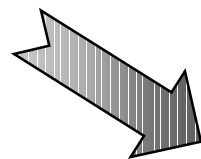
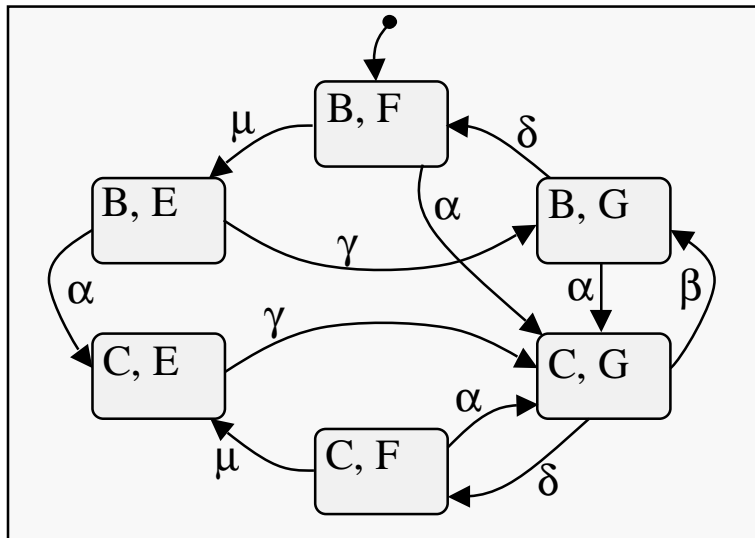
ORTHOGONALITY - 1

- Boxes splitted by a dashed line express orthogonality
- Independency and/or Concurrency

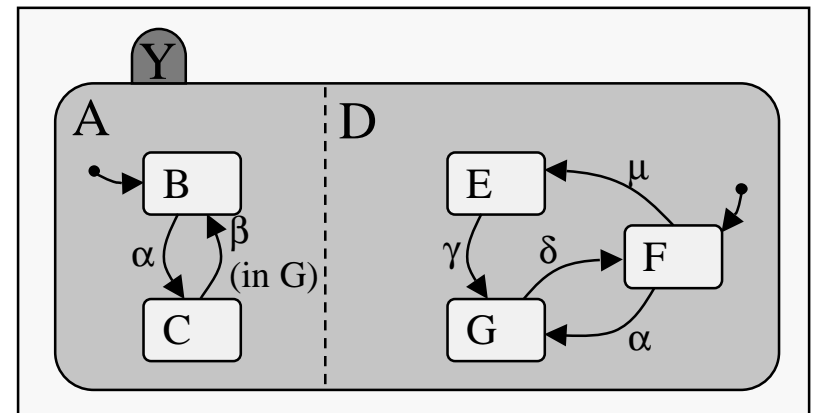


ORTHOGONALITY - 2

- AND Decomposition
- Way to economize states



$$N = N_1 \times N_2$$



$$N = N_1 + N_2$$



MORE FEATURES....

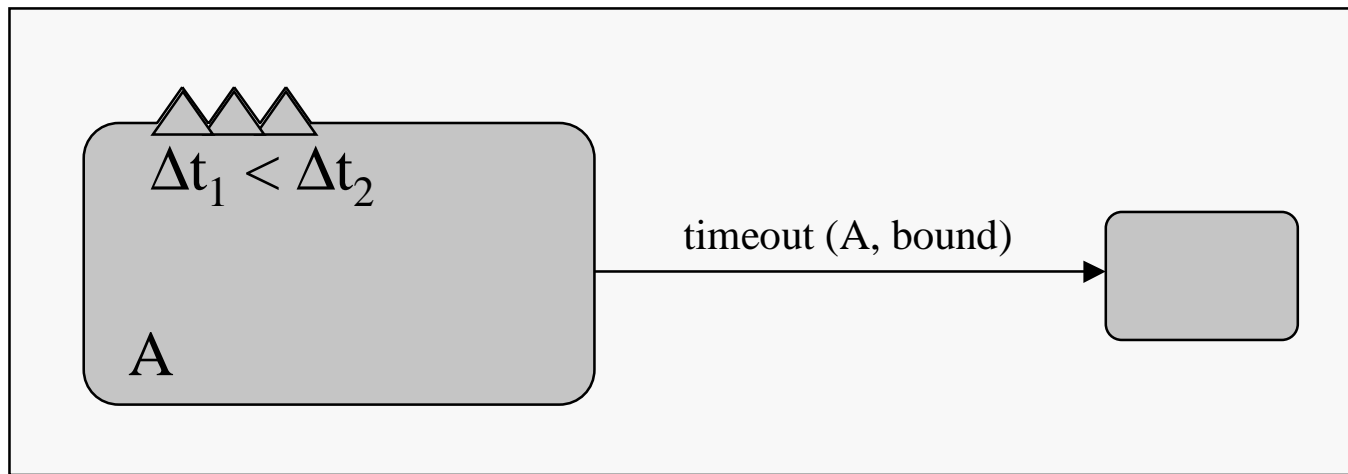
- Time Constraints
- Unclustering
- Actions and Activities

TIME CONSTRAINTS

■ Delays

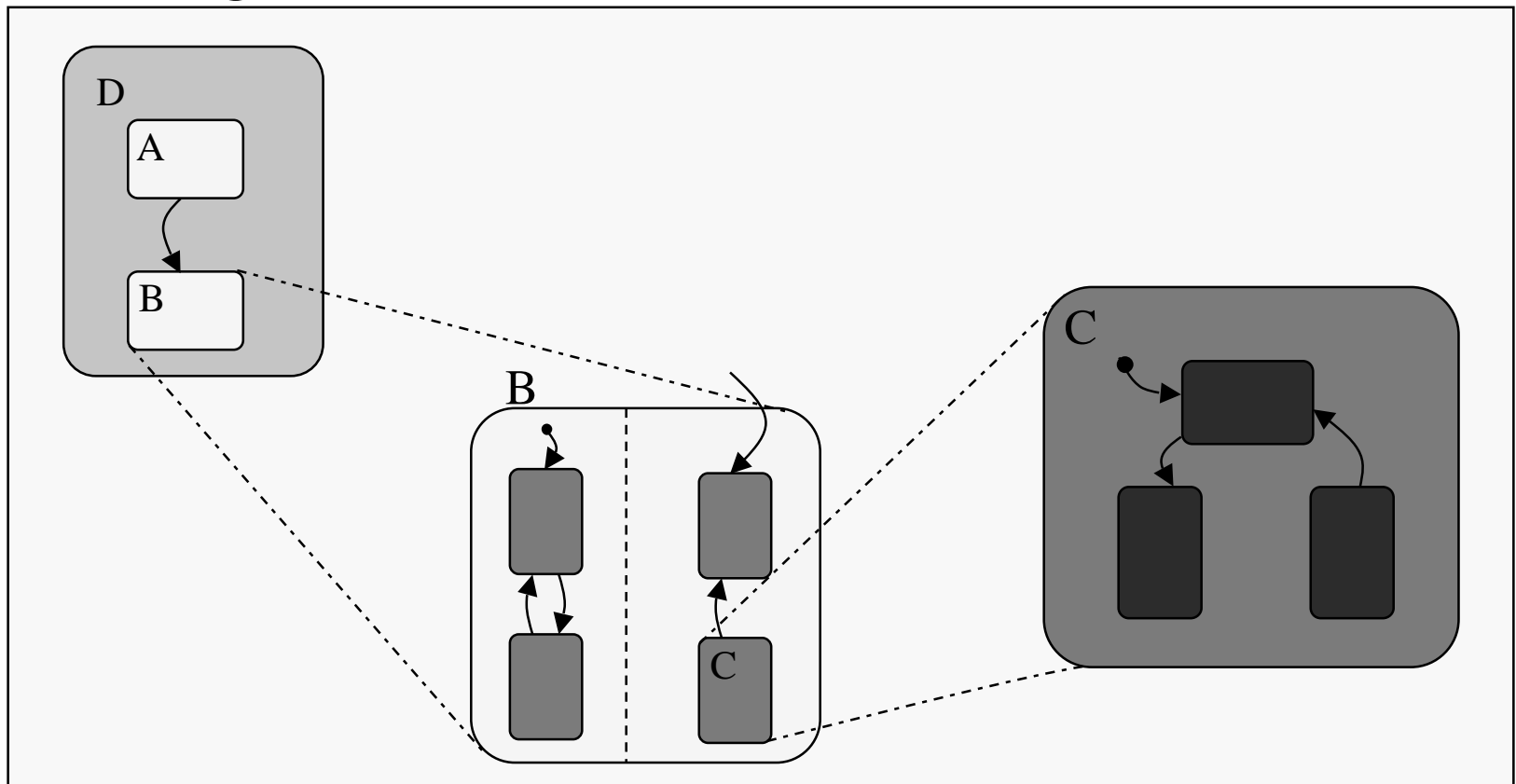
– timeout(event, number)

■ Time Bounds



UNCLUSTERING

- Laying out parts outside the natural neighborhood



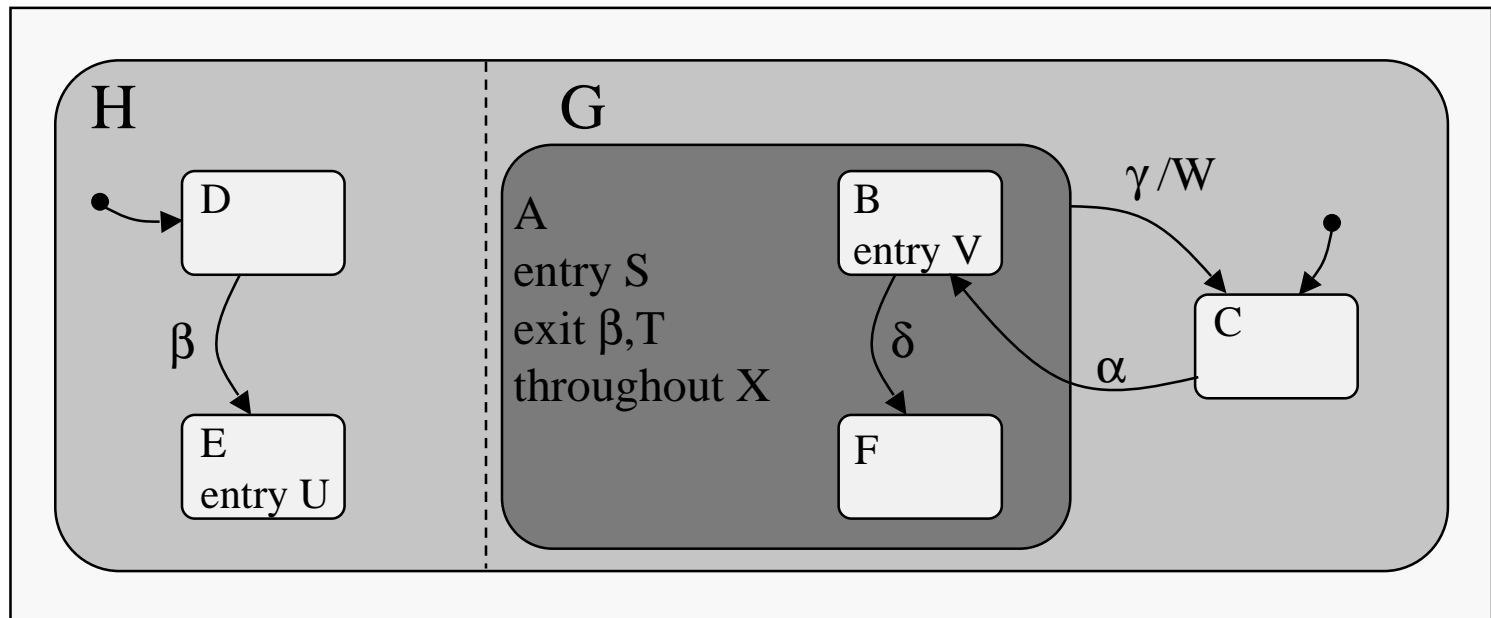


ACTIONS AND ACTIVITIES

- Connection with the “real world”
 - ACTIONS (zero time)
 - ACTIVITIES (non-zero time)
- Actions to control activity X
 - start(X)
 - stop(X)

ACTIONS AND ACTIVITIES

■ Extension of State Transition Diagram





SUMMARY

- Extended State Transition Diagram
 - Hierarchy
 - Orthogonality
- Default states
 - enter by (history, condition, selection)
- Time constraints
- Actions and Activities

WATCH





WATCH INTERFACE

■ OUTPUT

- ❑ Main Display Area, 4 Smaller Display Areas
- ❑ Two-Tone Beeper

■ INPUT

- ❑ 4 Control Buttons: a, b, c, and d
- ❑ Battery



WATCH FUNCTION

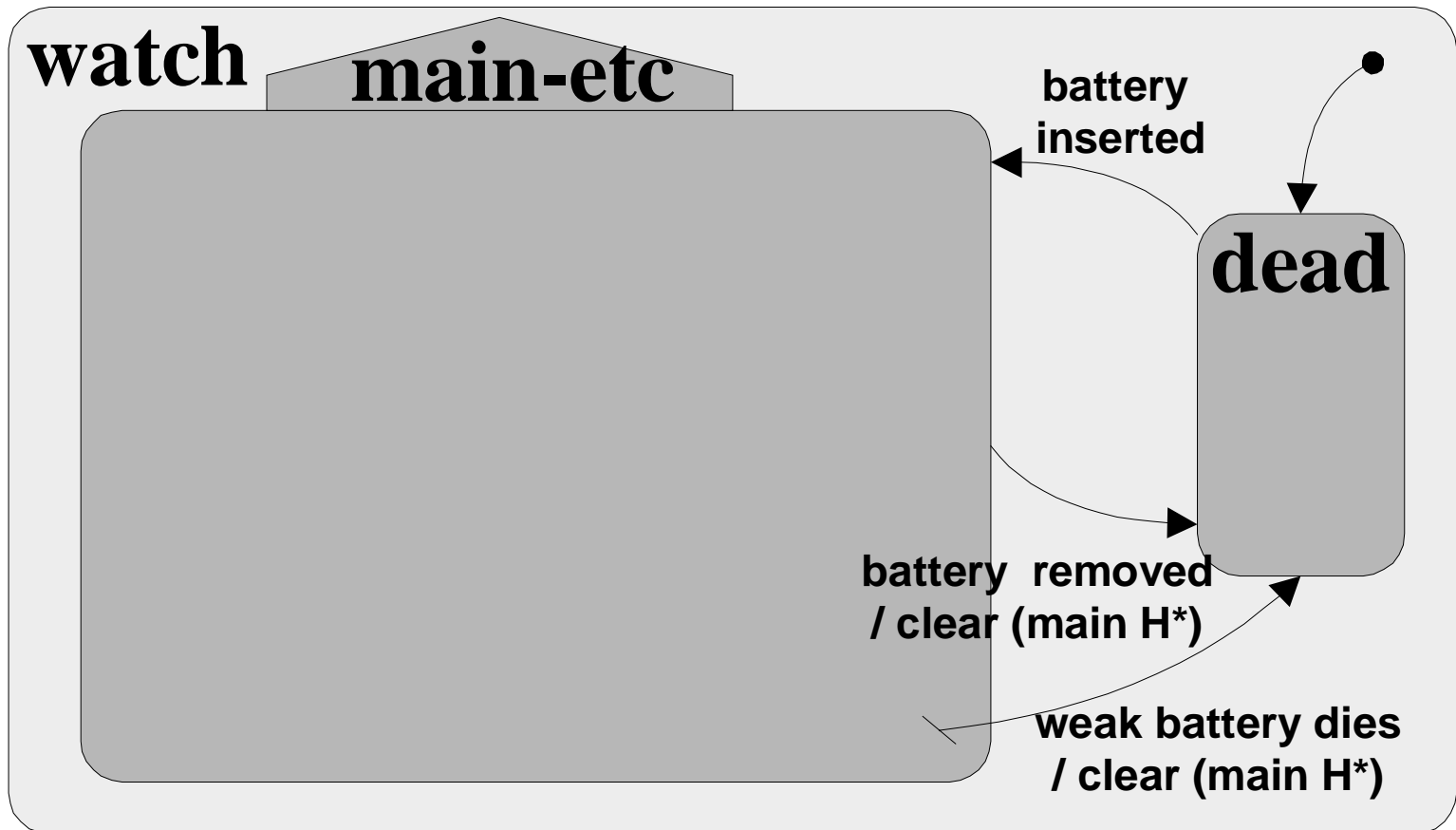
- Display Time (am/pm or 24 hour)
or Display Date (day, month, date of week)
- Two Independent Alarms
- A Stopwatch (lap and regular modes, and a 1/100 s display)
- A Light for Illumination
- A Weak Battery Blinking Indication
- Beeper Test



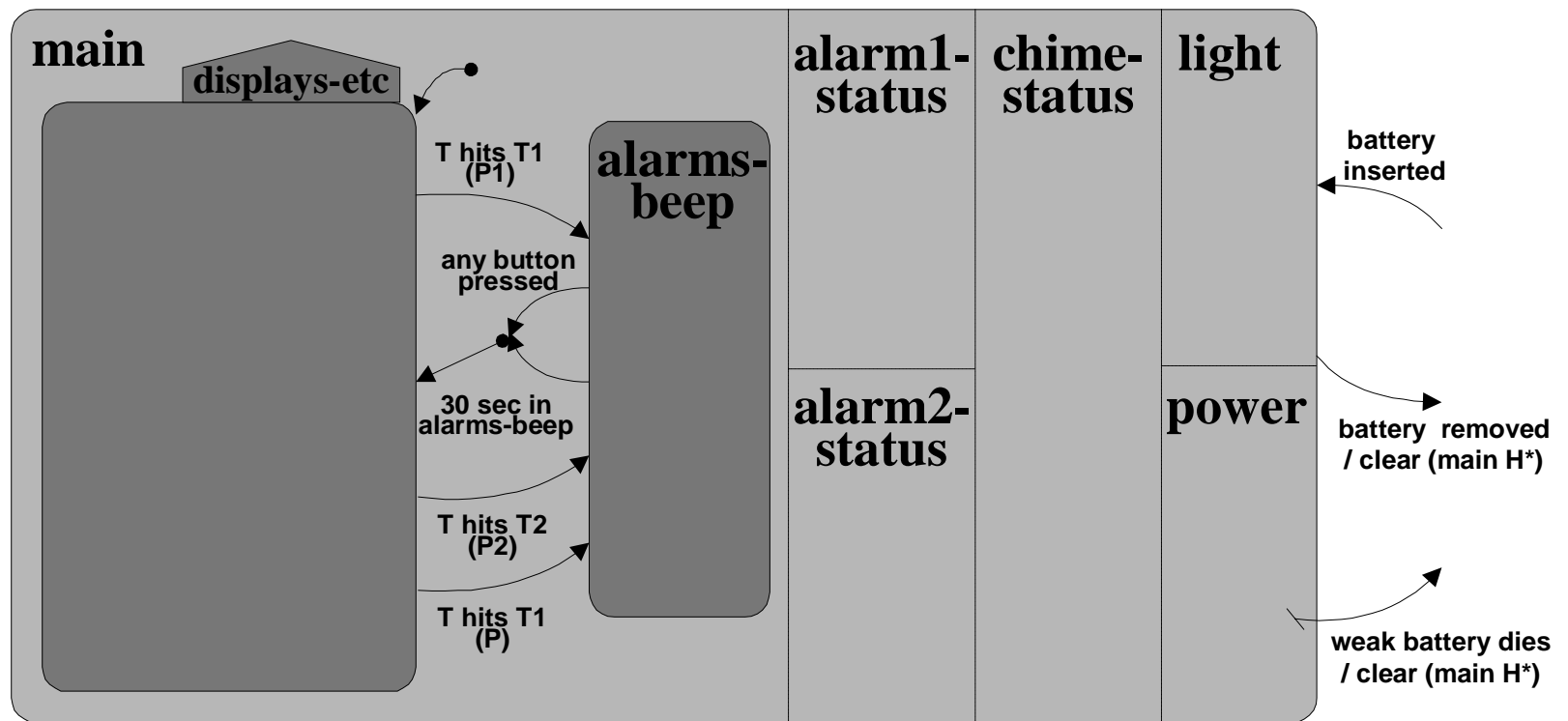
WATCH USER'S GUIDE

Button	Current State	Function
a	Normal	Select displays (time/date, alarm setting, chime setting, stopwatch)
b	Normal Update or Setting Stopwatch	Press with button d enter Beep-test Exit current update or setting Run/Stop
c	Update or Setting	Select Update Content
d	Time Alarm Setting Chime Setting Normal Stopwatch Run Stopwatch Stop Update or Setting	Time/Date switch Turn on/off Turn on/off Press with button b to enter Beep-test Regular/Lap display Clear stopwatch to zero Resume

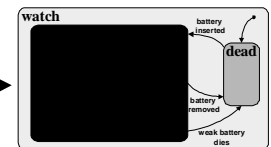
STATECHART: watch



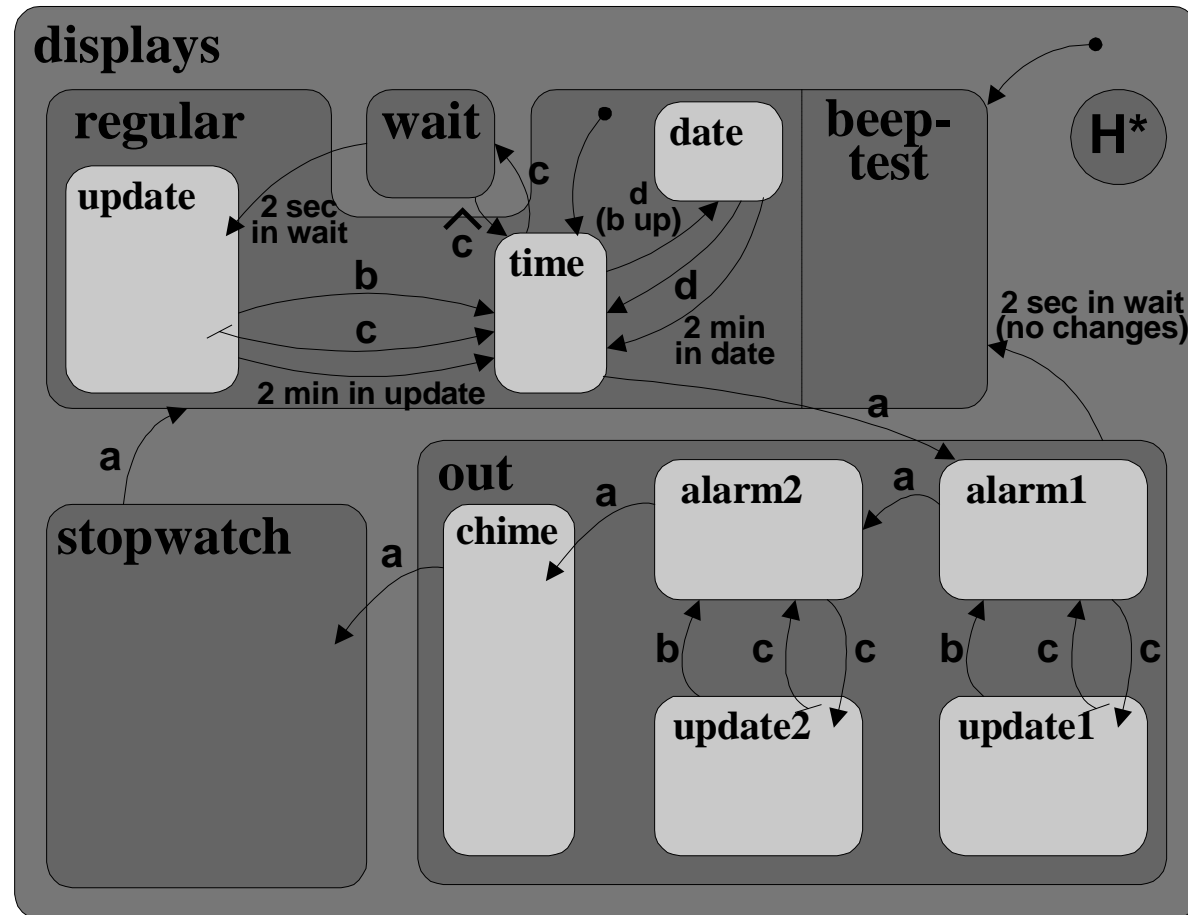
STATECHART: main//alarm1-status//alarm2-status// chime-status//light//power



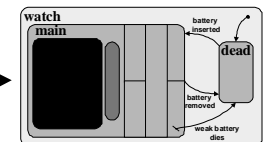
We are here →



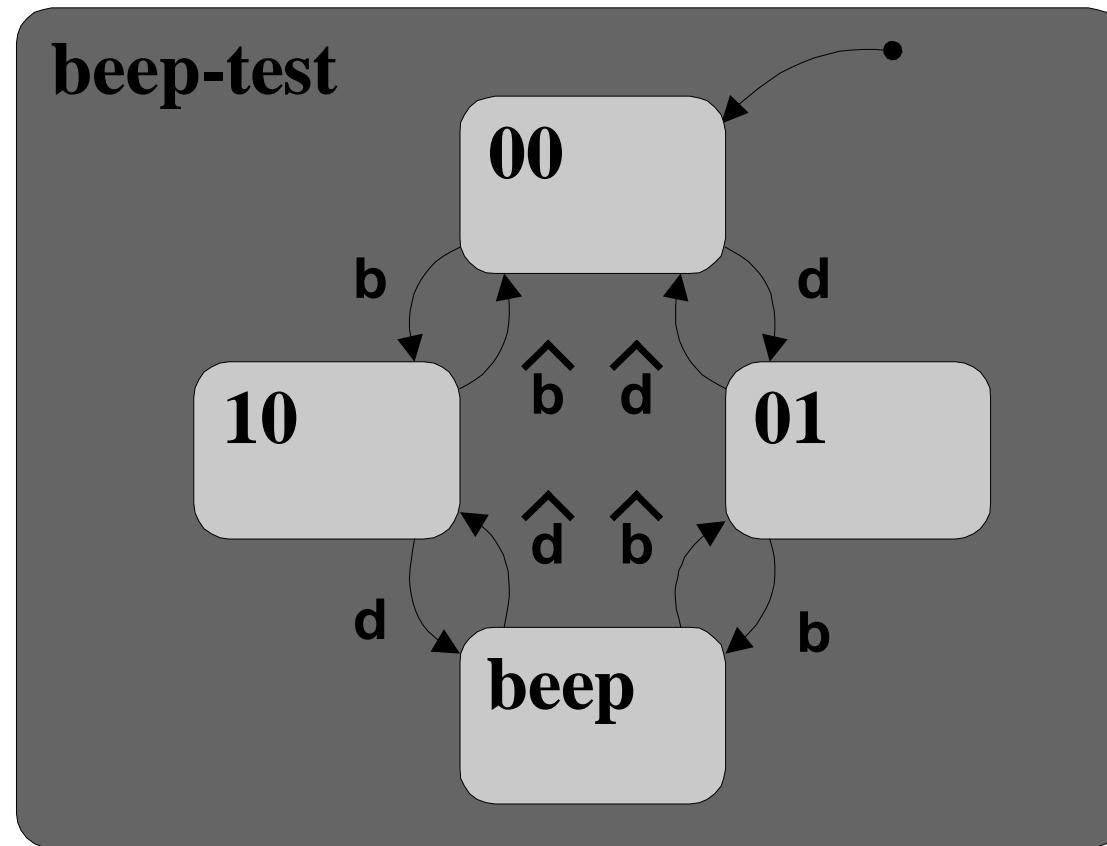
STATECHART: displays



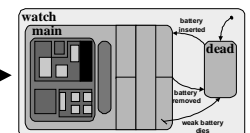
We are here →



STATECHART: beep-test

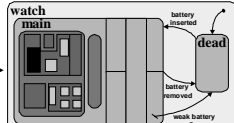


We are here →

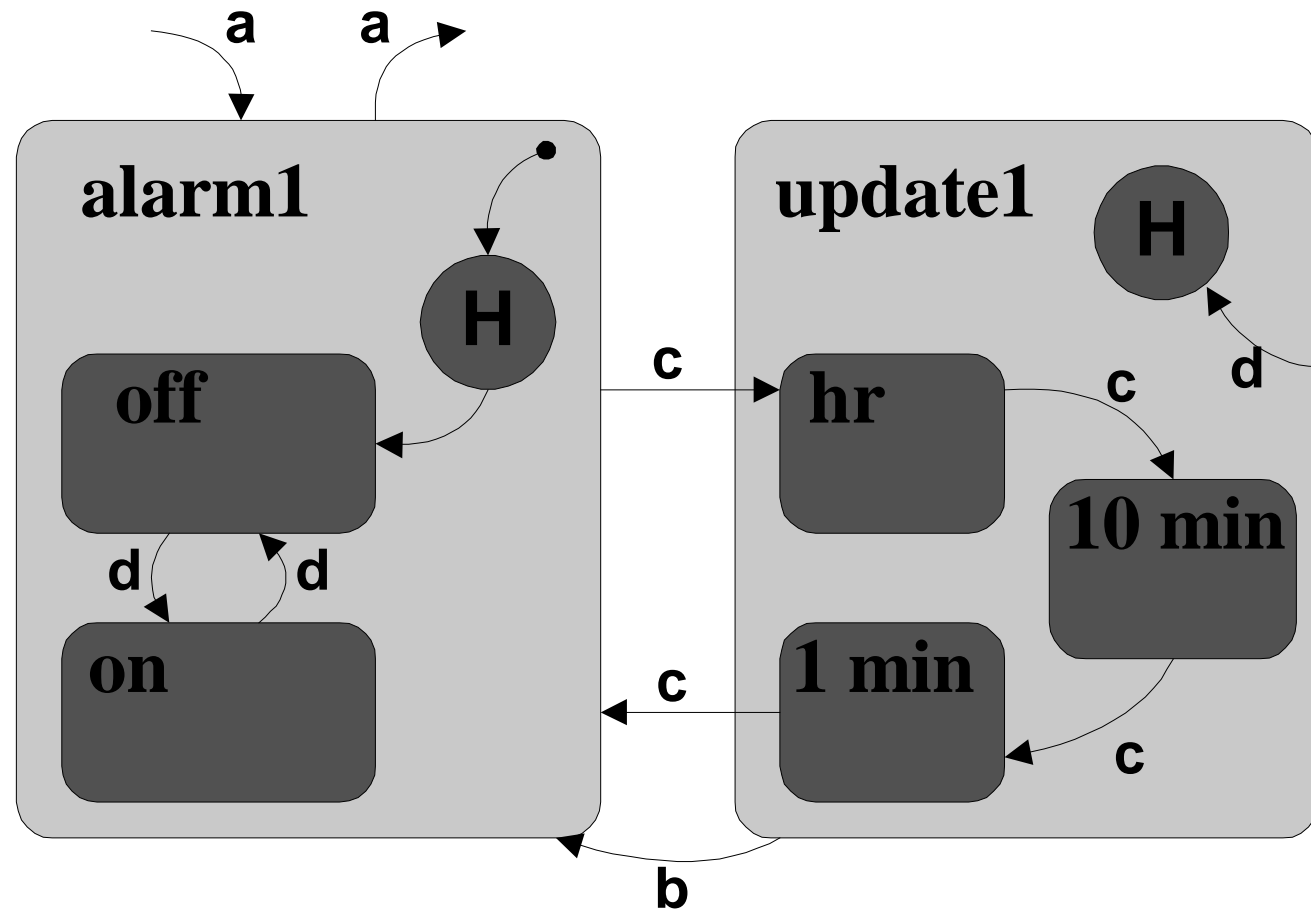




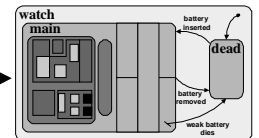
***We are here* →**



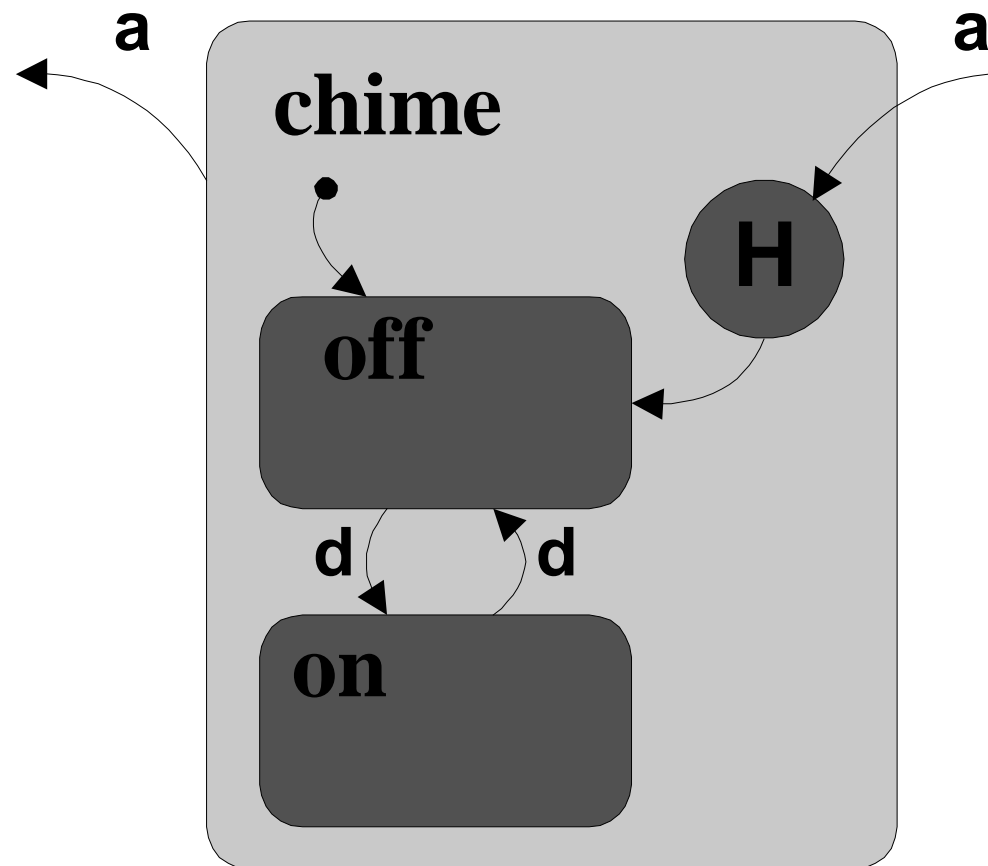
STATECHART: alarm1, update1



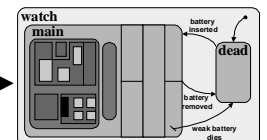
We are here →



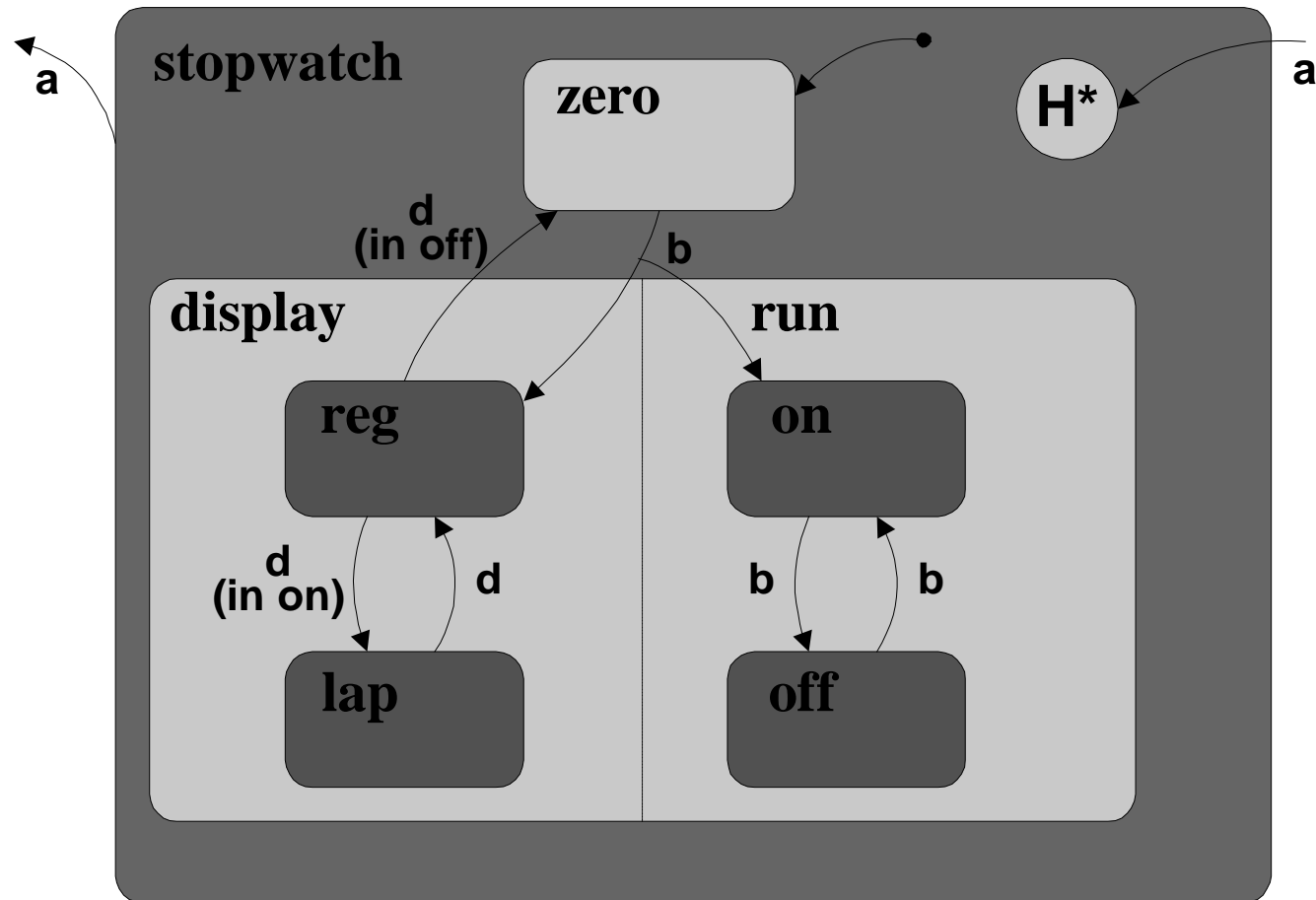
STATECHART: chime



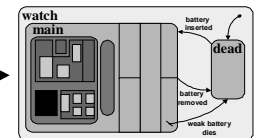
We are here →



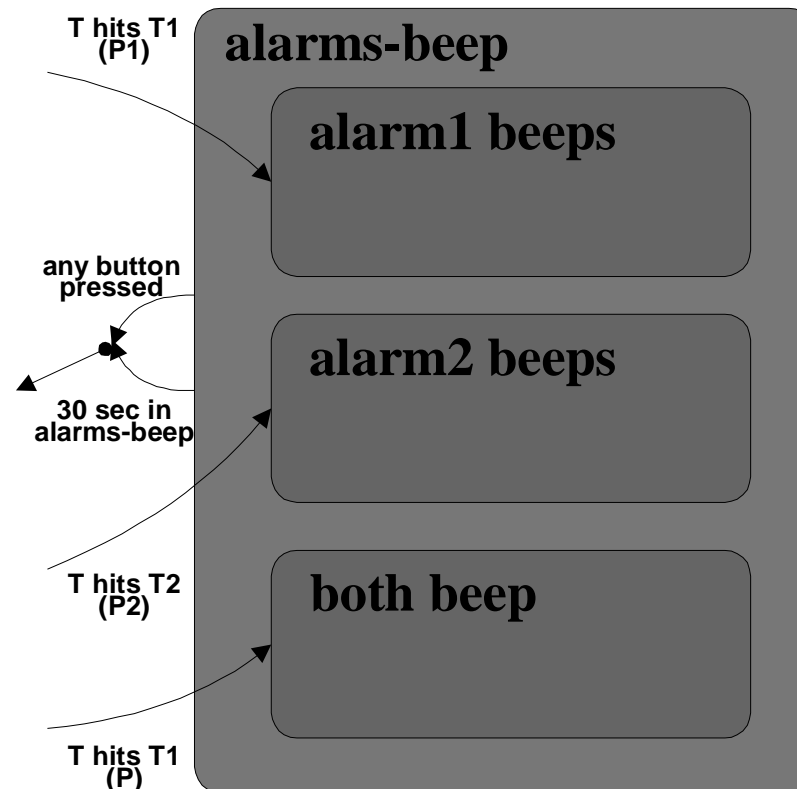
STATECHART: stopwatch



We are here →



STATECHART: alarms-beep



T1: time setting of alarm1

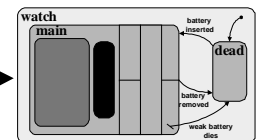
T2: time setting of alarm2

P1: $\text{alarm1_enabled} \wedge (\text{alarm2_disabled} \vee T1 \neq T2)$

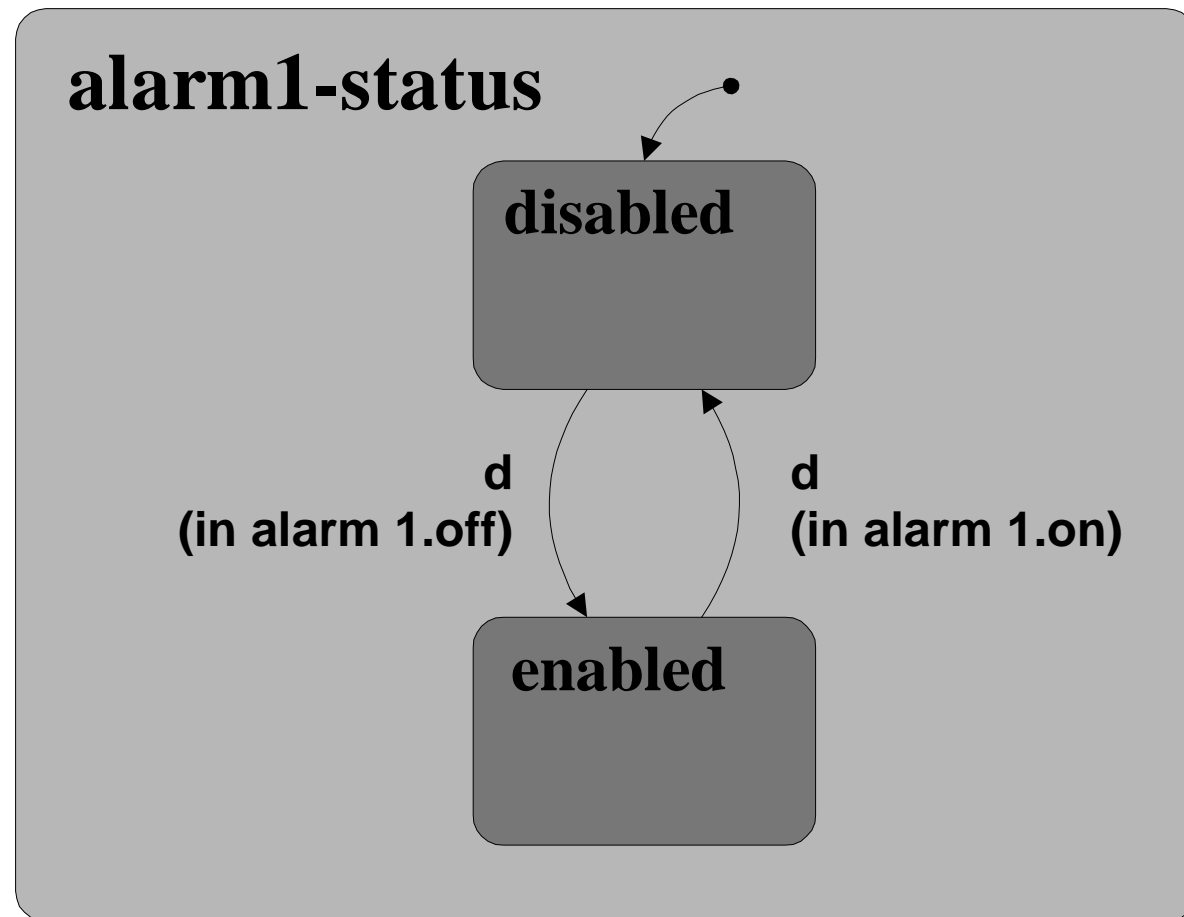
P2: $\text{alarm2_enabled} \wedge (\text{alarm1_disabled} \vee T1 \neq T2)$

P: $\text{alarm1_enabled} \wedge \text{alarm2_enabled} \wedge T1 = T2$

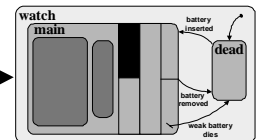
We are here →



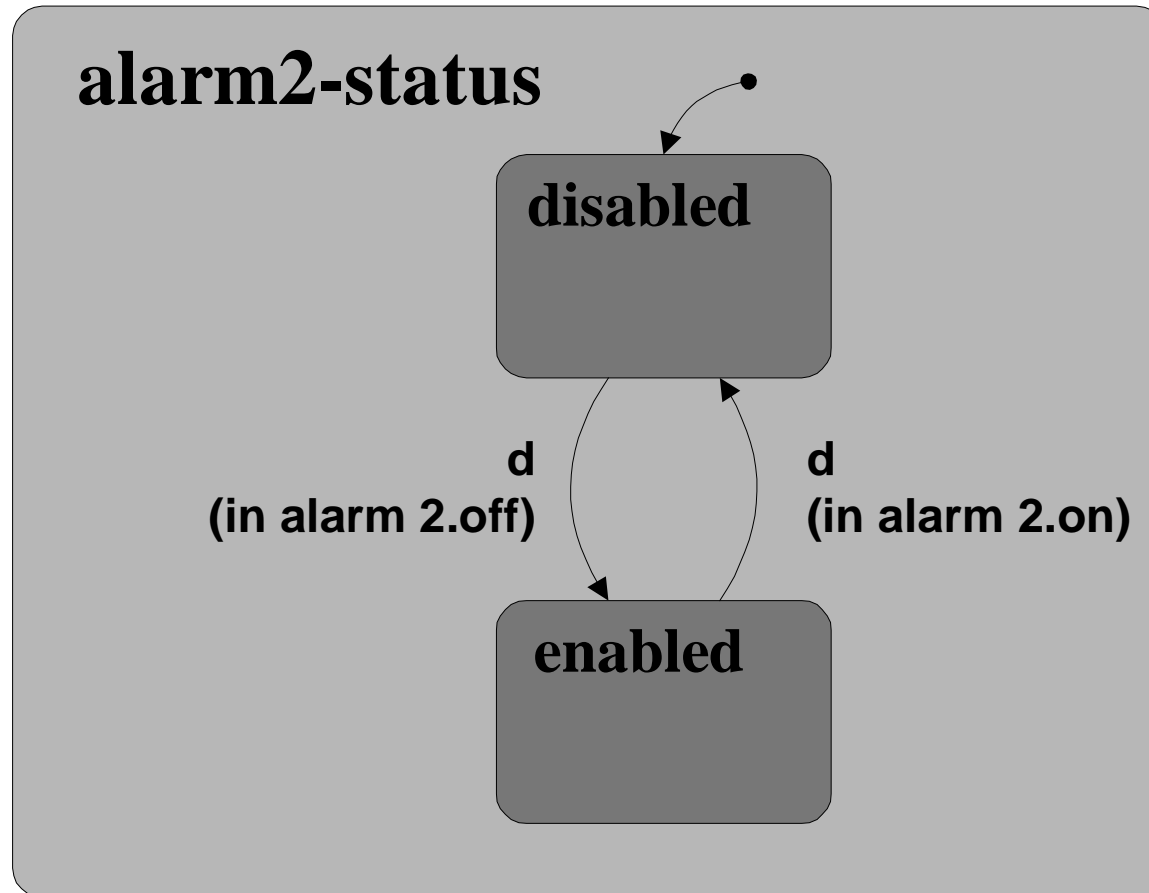
STATECHART: alarm1-status



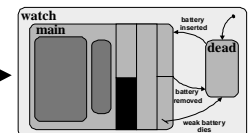
We are here →



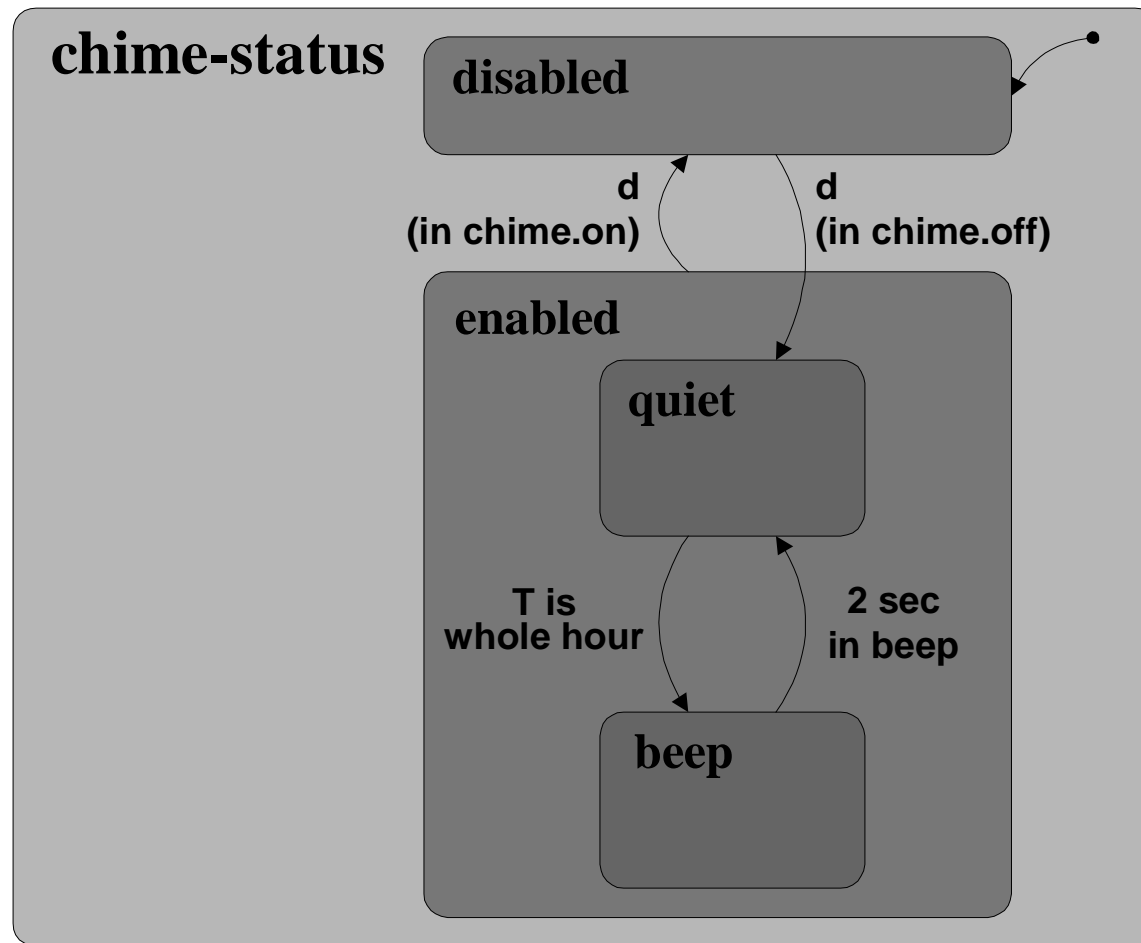
STATECHART: alarm2-status



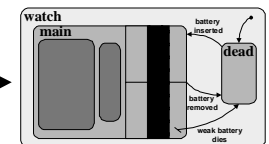
We are here →



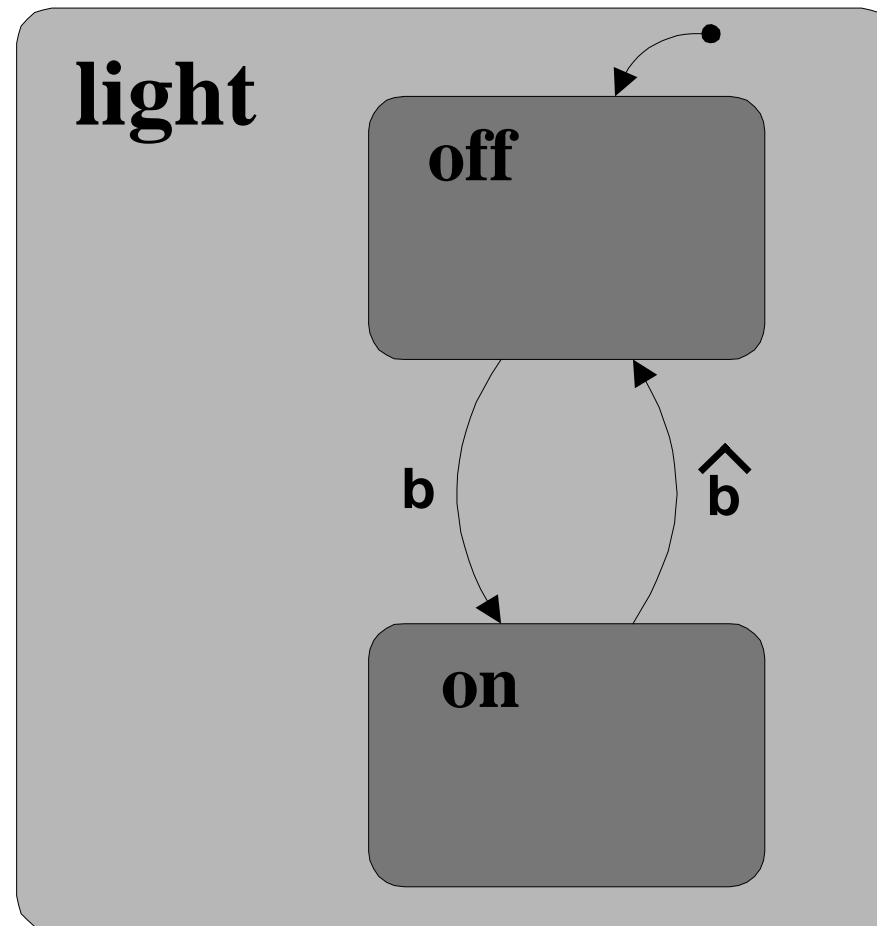
STATECHART: chime-status



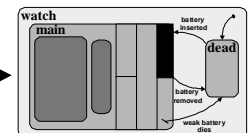
We are here →



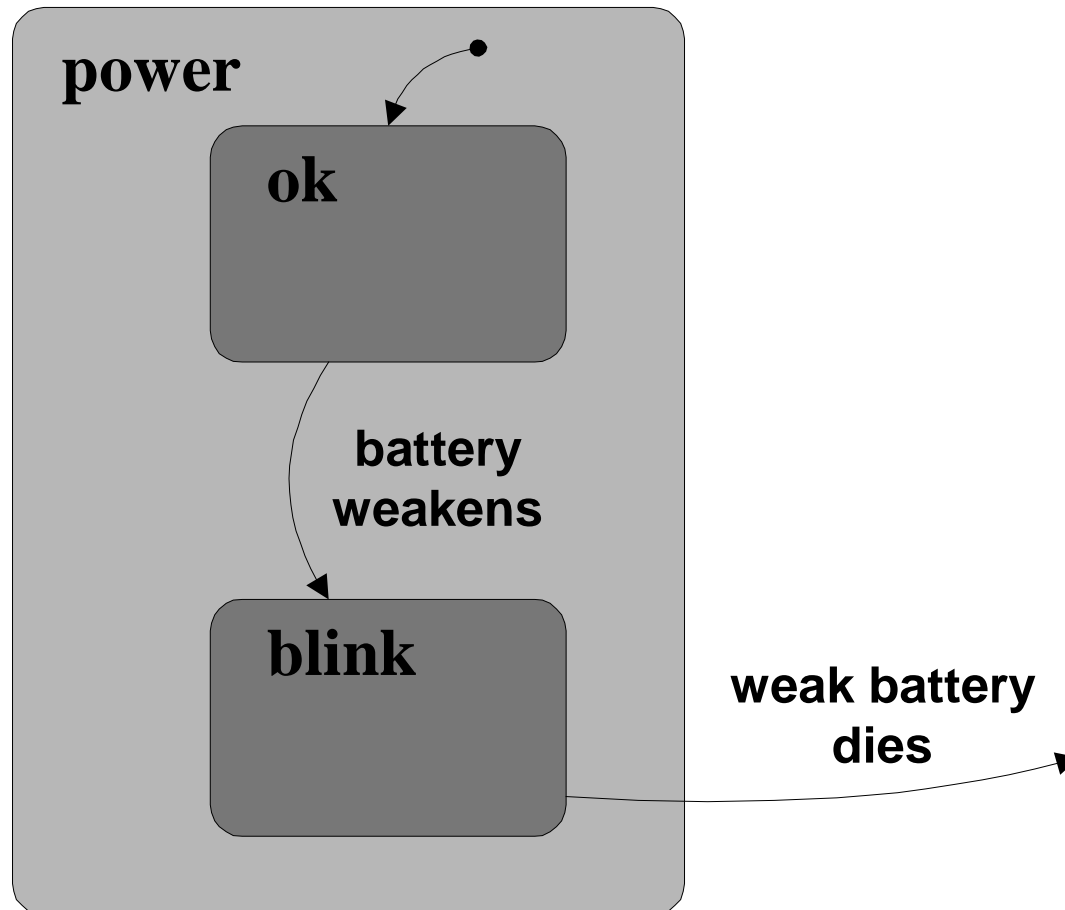
STATECHART: light



We are here →



STATECHART: power



We are here →

