

Specification of Control Flow in Ptolemy and Application to Lego Mindstorms



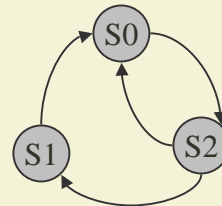
Xiaojun Liu
EECS, UC Berkeley
liuxj@eecs.berkeley.edu

Ptolemy Miniconference
Berkeley, CA, March 22-23, 2001

Specify Control Flow with Finite State Machines

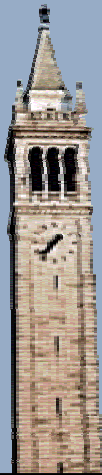


- FSM is an intuitive description of sequential control logic.
 - System states and transitions among states are represented explicitly.
- FSM is used extensively in control logic design.
 - Solid theoretical foundation
 - Many formal analysis and verification methods available

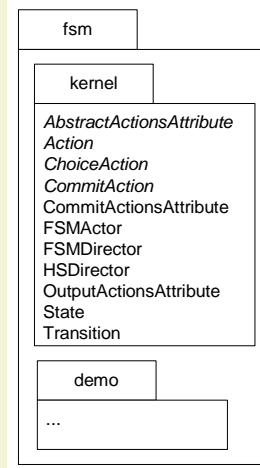


Ptolemy Miniconference, Berkeley, 2

The FSM Domain in Ptolemy II



- Support writing actors with FSMs
- Support the **charts* formalism
 - Proposed by A. Girault, B. Lee, E. A. Lee, 1999
 - Allow FSMs to be hierarchically combined with various domains



Ptolemy Miniconference, Berkeley, 3


Write Actors Using FSMs



- Three easy steps
 - Specify the behavior of the actor with an FSM
 - Construct the FSM in Vergil
 - Use the actor in Ptolemy models
- No code writing
- Domain polymorphic
 - Some domain-specific constraints on actors can only be satisfied by designing the FSM properly. For example, in the SDF domain, on any transition an FSM actor must produce exactly one token to each output port.
- Limitation
 - Minimal support for type resolution

Ptolemy Miniconference, Berkeley, 4


A Run-length Coder



0110110000110110
 1 21 2 4 21 2

Ptolemy Miniconference, Berkeley, 5

Control Modes of Operation

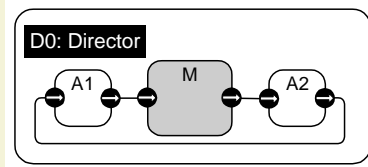


- Many systems exhibit multiple modes of operation.
 - We call such systems *Modal* systems.
 - For example, a communication endpoint switches detection mechanism when channel condition changes.

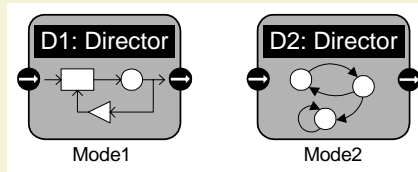
Ptolemy Miniconference, Berkeley, 6

Model Modal Systems with *charts

- Create a model of the environment



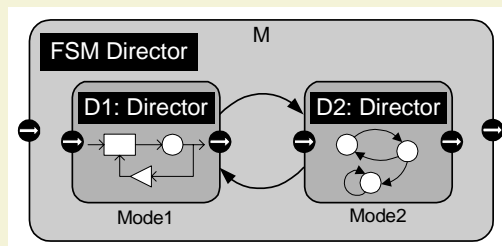
- Identify the modes of operation, create a model for each mode



Ptolemy Miniconference, Berkeley, 7

Model Modal Systems with *charts

- Construct the mode controller as an FSM, assemble the modal model
 - The states of the mode controller represent the modes of operation.
 - The model of a mode is active if and only if the mode controller is in its corresponding state.



Ptolemy Miniconference, Berkeley, 8

Some *charts Applications in Ptolemy II



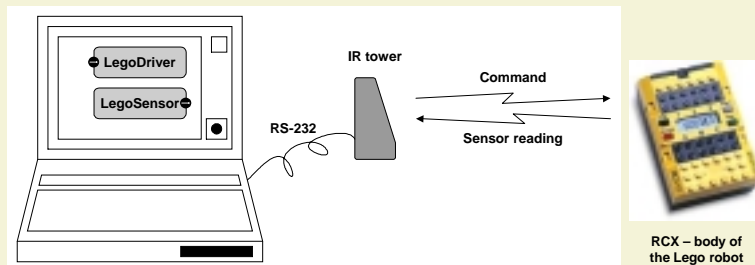
- Hybrid systems (with Jie Liu)
 - CT, FSM
- Heterochronous dataflow (Brian K. Vogel)
 - SDF, FSM
- Modal time-triggered programming
 - Giotto, FSM
- Communication protocol modeling
 - DE, hierarchical FSMs
- Modal scene graph in 3D modeling (Chamberlain Fong)
 - GR, FSM

Ptolemy Miniconference, Berkeley, 9

A Ptolemy II Model Driving Lego Robots

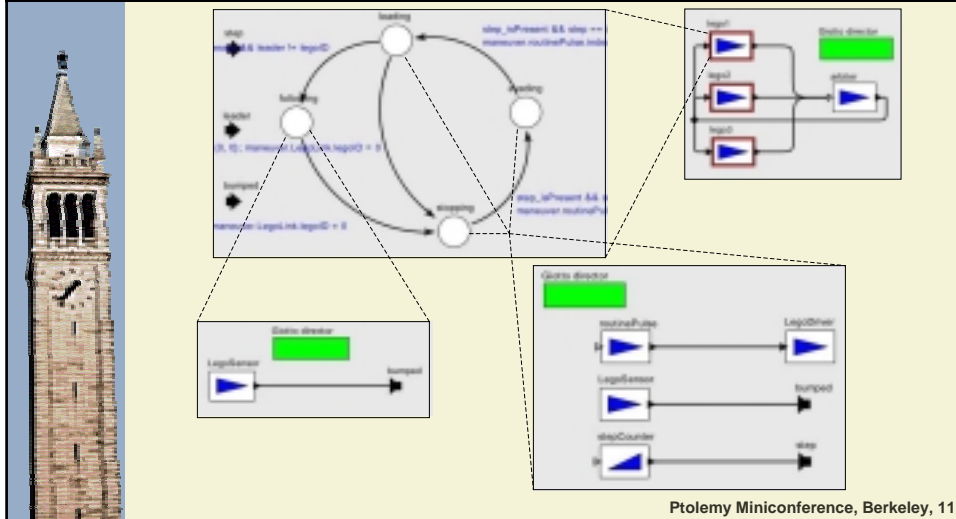


- Lego robots
 - Pros: robust, easy to acquire and maintain, cheap
 - Cons: limited processing power, communication bandwidth, and sensing capability



Ptolemy Miniconference, Berkeley, 10

A Ptolemy II Model Driving Lego Robots



Conclusion and Future Work

- Summary of major features
 - Support writing actors with FSMs
 - Support modal system modeling with the *charts formalism
- Future work
 - Support type resolution
 - Redesign the architecture of modal models using higher-order functions
 - Support code generation from FSM actors and modal models
 - Explore other applications