Embedded Systems Programming

Synchronous Model
(Module 26)

Yann-Hang Lee
Arizona State University
yhlee@asu.edu
(480) 727-7507

Summer 2014
Actor-Oriented Design

Object orientation:

- class name
- data
- methods

What flows through an object is sequential control

Actor orientation:

- actor name
- data (state)
- parameters
- ports

Input data

Output data

What flows through an object is data streams

(http://ptolemy.eecs.berkeley.edu/presentations/04/Parc_Lee.ppt)
Actor Orientation vs. Object Orientation

**Object oriented**

<table>
<thead>
<tr>
<th>TextToSpeech</th>
</tr>
</thead>
<tbody>
<tr>
<td>initialize(): void</td>
</tr>
<tr>
<td>notify(): void</td>
</tr>
<tr>
<td>isReady(): boolean</td>
</tr>
<tr>
<td>getSpeech(): double[]</td>
</tr>
</tbody>
</table>

**Actor oriented**

- **OO interface definition** gives procedures that have to be invoked in an order not specified as part of the interface definition.
- **actor-oriented interface definition** says “Give me text and I’ll give you speech”

- **Identified limitations of object orientation:**
  - Says little or nothing about concurrency and time
  - Concurrency typically expressed with threads, monitors, semaphores
  - Components tend to implement low-level communication protocols
  - Re-use potential is disappointing

(http://ptolemy.eecs.berkeley.edu/presentations/04/Parc_Lee.ppt)
Example of an Actor-Oriented Framework

Signal flow graph with linear, time-invariant components

(http://ptolemy.eecs.berkeley.edu/presentations/04/Parc_Lee.ppt)
Synchronous Execution Model – Simulink

- **Block: an actor**
  - Consists of some functionality and an arbitrary number of ports
  - can be pre-defined blocks from Simulink library, S-function blocks (writing your own function in C, Fortran, etc.), or subsystem blocks
    - S-functions are dynamically linked subroutines that the MATLAB interpreter can automatically load and execute
  - Signals connect block's ports to pass data between blocks
  - To calculate the values of the output ports based on the values of the input ports and the internal states.
  - Sample time: how often and when the functionality of a block is evaluated.
Synchronous or Asynchronous (1)

- **Synchronous:**
  - Atomic reactions indexed by a global logical clock,
  - Each reaction computes new events for its outputs based on its internal state and on the input values,
  - The communication of all events between components occur synchronously during each reaction.

- **Cycles of reading inputs, computing reaction and producing outputs**
  - Synchronous = 0-delay = within the same cycle
  - No interference between I/O and computation

- **Simulink® and other synchronous languages**

- **Why?**
  - Deterministic semantics in the presence of concurrency.
Synchronous or Asynchronous (2)

A high priority task B arrives and receives inputs from A (from A_1 or A_2?)

If Pri(A) > Pri(C) > Pri(B), depending upon the execution time of C, B may receive inputs from A_1 or A_2.
Synchronous or Asynchronous (3)

- If execution time = 0, then the computation is determined by the order of arrivals, not the arrival instances, nor execution time.
- Can we memorize the arrival order and then fetch data from buffer.