Introduction to FMI3 Clocked Simulations

Based on Gomes et al., 2021. The FMI 3.0 Standard Interface for Clocked and Scheduled Simulations. DOI: https://doi.org/10.3384/ecp2118127

Santiago Gil
Cláudio Gomes
Yon Vannomeslaeghe

FMI Tutorial - Example incubator using UniFMU with FMI3 and clocks

Some considerations:

- This presentation is related to the FMI tutorial.
Go to this repository:
https://github.com/INTO-CPS-Association/example-incubator-fmi3

- Focus on *Co-simulation* interfaces only.

- We're skipping some details.

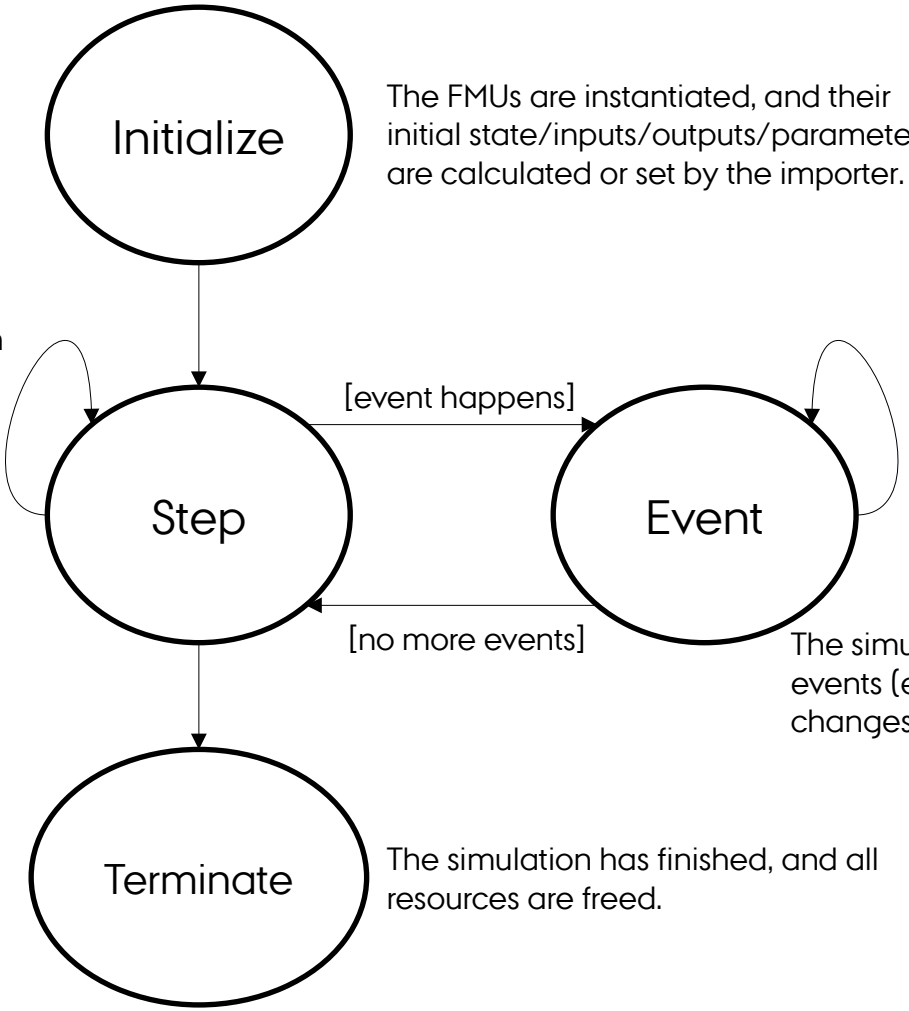
Formulation of Super-dense Time

Super-dense time is defined by the tuple*:

$$t = (t_R, t_I), \quad t_R \in \mathbb{R}, t_I \in \mathbb{N}$$

* Lee and Zheng, 2007. Leveraging synchronous language principles for heterogeneous modeling and design of embedded systems.

Co-simulation Loop



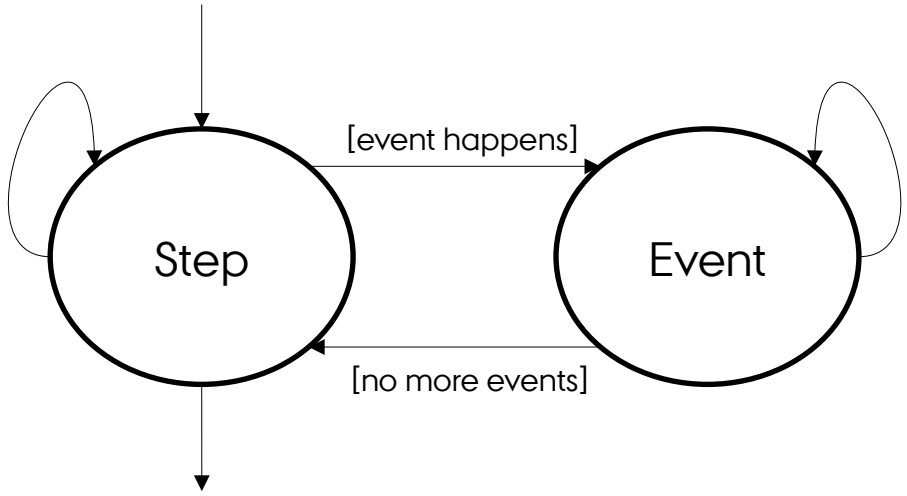
The FMUs are instantiated, and their initial state/inputs/outputs/parameters are calculated or set by the importer.

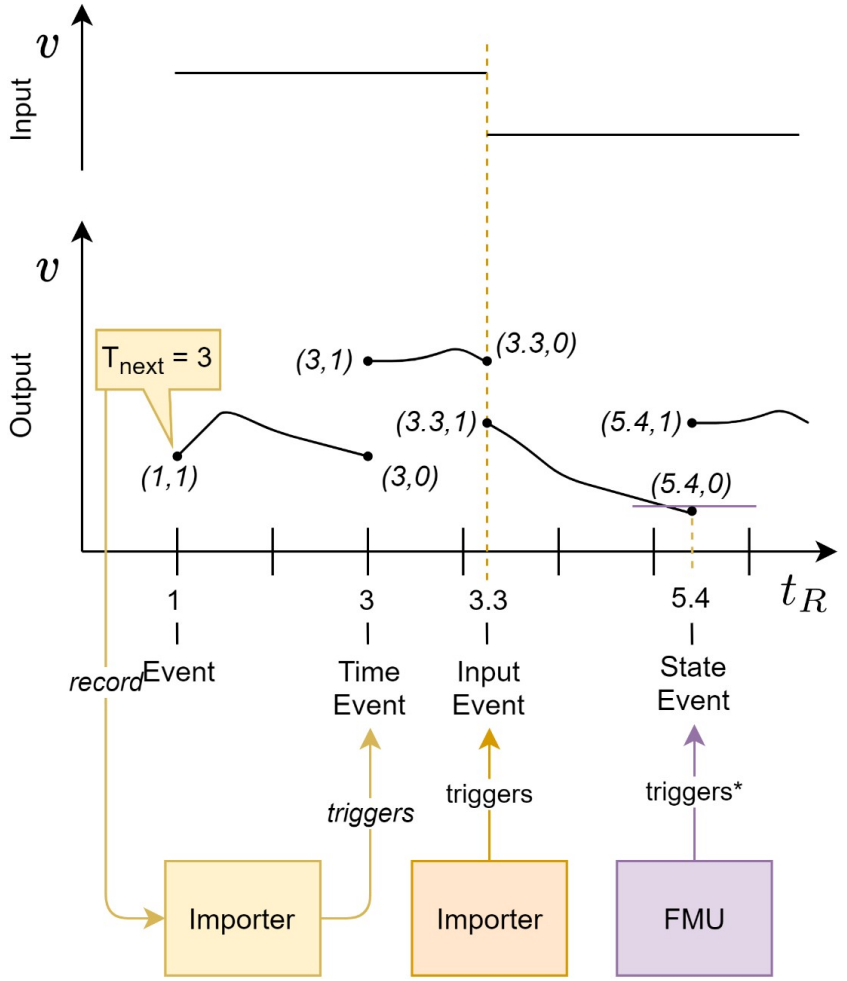
The simulation is progressing in simulated time, and FMUs that represent ODEs are being numerically integrated.

The simulated time is stopped and events (e.g., clock ticks, parameter changes) are being processed.

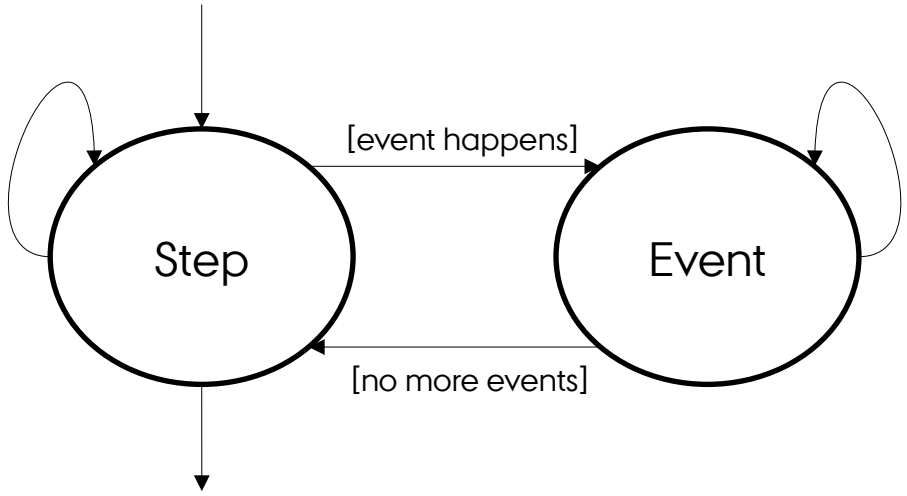
The simulation has finished, and all resources are freed.

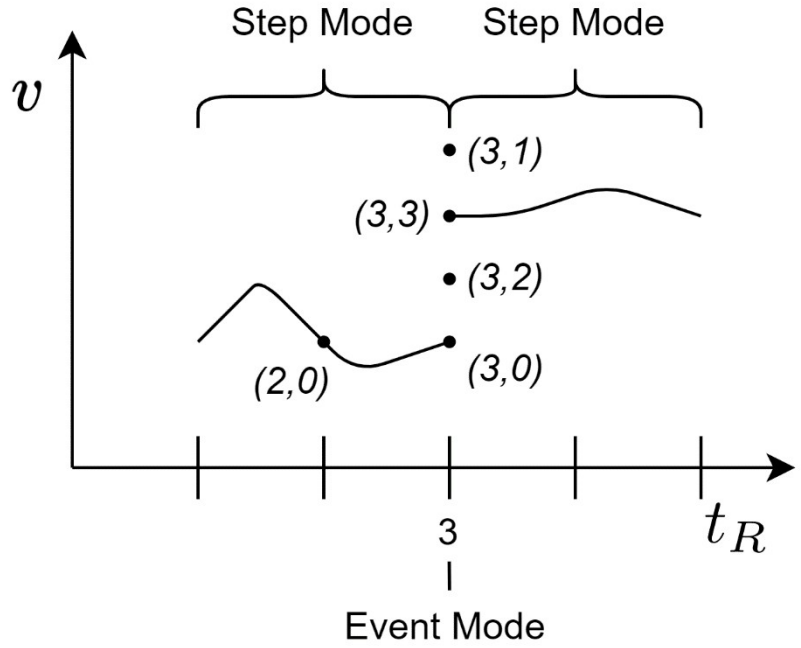
Types of Events





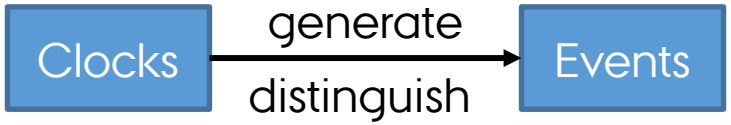
Consecutive Events





Introduction to Clocks

- Clocks:
  - abstraction of activities whose occurrence is tied to specific points in time.
  - At any super-dense time instant, a clock is either active (ticking) or inactive.
- Declared in the ModelDescription.xml



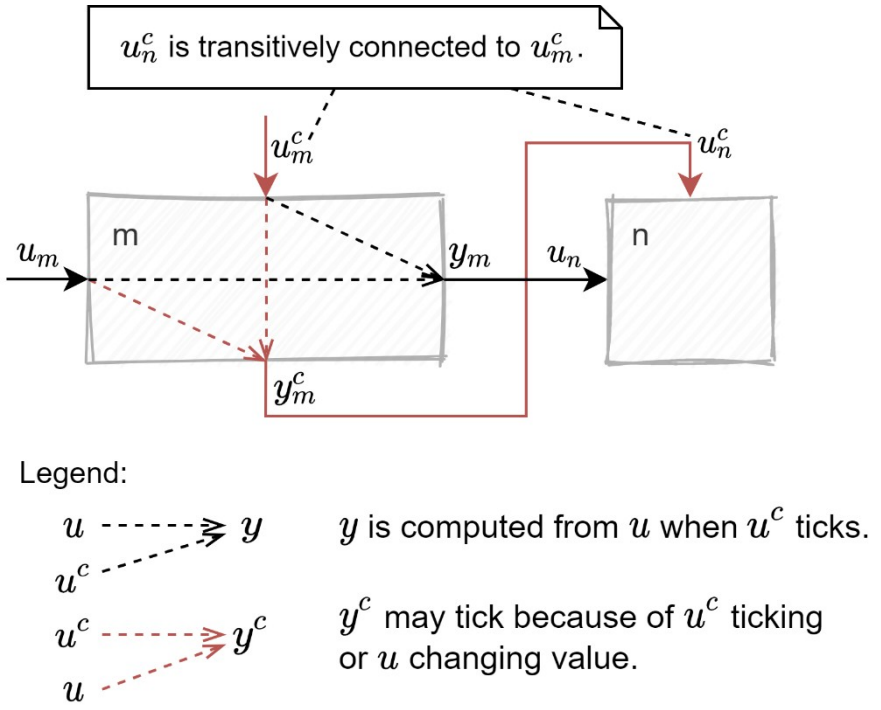
Clock Types

Clock Type	Period	Interval	Interpretation
time-based	periodic	constant	FMU declares period in XML.
		fixed	Importer sets the interval during Initialize.
		calculated	FMU calculates period in Initialize mode.
		tunable	FMU calculates period in Event mode (CS) or after executing model partition (SE).
	aperiodic	changing	FMU calculates interval after each clock tick.
		countdown	FMU calculates interval after an event.
triggered	–	triggered	There's no known interval. The clock ticks unpredictably, either due to FMU current state/inputs, or due to events.

All time-based clocks are inputs.

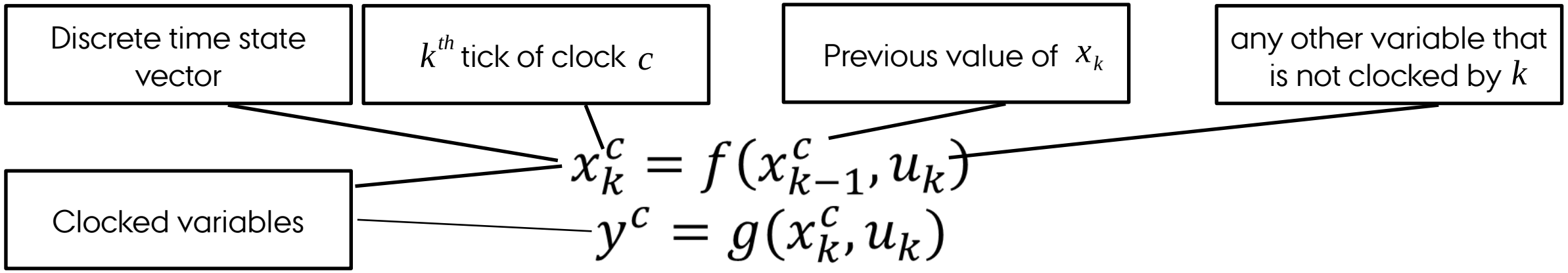
Clock Connections

- Clock y_m^c connected to u_n^c means the two always tick at the same super-dense time.
  - The importer must enforce this.
- Output triggered clocks can declare dependencies.



Clocked Variables

- Variable v associated to a set of Clocks C, where v's value changes only when one $c \in C$ ticks.
- Each clock c is associated to a set of variables V and equations spanning V, forming a discrete-time system that evolves on c's ticks:

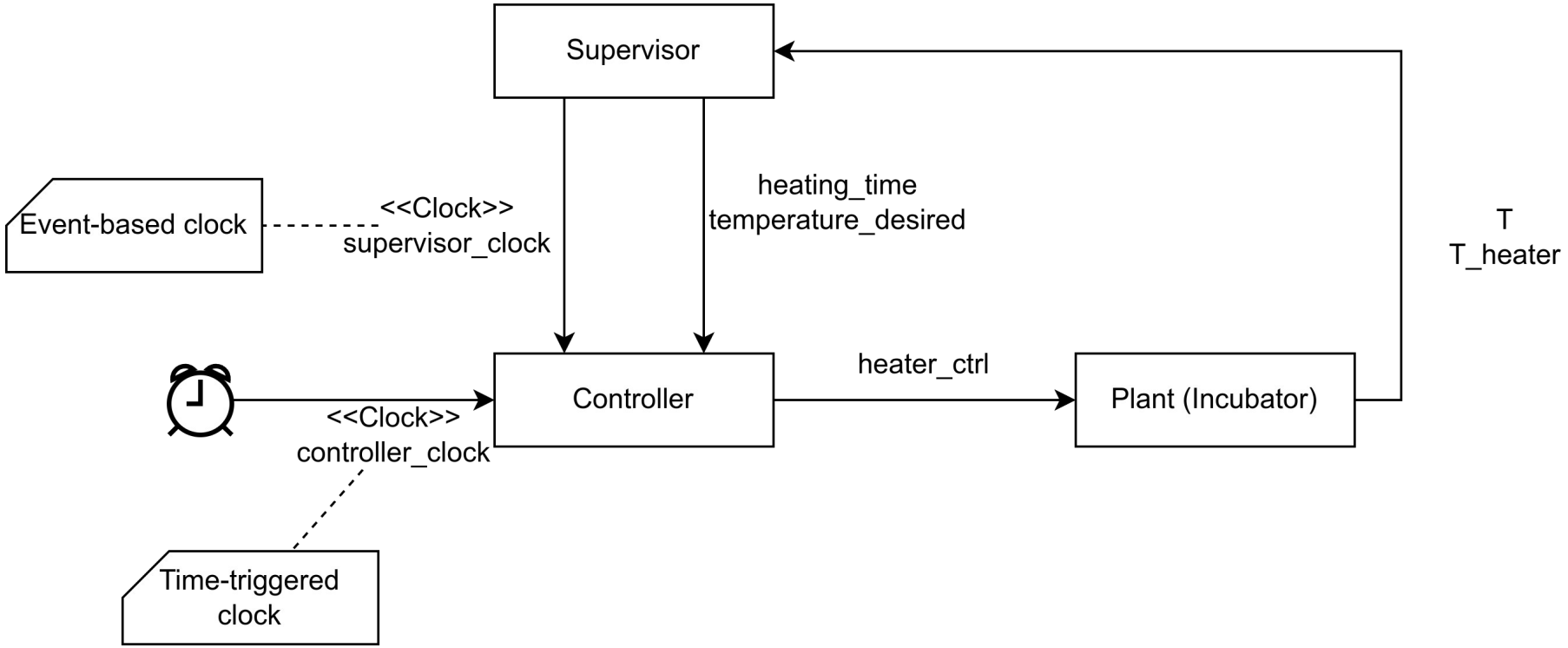


- Clocked variables can be discrete-time states.

Our example

We're using the incubator Digital twin example:

https://github.com/INTO-CPS-Association/example_digital-twin_incubator



Thanks!

Department of Electrical and Computer Engineering - Cyber-Physical Systems

Helsingforsgade 10, office 5123-318
8200 Aarhus N, Denmark

sgil@ece.au.dk

Department of Electrical and Computer Engineering - Cyber-Physical Systems

Helsingforsgade 10, office 5123-425
8200 Aarhus N, Denmark

claudio.gomes@ece.au.dk

Cosys Lab – University of Antwerp

Groenenborgerlaan 171
B-2020 Antwerpen, Belgium

yon.vanommeslaeghe@uantwerpen.be

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING