

SYSC 5906 G

Methodologies for Discrete Event Modeling and Simulation

# **A DEVS Based Simulation of Mitosis Using Cadmium**

Team Members:

Hamnah Qureshi, 101225634

Naomi Amaya Lovett, 101223562

Grant Achuzia, 101222695

Due: March 2nd, 202

# Table of Contents

<b>Part I: Defining the Mitosis System.....</b>	<b>3</b>
Model Choice Overview.....	3
Model Diagram.....	3
Model Structure.....	4
Level 1: Top Coupled Model.....	4
Level 2: Coupled Subsystems.....	4
Level 3: Atomic Models in Subsystems.....	4
Technical Definitions.....	5
<b>PART II: A DEVS Formal Specification of the Mitosis Model.....</b>	<b>6</b>
Updated Model Diagram.....	6
Formal Specifications for Coupled Models.....	6
Formal Specifications for Atomics Models.....	8
Atomic Model Test Cases.....	16
<b>PART III: Execution Results and Analysis of the Mitosis Simulation.....</b>	<b>21</b>
Simulation Results.....	21
Analysis of Output of Top Model.....	26
Conclusion.....	28

# Part I: Defining the Mitosis System

## Model Choice Overview

Cell division through mitosis is a biological process that uses a finite sequence of phases (interphase, prophase, metaphase, anaphase, telophase, and cytokinesis). Each phase is made up of discrete structural changes in cellular components throughout the cell.

The goal of this project is to model mitosis as a Discrete Event Dynamic System (DEDS) with the DEVS formalism. The model will aim to capture the following:

- The sequence of mitotic phases
- The event driven interactions between cellular parts
- The state changes of each component throughout mitosis

This model can be used to simulate a normal mitotic progression, verify that the phases are ordered correctly, and serve as a foundation to extend the model to abnormal cell division scenarios.

## Model Diagram

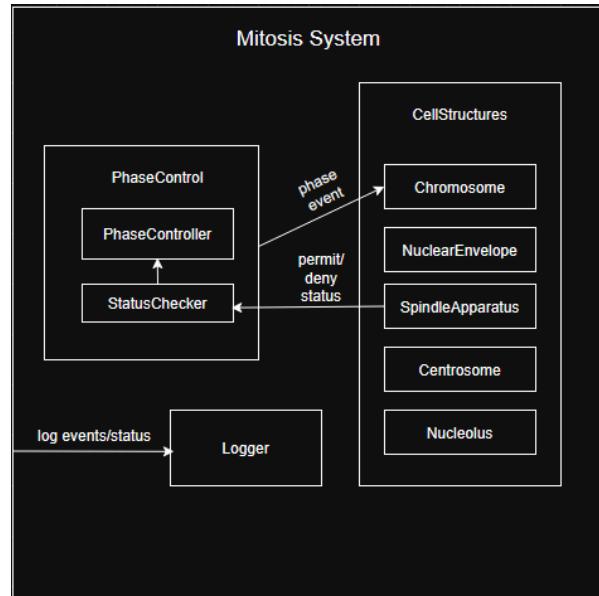


Figure 1: Draft Mitosis System Model Diagram

## Model Structure

### Level 1: Top Coupled Model

**MitosisSystem:** Top level model representing the complete mitosis process. It connects control logic, cell components, and outputs. It does not have state or behavior of its own, rather, it defines structures and connections

### Level 2: Coupled Subsystems

- **PhaseControl:** Coordinates the system's attempts to move from one phase to the next, letting the system know if a transition is allowed.
- **CellStructures:** Groups the models that represent the physical parts of the cell involved in mitosis. Submodels respond to phase changes.
- **Logger:** Records phase transitions and events in the simulation.

### Level 3: Atomic Models in Subsystems

- **Chromosome:** Tracks the varying behaviors/actions in the chromosome. Updates state based on phase events and reports changes in the chromosome.
- **NuclearEnvelope:** Updates state based on phase and reports whether the envelope is broken or reformed.
- **SpindleApparatus:** Represents spindle formation and attachment to chromosomes. Reports when attachment is complete.
- **Centrosome:** Tracks centrosome duplication and movement to opposite poles. Reports whether centrosomes have reached opposite poles.
- **Nucleolus:** Represents the appearance/disappearance of the nucleolus as a visible indication of early and late mitosis phases.
- **PhaseController:** Maintains the current mitosis phase, requests transitions to the next phase, and broadcasts a phase event when the transition is allowed.
- **StatusChecker:** Receives updates from cell components and decides if the system can move on to the next phase. It sends a permit or deny signal to the PhaseController.

## **Technical Definitions**

**Chromosome:** It is a tightly packed structure made of DNA and proteins. It carries the genetic information and needs to make sure the DNA is probably distributed during cell division. They condense in prophase, align in metaphase, separate in anaphase and decondense in telophase.

**DNA:** DNA is the molecule that carries the genetic instructions for how a cell functions and grows. DNA is organized differently depending on the stage of the cell cycle. During prophase, chromatin condenses into chromosomes. They align and separate in metaphase and anaphase, then decondense back into chromatin during telophase.

**Nucleus:** It controls the cell's activities and protects its DNA. During prophase, the nuclear membrane breaks down. It remains absent in metaphase and anaphase to allow chromosome movement, then reforms around the chromosomes in telophase.

**Spindle apparatus:** This component is made of microtubules that form during mitosis and pulls sister chromatids apart. This forms during prophase, attaches to chromosomes in metaphase, pulls sister chromatids in anaphase, and breaks down in telophase.

**Centrosome:** The centrosome organizes the spindle fibers for chromosome movement. It duplicates before mitosis, moves to opposite poles during prophase, helps pull chromosomes apart in anaphase, and remains at the poles until the end of telophase.

## PART II: A DEVS Formal Specification of the Mitosis Model

### Updated Model Diagram

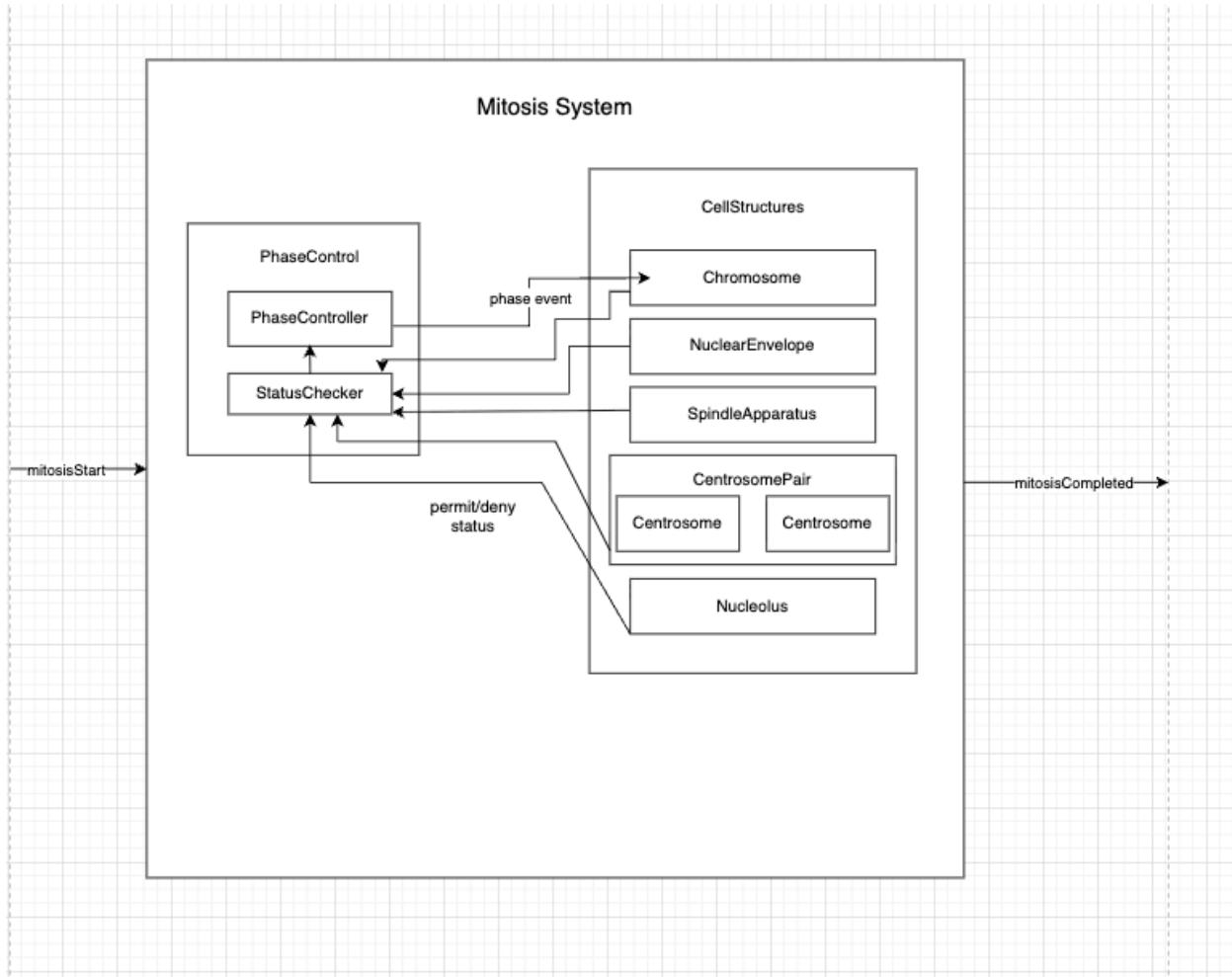


Figure 2: Updated Model Diagram

### Formal Specifications for Coupled Models

For a coupled model,  $CM = \langle X, Y, D, \{M_d \mid d \in D\}, EIC, EOC, IC, \text{select} \rangle$  where ,  $X$  is the input event set,  $Y$  is the output event set,  $D$  is the set of components,  $M_d$  is the component models,  $EIC$  is the external input couplings,  $EOC$  is the external output couplings,  $IC$  is the internal input couplings, and  $\text{Select}$  is the tiebreaker function.

## **MitosisSystem**

X = {start}

Y = {mitosis\_complete | mitosis\_complete ∈ {true, false}}

D = {PhaseControl, CellStructures}

M<sub>d</sub> = {M<sub>PhaseControl</sub>, M<sub>CellStructure</sub>}

EIC = {((Self, start), (PhaseController, start))}

EOC = {((PhaseControl, mitosis\_complete), (Self, mitosis\_complete))}

IC = {

    ((PhaseControl, phase\_event), (CellStructures, phase\_in)),  
    ((CellStructures, chrom\_status), (PhaseControl, chrom\_status)),  
    ((CellStructures, ne\_status), (PhaseControl, ne\_status)),  
    ((CellStructures, sp\_status), (PhaseControl, sp\_status)),  
    ((CellStructures, cen\_status), (PhaseControl, cen\_status)),  
    ((CellStructures, nucleo\_status), (PhaseControl, nucleo\_status))

}

Select = {PhaseControl > CellStructures}

## **PhaseControl**

X = {start}

Y = {phase\_out ∈ {Interphase, Prophase, Metaphase, Anaphase, Telophase, Cytokinesis}}

D = {PhaseController, StatusChecker}

M<sub>d</sub> = {M<sub>PhaseController</sub>, M<sub>StatusChecker</sub>}

EIC = {((Self, start), (PhaseController, start))}

EOC = {((PhaseController, phase\_event), (Self, phase\_out))}

IC = {((StatusChecker, allow\_transition), (PhaseController, transition)),  
      ((StatusChecker, deny\_transition), (PhaseController, do\_not\_transition))}

Select = {StatusChecker > PhaseController}

## **CellStructures**

X = {phase\_in ∈ { Interphase, Prophase, Metaphase, Anaphase, Telophase, Cytokinesis}}

Y = {chrom\_status, ne\_status, sp\_status, cen\_status, nucleo\_status}

$D = \{\text{Chromosome}, \text{NuclearEnvelope}, \text{SpindleApparatus}, \text{CentrosomePair}, \text{Nucleolus}\}$

$M_d = \{M_{\text{Chromosome}}, M_{\text{NuclearEnvelope}}, M_{\text{SpindleApparatus}}, M_{\text{CentrosomePair}}, M_{\text{Nucleolus}}\}$

$EIC = \{((\text{Self}, \text{phase\_in}), (d, \text{phase\_in}))\}$

$EOC = \{ ((\text{Chromosome}, \text{status\_out}), (\text{Self}, \text{chrom\_status})),$   
 $\quad ((\text{NuclearEnvelope}, \text{status\_out}), (\text{Self}, \text{ne\_status})),$   
 $\quad ((\text{SpindleApparatus}, \text{status\_out}), (\text{Self}, \text{sp\_status})),$   
 $\quad ((\text{CentrosomePair}, \text{status\_out}), (\text{Self}, \text{cen\_status})),$   
 $\quad ((\text{Nucleolus}, \text{status\_out}), (\text{Self}, \text{nucleo\_status})) \}$

$IC = \emptyset$

$Select = \{\text{Chromosome} > \text{NuclearEnvelope} > \text{SpindleApparatus} > \text{CentrosomePair} > \text{Nucleolus}\}$

### **CentrosomePair**

$X = \{\text{phase\_in} \in \{\text{Interphase}, \text{Prophase}, \text{Metaphase}, \text{Anaphase}, \text{Telophase}, \text{Cytokinesis}\}\}$

$Y = \{\text{status\_out} \in \{\text{ready}, \text{not\_ready}\}\}$

$D = \{\text{CentrosomeTop}, \text{CentrosomeBottom}\}$

$M_d = \{M_{\text{CentrosomeTop}}, M_{\text{CentrosomeBottom}}\}$

$EIC = \{((\text{Self}, \text{phase\_in}), (d, \text{phase\_in}))\}$

$EOC = \{ ((M_{\text{CentrosomeTop}}, \text{status\_out}, M_{\text{CentrosomeTop}}, \text{status\_out}), (\text{Self}, \text{status\_out})) \}$

$IC = \emptyset$

$Select = \{M_{\text{CentrosomeTop}} > M_{\text{CentrosomeBottom}}\}$

## **Formal Specifications for Atomics Models**

For an atomic model,  $M = \langle X, Y, S, \delta_{\text{int}}, \delta_{\text{ext}}, \lambda, \text{ta} \rangle$  where  $X$  is the input event set,  $Y$  is the output event set,  $S$  is the set of states,  $\delta_{\text{int}}$  is the internal transition function,  $\delta_{\text{ext}}$  is the external transition function,  $\lambda$  is the output function, and  $\text{ta}$  is the time advance function.

**Note:** The port for phase events was originally called `input\_phase` as seen in the atomic model diagrams below. However, the specifications have since been updated to use the name `phase\_in`.

## Chromosome

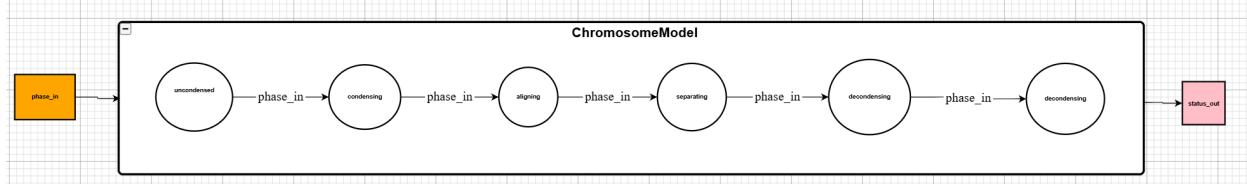


Figure 3: Chromosome Atomic Model Diagram

$X = \{ \text{phase\_in} \in \{ \text{Interphase, Prophase, Metaphase, Anaphase, Telophase, Cytokinesis} \} \}$

$Y = \{ \text{status\_out} \in \{ \text{ready, not\_ready} \} \}$

$S = \{ \text{uncondensed, condensed, aligned, separated, condensing, aligning, separating, decondensing} \}$

$\delta_{\text{int}} (\text{Active}) =$

- condensing  $\rightarrow$  condensed
- aligning  $\rightarrow$  aligned
- separating  $\rightarrow$  separated
- decondensing  $\rightarrow$  uncondensed

$\delta_{\text{ext}} =$

- Prophase  $\rightarrow$  condensing
- Metaphase  $\rightarrow$  aligning
- Anaphase  $\rightarrow$  separating
- Telophase  $\rightarrow$  decondensing
- Cytokinesis  $\rightarrow$  decondensing

$\lambda =$

```

if ( $S == \text{condensing} \parallel S == \text{aligning} \parallel S == \text{separating} \parallel S == \text{decondensing}$ )
    status_out = ready
else
    status_out = not_ready
  
```

$ta =$

```

ta(condensing) = t1
ta(aligning) = t1
ta(separating) = t1
ta(decondensing) = t1
  
```

$ta(\text{otherwise}) = \infty$

## NuclearEnvelope

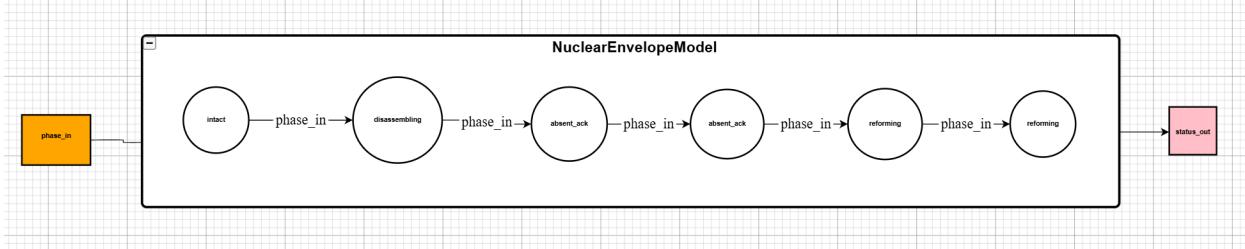


Figure 4: Nuclear Envelope Atomic Model Diagram

$X = \{ \text{phase\_in} \in \{ \text{Interphase, Prophase, Metaphase, Anaphase, Telophase, Cytokinesis} \} \}$

$Y = \{ \text{status\_out} \in \{ \text{ready, not\_ready} \} \}$

$S = \{ \text{intact, disassembling, absent\_ack, absent, reforming, restored} \}$

$\delta_{\text{int}} (\text{Active}) =$

disassembling  $\rightarrow$  absent

absent\_ack  $\rightarrow$  absent

reforming  $\rightarrow$  restored

$\delta_{\text{ext}} =$

Interphase  $\rightarrow$  intact

Prophase  $\rightarrow$  disassembling

Metaphase & Anaphase  $\rightarrow$  absent\_ack

Telophase & Cytokinesis  $\rightarrow$  reforming

$\lambda =$

if ( $S == \text{disassembling} \parallel S == \text{reforming} \parallel S == \text{absent\_ack}$ )

status\_out = ready

else

status\_out = not\_ready

$ta =$

$ta(\text{disassembling}) = t_1$

$ta(\text{reforming}) = t_1$

$ta(\text{absent\_ack}) = t_2$

$ta(\text{otherwise}) = \infty$

## SpindleApparatus

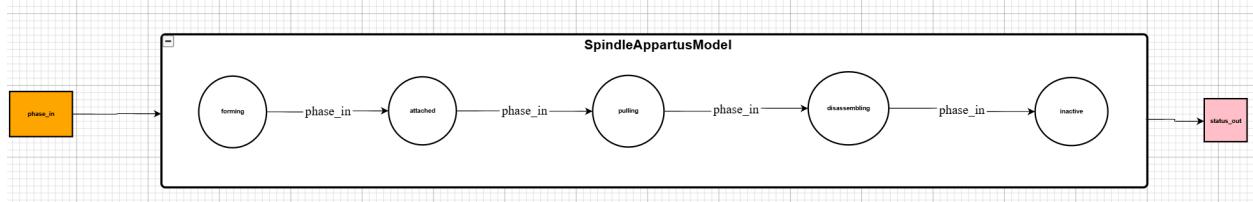


Figure 5: SpindleApparatus Atomic Model Diagram

$X = \{ \text{phase\_in} \in \{ \text{Interphase, Prophase, Metaphase, Anaphase, Telophase, Cytokinesis} \} \}$

$Y = \{ \text{status\_out} \in \{ \text{ready, not\_ready} \} \}$

$S = \{ \text{forming, attached, pulling, disassembling, inactive} \}$

$\delta_{\text{int}} (\text{Active}) = \text{Passive}$

$\delta_{\text{ext}} =$

Prophase  $\rightarrow$  forming

Metaphase  $\rightarrow$  attached

Anaphase  $\rightarrow$  pulling

Telophase  $\rightarrow$  disassembling

Cytokinesis  $\rightarrow$  inactive

$\lambda =$

if ( $S == \parallel S == \text{reforming} \parallel S == \text{absent\_ack}$ )

status\_out = ready

else

status\_out = not\_ready

$ta =$

$ta(\text{forming}) = t_1$

$ta(\text{attached}) = t_2$

$ta(\text{pulling}) = t_2$

$ta(\text{disassembling}) = t_2$

$ta(\text{inactive}) = 0.0$

$ta(\text{otherwise}) = \infty$

## Centrosome

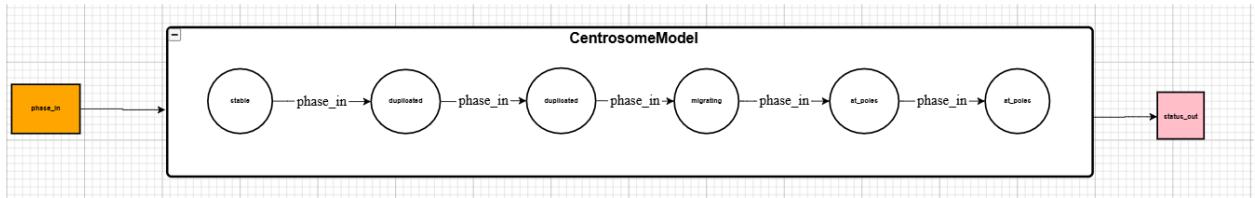


Figure 6: Centrosome Atomic Model Diagram

$X = \{ \text{phase\_in} \in \{ \text{Interphase, Prophase, Metaphase, Anaphase, Telophase, Cytokinesis} \} \}$

$Y = \{ \text{status\_out} \in \{ \text{ready, not\_ready} \} \}$

$S = \{ \text{duplicated, migrating, at\_poles, stable} \}$

$\delta_{\text{int}} (\text{Active}) = \text{Passive}$

$\delta_{\text{ext}} =$

Interphase  $\rightarrow$  stable

Prophase  $\rightarrow$  duplicated

Metaphase  $\rightarrow$  migrating

Anaphase  $\rightarrow$  at\_poles

Telophase  $\rightarrow$  at\_poles

$\lambda =$

if ( $S == \text{at\_poles} \parallel S == \text{stable} \parallel S == \text{duplicated} \parallel S == \text{migrating}$ ):

status\_out = ready

else:

status\_out = not\_ready

$ta =$

$ta(\text{duplicated}) = t_1$

$ta(\text{migrating}) = t_2$

$ta(\text{at\_poles}) = 0.0$

$ta(\text{otherwise}) = \infty$

## Nucleolus

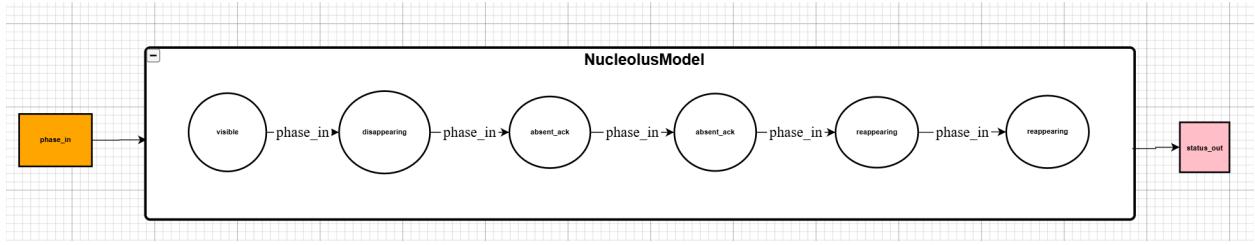


Figure 7: Nucleolus Atomic Model Diagram

$$X = \{ \text{phase\_in} \in \{ \text{Interphase, Prophase, Metaphase, Anaphase, Telophase, Cytokinesis} \} \}$$

$$Y = \{ \text{status\_out} \in \{ \text{ready, not\_ready} \} \}$$

$$S = \{ \text{visible, disappeared, reappearing, disappearing, absent\_ack} \}$$

$$\delta_{\text{int}} (\text{Active}) =$$

disappearing → disappeared

absent\_ack → disappeared

reappearing → visible

$$\delta_{\text{ext}} =$$

Interphase → visible

Prophase → disappearing

Metaphase & Anaphase → absent\_ack

Telophase & Cytokinesis → reappearing

$$\lambda =$$

if ( $S == \text{disappearing} \parallel S == \text{absent\_ack} \parallel S == \text{reappearing}$ ):

status\_out = ready

else:

status\_out = not\_ready

$$ta =$$

ta(disappeared) =  $t_1$

ta(reappearing) =  $t_2$

ta(absent\_ack) =  $t_3$

ta(otherwise) =  $\infty$

## StatusChecker

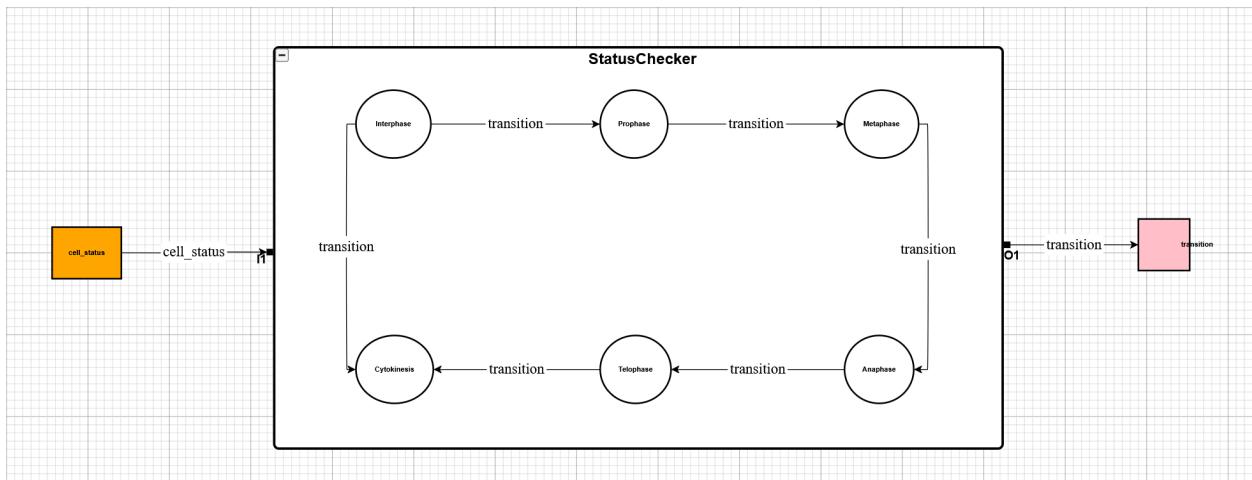


Figure 8: Status Checker Atomic Model Diagram

$X = \text{cell\_status} \in \{\text{chrom\_status}, \text{ne\_status}, \text{sp\_status}, \text{cen\_status}, \text{nucleo\_status}\}$ ,  $\text{start\_in} \in \{\text{start}\}$

$Y = \text{transition} \in \{\text{allow\_transition}, \text{deny\_transition}\}$

$S = \{\text{Interphase}, \text{Prophase}, \text{Metaphase}, \text{Anaphase}, \text{Telophase}, \text{Cytokinesis}\}$

$\delta_{\text{int}}(\text{Active}) = \text{Passive}$

$\delta_{\text{ext}} = \text{When all of the inputs are set to "ready", switch to the next state}$

Interphase → Prophase

Prophase → Metaphase

Metaphase → Anaphase

Anaphase → Telophase

Telophase → Cytokinesis

$\lambda =$

if ALL cell\_status == ready:

    if S == Cytokinesis return "Simulation Complete"

    return allow\_transition

else:

    stay in deny\_transition

$ta = \infty$ , depends on the external inputs from the cell structures

## PhaseController

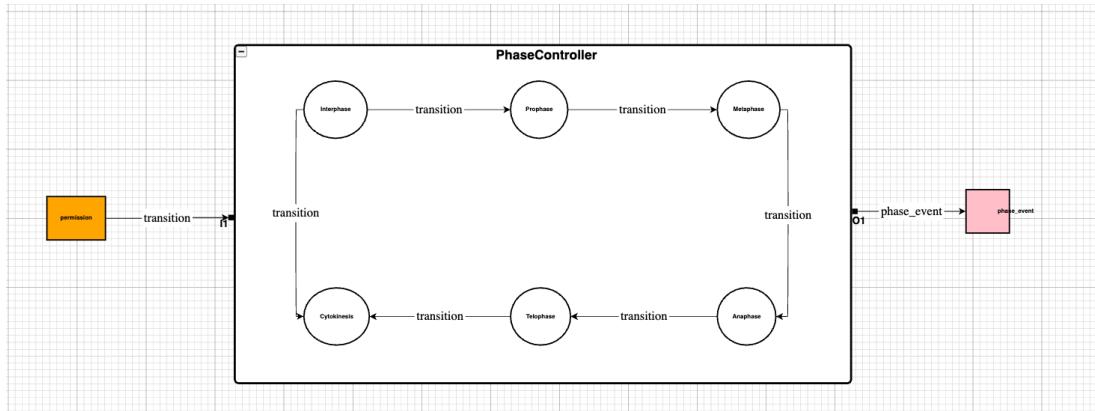


Figure 9: Phase Controller Atomic Model Diagram

$X = \text{permission} \in \{\text{transition}, \text{do\_not\_transition}\}$

$Y = \text{phase\_event} \in \{\text{Interphase}, \text{Prophase}, \text{Metaphase}, \text{Anaphase}, \text{Telophase}, \text{Cytokinesis}\}$

$S = \{\text{Interphase}, \text{Prophase}, \text{Metaphase}, \text{Anaphase}, \text{Telophase}, \text{Cytokinesis}\}$

$\delta_{\text{int}}(\text{Active}) = \text{Passive waiting}$

$\delta_{\text{ext}} =$

if transition:

- Interphase  $\rightarrow$  Prophase
- Prophase  $\rightarrow$  Metaphase
- Metaphase  $\rightarrow$  Anaphase
- Anaphase  $\rightarrow$  Telophase
- Telophase  $\rightarrow$  Cytokinesis
- Cytokinesis  $\rightarrow$  Interphase

else if do\_not\_transition:

- stay in current state

$\lambda = \text{outputs current phase event}$

$ta = \infty$  (waiting), 0 on transition

## Atomic Model Test Cases

### Chromosome Test Cases

Initial State: uncondensed

#### **TC-C1: Prophase Trigger**

Starts from the uncondensed state and goes to condensed based on a single Prophase at t =0.

After an internal delay to aligned, it should output 'not\_ready' until the full progression is done; then it goes to ready.

Item	Value
Initial State	uncondensed
Input	prophase at t=0
Expected State	condensed
After ta(condensed) = t1	aligned
Output	not_ready until completed
Final Output	ready when completed

#### **TC-C2: Full Phase Progression**

Feeds the full phase sequence (Prophase -> Metaphase -> Anaphase -> Telophase) and checks that the chromosome goes through its full sequence of states (and outputs ready when completed).

Input sequence:

```
0.0 Prophase  
10.0 Metaphase  
15.0 Anaphase  
20.0 Telophase
```

Expected Output:

condensed → aligned → separating → decondensing → completed

- Output = ready when progression complete

### ***TC-C3: Invalid Phase Jump***

This test sends an Anaphase at t = 0, which should cause the chromosome to stay uncondensed because it never received Prophase/Metaphase (so no transition).

Input Sequence

0.0 Anaphase

Expected Output

## **NuclearEnvelope Test Cases**

### ***TC-NE1: Prophase Breakdown***

One Prophase at t=0, which means the nuclear envelope should go to disassembling then (after ta) to absent and output 'not\_ready'.

Input Sequence:

0.0 Prophase

Expected Output:

- State = assembling
- Eventually absent
- Output = not\_ready

### ***TC-NE2: Telophase Reforming***

One Telophase at t=0; tests that the nuclear envelope goes from reforming to restored and outputs ready.

Input Sequence:

0.0 Telophase

Expected Output:

reforming → restored

- Output = ready

## SpindleApparatus Test Case

### *TC-SP1: Normal Sequence*

This test goes through the full phase sequence for the spindle apparatus (forming -> attached -> pulling -> disassembling -> inactive) and outputs ready when it is complete.

Input sequence:

0.0 Prophase  
10.0 Metaphase  
20.0 Anaphase  
30.0 Telophase

Expected Output:

forming → attached → pulling → disassembling → inactive

- Output = ready when progression complete

## Centrosome Test Case

### *TC-CT1: Normal Pole Migration*

The test feeds the centrosome the phases Interphase, then Prophase, Metaphase, then Anaphase at the given times and checks that its state goes from stable-> duplicated -> migrating -> at\_poles, and that it reports ready once it is at\_poles.

Input sequence:

0.0 Interphase  
5.0 Prophase  
10.0 Metaphase  
15.0 Anaphase

Expected Output:

stable → duplicated → migrating → at\_poles

- Output = ready when at \_poles

## Nucleolus Test Case

### *TC-NU1: Disappearance and Reappearance*

Sends Prophase at time 0 and Telophase at time 20 and checks that the nucleolus goes from visible -> disappeared -> reappearing -> visible. It should then report ready once that full sequence is done.

Input sequence:

```
0.0 Prophase
20.0 Telophase
```

Expected Output:

```
visible → disappeared → reappearing → visible
```

- output = ready when progression complete

## StatusChecker Test Case

### *TC-SC1: All Components Ready*

This test gives the StatusChecker “ready” from all components (chromosome, nuclear envelope, spindle, centrosome, nucleolus) at the same time and checks that it outputs allow\_transition.

Inputs at the same time:

```
chrom_status = ready
ne_status = ready
sp_status = ready
cen_status = ready
nucleo_status = ready
```

Expected Output:

- Output: allow\_transition

## PhaseControllerTest Case

### *TC-PC1: All Components Ready*

The test sends “transition” and checks that the PhaseController moves to the next phase (e.g, from Interphase -> Prophase), outputs that phase, and that it’s next internal transition happens immediately.

Input:

Permission = transition

Expected Output:

- Output: phase\_event at which the model is transitioning
- Ex : Interphase → Prophase
- ta = 0
- Ex: Phase\_event = Prophase

### ***TC-PC2: Deny Transition***

The test sends a “do not transition” and checks that the PhaseController does not change phase, does not produce an output, and keeps waiting until it receives “transition”.

Inputs at the same time:

Permission = do\_not\_transition

Expected Output

- Output: None
- ta =  $\infty$  (waiting)
- deny transition until Permission = transition

**Note:** The test result files for these simulation results can be found at

[https://github.com/GAchuzia/Cadmium-Mitosis/tree/main/simulation\\_results](https://github.com/GAchuzia/Cadmium-Mitosis/tree/main/simulation_results)

## PART III: Execution Results and Analysis of the Mitosis Simulation

### Simulation Results

For the atomic models, we ran the above mentioned test cases and for all of them received the expected output. This helped ensure the Cadmium DEVS atomic models were done correctly but that still meant we can run into issues when combining these atomic models to work together synchronously. Before testing out the entire model, we tested out the coupled models separately to make sure their behavior is correct and ensure our time advance values were ideal.

#### CentrosomePair (Coupled Model)

The CentrosomePair is made up of two Centrosome atomic models, Centrosome<sub>Top</sub> and Centrosome<sub>Bottom</sub>. These models are synched by the CentrosomePair as it takes in input\_phase and passed into the both atomic models as their input. The model takes the output of both Centrosomes which returns either "ready" or "not\_ready". If both of them are "ready", it will return "ready". If only one of them or both are not, it will return "not\_ready".

Input Sequence:

```
0.0 Interphase
5.0 Prophase
10.0 Metaphase
15.0 Anaphase
20.0 Telophase
```

Simulation result:

```
time;model_id;model_name;port_name;data
0;1;centrosome;;stable
0;2;input_reader;;0
0;1;centrosome;;stable
0;2;input_reader;out;Interphase
0;2;input_reader;;5
5;1;centrosome;;duplicated
5;2;input_reader;out;Prophase
5;2;input_reader;;5
7;1;centrosome;status_out;not_ready
7;1;centrosome;;duplicated
10;1;centrosome;;migrating
10;2;input_reader;out;Metaphase
10;2;input_reader;;5
13;1;centrosome;status_out;not_ready
13;1;centrosome;;migrating
15;1;centrosome;;at_poles
15;2;input_reader;out;Anaphase
15;2;input_reader;;5
15;1;centrosome;status_out;ready
15;1;centrosome;;at_poles
20;1;centrosome;;at_poles
20;2;input_reader;out;Telophase
20;2;input_reader;;inf
20;1;centrosome;status_out;ready
20;1;centrosome;;at_poles
20;1;centrosome;;at_poles
20;2;input_reader;;inf
```

Figure 10: Simulation Results for CentrosomePair

As we can see, at 0s, both Centrosome atomic models start at "stable" state. At 5s, the input\_phase is "Prophase". Then, both models switch to "duplicated" state and status\_out is printed as "not\_ready" at 7s as the time advance is 2.0 for that state. At 10s, the input\_phase is "Metaphase", then the model goes to "migrating" state and status\_out is printed as "not\_ready" at 13s as the time advance is 3.0 for that state. At 15s, the input\_phase is "Anaphase", then the model goes to "at\_poles" state and status\_out is printed as "ready" as the time advance is 0.0 for that state so it would log it. The same thing happens at t=20 when input\_phase is "Telophase".

## **PhaseControl(Coupled Model)**

The PhaseControl model is made up of two atomic models: the PhaseController and the StatusChecker models. The StatusChecker takes in all five outputs from the CellStructures model and compares their status as ready / not ready. Once all five CellStructure outputs are set to ready, the StatusChecker outputs an “allow\_transition” message to the PhaseController which then sends out the next mitosis state name to the CellStructures as “phase\_event”.

Input sequence example:

Spindle input:

```
0.0 ready
10.0 not_ready
20.0 ready
35.0 not_ready
45.0 ready
60.0 not_ready
70.0 ready
```

Centroid input:

```
0.0 ready
5.0 not_ready
20.0 ready
30.0 not_ready
45.0 ready
55.0 not_ready
70.0 ready
```

... same for the remaining three cell structures.

```

Lovett@devssim:~/Cadmium-Mitosis$ ./simulation
time;model_id;model_name;port_name;data
0;1;nucleo_reader;;0
0;3;phaseController;;Interphase
0;4;statusChecker;;Interphase
0;5;cen_reader;;0
0;6;sp_reader;;0
0;7;ne_reader;;0
0;8;chrom_reader;;0
0;1;nucleo_reader;out;ready
0;1;nucleo_reader;;10
0;4;statusChecker;;Prophase
0;5;cen_reader;out;ready
0;5;cen_reader;;5
0;6;sp_reader;out;ready
0;6;sp_reader;;15
0;7;ne_reader;out;ready
0;7;ne_reader;;10
0;8;chrom_reader;out;ready
0;8;chrom_reader;;5
0;3;phaseController;;Interphase
0;4;statusChecker;transition;allow_transition
0;4;statusChecker;;Prophase
5;4;statusChecker;;Prophase
5;5;cen_reader;out;not_ready
5;5;cen_reader;;15
5;8;chrom_reader;out;not_ready
5;8;chrom_reader;;15
10;1;nucleo_reader;out;not_ready
10;1;nucleo_reader;;10
10;4;statusChecker;;Prophase
10;7;ne_reader;out;not_ready
10;7;ne_reader;;10
15;4;statusChecker;;Prophase
15;6;sp_reader;out;not_ready
15;6;sp_reader;;5
20;1;nucleo_reader;out;ready
20;1;nucleo_reader;;15
20;4;statusChecker;;Metaphase
20;5;cen_reader;out;ready
20;5;cen_reader;;10
20;6;sp_reader;out;ready

```

*Figure 11: PhaseControl Simulation Results*

At  $t = 0$ s, the system initializes in Interphase. The PhaseControl outputs Interphase, and the statusChecker is also in the Interphase state. All structure readers (nucleo\_reader, cen\_reader, sp\_reader, ne\_reader, and chrom\_reader) initially report 0, then output “ready” to show that during the Interphase all cellular structures satisfy the readiness condition. Once they are all ready, the statusChecker produces a transition;allow\_transition event. This enables the

phaseController to move forward. At  $t = 10s$ , structures report not \_ ready which happens while the cell components are progressing. The statusChecker remains in Prophase since the readiness condition is still false. At  $t = 20s$ , all cell components are “ready” again and the statusChecker transitions to the Metaphase, confirming that all the conditions defined in the formal specification are satisfied.

### **CellStructures (Coupled Model)**

The CellStructures model is composed of four cell part atomic models, and the coupled centrosome model: CentrosomePair, Chromosome, NuclearEnvelope, Nucleolus, and Spindle. It takes its input from the PhaseControl output as the “phase\_event” or the name of the phase the model is simulating, and outputs “ready” for each Atomic part that finishes its internal transition.

Input sequence example:

```
0.0 Interphase
```

```

> Lovett@devssim:~/Cadmium-Mitosis$ ./simulation
time;model_id;model_name;port_name;data
0;2;nucleo;;visible
0;4;centrosome_Bottom;;stable
0;5;centrosome_Top;;stable
0;6;sp;;inactive
0;7;ne;;intact
0;8;chrom;;uncondensed
0;9;input_reader;;0
0;2;nucleo;;visible
0;4;centrosome_Bottom;;stable
0;5;centrosome_Top;;stable
0;6;sp;;inactive
0;7;ne;;intact
0;8;chrom;;uncondensed
0;9;input_reader;out;Interphase
0;9;input_reader;;5
5;2;nucleo;;disappearing
5;4;centrosome_Bottom;;duplicated
5;5;centrosome_Top;;duplicated
5;6;sp;;forming
5;7;ne;;disassembling
5;8;chrom;;condensing
5;9;input_reader;out;Prophase
5;9;input_reader;;5
6;7;ne;status_out;ready
6;7;ne;;absent
6;8;chrom;status_out;ready
6;8;chrom;;condensed
7;2;nucleo;status_out;ready
7;2;nucleo;;disappeared
7;4;centrosome_Bottom;status_out;ready
7;4;centrosome_Bottom;;duplicated
7;5;centrosome_Top;status_out;ready
7;5;centrosome_Top;;duplicated
7;6;sp;status_out;ready
7;6;sp;;forming
10;2;nucleo;;absent_ack
10;4;centrosome_Bottom;;migrating
10;5;centrosome_Top;;migrating
10;6;sp;;attached
10;7;ne;;absent_ack
10;8;chrom;;aligning
10;9;input_reader;out;Metaphase
10;9;input_reader;;5

```

*Figure 12: CellStructures Simulation Results*

## Analysis of Output of Top Model

The top model runs a single simulation on a fixed time horizon ( $t = 100$ ), with multiple atomic models in parallel. Each atomic model is driven by an IEStream reader that injects phase events from the same input files used in atomic atomic test cases. The coordinator advances simulation time and processes internal and external events for all components together, so the global log is one timeline with events from every model interwoven by time.

```

● Qureshi@devssim:~/Cadmium-Mitosis$ ./build_sim.sh
Build successful! Run it using: ./simulation
● Qureshi@devssim:~/Cadmium-Mitosis$ ./simulation
time;model_id;model_name;port_name;data
0;3;nucleo;;visible
0;5;centrosome_Bottom;;stable
0;6;centrosome_Top;;stable
0;7;sp;;inactive
0;8;ne;;intact
0;9;chrom;;uncondensed
0;11;phaseController;;Interphase
0;12;statusChecker;;Interphase
0;13;start_reader;;1
1;12;statusChecker;;Prophase
1;13;start_reader;out;"start"
1;13;start_reader;;inf
1;11;phaseController;;Prophase
1;12;statusChecker;transition;allow_transition
1;12;statusChecker;;Prophase
1;3;nucleo;;disappearing
1;5;centrosome_Bottom;;duplicated
1;6;centrosome_Top;;duplicated
1;7;sp;;forming
1;8;ne;;disassembling
1;9;chrom;;condensing
1;11;phaseController;phase_event;Prophase
1;11;phaseController;;Prophase
2;8;ne;status_out;ready
2;8;ne;;absent
2;9;chrom;status_out;ready
2;9;chrom;;condensed
2;12;statusChecker;;Prophase
3;3;nucleo;status_out;ready
3;3;nucleo;;disappeared
3;5;centrosome_Bottom;status_out;ready
3;5;centrosome_Bottom;;duplicated
3;6;centrosome_Top;status_out;ready
3;6;centrosome_Top;;duplicated
3;7;sp;status_out;ready
3;7;sp;;forming
3;12;statusChecker;;Metaphase
3;11;phaseController;;Metaphase
3;12;statusChecker;transition;allow_transition
3;12;statusChecker;;Metaphase
3;3;nucleo;;absent_ack
3;5;centrosome_Bottom;;migrating
3;6;centrosome_Top;;migrating
3;7;sp;;attached
3;8;ne;;absent_ack

```

```

Qureshi@devssim:~/Cadmium-Mitosis$ ./simulation
3;8;ne;;absent_ack
3;9;chrom;;aligning
3;11;phaseController;phase_event;Metaphase
3;11;phaseController;;Metaphase
3.1;3;nucleo;status_out;ready
3.1;3;nucleo;;disappeared
3.1;8;ne;status_out;ready
3.1;8;ne;;absent
3.1;12;statusChecker;;Metaphase
4;9;chrom;status_out;ready
4;9;chrom;;aligned
4;12;statusChecker;;Metaphase
6;5;centrosome_Bottom;status_out;ready
6;5;centrosome_Bottom;;migrating
6;6;centrosome_Top;status_out;ready
6;6;centrosome_Top;;migrating
6;7;sp;status_out;ready
6;7;sp;;attached
6;12;statusChecker;;Anaphase
6;11;phaseController;;Anaphase
6;12;statusChecker;transition;allow_transition
6;12;statusChecker;;Anaphase
6;3;nucleo;;absent_ack
6;5;centrosome_Bottom;;at_poles
6;6;centrosome_Top;;at_poles
6;7;sp;;pulling
6;8;ne;;absent_ack
6;9;chrom;;separating
6;11;phaseController;phase_event;Anaphase
6;11;phaseController;;Anaphase
6;5;centrosome_Bottom;status_out;ready
6;5;centrosome_Bottom;;at_poles
6;6;centrosome_Top;status_out;ready
6;6;centrosome_Top;;at_poles
6;12;statusChecker;;Anaphase
6.1;3;nucleo;status_out;ready
6.1;3;nucleo;;disappeared
6.1;8;ne;status_out;ready
6.1;8;ne;;absent
6.1;12;statusChecker;;Anaphase
7;9;chrom;status_out;ready
7;9;chrom;;separated
7;12;statusChecker;;Anaphase
9;7;sp;status_out;ready
9;7;sp;;pulling
9;12;statusChecker;;Telophase
9;11;phaseController;;Telophase
9;12;statusChecker;transition;allow_transition

```

```

9;12;statusChecker;;Telophase
9;3;nucleo;;reappearing
9;5;centrosome_Bottom;;at_poles
9;6;centrosome_Top;;at_poles
9;7;sp;;disassembling
9;8;ne;;reforming
9;9;chrom;;decondensing
9;11;phaseController;phase_event;Telophase
9;11;phaseController;;Telophase
9;5;centrosome_Bottom;status_out;ready
9;5;centrosome_Bottom;;at_poles
9;6;centrosome_Top;status_out;ready
9;6;centrosome_Top;;at_poles
9;12;statusChecker;;Telophase
10;8;ne;status_out;ready
10;8;ne;;restored
10;9;chrom;status_out;ready
10;9;chrom;;uncondensed
10;12;statusChecker;;Telophase
12;3;nucleo;status_out;ready
12;3;nucleo;;visible
12;7;sp;status_out;ready
12;7;sp;;disassembling
12;12;statusChecker;;Cytokinesis
12;11;phaseController;;Telophase
12;12;statusChecker;transition;Simulation Complete
12;12;statusChecker;;Cytokinesis
12;3;nucleo;;visible
12;5;centrosome_Bottom;;at_poles
12;6;centrosome_Top;;at_poles
12;7;sp;;disassembling
12;8;ne;;restored
12;9;chrom;;uncondensed
12;11;phaseController;;Telophase
12;12;statusChecker;;Cytokinesis
12;13;start_reader;;inf

```

*Figure 13: Final Mitosis System Simulation Results*

From the three above images, we can see the simulation results of the entire mitosis system. As discussed earlier, the stages of mitosis are interphase, prophase, metaphase, anaphase, telophase and cytokinesis. The system is initialized with interphase and once it reads the input.txt file to start simulation. Then the phase controller and status checker will move to the next phase, prophase. Once in prophase, the atomic models will do their external transition and time advance and output ready once they are done. This will move the system to metaphase, and repeat the same process and will move on to the next phase when all models in cell structures are ready. Once it reaches the Cytokinesis phase, the status checker will print out "Simulation Complete" and the status of all the models will print out which is their last state. From this simulation, we can see that atomic and coupled models are able to work together in the DEVS framework. It displayed the phases of the mitosis system and states for each part of the cell as the phases progressed.

## **Conclusion**

In conclusion, applying the DEVS (Discrete Event System Specification) framework to mitosis was an invaluable learning experience for the team as it was able to show how theoretical system design can be used to model biological complexity. We were able to validate the output behavior involved comparing the simulation's event log against experiment data to ensure internal transition and time advance were producing biologically accurate results.