

**ASSIGNMENT NO. 1**

**CADMIUM SIMULATION ON**

**ASSIGNMENT SUBMISSION CHECKER**

**MODEL**

**Course Title:**

**Methodologies for Discrete Event**

**Modelling and Simulation**

**Course Code: SYSC 5104**

**SUBMITTED TO:**

**Dr. Gabriel A. Wainer**

**Professor, Ph.D.**

**Systems and Computer Engineering**

**Carleton University, Ottawa, Canada**

**SUBMITTED BY:**

**Shubham Thakur**

**Student ID: 101134521**

**DATE OF SUBMISSION:**

**18/10/2019**

**PART I - Description of the Selected Model:**

The document describes a DEVS model that checks whether the assignment can be submitted by checking three things,

1. Submission time of the assignment.

2. If there is plagiarism in the assignment.

3. The email ID to which the assignment is submitted.

A successful response is provided if the assignment is submitted before the time (deadline), does not contain plagiarism, and is sent to the right email address. If any of the three checks came out to be false then an unsuccessful response is provided by the model. The figure below shows the block diagram for the Assignment Submission Checker.

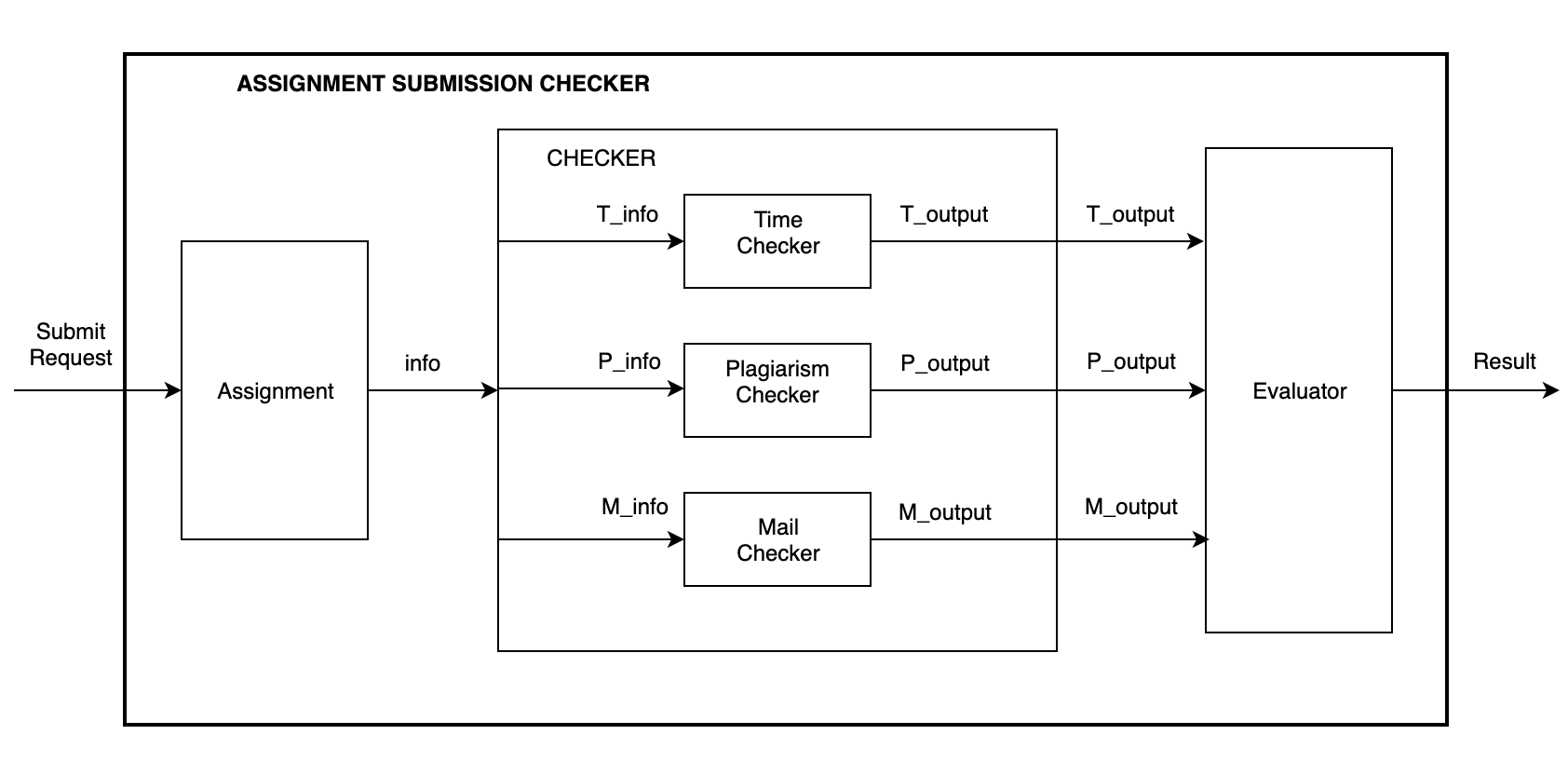


Figure 1: Block Diagram for Assignment Submission Checker Model

1. **Assignment:** The *Assignment* block represents that the submit request has been submitted. As soon as *Submit Request* (6-digit)input is provided, the *Assignment* block goes to active state and the information (*info*) is passed onto the *Checker* block. Otherwise it is always is passive state. The 1st digit of the input is always 1.
2. **Checker:** This block is used to check/validate the information (*info*). It consists of three sub-blocks namely, *Time Checker, Plagiarism Checker and Mail Checker.*

**2.1 Time Checker:** This block checks the time at which the *Submit Request* was submitted by extracting the data at 2nd and 3rd digit. The 24-hour format (1-24) will be followed. If the data is less than 10 then the submission is valid and will be accepted and *T\_output* will be 1, otherwise it will be 0.

**2.2 Plagiarism Checker:** This block will check whether there is plagiarism in the submitted assignment by extracting the data at 4th and 5th digit. If the value is 11 then there is no plagiarism and if the value is anything other than 11 then the assignment contains plagiarism. If there is no plagiarism in the assignment then *P\_output* will be given as 1, otherwise it will be 0.

**2.3 Email Checker:** This block will check if the assignment has been submitted to the correct email address by extracting the data at 6th digit. It can have two values 0 or 1. If the email address is correct (has value 1) then *M\_output* will be provided as 1 and if the email address is incorrect (has value 0) then *M\_output* will be given as 0.

1. **Evaluator:** This block will get the output of three sub-blocks of *Checker* block as inputs. It will evaluate the outputs of *Time Checker, Plagiarism Checker and Email Checker* blocks and give the final output. If all the inputs to this block are 1, then the assignment is valid and can be accepted and *Result* will be given as 1. On the other hand, if any one of the inputs is 0 then *Result* will be evaluated to 0. The *Evaluator* decides whether the final result came out to be successful or unsuccessful.

**PART II - DEVS Specification for Atomic and Coupled Models:**

The Assignment Submission Checker takes one 6-digit integer input (submitRequest) and gives one output (Result). The first digit of the input will always be 1. The second and third digits represents the value of submission time (1 to 24). The fourth and fifth digits represents the plagiarism status of the assignment. The last digit shows if the assignment is submitted to the correct email address and can either be 0 or 1.

**1. DEVS Formal Specification for Atomic Models:**

The DEVS formal specification of all the atomic models in the Assignment Submission Checker Model are given below:

DEVS Structure = <X, Y, S, δint, δext, λ, ta>

**1. Assignment Component:**

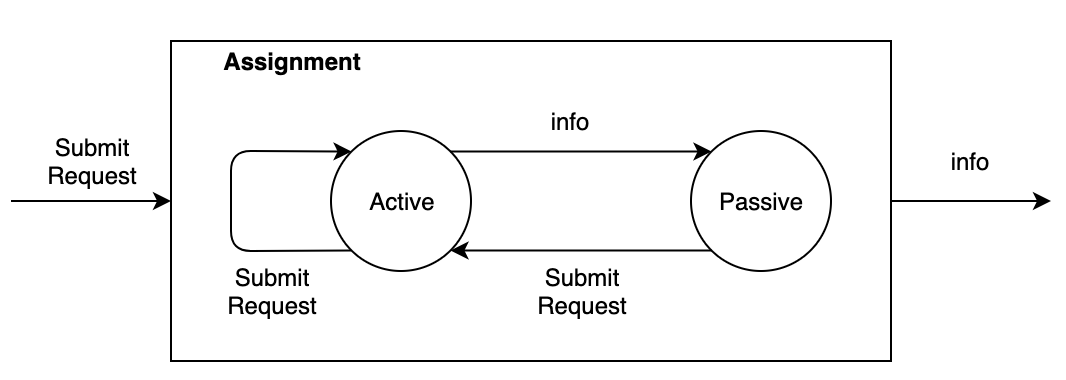


Figure 2: Atomic Model for Assignment

X = {SubmitRequest}

Y = {info}

S = {Active, Passive}

δint (Active) = Passive

δext (Submit Request, Passive) = Active

δext (Submit Request, Active) = Active

λ (Active) {

Send *“info”* as input to the *Checker* block when *“Submit Request”* is given.

}

ta (Passive) = INFINITY

ta (Active) = A\_time //Time taken by the *Assignment* to forward *“info”* to *Checker*.

**2. Checker Component:**

This block has three sub-modules namely, *Time Checker, Plagiarism Checker and Mail Checker.*

**2.1 Time Checker:**

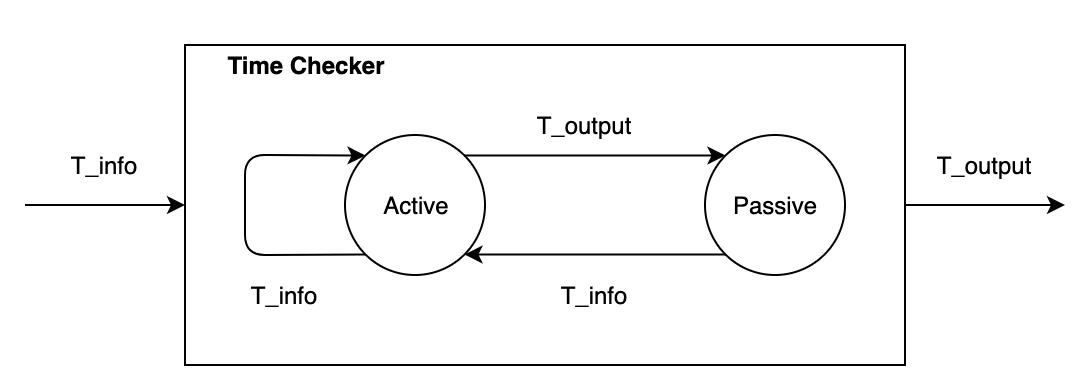


Figure 3: Atomic Model for Time Checker

X = {T\_info}

Y = {T\_output}

S = {Active, Passive}

δint (Active) = Passive

δext (T\_info, Passive) = Active

δext (T\_info, Active) = Active

λ (Active) {

Extract and check the time field (2nd & 3rd digit) from input *“info”*.

If it is smaller than 10 then the output is 1 else the output is 0.

}

ta (Passive) = INFINITY

ta (Active) = T\_time //Time taken by *Time Checker* to generate output.

**2.2 Plagiarism Checker:**

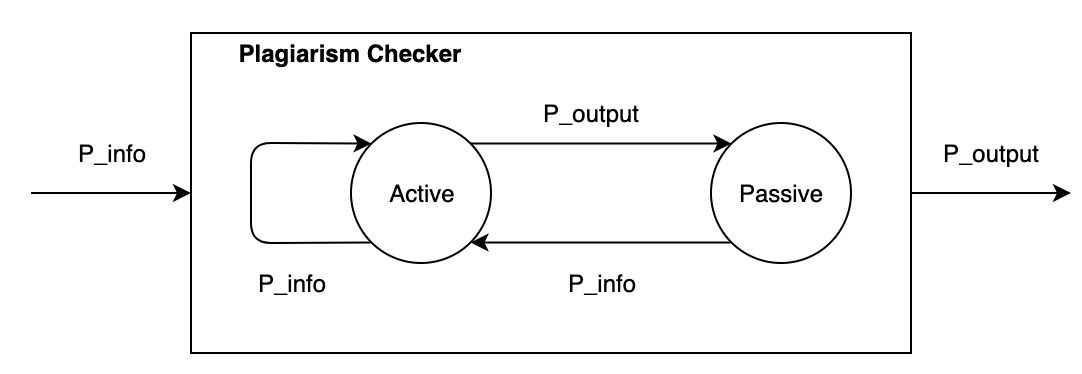


Figure 4: Atomic Model for Plagiarism Checker

X = {P\_info}

Y = {P\_output}

S = {Active, Passive}

δint (Active) = Passive

δext (P\_info, Passive) = Active

δext (P\_info, Active) = Active

λ (Active) {

Extract and check the plagiarism field (4th & 5th digit) from input *“info”*.

If it is equal to 11 then the output is 1 (no plagiarism) , else the output is always 0 (plagiarism is present).

}

ta (Passive) = INFINITY

ta (Active) = P\_time //Time taken by *Plagiarism Checker* to generate output.

**2.3 Mail Checker:**

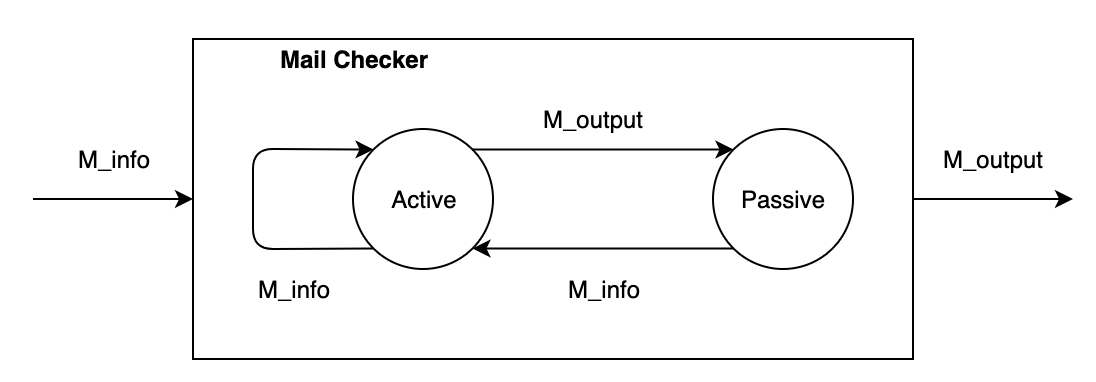


Figure 5: Atomic Model for Main Checker

X = {M\_info}

Y = {M\_output}

S = {Active, Passive}

δint (Active) = Passive

δext (M\_info, Passive) = Active

δext (M\_info, Active) = Active

λ (Active) {

Extract and check the email field (6th digit) from input *“S\_info”*.

If it is equal to 1 then the output is 1 and if it is equal to 0 then output is 0.

}

ta (Passive) = INFINITY

ta (Active) = E\_time //Time taken by *Email Checker* to generate output.

**3. Evaluator Component:**

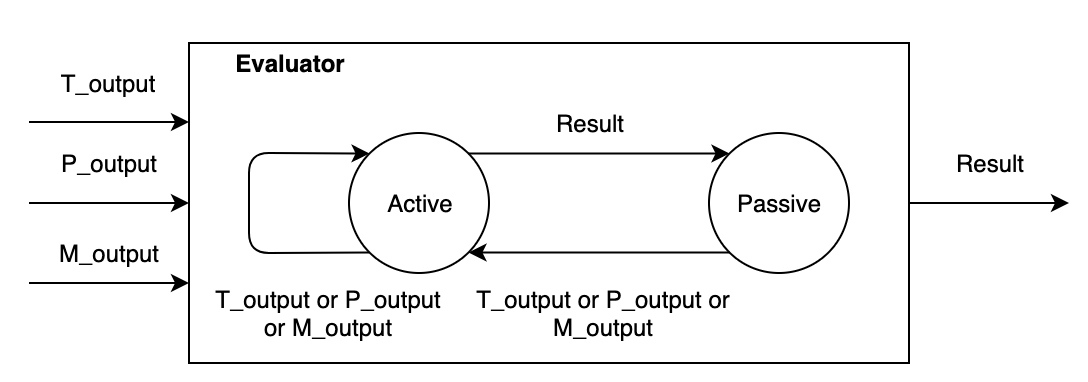
****

Figure 6: Atomic Model for Evaluator

X = {T\_output, P\_output, M\_output}

Y = {Result}

S = {Active, Passive}

δint (Active) = Passive

δext (T\_output or P\_output or M\_output, Passive) = Active

δext (T\_output or P\_output or M\_output, Active) = Active

λ (Active) {

Check the inputs *“T\_output”, “P\_output” and “M\_output”*

If all of them are equal to 1 then the *“Result”* is 1, otherwise the *“Result”* is 0.

}

ta (Passive) = INFINITY

ta (Active) = Eval\_time //Time taken by *Evaluator* to generate *“Result”.*

**2. DEVS Formal Specification for Coupled Models:**

The DEVS formal specifications for all coupled models are given below.

DEVS Coupled Structure = <I, X, Y, D, {Mi}, {Ii}, {Zij}>

**1. Checker Component:**

I = {T\_info, P\_info, M\_info, T\_output, P\_output, M\_output}

X = {info}

Y = {T\_output, P\_output, M\_output}

M1 = {TimeChecker}

M2 = {PlagiarismChecker}

M3 = {EmailChecker}

I1 = {Assignment}

I2 = {Evaluator}

Zij =

info T\_info@TimeChecker

info P\_info@PlagiarismChecker

info M\_info@MailChecker

T\_output@TimeChecker T\_output

P\_output@PlagiarismChecker P\_output

M\_output@MailChecker M\_output

**2. Assignment Submission Checker:**

This is the highest level of the model.

I = {Submit Request, Result}

X = {Submit Request}

Y = {Result}

M1 = {Assignment}

M2 = {Checker}

M3 = {Evaluator}

Zij =

SubmitRequest SubmitRequest@Assignment

info@Assignmnet info Checker

T\_output@Checker T\_output@Evaluator

P\_output@Checker P\_output@Evaluator

M\_output@Checker M\_output@Evaluator

Result@Evaluator Result

**PART III - Test Case Execution and Simulation Results:**

**1. Test Strategy:**

The atomic models and coupled will be tested using the “Black Box” testing method**.** An input file (.txt) containing test cases will be given to every atomic and coupled model. Various test case scenarios are created inside these input test files which. All of the input files are stored in the *“ASC/input”* folder. Based on the input file given, different outputs files are generated and stored in the *“ASC/results”*. After running the tests, it was found that all the atomic and coupled models are working as expected.

For unit testing purposes, a separate input file is provided to every atomic model. After making sure that unit testing is successful and outputs comes out to be exactly same as expected, the models were integrated and tested as a single entity using only one input file named *“asc\_input.txt”* which will give us two output files namely, *“ASC\_output\_messages.txt”* and *“ASC\_output\_state.txt”*.

**2. Test Cases Execution:**

**2.1 Assignment Atomic Model:**

This atomic model takes input from *“SubmitRequest”* port. An input file from *“ASC/input”* folder with name *“assignment\_input\_test.txt”* is provided and two output files are obtained namely, *“assignment\_test\_output\_messages.txt”* and *“assignment\_test\_output\_state.txt”* inside the *“ASC/results”* folder.

This model will read the data from the input file and will pass on the data to *Checker Model* through the *“info”* port. Below are the contents of the assignment input file.

00:00:20 101043

00:00:30 102432

00:00:40 111000

00:00:50 115111

00:01:50 123311

00:02:00 103420

00:02:10 103111

00:02:20 105640

00:02:30 116051

00:02:40 107460

00:02:50 110111

00:03:00 108180

00:03:20 109652

assignment\_input\_test.txt

To see the complete output of the model, open *“results/assignment\_test\_output\_messages.txt*”.

**2.2 Time Checker Atomic Model:**

The input of the Time Checker Model comes through the *“T\_info”*. The file contains a 6-digit input which consists of the information regarding the time of submission, plagiarism status and the submission mail ID. The Time Checker Model extracts the time field (2nd & 3rd digit) from the input and checks whether it is smaller than 10 which represents that the assignment is submitted before time and results in output *“T\_output”* to be 1. Otherwise the output *“T\_output”* is 0. The output is generated after a fixed time duration of 1-time unit. *“timechecker\_input\_test.txt”* is the input file to Time Checker Model and its contents are given below:

00:00:10 100111

**00:00:20 101043**

**00:00:30 102432**

00:00:40 111000

00:00:50 115111

00:01:50 123311

**00:02:00 103420**

**00:02:10 103111**

**00:02:20 105640**

00:02:30 116051

**00:02:40 107460**

00:02:50 110111

**00:03:00 108180**

**00:03:20 109652**

timechecker\_input\_test.txt

The inputs in bold are the cases which results in the output *“T\_output”* to be 1 representing that the assignment is submitted on or before time. The check the complete output, kindly open *“ASC/results/timechecker\_test\_output\_messages.txt”*. Some chunk of the output files is given below:

00:00:10:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<Message\_t>::out: {100111}] generated by model input\_reader

00:00:11:000

[TimeChecker\_defs::T\_output: {0}] generated by model timechecker1

00:00:20:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<Message\_t>::out: {101043}] generated by model input\_reader

00:00:21:000

**[TimeChecker\_defs::T\_output: {1}] generated by model timechecker1**

00:00:30:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<Message\_t>::out: {102432}] generated by model input\_reader

00:00:31:000

**[TimeChecker\_defs::T\_output: {1}] generated by model timechecker1**

00:00:40:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<Message\_t>::out: {111000}] generated by model input\_reader

00:00:41:000

[TimeChecker\_defs::T\_output: {0}] generated by model timechecker1

............

**2.3 Plagiarism Checker Model:**

The input of the Plagiarism Checker Model comes through the *“P\_info”*. The file contains a 6-digit input which consists of the information regarding the time of submission, plagiarism status and the submission mail ID. The Plagiarism Checker Model extracts the serial number field (4th & 5th digit) from the input and checks whether it is equal to 11 which represents that the assignment submitted is not plagiarized and results in output *“P\_output”* to be 1. Otherwise the *“P\_output”* is 0. The output is generated after a fixed time duration of 1-time unit. The input file, *“plagiarismchecker\_input\_test.txt”* is created for checking various possible values as follows:

**00:00:10 100111**

00:00:20 101043

00:00:30 102432

00:00:40 111000

**00:00:50 115111**

00:01:50 123311

00:02:00 103420

**00:02:10 103111**

00:02:20 105640

00:02:30 116051

00:02:40 107460

**00:02:50 110111**

00:03:00 108180

00:03:20 109652

plagiarism\_input\_test.txt

The inputs in bold are the cases which results in the output *“P\_output”* to be 1. To check the complete output, kindly open *“ASC/results/plagiarismchecker\_test\_output\_messages.txt”*. Some chunk of the output file is given below:

00:00:10:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<Message\_t>::out: {100111}] generated by model input\_reader

00:00:11:000

**[PlagiarismChecker\_defs::P\_output: {1}] generated by model plagiarismchecker1**

00:00:20:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<Message\_t>::out: {101043}] generated by model input\_reader

00:00:21:000

[PlagiarismChecker\_defs::P\_output: {0}] generated by model plagiarismchecker1

00:00:30:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<Message\_t>::out: {102432}] generated by model input\_reader

00:00:31:000

[PlagiarismChecker\_defs::P\_output: {0}] generated by model plagiarismchecker1

............

**2.4 Mail Checker Atomic Model:**

The input of the Mail Checker Model comes through the *“M\_info”*. The file contains a 6-digit input which consists of the information regarding the time of submission, plagiarism status and the submission mail ID. The model extracts the mail id field (6th digit) from the input and checks whether it is equal to 1 which represents that the assignment is submitted to the right mail id and results in output *“M\_output”* to be 1. The field 0 represents that the assignment is submitted to the wrong mail id and *“M\_output”* is 0. The time duration for output generation for this model is 1-time unit. The input file, *“mailchecker\_input\_test.txt”* is created for checking various possible values as follows:

**00:00:10 100111**

00:00:20 101043

00:00:30 102432

00:00:40 111000

**00:00:50 115111**

**00:01:50 123311**

00:02:00 103420

**00:02:10 103111**

00:02:20 105640

**00:02:30 116051**

00:02:40 107460

**00:02:50 110111**

00:03:00 108180

00:03:20 109652

mailchecker\_input\_test.txt

The inputs in bold are the cases which results in the output *“M\_output”* to be 1. To check the complete output, kindly open *“ASC/results/mailchecker\_test\_output\_messages.txt”*. Some chunk of the output file is given below:

00:00:40:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<Message\_t>::out: {111000}] generated by model input\_reader

00:00:41:000

[MailChecker\_defs::M\_output: {0}] generated by model mailchecker1

00:00:50:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<Message\_t>::out: {115111}] generated by model input\_reader

00:00:51:000

**[MailChecker\_defs::M\_output: {1}] generated by model mailchecker1**

00:01:50:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<Message\_t>::out: {123311}] generated by model input\_reader

00:01:51:000

**[MailChecker\_defs::M\_output: {1}] generated by model mailchecker1**

..........

**2.5 Evaluator Atomic Model:**

The input of the Evaluator Model is the output of the coupled Checker Model i.e. the output of the atomic models Time Checker, Plagiarism Checker and Mail Checker. The output of the Evaluator *“Result”* is 1 if and only if all three inputs are 1 (Successful Submission). Otherwise, the output *“Result”* is 0 (Unsuccessful Submission). The output is generated after a fixed time duration after the inputs are received. The time duration in our case is 1-time unit. This model takes three input files and their names are given below.

1. *“evaluator\_input\_test\_timechecker.txt”* //T\_output value

2. *“evaluator\_input\_test\_plagiarismchecker.txt”*  //P\_output value

3. *“evaluator\_input\_test\_mailchecker.txt”* //M\_output value

|  |  |  |
| --- | --- | --- |
| 00:00:10 0  00:00:20 1  00:00:30 1  00:00:40 0  00:00:50 0  00:01:50 0  00:02:00 1  **00:02:10 1**  00:02:20 1  00:02:30 0  00:02:40 1  00:02:50 0  00:03:00 1  00:03:20 1  evaluator\_input\_test\_timechecker.txt | 00:00:10 1  00:00:20 0  00:00:30 0  00:00:40 0  00:00:50 1  00:01:50 0  00:02:00 0  **00:02:10 1**  00:02:20 0  00:02:30 0  00:02:40 0  00:02:50 1  00:03:00 0  00:03:20 0  evaluator\_input\_test\_plagiarismchecker.txt | 00:00:10 1  00:00:20 0  00:00:30 0  00:00:40 0  00:00:50 1  00:01:50 1  00:02:00 0  **00:02:10 1**  00:02:20 0  00:02:30 1  00:02:40 0  00:02:50 1  00:03:00 0  00:03:20 0  evaluator\_input\_test\_mailchecker.txt |

The inputs in the bold are the only case which results in the successful submission of the assignment shown by output value *“Result”* equal to 1. Any other input combination will result in *“Result”* 0 representing unsuccessful submission. To check the complete output, kindly open *“ASC/results/evaluator\_test\_output\_messages.txt”*. Some chunk of the output file is given below:

00:02:10:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<Message\_t>::out: {1}] generated by model input\_reader\_timechecker

[cadmium::basic\_models::pdevs::iestream\_input\_defs<Message\_t>::out: {1}] generated by model input\_reader\_plagiarismchecker

[cadmium::basic\_models::pdevs::iestream\_input\_defs<Message\_t>::out: {1}] generated by model input\_reader\_mailchecker

00:02:11:000

**[Evaluator\_defs::Result: {1}] generated by model evaluator1**

.........

**2.6 Coupled Assignment Submission Checker Model:**

The coupled Assignment Submission Checker Model is the top model which consists of atomic

Assignment Model, Evaluator Model and Coupled Checker Model. The Checker consists of atomic models Time Checker, Plagiarism Checker and Mail Checker. The input *“info”* is sent by the Assignment Model to the Checker when the input *“Submit Request”* is received. The Checker checks the time, plagiarism status and mail id of the assignment and sends the corresponding output to the Evaluator Model, which generates the final output *“Result”* based on all three inputs. If all three inputs are 1, the output is 1 representing that the assignment is submitted successfully. Otherwise the output is 0 representing that the assignment submission is unsuccessful. The time duration for output generation for this model is 1-time unit. The input file, *“asc\_input.txt”* is created for checking various possible values as follows:

00:00:10 100111

00:00:20 101043

00:00:30 102432

00:00:40 111000

00:00:50 115111

00:01:50 123311

00:02:00 103420

**00:02:10 103111**

00:02:20 105640

00:02:30 116051

00:02:40 107460

00:02:50 110111

00:03:00 108180

00:03:20 109652

asc\_input.txt

The inputs in bold are the cases which results in the output *“Result”* to be 1. To check the complete output, kindly open *“ASC/results/mailchecker\_test\_output\_messages.txt”*. Some chunk of the output file is given below:

00:02:10:000

[cadmium::basic\_models::pdevs::iestream\_input\_defs<Message\_t>::out: {103111}] generated by model input\_reader

00:02:11:000

[Assignment\_defs::info: {103111}] generated by model assignment1

00:02:12:000

[TimeChecker\_defs::T\_output: {1}] generated by model timechecker1

[PlagiarismChecker\_defs::P\_output: {1}] generated by model plagiarismchecker1

[MailChecker\_defs::M\_output: {1}] generated by model mailchecker1

00:02:13:000

[Evaluator\_defs::Result: {1}] generated by model evaluator1

..........