Assignment 3

Formal Methods

Noor ul Sehr Zia | BESE 5A

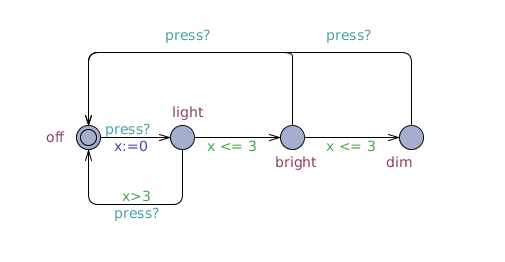
Link to Github repository: <https://github.com/NoorZia/FMassignment3>

# Extension of Light Model:

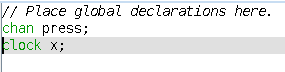
The light model had the following states:

* Off
* Light
* Bright

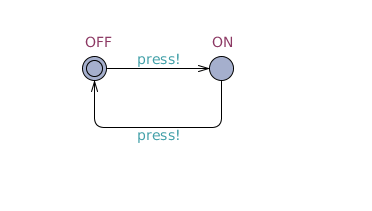
1. I’ve added the state **Dim** which will be assigned only when the button is pressed 3 times within the initial 3 time units. Otherwise it will go to Off state.



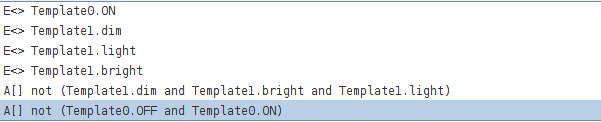
1. A **clock** which is needed in a timed automaton. The clock is denoted by **x**.

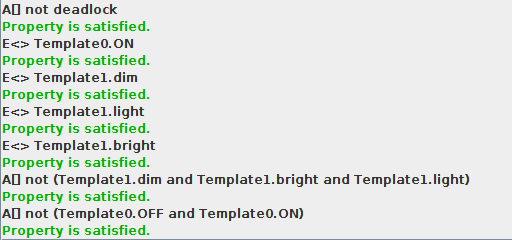


1. Corresponding output channel for the input channel **press** associated with the Light automaton. This is a button with two states OFF and ON and a transition is fired every time it is pressed.

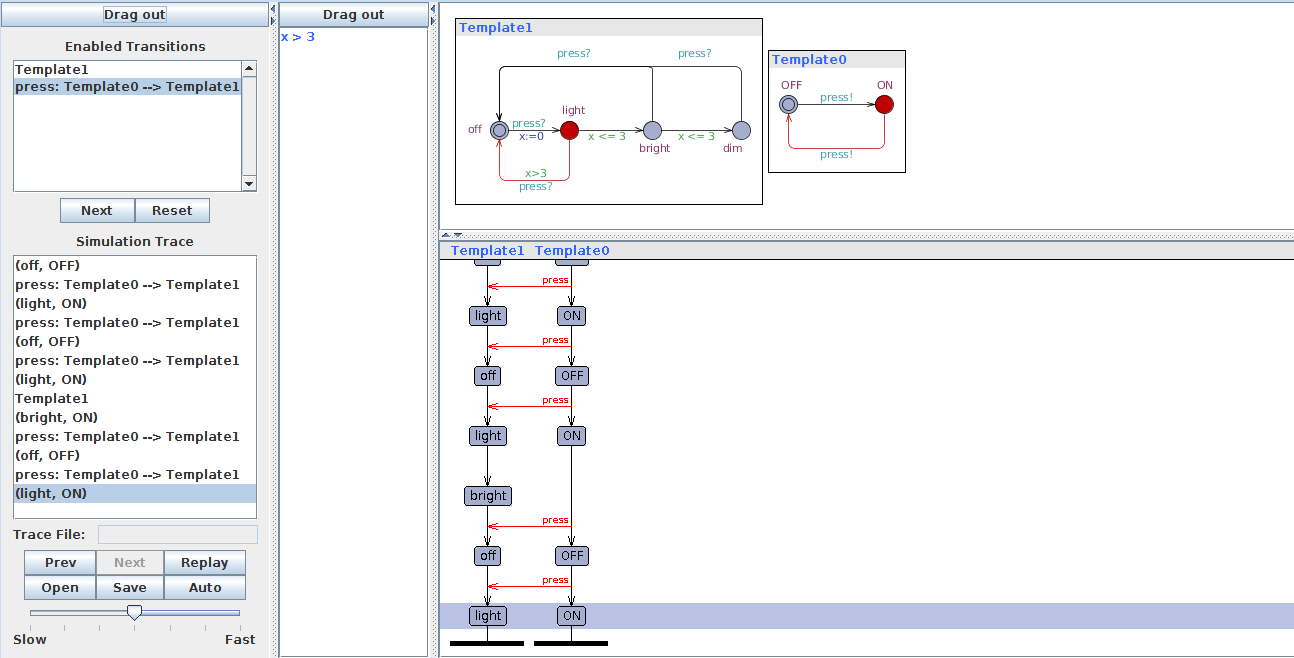


1. Properties are checked and satisfied including the synchronization properties such as deadlock, reachability and mutual exclusion.





## Simulation:



# Gear Box Simulation:

## Problem Statement:

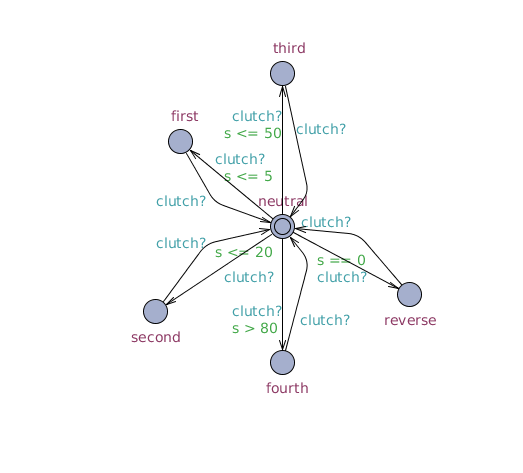
A typical single clutch gearbox of a car consist of 5 gears in total including the reverse. These gears have some properties that need to be fulfilled for the smooth working of the gearbox. The initial state is neutral. A gear can only be shifted from neutral i.e it is not the case that that a gear can be shifted without passing through the neutral state. The gears come with their optimum speeds. The reverse gear can only be engaged when the speed is zero. Similarly, the first gear can only be engaged when the speed is less than 10 units, second when speed is less than 20, third when speed is less than 50. Fourth gear can only be engaged when speed is greater than 80.

A gear can only be shifted when the clutch has been pressed. These constraints are essential for the working of the gearbox as if any of these do not satisfy the gear box can be harmed significantly.

## Modelling on UPPAAL:

Two templates are required for modelling the gearbox.

1. GearBox itself which has an input channel as clutch and six states in total.

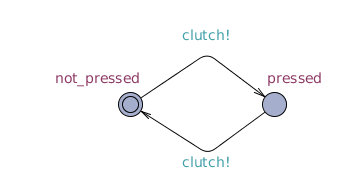


The guards are the speed constraints and the synchronization is the clutch input channel.

Since there is an input channel in the template above it is necessary that an output channel exist for that automaton.

1. ClutchPlate

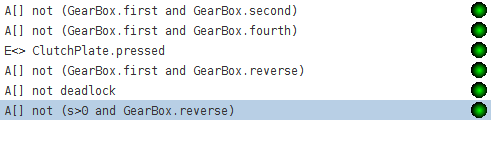
The clutch plate automaton consist of the clutch channel output which is used as a synchronization channel in the gearbox. This automaton upholds the constraint that the clutch must be pressed while shifting gear i.e whenever a transition is fired. It has only two states Pressed and Not Pressed.

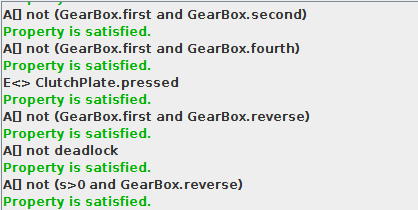


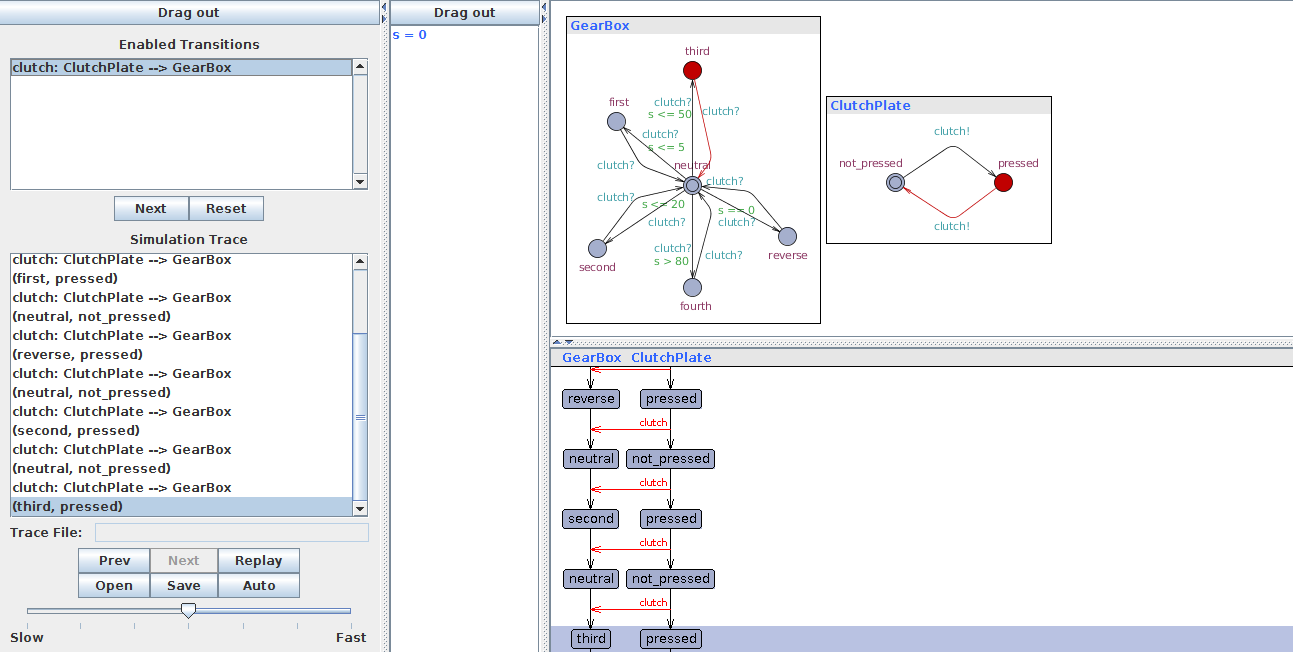
There are only two global declarations:

gear31.png

All of the constraints mentioned in the problem have been translated as properties for the Uppaal verifier and have been verified. These include deadlock avoidance, synchronization, safety, reachability, correctness properties as well. All of these properties are satisfied in the model. The query file is included with the code.







# Mutual Exclusion Example Implementation:

Implementation of Mutual exclusion protocol that checks that no two processes enter the critical section at the same time.

## Template of process

This template will be initialized twice, once for process P1 and then for P2.

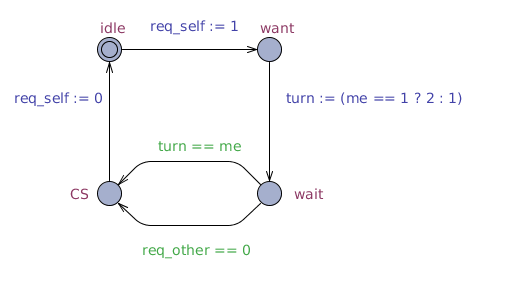
// Place template instantiations here.

P1 = Mutex(1, req1, req2);

P2 = Mutex(2, req2, req1);

// List one or more processes to be composed into a system.

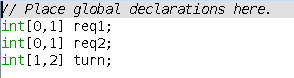
system P1, P2;



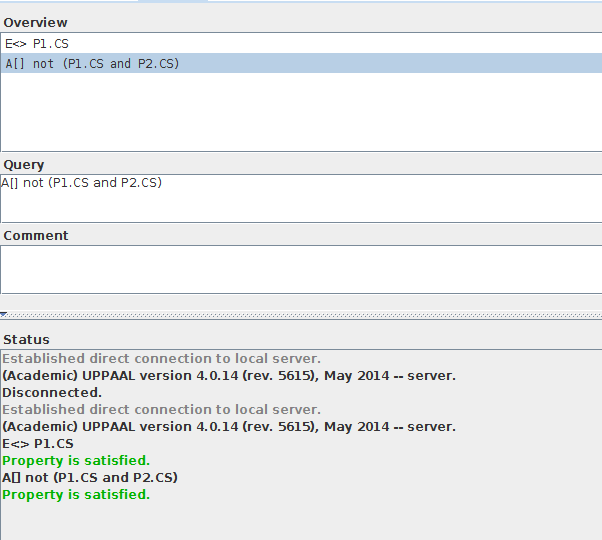
## Parameters of template are:

const int[1,2] me, int[0,1] &req\_self, int[0,1] &req\_other

Global variable turn indicates which process can enter the critical section.



## Properties checked:



## Simulation

