**BucharestIntersections:**

**PLACES:**

**DataCar:**

* P\_A\_CANTEMIRNORD
* P\_A\_CANTEMIRSUD
* P\_A\_MARASESTIEST
* P\_A\_MARASESTIEST\_BUS
* P\_A\_MARASESTIVEST
* P\_A\_MARASESTIVEST\_BUS
* P\_A1\_MARASESTIEST\_BUS
* P\_A1\_MARASESTIVEST\_BUS
* P\_B\_CANTEMIRNORD
* P\_B\_CANTEMIRSUD
* P\_B\_MARASESTIEST
* P\_B\_MARASESTIEST\_BUS
* P\_B\_MARASESTIVEST
* P\_B\_MARASESTIVEST\_BUS
* P\_OE\_CANTEMIRNORD
* P\_OE\_CANTEMIRSUD
* P\_OE\_MARASESTIEST
* P\_OE\_MARASESTIEST\_BUS
* P\_OE\_MARASESTIVEST
* P\_OE\_MARASESTIVEST\_BUS
* P\_A\_CANTEMIR
* P\_A\_LANARIEI
* P\_A\_PARKING
* P\_A\_POTERASI
* P\_A\_RADULESCU
* P\_A00\_CANTEMIR
* P\_A01\_CANTEMIR
* P\_A02\_CANTEMIR
* P\_A03\_CANTEMIR
* P\_A04\_CANTEMIR
* P\_A1\_LANARIEI
* P\_A1\_POTERASI
* P\_A1\_RADULESCU
* P\_A10\_CANTEMIR
* P\_A11\_CANTEMIR
* P\_A12\_CANTEMIR
* P\_A13\_CANTEMIR
* P\_OE\_CANTEMIR
* P\_OE\_PARKING
* P\_OE\_POTERASI
* P\_OE\_RADULESCU

**DataCarQueue:**

* P\_INTER\_1
* P\_O\_CANTEMIRNORD
* P\_O\_CANTEMIRSUD
* P\_O\_MARASESTIEST
* P\_O\_MARASESTIEST\_BUS
* P\_O\_MARASESTIVEST
* P\_O\_MARASESTIVEST\_BUS
* P\_STATION\_MARASESTIEST\_BUS
* P\_STATION\_MARASESTIVEST\_BUS
* P\_X\_CANTEMIRNORD
* P\_X\_CANTEMIRSUD
* P\_X\_MARASESTIEST
* P\_X\_MARASESTIVEST
* P\_X1\_MARASESTIEST\_BUS
* P\_X1\_MARASESTIVEST\_BUS
* P\_X2\_MARASESTIEST\_BUS
* P\_X2\_MARASESTIVEST\_BUS
* P\_X3\_MARASESTIEST\_BUS
* P\_X3\_MARASESTIVEST\_BUS
* P\_STATION0\_CANTEMIR
* P\_STATION1\_CANTEMIR
* P\_X00\_CANTEMIR
* P\_X01\_CANTEMIR
* P\_X02\_CANTEMIR
* P\_X03\_CANTEMIR
* P\_X04\_CANTEMIR
* P\_X05\_CANTEMIR
* P\_X06\_CANTEMIR
* P\_X10\_CANTEMIR
* P\_X11\_CANTEMIR
* P\_X12\_CANTEMIR
* P\_X13\_CANTEMIR
* P\_X14\_CANTEMIR
* P\_X15\_CANTEMIR
* P\_O\_CANTEMIR
* P\_O\_SERBANVODA
* P\_O\_SERBANVODA\_BUS
* P\_O\_SINCAI
* P\_O\_SINCAI\_BUS
* P\_O\_TINERETULUI
* P\_X\_CANTEMIR
* P\_X\_SERBANVODA
* P\_X\_SERBANVODA\_BUS
* P\_X\_SINCAI
* P\_X\_TINERETULUI
* P\_X1\_SINCAI\_BUS
* P\_X1\_SINCAINORD\_BUS
* P\_X2\_SINCAI\_BUS
* P\_X2\_SINCAINORD\_BUS
* P\_X3\_SINCAINORD\_BUS
* P\_INTER\_2

**TRANSITIONS:**

**1. T\_U\_… Transitions -**

**1. Arrival place is not empty**

**new Condition(T\_U\_..., "P\_A\_...", TransitionCondition.NotNull)**

**2. Downstream queue can accept more cars**

**new Condition(T\_U\_..., "P\_X\_...", TransitionCondition.CanAddCars)**

**These conditions are usually combined with a logical \*\*AND\*\* connector:**

**Condition ct1 = new Condition(..., "P\_A\_...", TransitionCondition.NotNull);**

**Condition ct2 = new Condition(..., "P\_X\_...", TransitionCondition.CanAddCars);**

**ct1.SetNextCondition(LogicConnector.AND, ct2);**

**Typical Activations:**

**AddElement from arrival place to the queue.**

**grd.Activations.add(new Activation(**

**T\_U\_...,**

**"P\_A\_...",**

**TransitionOperation.AddElement,**

**"P\_X\_..."**

**));**

T\_U00\_CANTEMIR

T\_U01\_CANTEMIR

T\_U02\_CANTEMIR

T\_U03\_CANTEMIR

T\_U04\_CANTEMIR

T\_U05\_CANTEMIR

T\_U10\_CANTEMIR

T\_U11\_CANTEMIR

T\_U12\_CANTEMIR

T\_U13\_CANTEMIR

T\_U14\_CANTEMIR

T\_U\_CANTEMIR

T\_U\_CANTEMIRNORD

T\_U\_CANTEMIRSUD

T\_U\_LANARIEI

T\_U\_MARASESTIEST

T\_U\_MARASESTIVEST

T\_U\_MARASESTIEST\_BUS

T\_U\_MARASESTIVEST\_BUS

T\_U\_POTERASI

T\_U\_RADULESCU

T\_U\_SERBANVODA

T\_U\_SINCAI

T\_U\_TINERETULUI

T\_U\_SINCAI\_BUS

T\_U\_SERBANVODA\_BUS

T\_U\_SINCAINORD\_BUS

T\_U1\_LANARIEI

T\_U1\_MARASESTIEST\_BUS

T\_U1\_MARASESTIVEST\_BUS

T\_U1\_POTERASI

T\_U1\_RADULESCU

T\_U1\_SINCAINORD\_BUS

T\_U2\_MARASESTIEST\_BUS

T\_U2\_MARASESTIVEST\_BUS

T\_U2\_SINCAINORD\_BUS

**2. T\_E\_… Transitions**

Typical Conditions

1. Traffic light is green

new Condition(T\_E\_..., "P\_...\_TL", TransitionCondition.Equal, "green")

2. Queue has at least one car (or a “priority car”)

new Condition(T\_E\_..., "P\_X\_...", TransitionCondition.HaveCar)

new Condition(T\_E\_..., "P\_X\_...", TransitionCondition.HavePriorityCar)

Condition ct1 = new Condition(..., "P\_...\_TL", TransitionCondition.Equal, "green");

Condition ct2 = new Condition(..., "P\_X\_...", TransitionCondition.HaveCar);

ct1.SetNextCondition(LogicConnector.AND, ct2);

Typical Activations

PopElementWithoutTarget from the queue and place it in the “B” place (the exit buffer).

grd.Activations.add(new Activation(

T\_E\_...,

"P\_X\_...",

TransitionOperation.PopElementWithoutTarget,

"P\_B\_..."

));

Move traffic light token (optional) so the light place remains consistent.

grd.Activations.add(new Activation(

T\_E\_...,

"P\_...\_TL",

TransitionOperation.Move,

"P\_...\_TL"

));

T\_E00\_CANTEMIR

T\_E01\_CANTEMIR

T\_E02\_CANTEMIR

T\_E03\_CANTEMIR

T\_E04\_CANTEMIR

T\_E10\_CANTEMIR

T\_E11\_CANTEMIR

T\_E12\_CANTEMIR

T\_E13\_CANTEMIR

T\_E\_CANTEMIR

T\_E\_CANTEMIRNORD

T\_E\_CANTEMIRSUD

T\_E\_MARASESTIEST

T\_E\_MARASESTIVEST

T\_E\_SERBANVODA

T\_E\_SINCAI

T\_E\_SINCAINORD\_BUS

T\_E\_SINCAI\_BUS

T\_E\_TINERETULUI

T\_E\_SERBANVODA\_BUS

**3. T\_O\_… Transitions**

Typical Conditions

1. Intersection (or intermediate) place has a car “for me”

new Condition(T\_O\_..., "P\_INTER\_...", TransitionCondition.HaveCarForMe)

2. Output place can add more cars

new Condition(T\_O\_..., "P\_O\_...", TransitionCondition.CanAddCars)

Condition ct1 = new Condition(..., "P\_INTER\_...", TransitionCondition.HaveCarForMe);

Condition ct2 = new Condition(..., "P\_O\_...", TransitionCondition.CanAddCars);

ct1.SetNextCondition(LogicConnector.AND, ct2);

Typical Activations

PopElementWithTargetToQueue from the intersection place to an output place.

grd.Activations.add(new Activation(

T\_O\_...,

"P\_INTER\_...",

TransitionOperation.PopElementWithTargetToQueue,

"P\_O\_..."

));

T\_O\_CANTEMIR

T\_O\_CANTEMIRNORD

T\_O\_CANTEMIRSUD

T\_O\_MARASESTIEST

T\_O\_MARASESTIVEST

T\_O\_SERBANVODA

T\_O\_SINCAI

T\_O\_SINCAINORD\_BUS

T\_O\_SINCAI\_BUS

T\_O\_TINERETULUI

T\_O\_SERBANVODA\_BUS

**4. T\_I\_… Transitions**

Typical Conditions

1. Buffer place (e.g. `P\_B\_...`) is not null

new Condition(T\_I\_..., "P\_B\_...", TransitionCondition.NotNull)

2. Intersection place (`P\_INTER\_...`) can add more cars

new Condition(T\_I\_..., "P\_INTER\_...", TransitionCondition.CanAddCars)

Condition ct1 = new Condition(..., "P\_B\_...", TransitionCondition.NotNull);

Condition ct2 = new Condition(..., "P\_INTER\_...", TransitionCondition.CanAddCars);

ct1.SetNextCondition(LogicConnector.AND, ct2);

Typical Activations

AddElement from `P\_B\_...` to `P\_INTER\_...`.

grd.Activations.add(new Activation(

T\_I\_...,

"P\_B\_...",

TransitionOperation.AddElement,

"P\_INTER\_..."

));

T\_I\_CANTEMIR

T\_I\_CANTEMIRNORD

T\_I\_CANTEMIRSUD

T\_I\_MARASESTIEST

T\_I\_MARASESTIVEST

T\_I\_MARASESTIEST\_BUS

T\_I\_MARASESTIVEST\_BUS

T\_I\_SERBANVODA

T\_I\_SERBANVODA\_BUS

T\_I\_SINCAI

T\_I\_SINCAINORD\_BUS

T\_I\_TINERETULUI

**5. T\_OUT\_… Transitions**

Typical Conditions

1. Arrival place is not null

new Condition(T\_OUT\_..., "P\_A\_...", TransitionCondition.NotNull)

2. Queue is unable to add more cars (i.e., “full” scenario)

new Condition(T\_OUT\_..., "P\_X\_...", TransitionCondition.CanNotAddCars)

Condition ct1 = new Condition(..., "P\_A\_...", TransitionCondition.NotNull);

Condition ct2 = new Condition(..., "P\_X\_...", TransitionCondition.CanNotAddCars);

ct1.SetNextCondition(LogicConnector.AND, ct2);

Typical Activations

SendOverNetwork a “full” signal, or some message, to an external output.

grd.Activations.add(new Activation(

T\_OUT\_...,

"full",

TransitionOperation.SendOverNetwork,

"OP\_..."

));

T\_OUT\_CANTEMIR

T\_OUT\_CANTEMIRNORD

T\_OUT\_CANTEMIRSUD

T\_OUT\_MARASESTIEST

T\_OUT\_MARASESTIVEST

T\_OUT\_MARASESTIEST\_BUS

T\_OUT\_MARASESTIVEST\_BUS

T\_OUT\_SERBANVODA

T\_OUT\_SINCAI

T\_OUT\_SINCAINORD\_BUS

T\_OUT\_TINERETULUI

T\_OUT\_SERBANVODA\_BUS

**T\_OE\_… Transitions**

**Typical Conditions**

**Place (P\_O\_...) has cars.**

**new Condition(T\_OE\_..., "P\_O\_...", TransitionCondition.HaveCar);**

**Typical Activations**

1. **Move a car from P\_O\_... to P\_OE\_....**

**grd.Activations.add(new Activation(**

**T\_OE\_...,**

**"P\_O\_...",**

**TransitionOperation.PopElementWithoutTarget,**

**"P\_OE\_..."**

**));**

T\_OE\_CANTEMIRSUD

T\_OE\_CANTEMIRNORD

T\_OE\_MARASESTIEST

T\_OE\_MARASESTIVEST

T\_OE\_CANTEMIR

T\_OE\_SERBANVODA

T\_OE\_SINCAI

T\_OE\_TINERETULUI

T\_OE\_MARASESTIEST\_BUS

T\_OE\_MARASESTIVEST\_BUS

T\_OE\_SERBANVODA\_BUS

T\_OE\_PARKING

T\_OE\_POTERASI

T\_OE\_RADULESCU

T\_OE1\_SINCAI\_BUS

**T\_S\_… Transitions**

**Typical Conditions**

Source place (P\_X\_...) has cars.

new Condition(T\_S\_..., "P\_X\_...", TransitionCondition.HaveCarForMe);

The cars belong to a specific category (e.g., bus).

new Condition(T\_S\_..., "P\_X\_...", TransitionCondition.HaveBus);

**Typical Activations**

Move a car from P\_X\_... to the station P\_STATION\_....

grd.Activations.add(new Activation(

T\_S\_...,

"P\_X\_...",

TransitionOperation.PopElementWithoutTargetToQueue,

"P\_STATION\_..."

));

T\_S0\_CANTEMIR

T\_S1\_CANTEMIR

T\_S\_MARASESTIEST\_BUS

T\_S\_MARASESTIVEST\_BUS

T\_S\_SINCAINORD\_BUS

T\_S\_SINCAI\_BUS

**T\_PED\_…**

**Typical Conditions**

Pedestrian place (P\_PED\_...) is not empty.

new Condition(T\_PED\_..., "P\_PED\_...", TransitionCondition.NotNull);

1. Traffic light (P\_TL\_...) is red.

new Condition(T\_PED\_..., "P\_TL\_...", TransitionCondition.Equal, "red");

**Typical Activations**

Allow a pedestrian to cross.

grd.Activations.add(new Activation(

T\_PED\_...,

"P\_PED\_...",

TransitionOperation.Move,

"P\_CROSSWALK\_..."

));

T\_PED\_0\_CANTEMIR

T\_PED\_1\_CANTEMIR

**T\_ES\_… Transitions**

**Typical Conditions**

**Source place (P\_STATION\_...) has cars.**

**new Condition(T\_ES\_..., "P\_STATION\_...", TransitionCondition.HaveCar);**

**User request (USERREQ\_TAXI\_...) is not null.**

**new Condition(T\_ES\_..., "USERREQ\_TAXI\_...", TransitionCondition.NotNull);**

**Typical Activations**

**Move a car from P\_STATION\_... to a queue (P\_X...).**

**grd.Activations.add(new Activation(**

**T\_ES\_...,**

**Arrays.asList("P\_STATION\_...", "USERREQ\_TAXI\_..."),**

**TransitionOperation.PopTaxiToQueue,**

**"P\_X..."**

**));**

T\_ES0\_CANTEMIR

T\_ES1\_CANTEMIR

T\_ES\_MARASESTIEST\_BUS

T\_ES\_MARASESTIVEST\_BUS

T\_ES\_SINCAINORD\_BUS

T\_ES\_SINCAI\_BUS

**T\_C\_… Transitions**

**Typical Conditions**

Source place (P\_X1\_...) has cars for the target.

new Condition(T\_C\_..., "P\_X1\_...", TransitionCondition.HaveCarForMe);

Target place (P\_X2\_...) can accept cars.

new Condition(T\_C\_..., "P\_X2\_...", TransitionCondition.CanAddCars);

**Typical Activations**

Transfer a car from P\_X1\_... to P\_X2\_....

grd.Activations.add(new Activation(

T\_C\_...,

"P\_X1\_...",

TransitionOperation.PopElementWithTargetToQueue,

"P\_X2\_..."

));

T\_C0\_CANTEMIR

T\_C1\_CANTEMIR

T\_C\_MARASESTIEST\_BUS

T\_C\_MARASESTIVEST\_BUS

T\_C\_SINCAINORD\_BUS

T\_C\_SINCAI\_BUS

**PedestrianController\_0:**

**PLACES:**

**DataString:**

* USREQ\_CANTEMIR\_1
* GR
* YR
* RG
* RY
* P\_INI
* WAIT

**DataTransfer:**

* P\_CANTEMIR\_TL\_1 (Value: TransferOperation("localhost", "1080", "P\_CANTEMIR\_TL\_11"))
* P\_CANTEMIR\_PTL\_1 (Value: TransferOperation("localhost", "1080", "P\_CANTEMIR\_PTL\_11"))

**TRANSITIONS:**

**T\_INI:**

* **Conditions:**
  + P\_INI is not null
* **Activations:**
  + Send "green" to P\_CANTEMIR\_TL\_1
  + Send "red" to P\_CANTEMIR\_PTL\_1
* **Delay:** 0

**T1:**

* **Conditions:**
  + USREQ\_CANTEMIR\_1 is not null
  + WAIT is not null (combined with logical AND)
* **Activations:**
  + Move "WAIT" to YR
  + Send "yellow" to P\_CANTEMIR\_TL\_1
* **Delay:** 0

**T2:**

* **Conditions:**
  + YR is not null
* **Activations:**
  + Move "YR" to RG
  + Send "red" to P\_CANTEMIR\_TL\_1
  + Send "green" to P\_CANTEMIR\_PTL\_1
* **Delay:** 5

**T3:**

* **Conditions:**
  + RG is not null
* **Activations:**
  + Move "RG" to RY
  + Send "yellow" to P\_CANTEMIR\_PTL\_1
* **Delay:** 5

**T4:**

* **Conditions:**
  + RY is not null
* **Activations:**
  + Move "RY" to rr
  + Send "green" to P\_CANTEMIR\_TL\_1
  + Send "red" to P\_CANTEMIR\_PTL\_1
* **Delay:** 2

**T5:**

* **Conditions:**
  + GR is not null
* **Activations:**
  + Move "GR" to WAIT
* **Delay:** 5

**PedestrianController\_1:**

**PLACES:**

**DataString:**

* USREQ\_CANTEMIR\_2
* GR
* YR
* RG
* RY
* P\_INI
* WAIT

**DataTransfer:**

* P\_CANTEMIR\_TL\_2 (Value: TransferOperation("localhost", "1080", "P\_CANTEMIR\_TL\_21"))
* P\_CANTEMIR\_PTL\_2 (Value: TransferOperation("localhost", "1080", "P\_CANTEMIR\_PTL\_21"))

**TRANSITIONS:**

**T\_INI:**

* **Conditions:**
  + P\_INI is not null
* **Activations:**
  + Send "green" to P\_CANTEMIR\_TL\_2
  + Send "red" to P\_CANTEMIR\_PTL\_2
* **Delay:** 0

**T1:**

* **Conditions:**
  + USREQ\_CANTEMIR\_2 is not null
  + WAIT is not null (combined with logical AND)
* **Activations:**
  + Move "WAIT" to YR
  + Send "yellow" to P\_CANTEMIR\_TL\_2
* **Delay:** 0

**T2:**

* **Conditions:**
  + YR is not null
* **Activations:**
  + Move "YR" to RG
  + Send "red" to P\_CANTEMIR\_TL\_2
  + Send "green" to P\_CANTEMIR\_PTL\_2
* **Delay:** 5

**T3:**

* **Conditions:**
  + RG is not null
* **Activations:**
  + Move "RG" to RY
  + Send "yellow" to P\_CANTEMIR\_PTL\_2
* **Delay:** 5

**T4:**

* **Conditions:**
  + RY is not null
* **Activations:**
  + Move "RY" to rr
  + Send "green" to P\_CANTEMIR\_TL\_2
  + Send "red" to P\_CANTEMIR\_PTL\_2
* **Delay:** 2

**T5:**

* **Conditions:**
  + GR is not null
* **Activations:**
  + Move "GR" to WAIT
* **Delay:** 5

**Controller\_Intersection 1 and 2:**

**Controller: INTERSECTION\_1\_CONTROLLER**

**General Information:**

* **Petri Net Name:** INTERSECTION\_1\_CONTROLLER
* **Network Port:** 1081

**Constants:**

1. **P\_INI**: "red"
2. **red**: "red"
3. **green**: "green"
4. **yellow**: "yellow"
5. **FIVE**: 5
6. **TEN**: 10

**Places:**

**State Places:**

1. R1R2R3R4: Initial state for signals.
2. G1R2R3R4: Green light for lane 1, others red.
3. Y1R2R3R4: Yellow light for lane 1, others red.
4. R1G2R3R4: Red for lane 1, green for lane 2.
5. R1Y2R3R4: Red for lane 1, yellow for lane 2.
6. R1R2G3R4: Red for lanes 1 and 2, green for lane 3.
7. R1R2Y3R4: Red for lanes 1 and 2, yellow for lane 3.
8. R1R2R3G4: Red for lanes 1-3, green for lane 4.
9. R1R2R3Y4: Red for lanes 1-3, yellow for lane 4.

**Input Places:**

1. IN\_CANTEMIRNORD
2. IN\_MARASESTIEST
3. IN\_CANTEMIRSUD
4. IN\_MARASESTIVEST
5. IN\_MARASESTIEST\_BUS
6. IN\_MARASESTIVEST\_BUS

**Final Places:**

1. P\_F\_CANTEMIRNORD
2. P\_F\_MARASESTIEST
3. P\_F\_CANTEMIRSUD
4. P\_F\_MARASESTIVEST
5. P\_F\_MARASESTIEST\_BUS
6. P\_F\_MARASESTIVEST\_BUS

**Output Transfers:**

1. **OP\_CANTEMIRNORD**: Sends to "P\_CANTEMIRNORD\_TL"
2. **OP\_MARASESTIEST**: Sends to "P\_MARASESTIEST\_TL"
3. **OP\_CANTEMIRSUD**: Sends to "P\_CANTEMIRSUD\_TL"
4. **OP\_MARASESTIVEST**: Sends to "P\_MARASESTIVEST\_TL"
5. **OP\_MARASESTIEST\_BUS**: Sends to "P\_MARASESTIEST\_BUS\_TL"
6. **OP\_MARASESTIVEST\_BUS**: Sends to "P\_MARASESTIVEST\_BUS\_TL"

**Transitions:**

**T\_INI:**

* **Condition:** P\_INI is not null.
* **Activations:**
  + Send P\_INI to all traffic light outputs (OP\_\*).
  + Set P\_INI to null.
* **Delay:** 0

**T1:**

* **Condition:** R1R2R3R4 is not null.
* **Activations:**
  + Move R1R2R3R4 → G1R2R3R4.
  + Send "green" to OP\_CANTEMIRNORD.
* **Delay:** 5

**T2:**

* **Condition:** G1R2R3R4 is not null.
* **Activations:**
  + Move G1R2R3R4 → Y1R2R3R4.
  + Send "yellow" to OP\_CANTEMIRNORD.
* **Delay:** 5

**T3:**

* **Condition:** Y1R2R3R4 is not null.
* **Activations:**
  + Move Y1R2R3R4 → R1G2R3R4.
  + Send "green" to OP\_MARASESTIEST and OP\_MARASESTIEST\_BUS.
  + Send "red" to OP\_CANTEMIRNORD.
* **Delay:** 5

**T4:**

* **Condition:** R1G2R3R4 is not null.
* **Activations:**
  + Move R1G2R3R4 → R1Y2R3R4.
  + Send "yellow" to OP\_MARASESTIEST and OP\_MARASESTIEST\_BUS.
* **Delay:** 5

**T5:**

* **Condition:** R1Y2R3R4 is not null.
* **Activations:**
  + Move R1Y2R3R4 → R1R2G3R4.
  + Send "green" to OP\_CANTEMIRSUD.
  + Send "red" to OP\_MARASESTIEST and OP\_MARASESTIEST\_BUS.
* **Delay:** 5

**T6:**

* **Condition:** R1R2G3R4 is not null.
* **Activations:**
  + Move R1R2G3R4 → R1R2Y3R4.
  + Send "yellow" to OP\_CANTEMIRSUD.
* **Delay:** 5

**T7:**

* **Condition:** R1R2Y3R4 is not null.
* **Activations:**
  + Move R1R2Y3R4 → R1R2R3G4.
  + Send "green" to OP\_MARASESTIVEST and OP\_MARASESTIVEST\_BUS.
  + Send "red" to OP\_CANTEMIRSUD.
* **Delay:** 5

**T8:**

* **Condition:** R1R2R3G4 is not null.
* **Activations:**
  + Move R1R2R3G4 → R1R2R3Y4.
  + Send "yellow" to OP\_MARASESTIVEST and OP\_MARASESTIVEST\_BUS.
* **Delay:** 5

**T9:**

* **Condition:** R1R2R3Y4 is not null.
* **Activations:**
  + Move R1R2R3Y4 → R1R2R3R4.
  + Send "red" to OP\_MARASESTIVEST and OP\_MARASESTIVEST\_BUS.
* **Delay:** 5

**T\_F\_CANTEMIRNORD:**

* **Condition 1:**
  + P\_F\_CANTEMIRNORD is not null AND IN\_CANTEMIRNORD is null.
  + **Activation:** Dynamic delay with value "FIVE".
* **Condition 2:**
  + P\_F\_CANTEMIRNORD is not null AND IN\_CANTEMIRNORD is not null.
  + **Activation:** Dynamic delay with value "TEN".
* **Delay:** 0

**T\_F\_MARASESTIEST:**

* **Condition 1:**
  + P\_F\_MARASESTIEST is not null AND IN\_MARASESTIEST is null.
  + **Activation:** Dynamic delay with value "FIVE".
* **Condition 2:**
  + P\_F\_MARASESTIEST is not null AND IN\_MARASESTIEST is not null.
  + **Activation:** Dynamic delay with value "TEN".
* **Delay:** 0

**T\_F\_MARASESTIEST\_BUS:**

* **Condition 1:**
  + P\_F\_MARASESTIEST\_BUS is not null AND IN\_MARASESTIEST\_BUS is null.
  + **Activation:** Dynamic delay with value "FIVE".
* **Condition 2:**
  + P\_F\_MARASESTIEST\_BUS is not null AND IN\_MARASESTIEST\_BUS is not null.
  + **Activation:** Dynamic delay with value "TEN".
* **Delay:** 0

**T\_F\_CANTEMIRSUD:**

* **Condition 1:**
  + P\_F\_CANTEMIRSUD is not null AND IN\_CANTEMIRSUD is null.
  + **Activation:** Dynamic delay with value "FIVE".
* **Condition 2:**
  + P\_F\_CANTEMIRSUD is not null AND IN\_CANTEMIRSUD is not null.
  + **Activation:** Dynamic delay with value "TEN".

**T\_F\_MARASESTIVEST:**

* **Condition 1:**
  + P\_F\_MARASESTIVEST is not null AND IN\_MARASESTIVEST is null.
  + **Activation:** Dynamic delay with value "FIVE".
* **Condition 2:**
  + P\_F\_MARASESTIVEST is not null AND IN\_MARASESTIVEST is not null.
  + **Activation:** Dynamic delay with value "TEN".
* **Delay:** 0

**T\_F\_MARASESTIVEST\_BUS:**

* **Condition 1:**
  + P\_F\_MARASESTIVEST\_BUS is not null AND IN\_MARASESTIVEST\_BUS is null.
  + **Activation:** Dynamic delay with value "FIVE".
* **Condition 2:**
  + P\_F\_MARASESTIVEST\_BUS is not null AND IN\_MARASESTIVEST\_BUS is not null.
  + **Activation:** Dynamic delay with value "TEN".
* **Delay:** 0

**Controller Start:**

* **Delay:** 2000 ms
* **Start:** Disabled in the provided code.
* **Window:** A PetriNetWindow is initialized and made visible to represent the Petri net.

### Controller: INTERSECTION\_2\_CONTROLLER

#### General Information:

* **Petri Net Name:** INTERSECTION\_2\_CONTROLLER
* **Network Port:** 1082

#### Constants:

1. **P\_INI**: "red"
2. **red**: "red"
3. **green**: "green"
4. **yellow**: "yellow"
5. **FIVE**: 5
6. **TEN**: 10

#### Places:

**State Places:**

1. R1R2R3R4: Initial state for signals.
2. G1R2R3R4: Green light for lane 1, others red.
3. Y1R2R3R4: Yellow light for lane 1, others red.
4. R1G2R3R4: Red for lane 1, green for lane 2.
5. R1Y2R3R4: Red for lane 1, yellow for lane 2.
6. R1R2G3R4: Red for lanes 1 and 2, green for lane 3.
7. R1R2Y3R4: Red for lanes 1 and 2, yellow for lane 3.
8. R1R2R3G4: Red for lanes 1-3, green for lane 4.
9. R1R2R3Y4: Red for lanes 1-3, yellow for lane 4.

**Input Places:**

1. IN\_CANTEMIR
2. IN\_SINCAINORD
3. IN\_TINERETULUI
4. IN\_SERBANVODA
5. IN\_SINCAINORD\_BUS
6. IN\_SERBANVODA\_BUS

**Final Places:**

1. P\_F\_CANTEMIR
2. P\_F\_SINCAINORD
3. P\_F\_TINERETULUI
4. P\_F\_SERBANVODA
5. P\_F\_SINCAINORD\_BUS
6. P\_F\_SERBANVODA\_BUS

**Output Transfers:**

1. **OP\_CANTEMIR**: Sends to "P\_CANTEMIR\_TL"
2. **OP\_TINERETULUI**: Sends to "P\_TINERETULUI\_TL"
3. **OP\_SERBANVODA**: Sends to "P\_SERBANVODA\_TL"
4. **OP\_SINCAINORD\_BUS**: Sends to "P\_SINCAINORD\_BUS\_TL"
5. **OP\_SINCAI**: Sends to "P\_SINCAI\_TL"
6. **OP\_SERBANVODA\_BUS**: Sends to "P\_SERBANVODA\_BUS\_TL"

### Transitions:

#### T\_INI:

* **Condition:** P\_INI is not null.
* **Activations:**
  + Send P\_INI to all traffic light outputs (OP\_\*).
  + Set P\_INI to null.
* **Delay:** 0

#### T1:

* **Condition:** R1R2R3R4 is not null.
* **Activations:**
  + Move R1R2R3R4 → G1R2R3R4.
  + Send "green" to OP\_CANTEMIR.
  + Move R1R2R3R4 → P\_F\_CANTEMIR.
* **Delay:** 5

#### T2:

* **Condition:** G1R2R3R4 is not null.
* **Activations:**
  + Move G1R2R3R4 → Y1R2R3R4.
  + Send "yellow" to OP\_CANTEMIR.
* **Delay:** 5

#### T3:

* **Condition:** Y1R2R3R4 is not null.
* **Activations:**
  + Move Y1R2R3R4 → R1G2R3R4.
  + Send "green" to OP\_SINCAI and OP\_SINCAINORD\_BUS.
  + Send "red" to OP\_CANTEMIR.
  + Move Y1R2R3R4 → P\_F\_SINCAINORD and P\_F\_SINCAINORD\_BUS.
* **Delay:** 5

#### T4:

* **Condition:** R1G2R3R4 is not null.
* **Activations:**
  + Move R1G2R3R4 → R1Y2R3R4.
  + Send "yellow" to OP\_SINCAI and OP\_SINCAINORD\_BUS.
* **Delay:** 5

#### T5:

* **Condition:** R1Y2R3R4 is not null.
* **Activations:**
  + Move R1Y2R3R4 → R1R2G3R4.
  + Send "green" to OP\_TINERETULUI.
  + Send "red" to OP\_SINCAI and OP\_SINCAINORD\_BUS.
  + Move R1Y2R3R4 → P\_F\_TINERETULUI.
* **Delay:** 5

#### T6:

* **Condition:** R1R2G3R4 is not null.
* **Activations:**
  + Move R1R2G3R4 → R1R2Y3R4.
  + Send "yellow" to OP\_TINERETULUI.
* **Delay:** 5

#### T7:

* **Condition:** R1R2Y3R4 is not null.
* **Activations:**
  + Move R1R2Y3R4 → R1R2R3G4.
  + Send "green" to OP\_SERBANVODA and OP\_SERBANVODA\_BUS.
  + Send "red" to OP\_TINERETULUI.
  + Move R1R2Y3R4 → P\_F\_SERBANVODA and P\_F\_SERBANVODA\_BUS.
* **Delay:** 5

#### T8:

* **Condition:** R1R2R3G4 is not null.
* **Activations:**
  + Move R1R2R3G4 → R1R2R3Y4.
  + Send "yellow" to OP\_SERBANVODA and OP\_SERBANVODA\_BUS.
* **Delay:** 5

#### T9:

* **Condition:** R1R2R3Y4 is not null.
* **Activations:**
  + Move R1R2R3Y4 → R1R2R3R4.
  + Send "red" to OP\_SERBANVODA\_BUS.
* **Delay:** 5

#### T\_F\_CANTEMIR:

* **Condition 1:**
  + P\_F\_CANTEMIR is not null AND IN\_CANTEMIR is null.
  + **Activation:** Dynamic delay with value "FIVE".
* **Condition 2:**
  + P\_F\_CANTEMIR is not null AND IN\_CANTEMIR is not null.
  + **Activation:** Dynamic delay with value "TEN".
* **Delay:** 0

#### T\_F\_SINCAINORD:

* **Condition 1:**
  + P\_F\_SINCAINORD is not null AND IN\_SINCAINORD is null.
  + **Activation:** Dynamic delay with value "FIVE".
* **Condition 2:**
  + P\_F\_SINCAINORD is not null AND IN\_SINCAINORD is not null.
  + **Activation:** Dynamic delay with value "TEN".
* **Delay:** 0

#### T\_F\_SINCAINORD\_BUS:

* **Condition 1:**
  + P\_F\_SINCAINORD\_BUS is not null AND IN\_SINCAINORD\_BUS is null.
  + **Activation:** Dynamic delay with value "FIVE".
* **Condition 2:**
  + P\_F\_SINCAINORD\_BUS is not null AND IN\_SINCAINORD\_BUS is not null.
  + **Activation:** Dynamic delay with value "TEN".
* **Delay:** 0

#### T\_F\_TINERETULUI:

* **Condition 1:**
  + P\_F\_TINERETULUI is not null AND IN\_TINERETULUI is null.
  + **Activation:** Dynamic delay with value "FIVE".
* **Condition 2:**
  + P\_F\_TINERETULUI is not null AND IN\_TINERETULUI is not null.
  + **Activation:** Dynamic delay with value "TEN".
* **Delay:** 0

#### T\_F\_SERBANVODA:

* **Condition 1:**
  + P\_F\_SERBANVODA is not null AND IN\_SERBANVODA is null.
  + **Activation:** Dynamic delay with value "FIVE".
* **Condition 2:**
  + P\_F\_SERBANVODA is not null AND IN\_SERBANVODA is not null.
  + **Activation:** Dynamic delay with value "TEN".
* **Delay:** 0

#### T\_F\_SERBANVODA\_BUS:

* **Condition 1:**
  + P\_F\_SERBANVODA\_BUS is not null AND IN\_SERBANVODA\_BUS is null.
  + **Activation:** Dynamic delay with value "FIVE".
* **Condition 2:**
  + P\_F\_SERBANVODA\_BUS is not null AND IN\_SERBANVODA\_BUS is not null.
  + **Activation:** Dynamic delay with value "TEN".
* **Delay:** 0

### Controller Start:

* **Delay:** 2000 ms
* **Start:** Disabled in the provided code.
* **Window:** A PetriNetWindow is initialized and made visible to represent the Petri net.