Program Listings

Bold scripts are fundamental to the softwares functionality. The ordering is alphabetical.

Model

- **Intersection**: The class defining intersection objects, these are created whenever the user connects road segments. It is responsible for updating the traffic lights contained within it.
- LaneNode: Lanes are defined between two LaneNode objects.
- **LaneSegment**: Represents the data for the path vehicles will move down.
- **Line**: Defines a simple line between 2 points.
- RoadNode: Roads are built between 2 road nodes (points).
- **RoadSegment**: A collection of LaneSegments makes up a road. This script does a lot of calculations for the positions of lanes, lane divider lines and road corners.

Road Connecting

- ConnectButtonCustodian: Toggles connecting state.
- **ConnectLanes**: Responsible for the connecting lanes view. Shows and hides lane end markers accordingly, draws bezier curves between them, and connects lanes in the internal model.
- **ConnectRoadSegments**: Creates intersections upon user selection of appropriate markers. Draws lines between selected road segments to connect them.
- ItscMarkerManager: Intersection marker manager, manages the white rings used to select an intersection for editing.
- LaneMarkerManager: Similar to above, but for lane end markers. An animation is shown when the mouse is hovered.
- RoadEndMarkerManager: Controls the markers displayed when connecting roads, these toggle colour when clicked.
- **SelectIntersectionTool**: This manages the highest level view in the application, where the user can select intersections in the entire network to edit.

Traffic Lights

- EditLightsButtonManager: Implements on click actions of traffic lights enabled toggle and edit config button.
- **EditTrafficSchemeView**: Central script for editing traffic lights mode. Creates traffic light panels and enables the configuration panel.
- **LightDisplayPanel**: Creates a panel for storing traffic lights, which can be dragged.
- LightPhaseToggle: Small script which manages the state of ‘Toggle’ UI elements when configuring light phases.
- **TrafficLight**: Manages which traffic light asset is displayed inside of traffic light display panels. Also hosts the LightState enum, which is used all over the application.
- **TrafficPhaseRow**: Represents traffic phase configuration, controls toggle states and phase duration.
- **TrafficSchemePanel**: Main script for configuring traffic lights. Manages the configuration panel. Reads in and sets configurations into internal models using the ui components. Creates TrafficPhaseRows.

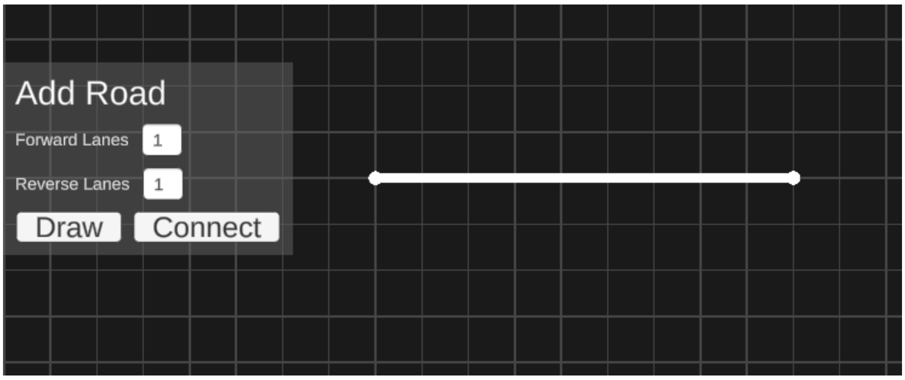
Main Folder

- BackgroundGridBuilder: The first script that executes in the program, draws the background canvas from scratch.
- **BezierCurveDrawer**: Does all calculations for Bezier curves, given start and end points.
- CameraManager: Moves the camera based on inputs.
- **Car**: Represents the internal model of the cars. Controls all vehicle movement behaviours, such as calculating stopping distance. Stores car properties like speed.
- **LineDrawer**: Script for drawing lines. Attached to line prefabs.
- **NewRoadDrawManager**: After a road selection has been made, this creates the visual representation of that road, and uses the internal model to find points to draw lines between.
- NewRoadPanelManager: Validation for number of lanes in new road panel.
- **NewRoadSelectionTool**: Main script for the new road drawing tool. Creates lines and dots to help users select points to make a new road between.
- **RoadNetworkManager**: One of the most central scripts. Stores the road network and vehicles on it. Orchestrates traffic lights and vehicle updates. Handles simulation statistics.
- StatusBarManager: Changes text displayed in status bar.
- **UIFlowManager**: Controls modes of application. Scripts call methods from this to switch to different tools and modes.

User Guide

The following procedure could be used to set up a functioning simulation environment. At any point the **arrow keys** are used to pan around and the **scroll wheel** or Q / E to zoom.
This guide is also available as a **video** demonstration: https://youtu.be/FFVYhXtyirI

1. Open Traffic Simulation.exe.
2. Click ‘**Draw**’ on the left of the screen.
3. Click on the grid to select two points horizontally.

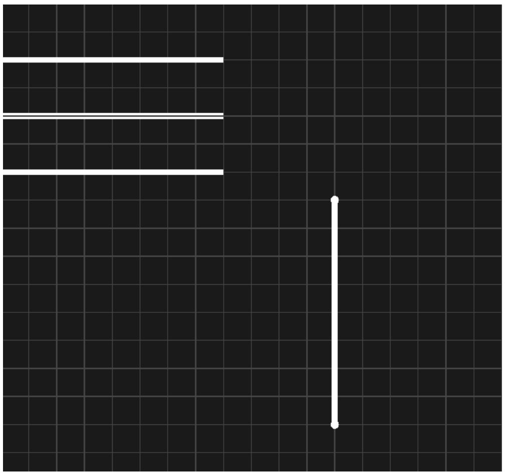


Add Road

Forward Lanes 1

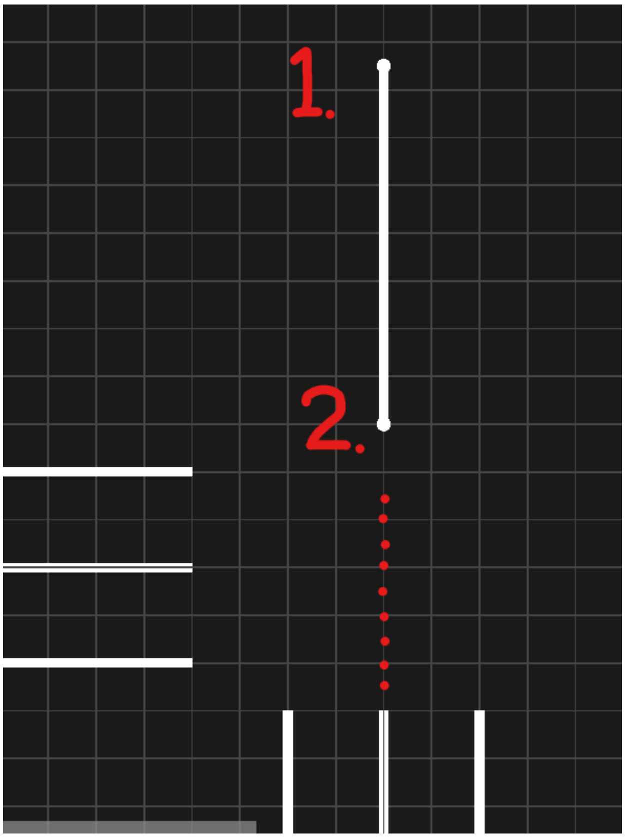
Reverse Lanes 1

4. Click on the **draw** button again.
5. Draw another line to the right of the existing road, starting just below and to the right, going downwards. (Making sure to leave enough room for the width of the road)

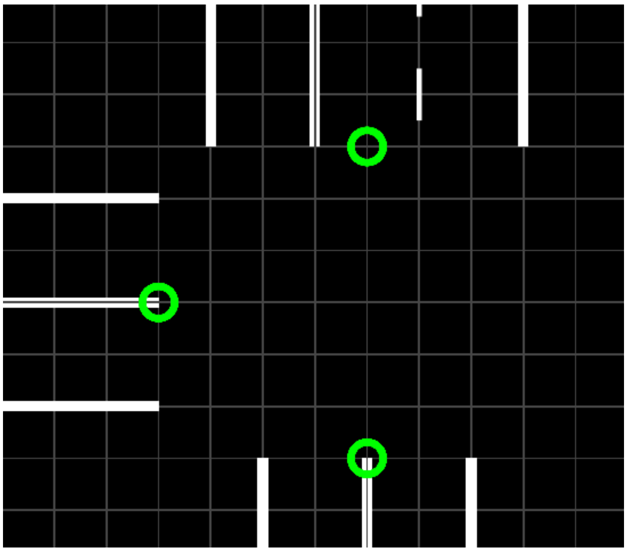


6. Change the ‘**Forward Lanes**’ value in the left panel to 2.
7. Click the **draw** button.

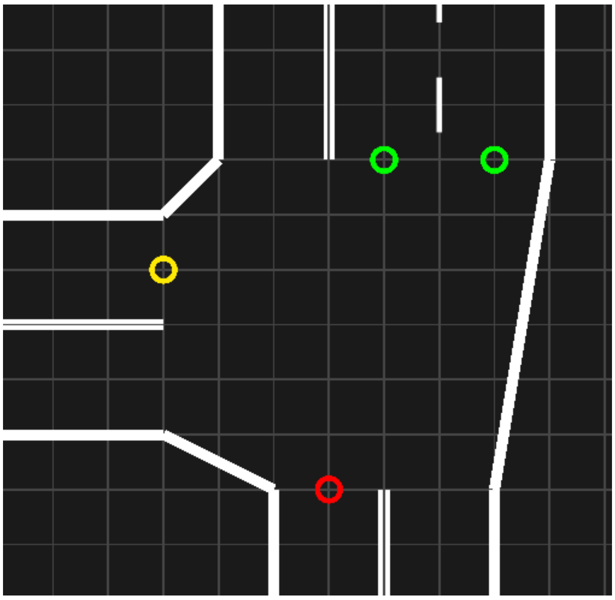
8. Draw another road from above, starting **far** and finishing close to the intersection.



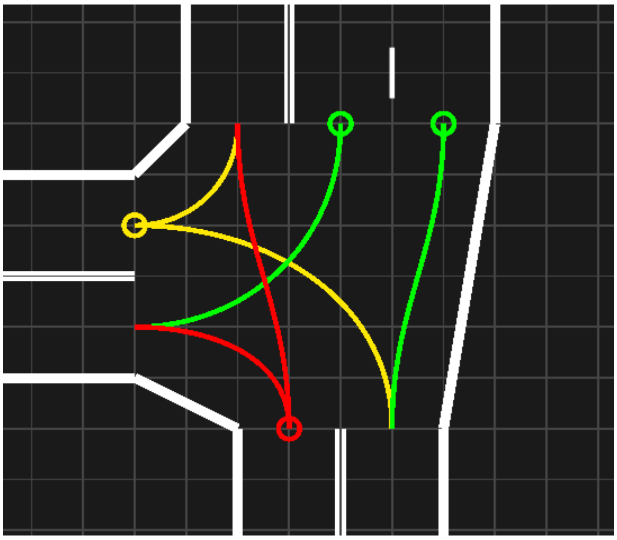
9. Click on the ‘**Connect**’ button in the left panel.
10. Click on each of the **three nodes** making up the intersection, so they are green.



11. Click ‘**Make Connection**’ on the left (this may take a second), the intersection should look like this:

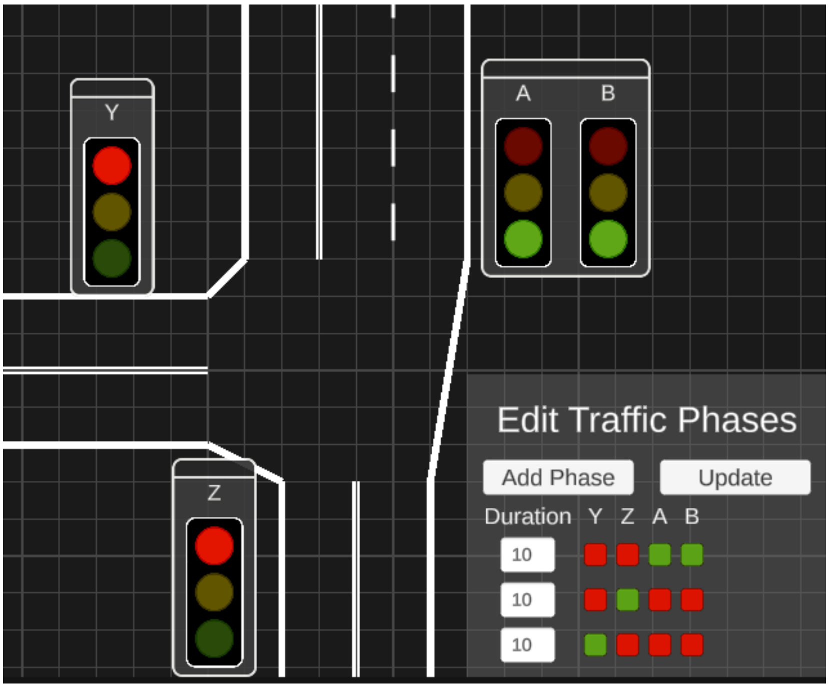


12. Click on the intersection entry markers, and connect each to at least one exit marker. **Right click** once to complete a selection.



13. Check the ‘**Traffic Lights Enabled**’ box, then click ‘**Edit Scheme**’.

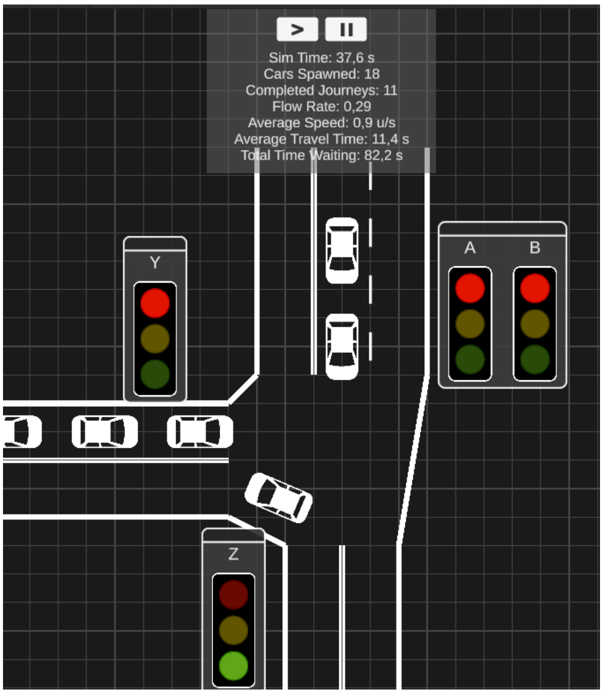
14. Click ‘**Add Phase**’ **twice** so there are 3 light phases. Then click on the **red boxes** so that one incoming road is green for each phase. In the example below, lights A and B are on the same road, so they can be green at the same time, in the first phase here.



Edit Traffic Phases

Duration	Y	Z	A	B
10	🟥	🟥	🟩	🟩
10	🟥	🟩	🟥	🟥
10	🟩	🟥	🟥	🟥

15. Click ‘**Update**’ in the bottom right. (Traffic lights can be moved by dragging)
16. Click the **play button** at the top of the screen to start the simulation.



Sim Time: 37,6 s
Cars Spawned: 18
Completed Journeys: 11
Flow Rate: 0,29
Average Speed: 0,9 u/s
Average Travel Time: 11,4 s
Total Time Waiting: 82,2 s

17. (**Optional**) Click pause at the top of the screen to stop the simulation, and see the results.