TransitFlow: Traffic Light Controller

Software Architecture Design (SAD)

April 11, 2024

SAD Version Final

Team 6

Authors: Sean Davies, Chris Jarek, Michael Pacheco, Ricardo Ponce, Ethan Puzak, Nathan Rowe

In Dedication to Froggy Froggerton

Table of Contents

1 Introduction

This document presents the Software Architecture Design (SAD) for the TransitFlow system. TransitFlow is a sophisticated traffic control system designed to integrate advanced modern technologies to ensure the safety and efficiency of traffic in urban and rural environments.

The SAD provides an in-depth analysis of the design and implementation of TransitFlow and its software, focusing on the software architecture. It's intended to serve as a reference to ensure that the system is robust, secure, and compliant with all relevant standards and regulations as outlined in the Manual on Uniform Traffic Control Devices (MUTCD), Chapter 4D. Traffic Control Signal Features.

2 Design Overview

This section provides an overview of the TransitFlow software architecture. The design approach, design decisions, and main subsystems that underlie TransitFlow will be described in detail.

2.1 Architecture Design Diagram

The architecture design diagram displays a high level overview of how each component of the virtual software will communicate with each other. The virtual TransitFlow system will function as a demonstration of the real practical version of the software.

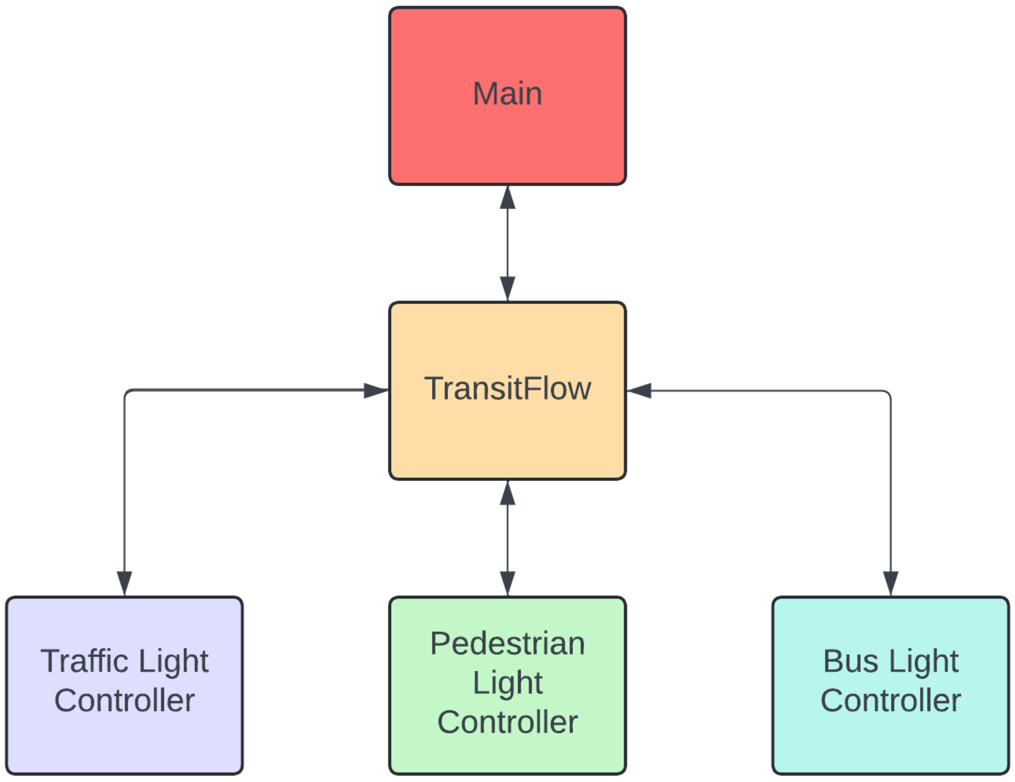

Figure 1: Architecture Design Diagram

2.1.1 Main

Main handles the initialization of the Graphical User interface and will interact with the TransitFlow component to handle updates to the scene. Main will handle all interactions with the user and will be the primary place of communication with TransitFlow.

2.1.2 TransitFlow

Will be the integration of TransitFlow's controls for various typical scenarios that will be handled during the life of the software. It will handle various inputs from its input controllers and will relay the outputs to the output controllers.

2.1.3 Traffic Light Controller

Handles communication with the traffic lights depending on what the TransitFlow component is requesting. The Traffic Light Controller component will make the appropriate light changes to all traffic lights. The Traffic Light Controller will only handle inputs and outputs to its corresponding systems.

2.1.4 Pedestrian Light Controller

Handles communication with pedestrian lights depending on what the TransitFlow component is requesting. The Pedestrian Light Controller component will make the appropriate light changes to pedestrian lights. The Pedestrian Light Controller component will only handle inputs and outputs to its corresponding systems.

2.1.5 Bus Light Controller

Handles communication with bus lights depending on what the TransitFlow component is requesting. The Bus Light Controller component will make the appropriate light changes to bus lights. The Bus Light Controller component will only handle inputs and outputs to its corresponding systems.

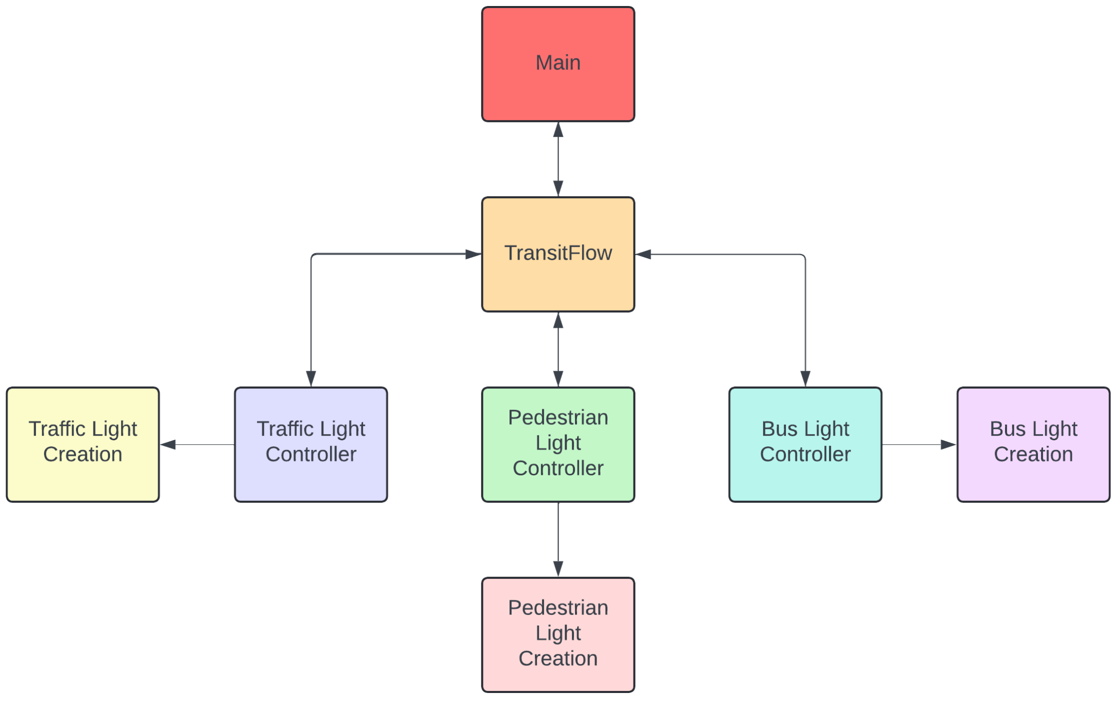

Figure 2: Design Overview Diagram

2.2 Design Approach

The architectural design pattern for TransitFlow uses states and logic to control the entire system. Dependent on the input devices and the sensors, TransitFlow knows its current state and what to expect regarding pedestrian and vehicle traffic. Signals (i.e. pedestrian crossing button, IR Sensor) are passed through the Traffic Light Controller component and the Pedestrian Light Controller component, which will allow the Traffic Light Controller component to display the necessary configuration of lights in the proper order to conduct traffic. Once a scenario is interpreted, the Traffic Light Controller's output component will provide feedback to the public, allowing interpretation of the implied instructions.

3 Component Specification

The following section details the components of the TransitFlow system. Each subsection defines the purpose of the required component, describing the functionalities relevant to TransitFlow.

3.1 TransitFlow

The TransitFlow component is the main component of the entire traffic system. The TransitFlow component consistently listens for inputs from traffic light signals, Pedestrian Lights, and Bus Lights. Upon receiving input, the TransitFlow component processes received data based upon various functions and algorithms, returning data to the Traffic Light Controller's subsystems dependent on the processed data.

3.1.1 Control System Receiver

The Control System Receiver (CSR) is a module connected to the TransitFlow system Controller designed to receive traffic data from Traffic Light Controller's subsystems over an Ethernet network for reliable communication. The receiver listens for input data over the network. Received data is immediately sent to the Data Conversion Unit for validation and processing.

3.1.2 Data Conversion Unit

The Data Conversion Unit (DCU) handles data packets received from the TransitFlow component's controller. Upon receiving a packet, the DCU creates a forked process to parse the individual packet. The forked process then handles the data packet in the following steps:

1. **Validation:** Received data packets from Traffic Light component subsystems have a specified header format to verify the data received. The process terminates if the packet's header does not match the checked format. Otherwise, the process continues with extraction.

2. **Extraction: Traffic Lights:** The contents of a data packet sent from a Traffic Light component subsystem at an intersection contain traffic data collected over the period since the previous packet had been sent in addition to the identifier for the intersection. The traffic data primarily consists of the number of vehicles that have passed through the intersection. The DCU extracts these contents to prepare for processing. If the extracted data is corrupt or missing after extraction, the process terminates with an error code. Otherwise, the extracted data is prepared for processing.

3. **Processing:** Extracted data is used to update the timed states of traffic lights at the designated intersection. These time updates are calculated through a specific set of parameters and functions, defined in 3.1.2 **State-Time Calculations**. Upon successful processing, the new state times for lights at the intersection will be prepared for packaging to be returned to the Traffic Light subsystem that sent the original packet.

   Pedestrian Light: Pedestrian light buttons will send data packets once pressed. The data packet will contain the intersection ID and the associated pedestrian light ID. The DCU will prepare a new data packet that instructs the Traffic Light Controller component to activate its attached pedestrian light at the next green interval. See 3.1.2 **State-Time Calculations** for pedestrian light's effect on timings.

   Bus Light: Bus sensors will send a data packet containing the bus light ID and bus intersection ID. The DCU will prepare a new data packet instructing the lights at the bus intersection to perform a pre-timed cycle.

4. **Packaging:** The DCU packages processed data into packets to be sent to the corresponding Traffic Light Controller component. Afterward, these packages are sent to the data transfer system. The forked process is terminated after successful execution.

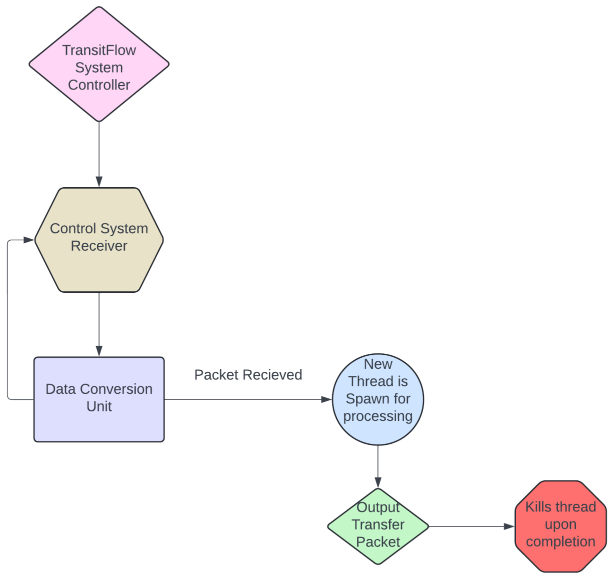
TransitFlow System Controller
Control System Receiver
Data Conversion Unit
Packet Recieved
New Thread is Spawn for processing
Output Transfer Packet
Kills thread upon completion

Figure 3: State diagram for the DCU. This diagram listens for information from the receiver then spawns a new thread to handle said received data. The spawned thread is killed after processing.

State-Time Calculations Within each 120-second cycle, the green state is adjusted depending on the current traffic density. Two timings for the green state will be calculated. These timings define the bounds for the actual green traffic light times. This information is then sent to the Traffic Light Controller's subsystems, which adjust actual timings per cycle at an intersection (see 3.3.2).

- Minimum Green: Will always be 15 seconds regardless of traffic data
- Maximum Green: A function of traffic density received from a Traffic Light subsystem.

$$G_{max} = G_{min} + (\frac{TD}{3.75}) \leq 80$$

where
G_{max} is the maximum green time in seconds.
G_{min} is the minimum green time in seconds.
TD is the number of vehicles detected in the received data packet.
80 is the hardbound on maximum green time.

3.1.3 Data Transfer System

The Data Transfer System (DTS) sends data packets created by the DCU to the corresponding Traffic Light Controller's subsystem. The ID for the corresponding subsystem is within the data packet to be readily identifiable by the ID map maintained by the DTS. The packet is then transmitted to the subsystem receiver matching the ID, defined in 3.2.1.

3.2 Input Devices

This section will detail the inputs and the data that generate which will be used by the TransitFlow Controller's subsystem. The TransitFlow component receives data from these input devices and processes it according to various functions and algorithms. The TransitFlow component then outputs and communicates to each intersection the signal timings for the traffic, bus, and pedestrian lights.

3.2.1 Subsystem Receiver

A Subsystem Receiver, connected to a Traffic Light Controller's subsystem, is responsible for receiving validated data packets from the DCU over the Ethernet network. Successfully received data packets are extracted, containing instructions for updated timed states at the intersection.

3.2.2 Vehicle Sensors

Each traffic light system within the TransitFlow system is equipped with sensors to collect data on current traffic density. After collecting this data, it will process it through functions to calculate how long the next green cycle timed state should be for the Traffic Light Controller. This number is then output to the proper traffic lights.

3.2.3 Bus Detection IR System

For Bus Lanes, buses will be equipped with an IR system on top of the roof, and Bus Lanes will have an IR Receiver specifically made to communicate with the IR system of each bus. When buses are a certain distance away, the IR Receiver will receive information from the bus's IR system stating that it needs to start a Yellow Light state for the perpendicular traffic. The perpendicular traffic lights will then turn Yellow then Red and the Bus Lights will turn Green allowing the bus to not have to stop at any time other than drop off/pick up areas.

3.2.4 Emergency Vehicle IR Receiver

An Infrared Receiver is attached to a traffic light for detection of emergency vehicles. When an Emergency Vehicle has its sirens on, the Emergency Vehicle IR System sends out a signal in front of it. When the Emergency Vehicle IR Receiver on top of a traffic light receives this input, it will send a signal to the TransitFlow Controller letting it know to activate a Yellow State on perpendicular lanes, allowing the parallel lanes to the emergency vehicle to turn Green.

3.2.5 Pedestrian Crossing Button

The pedestrian crossing button will schedule the pedestrian light to turn on. When the pedestrian crossing button is pressed it will send data to the TransitFlow Controller letting it know that a pedestrian is waiting. When the lanes parallel to that intersection's crosswalk turn green from a red state the pedestrian light will turn white with a countdown timer allowing them to cross.

3.2.6 Car Detector Camera

The Car Counter is a Camera that can detect vehicles and will count the total vehicles it sees crossing the intersection during a green light state. Along with counting, it will be used to detect a car at a red light state letting the TransitFlow Controller know there is a car waiting for a green light at that intersection.

3.3 Output Devices

The following subsections cover more detail about the components in which the TransitFlow Controller will be communicating with for output instructions. The following output devices are thus attached to the Traffic Light Controller's subsystems only, with all TransitFlow System output devices defined in section 3.1.

3.3.1 Subsystem Data Transfer System

The Data Transfer System attached to subsystems behaves similarly to the data transfer system defined in section 3.1. At a timed interval of 1 hour, a data packet containing traffic data will be built and sent to the TransitFlow Controller through the Ethernet network. For pedestrian light or bus light activation, the data packet transmitter will send the corresponding activation data packet to the TransitFlow Controller. To avoid potential pack conflicts within the data emitter, a queue structure will be implemented to send packets individually in the order they are received.

3.3.2 Traffic Light

A single traffic light changes states throughout one cycle, where a cycle consists of each state from green to red. Cycles are pre-timed, always existing in 120-second intervals. Within the fully actuated system, the output states will be split within a cycle interval and will change depending on traffic density for an intersection during the day. During night hours, a single cycle is only activated after receiving data from a sensor described in 3.2. The output states occur in the following order within a cycle.

Output States

1. **TURN ARROWS:** Turn arrows will be the first signal to occur at the beginning of each cycle. Through instructions provided by the associated Traffic Light Controller subsystem, a right-turn arrow will occur synchronously with the left-turn arrow of the left-adjacent traffic light. Afterwards, the same process will occur with the parallel traffic light within the intersection.
2. **GREEN:** The green state will be controlled by a maximum green time and minimum green time created by the TransitFlow Controller. The GREEN state will start at the minimum green time, which will then dynamically increase by 3 seconds for each passing vehicle detected by input sensors, up to the maximum green.
3. **YELLOW:** The yellow signal will always have a duration of six seconds regardless of traffic density.
4. **RED:** Once the red state is entered, the Traffic Light subsystem shall communicate with the perpendicular set of lights at an intersection to enter a new cycle.

   The following cycle will then repeat the procedure, starting at minimum green and adjusting accordingly.

System Clock The System clock attached to a Traffic Light Controller subsystem will be responsible for tracking two values: the current time of states within a cycle and the real local time. The local time shall be used to correlate traffic density at specific times of day in addition to determining when to implement night-time scheduling changes.

3.3.3 Pedestrian Light

The pedestrian light displays walk, walk-interval, and stop signals to users. The light remains in the stop state unless scheduled by the associated Traffic Light subsystem. Timings for walk intervals are fixed to 30-second intervals. During this, the minimum green interval of the scheduled cycle shall temporarily match the walk interval.

Proximity Speaker Pedestrian signals shall also be equipped with proximity-activated speakers to aid the visually impaired. These speakers will have the following voice instructions:

- Proximity activation will instruct pedestrians to press the pedestrian crossing button and will instruct a user to wait every few seconds after the button press until the scheduled cycle is reached.
- The speaker will instruct pedestrians to cross when right-of-way is determined and count down the remaining walk interval.
- After completion of the walk interval, the speaker will instruct pedestrians to no longer cross.

3.3.4 Bus Light

Bus Lights display right-of-way information to public vehicles within designated bus lanes. A green bus signal instructs a bus that it is safe to proceed while a red bus signal informs an approaching bus that they do not yet have the right-of-way.

4 Use Cases

This section details the different use cases that will be encountered during operation of the TransitFlow system.

4.0.1 Turning - Pressure Plate

Actor: Motor Vehicle
Goal: To allow a car to make a right turn.
Preconditions: The opposite left turn and the straight lane is set to red to prevent collisions.
Trigger: When a vehicle of a certain weight activates the pressure plate.

Scenario:

1. A vehicle enters the lane-direction split while coming to a stop before the painted and designated pedestrian crossing at the intersection where the pressure plate has been set.
   boolean plateActive()
2. The TransitFlow Controller updates the lights at that intersection to allow for the current car to proceed with a green light state and preventing perpendicular traffic.
   void UpdateLights()
3. A countdown timer starts the transition from the current proceed state of that lane to the slow down state.
   int countdownTime()
   void countdown(countdownTime)
4. A countdown timer starts the transition from the current slow down state to the stop state.
   int countdownTime()
   void countdown(countdownTime)
   void UpdateLights()

Exceptions:
If a pedestrian or emergency vehicle situation is active, the light will remain red until the situation is cleared for continuance of normal state.

void pedestrianActive() interrupts the normal traffic light cycles for the duration of an intermediary step. **void emergencyActive()** interrupts the normal traffic light cycles for the duration of

an intermediary step.

4.0.2 Car Counter - Camera

Actor: Motor Vehicle
Goal: To analyze the flow of traffic at a certain intersection or bus lane.
Preconditions: There are no preconditions as this a perpetual system to gather and analyze data.
Trigger: Presence of a vehicle detected by the camera at a intersection.

Scenario:

1. A vehicle comes to any one of several lanes at an intersection or bus lane.
   void cameraDetectVehicle()
2. Increment cardinal direction lane count.
   int intersectionCarCount()
   void increment(int intersectionCarCount)

3. The Traffic Light Controller sends the data packet to the TransitFlow Controller module to be analyzed.
   Direction intersectionDirectionWeight(int nortLaneCount, int southLaneCount, int eastLaneCount, int westLaneCount)

4. Analyze data packet at intersection and update the Traffic Light Controller's subsystems.
   void updateLightDefault(int countdownTime)

Exceptions:
Only in the event of systems malfunctions will the software be unavailable or altered from normal operating procedure.

bool statusPower() will check the current power input to ensure that the power is normal.
bool statusCamera() will check whether there is a camera equipment failure.

4.0.3 Emergency Vehicles - IR Sensor

Actor: Emergency Motor Vehicles
Goal: To allow the safe and quick passage of emergency vehicles on roads.
Preconditions: Emergency vehicles contain an IR Emitter Device.
Trigger: The IR Sensor at the traffic light is activated and within range.

Scenario:

1. Emergency vehicles are approaching the detection distance of the IR Sensor.
   boolean IRActive
   void checkIR()

2. Emergency vehicles is detected.
   void checkIR()

3. The Traffic Light Controller interrupts the current lights states to the slow down state.
   boolean IRActive
   void checkIR()
   void EmergencyInterrupt()

4. The software waits until the signal is out of range to resume normal procedure, continuing the previous proceed state.
   boolean IRActive
   void checkIR()
   void EmergencyInterrupt()

Exceptions:
Because this is an emergency situation created for the sole purpose of clearing traffic for the emergency vehicle there are no exceptions aside from systems malfunctions.

bool statusPower() will check the current power input to ensure that the power is normal.
bool statusIR() will check whether there is an IR Sensor equipment failure.

4.0.4 Bus Lanes

Actor: Public Buses
Goal: To expedite efficient publicly funded transit of buses.
Preconditions: Dedicated Bus Light Controller for bus travel that runs parallel to existing motor vehicle dedicated roads.
Trigger: Bus approaches a bus traffic light.

Scenario:

1. A public transit bus in its own dedicated lane and approaches the bus traffic light.
2. Once the bus reaches a certain distance from the bus traffic light a IR beam will be received.
3. Triggering the Bus Light Controller to display instructional light states to allow the bus through the intersection at the appropriate cycle.
4. Once the bus passes through the intersection the Bus Light Controller will wait until another bus is within range.
   void BusComingThrough()

Exceptions:
There is an emergency vehicle that is approaching the intersection where a bus is also approaching. The bus light will signal green at the start of the next cycle.

4.0.5 Pedestrian Crossings

Actor: Pedestrians
Goal: To facilitate safe crossings at intersections for pedestrians.
Preconditions: A pedestrian is waiting at an intersection.
Trigger: A pedestrian pushes the pedestrian crossing button to cross the intersection.

Scenario:

1. A pedestrian approaches a traffic intersection and signals their intention to cross by pushing the pedestrian crossing button.
   boolean IsPushed()
2. The pushed button alerts the Pedestrian Light Controller that a pedestrian needs to cross.
3. The TransitFlow Controller adjusts the timing of that intersection's traffic lights through the Traffic Light Controller and allows the pedestrian light to show the walk signal.
   void pedestrianCrossing()

Exceptions:
There is an emergency vehicle that is approaching the intersection where a pedestrian is crossing. TransitFlow will schedule the crossing signal at the start of the next light cycle.

4.0.6 Traffic Flow Cycles

Actor: TransitFlow
Goal: Allow efficient travel conditions for city traffic.
Preconditions: Current time of the day.
Trigger: Time

Scenario:

1. Depending on the time of day such as morning, afternoon, and evening traffic light timing will be adjusted.
2. Traffic light green signals of side roads will be shorter due to the increase of traffic of flow, while main roads green signals will be longer.
3. Once the time of day is changed the timing intervals of the light states and cycles will reflect the change.

Exceptions:
Depending on inputs such as the IR sensors, pedestrian crossing button, and pressure plate will override the timing due to a car being physically waiting.

5 Definition of Terms

CSR - 3.1.1 The Control System Receiver a module connected to the TransitFlow system Controller designed to receive traffic data from the Traffic Light Controller's subsystems.

DCU - 3.1.2 The Data Conversion Unit handles data packets received from the TransitFlow system Controller

DTS - The Data Transfer System sends data packets created by the DCU to the corresponding Traffic Light subsystem

Cycle - Execution of one instruction set by the TransitFlow system.

State - The current status of said traffic light such as green, yellow, red, etc.

Maximum Green - The longest duration that an intersection will allow a green light signal.

Minimum Green - The shortest duration that an intersection will allow a green light signal.

Actual Green - The average time for a green signal at an intersection.

TF Controller - TransitFlow central controller.

Traffic Light Subsystem - General term for any system that communicates with the TransitFlow central controller.