



A FOUR – WAY TRAFFIC LIGHT CONTROLLER

Presented by:
Venkata Sai Hari Krishna Maddi

CONTENTS

01 Traffic Light Controller

02 Project Specifications

03 Pin Diagram

04 Pin Description

05 Idea of Implementation

06 Pictorial Representation

07 State Diagram



TRAFFIC LIGHT CONTROLLER



- A Four way traffic signalling controller is the implementation of traffic lights to cross a intersection point by reducing collisions between vehicles while entering into different directions via junction.
- Usage of vehicles are increasing day by day, thereby traffic is becoming hectic especially at four way junctions.
- Generally, Round Robin Signalling method is used in these four way junctions but it has its own pros and cons.

➤ DISADVANTAGES OF ROUND ROBIN MODEL:

- Waiting time is more.
- Due to traffic gathering in waiting is also more, whole traffic in one direction may or may not be cleared in allotted allowed time.



PROJECT SPECIFICATIONS



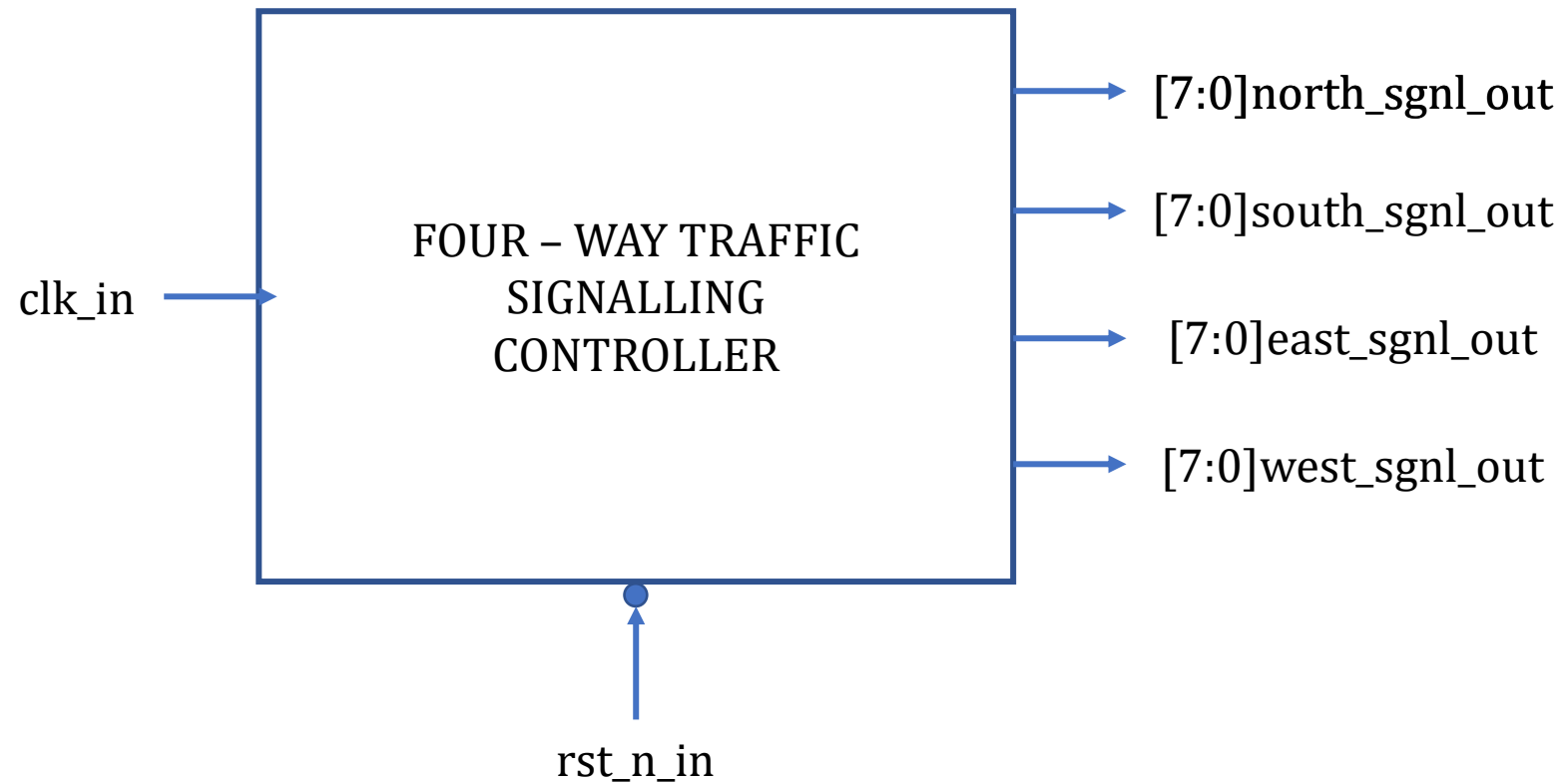
- This project is implementation of four – way traffic light controller with following specifications.
 - Free Left
 - Need to avoid routine round robin method of allowing traffic
 - It should have an optimal delay in all directions
 - The following are the signal representations.
 - ✓ **RED:** Stop
 - ✓ **GREEN:** Allow
 - ✓ **YELLOW:** About to turn red



An aerial, slightly blurred view of a city intersection. The image shows multiple lanes of traffic, with cars and buses visible. Traffic lights are prominent at the corners of the intersection. The overall color palette is dominated by the blues and greys of the pavement and buildings, with splashes of color from the vehicles and traffic signals.

IMPLEMENTATION

PIN DIAGRAM



PIN DESCRIPTION



PIN	DESCRIPTION
rst_n_in	Asynchronous Active Low Reset
clk_in	Clock
[7:0]north_sgnl_out	Signals facing NORTH direction
[7:0]south_sgnl_out	Signals facing SOUTH direction
[7:0]east_sgnl_out	Signals facing EAST direction
[7:0]west_sgnl_out	Signals facing WEST direction

PINS IN VECTOR VARIABLES	DESCRIPTION
north_sgnl_out[0], south_sgnl_out[0], east_sgnl_out[0], west_sgnl_out[0]	Straight RED Signals
north_sgnl_out[1], south_sgnl_out[1], east_sgnl_out[1], west_sgnl_out[1]	Right RED Signals
north_sgnl_out[2], south_sgnl_out[2], east_sgnl_out[2], west_sgnl_out[2]	YELLOW Signals
north_sgnl_out[3], south_sgnl_out[3], east_sgnl_out[3], west_sgnl_out[3]	Straight GREEN Signals
north_sgnl_out[4], south_sgnl_out[4], east_sgnl_out[4], west_sgnl_out[4]	Right GREEN Signals
north_sgnl_out[5], south_sgnl_out[5], east_sgnl_out[5], west_sgnl_out[5]	Left GREEN Signals
north_sgnl_out[6], south_sgnl_out[6], east_sgnl_out[6], west_sgnl_out[6]	Pedestrian RED Signals
north_sgnl_out[7], south_sgnl_out[7], east_sgnl_out[7], west_sgnl_out[7]	Pedestrian GREEN Signals



IDEA OF IMPLEMENTATION



- Consider asserting time for signals:

- Straight Green: 60sec
- Right Green: 30sec
- Yellow: 10sec

- So, if in round robin method above asserting time are considered then waiting time will be 210sec in each direction

- In this model, waiting time for

- Straight green is decreased to 150sec
- Right green is decreased to 180sec



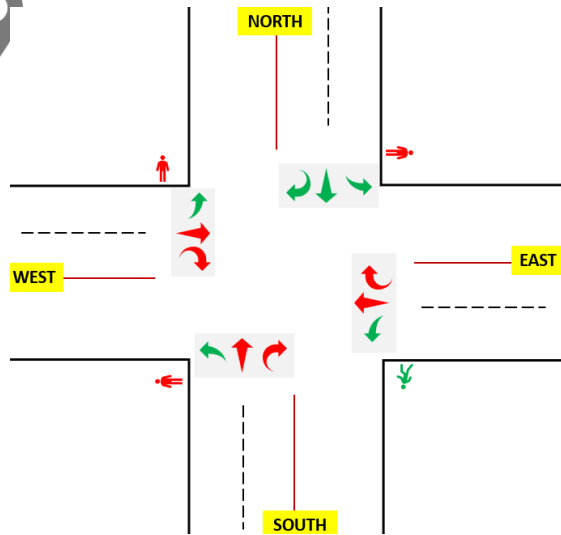
- **PROCEDURE:**

- Allow North straight (60sec) in the same time allow North right(first 30 sec) + West Right(next 30sec), East Pedestrian(60sec)
- Assert North YELLOW(30sec)
- Allow South straight (60sec) in the same time allow South right(first 30 sec) + East Right(next 30sec), West Pedestrian(60sec)
- Assert South YELLOW(30sec)
- Allow East and West straight(60sec), North and South Pedestrian(60sec)
- Assert East and West YELLOW (10sec)

PICTORIAL REPRESENTATION

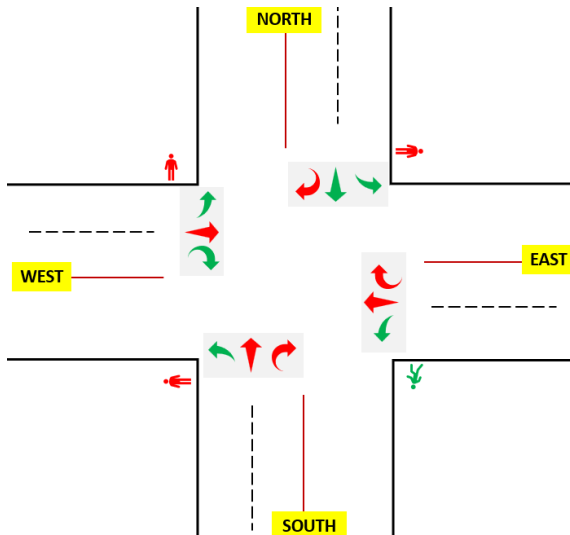


S_0



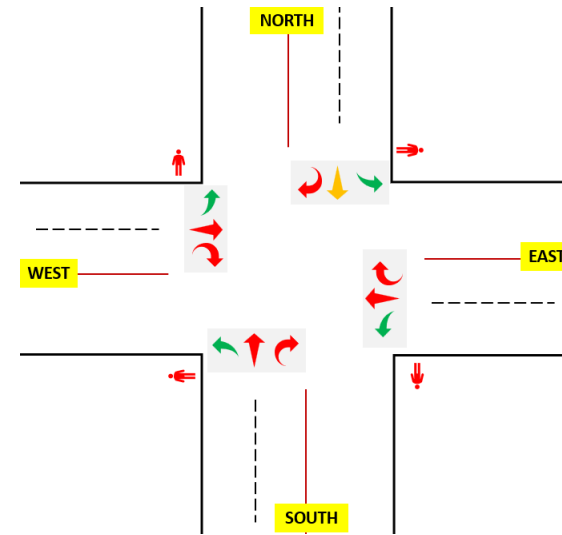
Assert
NORTH Straight (60-31sec)
NORTH Right (30sec)

S_1



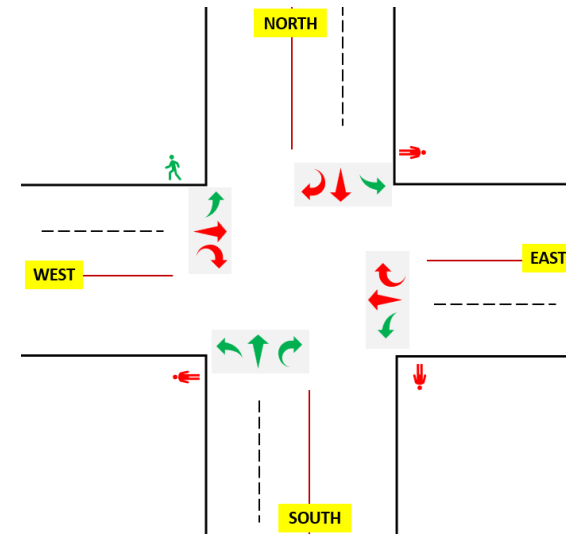
Assert
NORTH Straight (30sec)
WEST Right (30sec)

S_2



Assert
NORTH Yellow (10sec)

S_3

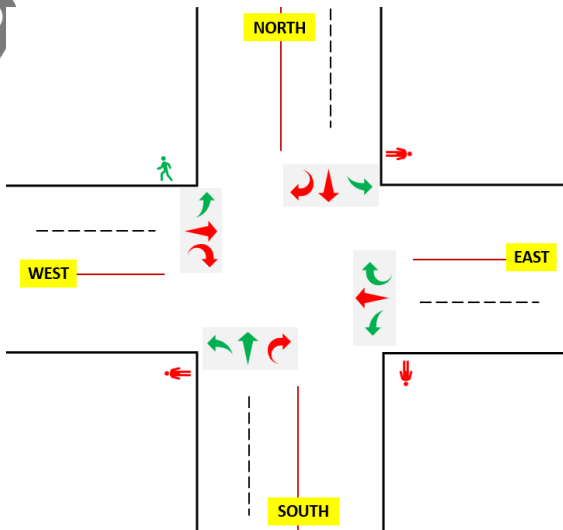


Assert
SOUTH Straight (60-31sec)
SOUTH Right (30sec)

PICTORIAL REPRESENTATION

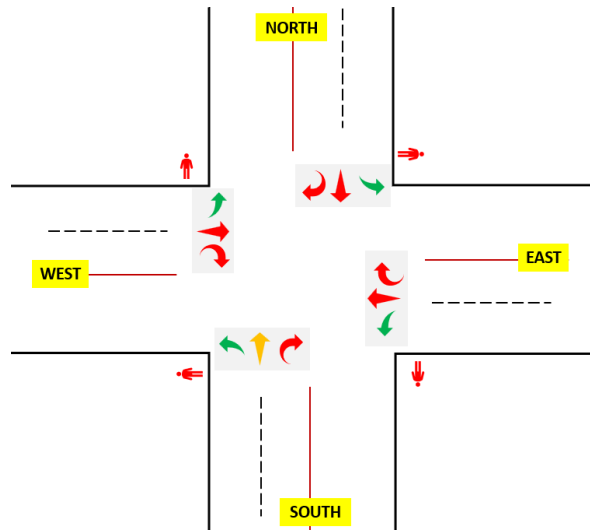


S_4



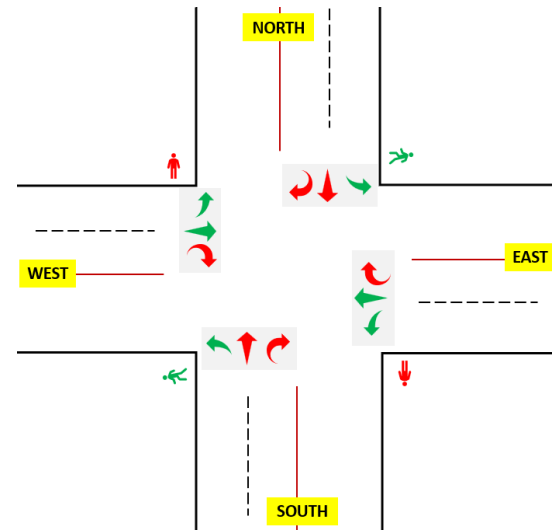
Assert
SOUTH Straight (30sec)
EAST Right (30sec)

S_5



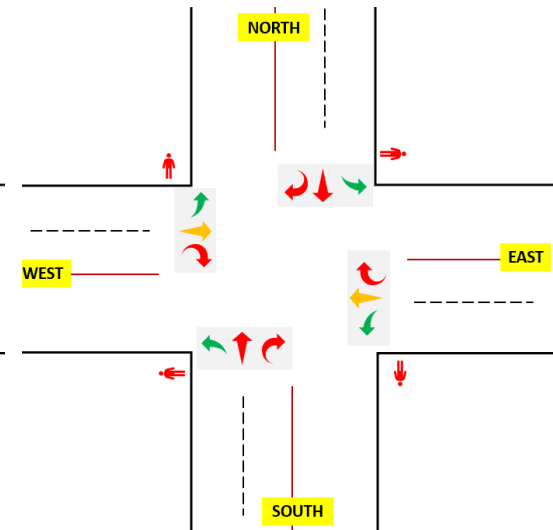
Assert
SOUTH Yellow (10sec)

S_6



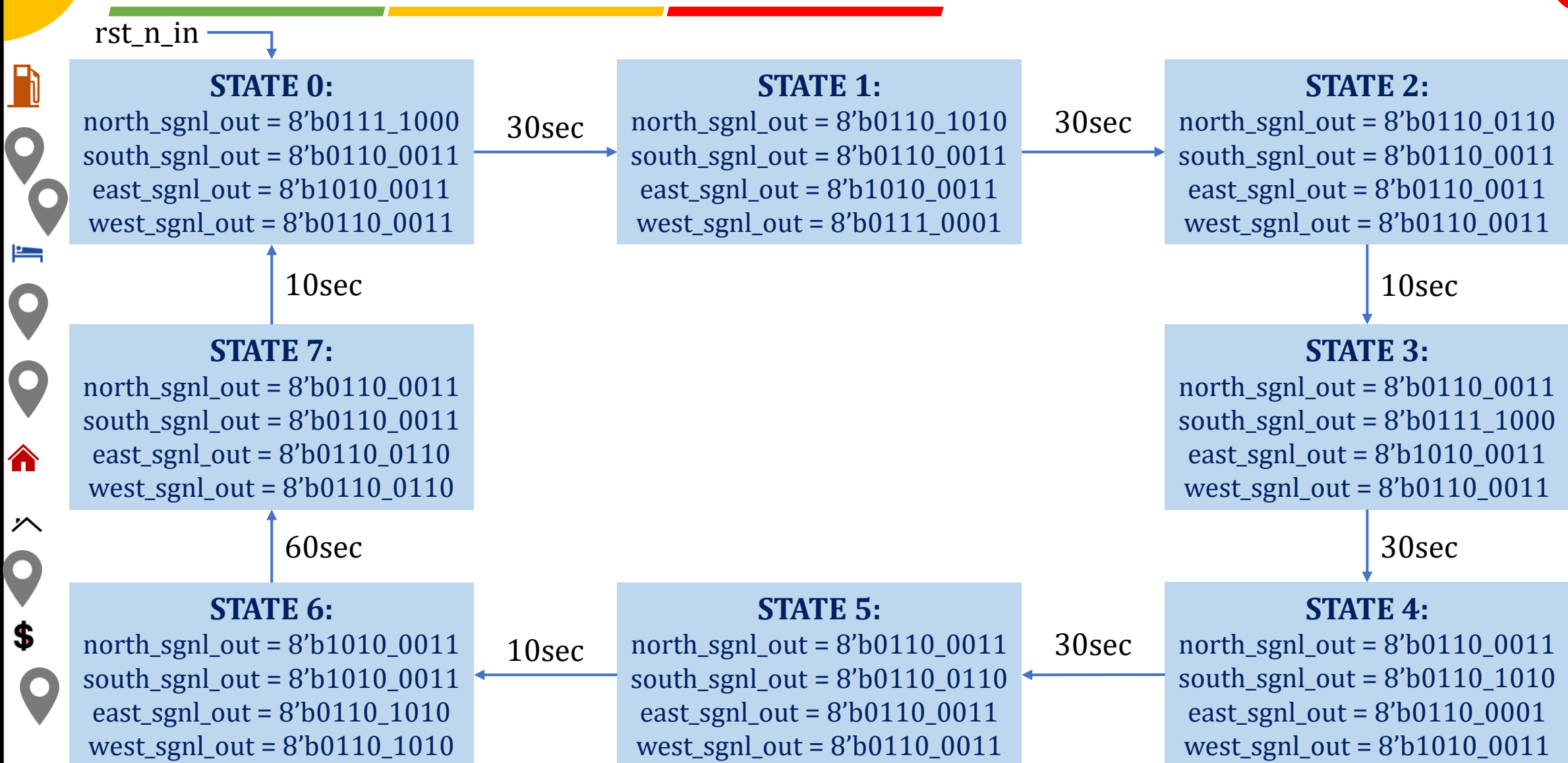
Assert
EAST Straight (60sec)
WEST Straight (60sec)

S_7



Assert
EAST Yellow (10sec)
WEST Yellow (30sec)

STATE DIAGRAM



An aerial, slightly blurred photograph of a dense urban street, likely in New York City. The street is filled with a variety of vehicles, including cars, buses, and trucks, moving in both directions. Tall, multi-story buildings line both sides of the street, their facades showing a mix of architectural styles and colors. The sky is overcast with soft, grey clouds. In the center of the image, there is a black rectangular box with a white border containing the text "TEST PLAN" in a white, serif font.

TEST PLAN



TEST PLAN FOR TRAFFIC LIGHT CONTROLLER				
Inputs: rst_in clk_in		Intermediate Signals: Count_st count_rt count_yellow	Outputs: [7:0]north_sgnl_out [7:0]south_sgnl_out [7:0]east_sgnl_out [7:0]west_sgnl_out	
S.no	Scenarios	Test Case	Test Description	Expecatations as per Specification
1	Test 01	Reset	initialise clock & reset	Resets the system
			Set rst_in = 0	Current state = reset state
				count_st = 60 count_rt = 30 count_yellow = 0
2	Test 02	Allowing North straight and right, East Pedistrian	initialise clock & reset	upto 30 time units: north_sgnl_out = 8'b0111_1000 south_sgnl_out = 8'b0110_0011 east_sgnl_out = 8'b1010_0011 west_sgnl_out = 8'b0110_0011
			Reset system set rst_in = 0	
			Before a posedge clk_in, set rst_in = 1	
3	Test 03	Allowing North straight, west right and East Pedistrian	initialise clock & reset	from 31 to 60 time units: north_sgnl_out = 8'b0110_1010 south_sgnl_out = 8'b0110_0011 east_sgnl_out = 8'b1010_0011 west_sgnl_out = 8'b0111_0001
			Reset system set rst_in = 0	
			Before a posedge clk_in, set rst_in = 1	
4	Test 04	Activation North yellow	initialise clock & reset	from 61 to 70 time units: north_sgnl_out = 8'b0110_0110 south_sgnl_out = 8'b0110_0011 east_sgnl_out = 8'b0110_0011 west_sgnl_out = 8'b0110_0011
			Reset system set rst_in = 0	
			Before a posedge clk_in, set rst_in = 1	

5	Test 05	Allowing South straight and right, West Pedestrian	initialise clock & reset	from 71 to 100 time units: north_sgnl_out = 8'b0110_0011 south_sgnl_out = 8'b0111_1000 east_sgnl_out = 8'b0110_0011 west_sgnl_out = 8'b1010_0011
			Reset system set rst_in = 0	
			Before a posedge clk_in, set rst_in = 1	
6	Test 06	Allowing South straight, east right and west Pedestrian	initialise clock & reset	from 101 to 130 time units: north_sgnl_out = 8'b0110_0011 south_sgnl_out = 8'b0110_1010 east_sgnl_out = 8'b0111_0001 west_sgnl_out = 8'b1010_0011
			Reset system set rst_in = 0	
			Before a posedge clk_in, set rst_in = 1	
7	Test 07	Activation South yellow	initialise clock & reset	from 131 to 140 time units: north_sgnl_out = 8'b0110_0011 south_sgnl_out = 8'b0110_0110 east_sgnl_out = 8'b0110_0011 west_sgnl_out = 8'b0110_0011
			Reset system set rst_in = 0	
			Before a posedge clk_in, set rst_in = 1	
8	Test 08	Allowing East & West straight, North & South Pedestrian	initialise clock & reset	from 141 to 200 time units: north_sgnl_out = 8'b1010_0011 south_sgnl_out = 8'b1010_0011 east_sgnl_out = 8'b0110_1010 west_sgnl_out = 8'b0110_1010
			Reset system set rst_in = 0	
			Before a posedge clk_in, set rst_in = 1	
9	Test 09	Activation of East & West Yellow	initialise clock & reset	from 200 to 210 time units: north_sgnl_out = 8'b1010_0011 south_sgnl_out = 8'b1010_0011 east_sgnl_out = 8'b0110_0110 west_sgnl_out = 8'b0110_0110
			Reset system set rst_in = 0	
			Before a posedge clk_in, set rst_in = 1	



The background of the image is a light gray network graph. It consists of numerous small circular nodes, some of which are solid gray and others are hollow with a gray outline. These nodes are interconnected by a web of thin, light gray lines, creating a complex, organic pattern that fills the entire frame. In the center of this pattern, the words "Thank you" are written in a large, elegant, black cursive script. The word "Thank" is on the top line, and "you" is on the line below it, with the two words overlapping slightly. The overall aesthetic is clean, modern, and professional, suggesting a theme of connectivity and gratitude.

*Thank
you*