**APPENDIX A**

**SERVER PROGRAM WITH PYTHON**

#!/usr/bin/env python3

# -\*- coding: utf-8 -\*-

"""

Created on Tue Mar 3 18:36:38 2020

@author: AungMinKhant

"""

import mysql.connector

from mysql.connector import errorcode

import socket

import sys

from random import randint

import time

# Create a TCP/IP socket

sock = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

# Bind the socket to the port

server\_address = ('192.168.0.100', 5000)

print ('starting up on %s port %s' % server\_address)

sock.bind(server\_address)

check = 1

lane1 = 10

lane2 = 15

ip = 0

p1 =0

p2 =0

p3 = 0

p4 = 0

def InsertMySQL(lane1,lane2):

interval\_time = int(time.time() % 60)

lane1 = int(lane1)

lane2 = int(lane2)

print(interval\_time)

print(lane1)

print(lane2)

try:

connection = mysql.connector.connect(host='localhost',database='traffic',user='minkhant',password='root')

if connection.is\_connected():

cursor = connection.cursor()

cursor.execute("select database();")

recoord = cursor.fetchone()

# cursor.execute("TRUNCATE TABLE cycle")

# cursor.execute("TRUNCATE TABLE lane1")

#cursor.execute("TRUNCATE TABLE lane2")

query = """INSERT INTO cycle (lane1,lane2,cycletime,interval\_time)

VALUES

(%s,%s,%s,%s)"""

\_data = (lane1,lane2,lane1+lane2,interval\_time)

cursor=connection.cursor()

result = cursor.execute(query,\_data)

connection.commit()

except IOError as e:

print("Error While connecting to MySQL",e)

finally:

if(connection.is\_connected()):

cursor.close()

connection.close()

def InsertLane1(lane1,lane2):

interval\_time = int(time.time())

lane1 = int(lane1)

lane2 = int(lane2)

print(interval\_time)

print(lane1)

print(lane2)

try:

connection = mysql.connector.connect(host='localhost',database='traffic',user='minkhant',password='root')

if connection.is\_connected():

cursor = connection.cursor()

cursor.execute("select database();")

recoord = cursor.fetchone()

query = """INSERT INTO lane1(delay,flowrate,saturation)

VALUES

(%s,%s,%s)"""

\_data = (lane1+lane2,0,0)

cursor=connection.cursor()

result = cursor.execute(query,\_data)

connection.commit()

except IOError as e:

print("Error While connecting to MySQL",e)

finally:

if(connection.is\_connected()):

cursor.close()

connection.close()

def InsertLane2(lane1,lane2):

interval\_time = int(time.time())

lane1 = int(lane1)

lane2 = int(lane2)

print(interval\_time)

print(lane1)

print(lane2)

try:

connection = mysql.connector.connect(host='localhost',database='traffic',user='minkhant',password='root')

if connection.is\_connected():

cursor = connection.cursor()

cursor.execute("select database();")

recoord = cursor.fetchone()

query = """INSERT INTO lane2(delay,flowrate,saturation)

VALUES

(%s,%s,%s)"""

\_data = (lane2,0,0)

cursor=connection.cursor()

result = cursor.execute(query,\_data)

connection.commit()

except IOError as e:

print("Error While connecting to MySQL",e)

finally:

if(connection.is\_connected()):

cursor.close()

connection.close()

def SelectMySQL():

try:

connection = mysql.connector.connect(host='localhost',database='traffic',user='minkhant',password='root')

if connection.is\_connected():

cursor = connection.cursor()

cursor.execute("select database();")

recoord = cursor.fetchone()

query = """SELECT \* FROM lane1 CROSS JOIN lane2 ORDER BY id ASC"""

cursor=connection.cursor()

result = cursor.execute(query)

result = cursor.fetchall()

print(len(result))

print(result)

for row in result:

flow1 = row[2]

flow2 = row[6]

sat1 = row[3]

sat2 = row[7]

WebsterMethod(flow1,flow2,sat1,sat2)

except IOError as e:

print("Error : ", e)

finally:

if(connection.is\_connected()):

cursor.close()

connection.close()

def WebsterMethod(flow1,flow2,S1,S2):

t = 1

a = 15

v = 30

L = 20

clctime = math.floor(t + ((0.5)\*(v/a)\*(22/15))+ ((W + L)/v)\*(15/22))

AmberTime = 3

LossTime = 2 \* clctime

y1 = flow1/S1

y2 = flow2/S2

Y = y1 + y2

Cycle = (1.5\*L + 5)/(1-Y)

G1 = ((y1/Y)\*(Cycle - L))

G2 = ((y2/Y)\*(Cycle - L))

R1 = Cycle - (G1 + L)

R2 = Cycle - (G1 + L)

InsertMySQL(G1,R1)

InsertLane1(G1,R1)

InsertLane2(G1,R1)

p1 = G1

p2 = R1

if \_\_name\_\_ == '\_\_main\_\_':

InsertMySQL(10,15)

InsertLane1(10,15)

InsertLane2(10,15)

SelectMySQL()

while True:

print('\nwaiting to receive message')

data, address = sock.recvfrom(4096)

print('received %s bytes from %s' % (len(data), address))

ip,\_ = address

if ip == '192.168.0.101':

lane1 = 1

adr1 = address

print >>sys.stderr, '\nlane1'

elif ip == '192.168.0.102':

lane2 = 1

adr2 = address

print >>sys.stderr, '\nlane2'

if lane1 == 1 and lane2 == 1:

SelectMySQL()

sent = sock.sendto('%s:%s' % (p1, p2), adr1)

sent = sock.sendto('%s:%s' % (p2 , p1), adr2)

print('sent %s bytes back to %s' % (p1, adr1))

print('sent %s bytes back to %s' % (p2, adr2))

lane1 = 0

lane2 = 0

print('\nCycle update')