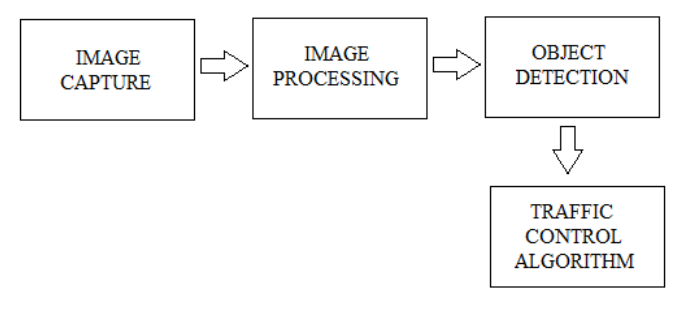
**Traffic Control and Management System using BMC 2835 Raspberry PI**

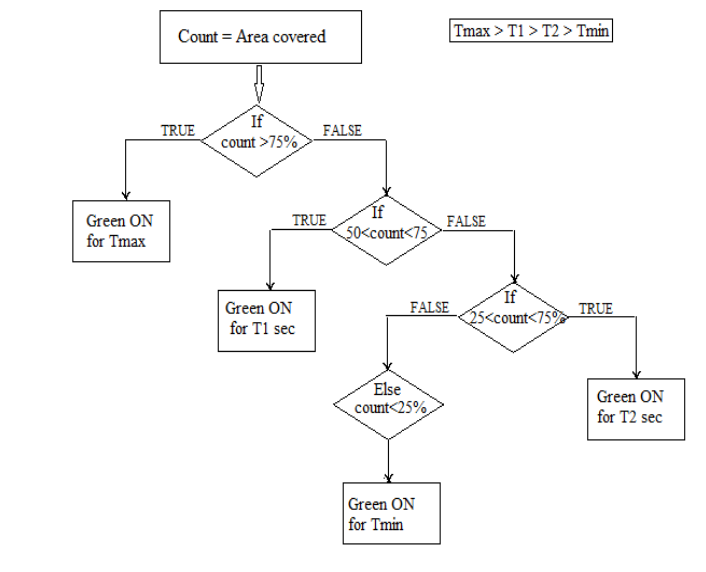
This project focuses on detecting, tracking and counting vehicles by using "[Blob Detection]” method. The unique ID is assigned to the each vehicle so it can not be counted more than once. This project was developed by OpenCV library. This vehicle count is taken as input to the Traffic Control System, which runs a algorithm to make traffic flow ease by increasing GREEN signal time.



Architecture Diagram

**Proposed Software Solution:-**

1. Image Capture by using Camera on each lane.
2. Image Processing uses Gaussian filter [11] to filter out the errors such as illumination and intensity variations, poor contrast etc. Conversion of RGB to HSV, background subtraction, conversion of HSV image to GRAY and threasholding the area captured for vehicles.
3. Traffic Control Algorithm :
   * Area count as the input data to control the traffic lights connected to the GPIO pins of the Raspberry pi.
   * For traffic is greater than 75% of the total area a maximum time TIMER\_MAX\_TRAFFIC is allotted to that lane,
   * similarly if the count is between 50%-75% a lesser time TIMER\_HIGH\_TRAFFIC
   * if the count is between 25%-50% TIMER\_MILD\_TRAFFIC is allotted
   * if density is less than 25% a minimum value of time i.e. TIMER\_NO\_TRAFFIC is allotted to that lane.
   * ( TIMER\_MAX\_TRAFFIC (Tmax) > TIMER\_HIGH\_TRAFFIC(T1) > TIMER\_MILD\_TRAFFIC(T2) > TIMER\_NO\_TRAFFIC(Tmin)) = (120 seconds > 90 seconds > 60 seconds > 30 seconds).



**Test-Driven Development (TDD) based approach using Unit Testing :**

#define BOOST\_TEST\_DYN\_LINK

#define BOOST\_TEST\_MODULE TrafficLights\_TM // must be define before "unit\_test.hpp"

#include <boost/test/unit\_test.hpp>

#include "TrafficLights.h"

using namespace TrafficLights;

BOOST\_AUTO\_TEST\_SUITE (Dummy\_Suite)

BOOST\_AUTO\_TEST\_CASE(Dummy\_Case )

{

//testing

BOOST\_TEST\_MESSAGE("Dummy test case for validation purpose ...");

}

BOOST\_AUTO\_TEST\_SUITE\_END()