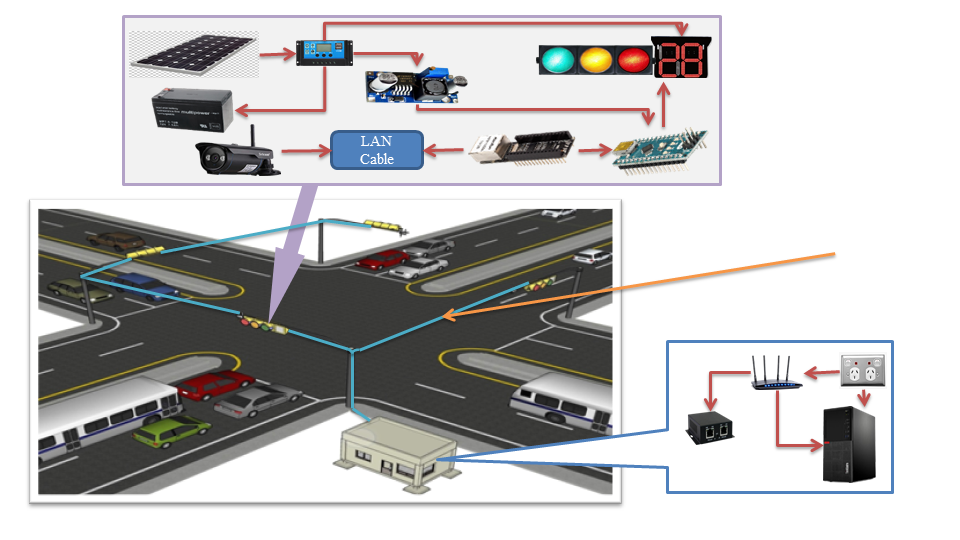
**CHAPTER 3**

**METHODOLGY**

3.1. **System Overview of Adaptive Traffic Signal Control System**

In Adaptive Traffic Signal Control System consists of two parts, main control unit and digit control units. Main control unit will collect traffic data from cameras and calculate required timing cycle. When main control unit get the optimum timing cycle, it send green split time and red time through the LAN to the digit control units. Each digit control unit receives data from the main control unit and task according to timing cycle. It also performs the traffic signal light such as green, amber and red. At the end of timing, digit control units send feedback to main control unit and request the new timing cycle for next round. While digit control units is running, main control unit collect data for each lane such as critical traffic saturation and average traffic flow volume by using image processing technique.

Traffic Control Unit



Main Control Unit

LAN communication

Figure 3.1 System Overview of Adaptive Traffic Signal Control System

3.2. **Video Live Streaming and Data Collections**

In this system, data collection is important role to calculate timing cycle. Image processing technique uses for collect traffic data. Video Live Streaming, Vehicles detection, classification and counting are utilized by openCV and python programming. Main Control unit connects to each camera with LAN for Video Live Streaming and collects traffic volume.

3.2.1. **Background Subtractions technique**

Background Subtractions is a popular method for isolating the moving parts of a scene by segmenting it into background and foreground. It is a widely used approach for detecting moving objects in videos from static cameras. The rationale in the approach is that of detecting the moving objects from the difference between the current frame and a reference frame, often called "background image", or "background model". Background subtraction is mostly done if the image in question is a part of a video stream. Background subtraction provides important cues for numerous applications in computer vision, for example surveillance tracking or human poses estimation.

Background subtraction can be generalized as three stages of the process ie, pre-processing, background modelling, and foreground detection. At the pre-processing stage involves simple image processing in the input video, such as format conversion, image resizing and others for the next steps. Then, the background modelling stage is responsible for constructing a statistical model of the image, followed by the classification of pixels in the foreground detection phase. The purpose of this stage is to establish a consistent background model but still be able to adapt to changes in the existing environment. The model should be able to tolerate the level of environmental change, but remain sensitive in detecting the movement of the relevant object. The next step is Foreground Detection, at this stage the foreground extraction process from the background. Simply can be explained in equation (3.1)

(3.1)

Where h(x, y) is background subtraction result, f (x , y) is image frame, and g ( x, y) is background modelling. One way to extract objects from the background is to select this model through a threshold value of T. Then, the image at the point g(x, y) at the value f(x, y) ≥ T is called the object (foreground), while the other is called the background. The equation of the thresholding value can be written according to equation (3.2)

(3.2)

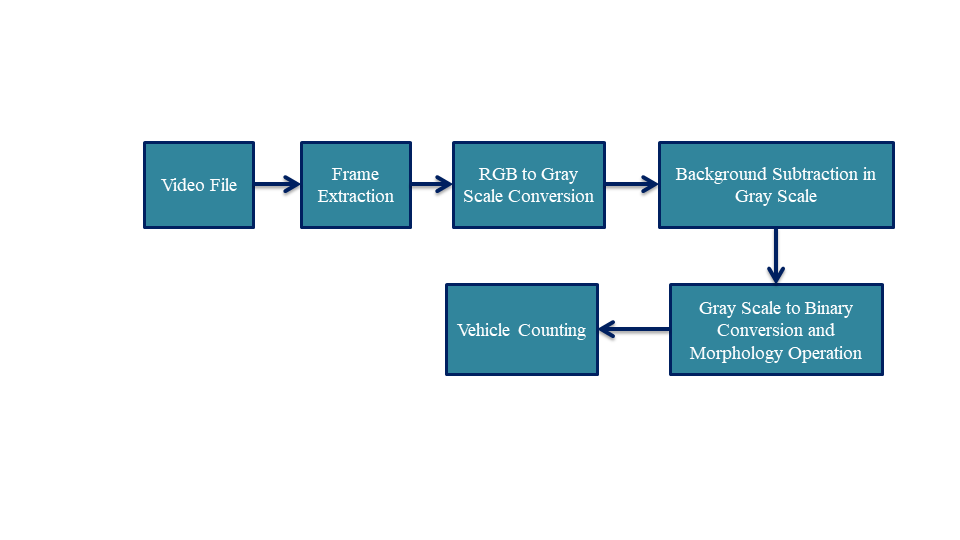


Figure 3.2. System Design for Vehicle Counting

**3.3. Vehicles Detection and Counting with YOLOV3**

You only look once (YOLO) is a state-of-the-art, real-time object detection system. On a Pascal Titan X it processes images at 30 FPS and has a mAP(mean Average Precision) of 57.9% on COCO test-dev.

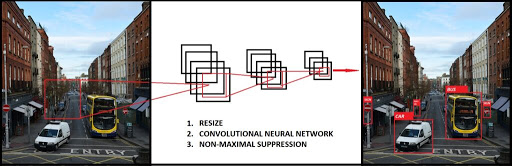


Figure 3.3 YOLO Detection System

YOLO is refreshingly simple. A single convolutional network simultaneously predicts multiple bounding boxes and class probabilities for those boxes. YOLO trains on full images and directly optimizes detection performance. This unified model has several benefits over traditional methods of object detection.

First, YOLO is extremely fast. Since we frame detection as a regression problem we don’t need a complex pipeline. We simply run our neural network on a new image at test time to predict detections. Our base network runs at 45 frames per second with no batch processing on a Titan X GPU and a fast version runs at more than 150 fps. This means we can process streaming video in real-time with less than 25 milliseconds of latency. Furthermore, YOLO achieves more than twice the mean average precision of other real-time systems.

Second, YOLO reasons globally about the image when making predictions. Unlike sliding window and region proposal-based techniques, YOLO sees the entire image during training and test time so it implicitly encodes contextual information about classes as well as their appearance.

Third, YOLO learns generalizable representations of objects. When trained on natural images and tested on art-work, YOLO outperforms top detection methods like DPM and R-CNN by a wide margin. Since YOLO is highly generalizable it is less likely to break down when applied to new domains or unexpected inputs.

All of our training and testing code is open source. A variety of pre trained models are also available to download.

3.3.1. **Theory**

As Yolo works with only one look at the image, sliding windows is not the right approach. Instead, the entire image can be spited into the grid. This grid will be S×S dimensions. Now, each cell is responsible for predicting a few different things.

First thing, each cell is responsible for predicting some number of [bounding boxes](http://datahacker.rs/025-cnn/). Also, each cell will predict confidence value for each bounding box. In other word, this is a probability that box contains an object. In case that there is no object in some grid cell, it is important that confidence value is very low for that cell.

When all of these predictions, we get a map of all the objects and a bunch of boxes which is ranked by their confidence value.

The second thing, each cell is responsible for predicting class probabilities. This is not saying that some grid cell contains some object, this is just a probability. So, if a grid cell predicts car, it is not saying that there is a car; it is just saying that if there is an object, than that object is a car.

In Yolo, **anchor boxes** are used to predict bounding boxes. The main idea of anchor boxes is to predefine two different shapes. They are called anchor boxes or anchor box shapes. In this way, we will be able to associate two predictions with the two anchor boxes. In general, we might use even more anchor boxes (five or even more). Anchors were calculated on the COCO (Common Objects in Context) dataset using k-means clustering.

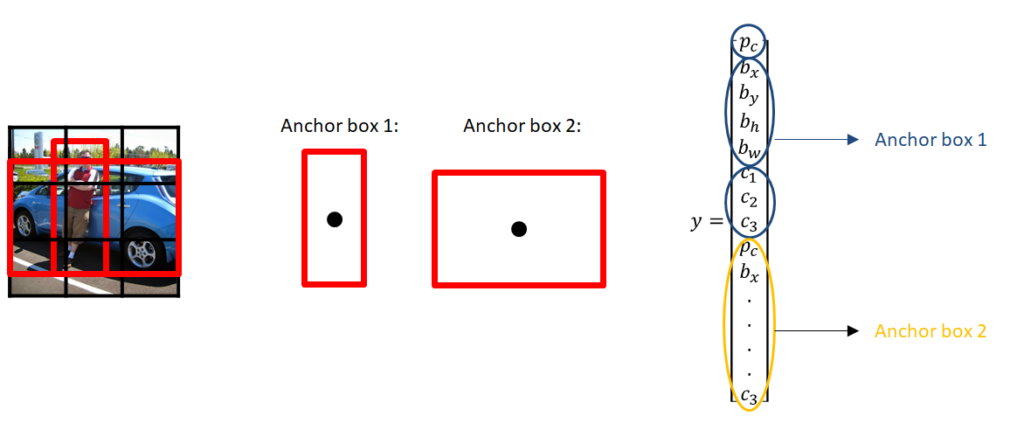


Figure 3.4 Anchor box in YOLO

3.3.2. **Unified Detection**

In this system divides the input image into an S×S grid. If the center of an object falls into a grid cell, that grid cell is responsible for detecting that object. Each grid cell predicts B bounding boxes and confidence scores for those boxes. These confidence scores reflect how confident the model is that the box contains an object and also how accurate it thinks the box is that it predicts. Formally define confidence as Pr (Object) x IOU (Intersection Over Union). If no object exists in that cell, the confidence scores should be zero. Otherwise we want the confidence score to equal the intersection over union (IOU) between the predicted box and the ground truth. Each bounding box consists of 5 predictions (x,y,w,h) and confidence. The (x,y) coordinates represent the center of the box relative to the bounds of the grid cell. The width and height are predicted relative to the whole image. Finally the confidence prediction represents the IOU between the predicted box and any ground truth box. Each grid cell also predicts C conditional class probabilities, Pr (Classi | Object). These probabilities are conditioned on the grid cell containing an object. The only predict one set of class probabilities per grid cell, regardless of the number of boxes B. At test time we multiply the conditional class probabilities and the individual box confidence predictions

which gives us class-specific confidence scores for each box. These scores encode both the probability of that class appearing in the box and how well the predicted box fits the object.

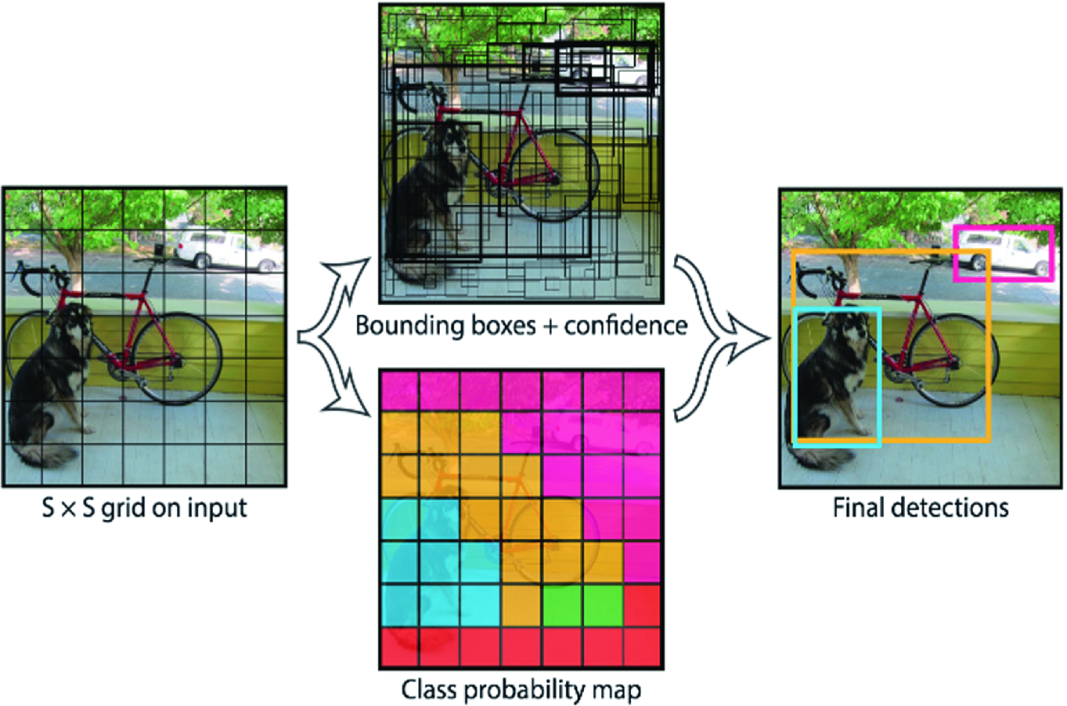


Figure 3.5 Divides the image into an S×S grid and for each grid cell predicts B bounding boxes, confidence for those boxes, and C class probabilities.

3.4**. Webster’s Method**

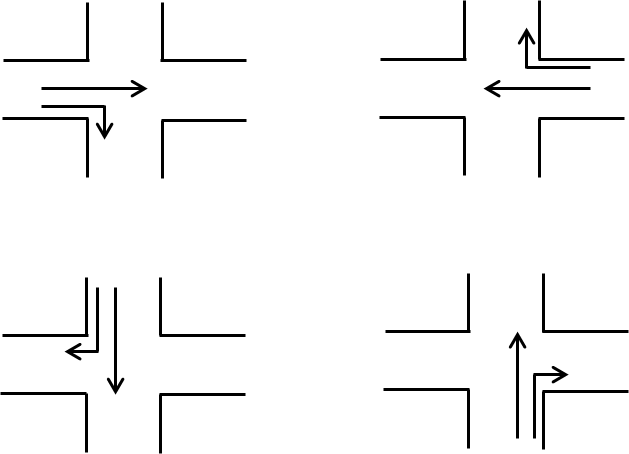
This method is used to compute the required data for determining suitable signal timing as it is suitable for timing of a pre-timed signal.

3.4.1. **Signal Phases**

The number of phases varies with the number of intersection’s approaches, the general intersection’s layout, the composition and the direction of traffic flows. Phasing can be used to minimize hazard risks by separation of movements. A large number of phases may be required if all conflicts are eliminated. Increasing the number of phases promotes safety but hinders efficiency because it results in increasing delays. The first issue is to decide how many phases are required. It is possible to have two, three, four or even more number of phases.

3.4.1.1. **Four phase signals**

There are at least three possible phasing options. Figure 2.3 shows the most simple and trivial phase plan. Where, flow from each approach is put into a single phase avoiding all conflicts. This type of phase plan is ideally suited in urban areas where the turning movements are comparable with through movements and when through traffic and turning traffic need to share the same lane. This phase plan could be very inefficient when turning movements are relatively low. Figure 2.4 shows a second possible phase plan option where opposing through traffic are put into the same phase. This type of phasing is very efficient when the intersection geometry permits to have at least one lane for each movement, and the through traffic volume is significantly high. Figure 2.5 shows yet another phase plan. However, this is rarely used in practice. There are five phase signals and six phase signals. They are normally provided if the intersection control is adaptive, that is, the signal phases and timing adapt to the real time traffic conditions.



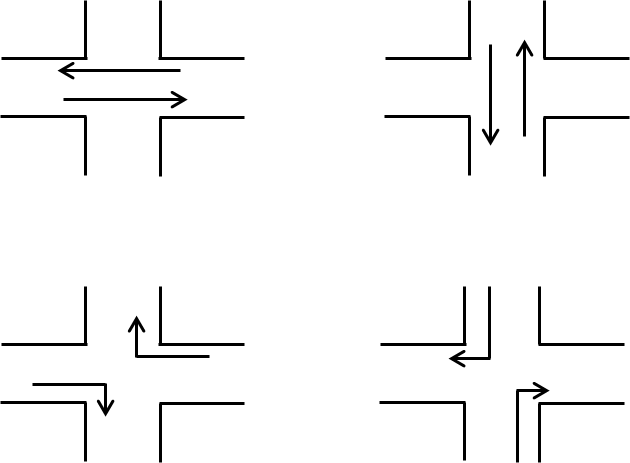
Phase 4

Phase 3

Phase 1

Phase 2

Figure 3.6. Movements in Four Phase Signal System: Option 1



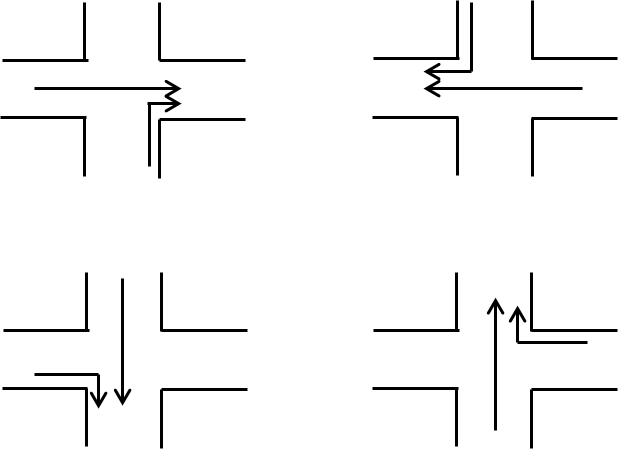
Phase 2

Phase 1

Phase 3

Phase 4

Figure 3.7. Movements in Four Phase Signal system: Option 2



Phase 1

Phase 2

Phase 4

Phase 3

Figure 3.8. Movements in Four Phase Signal System

3.4.1.2. **Passenger car unit (PCU)**

This unit is used for all traffic engineering design such as intersections, roadway, bridges, parking and traffic signals. The traffic volume and capacity are generally expressed as PCU per hour or PCU per hour or PUC/lane/hours and traffic density as PCU per kilometer length of lane.

On common roadway, different classes of vehicles such as cars, vans, buses, trucks, auto risk shaw, motor cycles, bicycles, bullock carts are found. Therefore, the mixed traffic flow characteristics are very complex. It is the method of expressing various types of vehicles having different characteristics that are very complex and common equipment unit. One car is considered as one unit. Traffic such as cycle, pedal cycle cause less inconvenience to other traffic than a car and hence it is considered as equipment to half car unit. A bus causes a lot of inconvenience to other traffic. It is estimated that a single bus causes inconvenience equivalent to three cars and hence it has three car units. Different vehicles having different vehicular and operational characteristics are also expressed in terms of standard unit called passenger car unit. The PCU for straight rural road and urban road are described in Table3.1.

Table 3.1. Suggested Values of PCU for Urban Roads

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr  No. | PCU Values of Vehicles Class | | | |
| Vehicle Class | Urban Roads Mid-block Section | Signalized Intersection | Curb Parking (Parallel and Angle) |
| 1 | Car | 1.0 | 1.0 | 1.0 |
| 2 | Bus and truck | 2.2 | 2.8 | 3.4 |
| 3 | Auto risk shaw | 0.5 | 0.4 | 0.4 |
| 4 | Two wheeler automobiles | 0.4 | 0.3 | 0.2 |
| 5 | Pedal cycles | 0.7 | 0.4 | 0.1 |
| 6 | Bullock cart | 4.6 | 3.2 | 1.2 |
| 7 | Hand cart | 4.6 | 3.2 | 0.3 |

3.4.2. **Vehicle Clearance Interval**

The objective of the amber signal indication following each green interval is to warn moving traffic facing the signals to come to a stop. It should provide enough time for vehicles to clear the intersections before cross traffic starts to move. If it is too short, it may constitute a hazard and increases rear-end collisions. The safe stopping distance can be calculated by the following formula.

 (3.3)

where, Y = clearance interval, in second

t = perception reaction time, in second, suggest value = 1

v = approach speed of clearing vehicle, feet per second

a = deceleration rate of clearing vehicle, in feet per second square, suggested value 15

W = intersection width (curb to curb), in feet

L = length of vehicle, in feet, suggest value = 20

3.4.3. **Minimum Cycle Length**

The minimum cycle time may be calculated by Webster’s Method for three and four phases signal cycle. In calculating minimum cycle length, the largest number of vehicles in a single lane entering the intersection must be considered during peak and off-peak hour. And then, approximate average headway between vehicles entering the intersection is also considered. The cycle lengths between 45 seconds and 180 seconds are used in the field [6].

 (3.4)

where, N = the number of phases

tl = lost time per phase in seconds

Xc = critical  ratio where v is the volume and c is the capacity

 = the ratio of critical volume to saturation flow

3.4.4. **Green Splitting**

Green splitting or apportioning of green time is the proportioning of effective green time in each of the signal phase. The green splitting is given by

 (3.5)

where, Vi = the critical lane volume for phase i

Vc = the critical lane volume

gi = the effective green time

C = the cycle time in seconds

L = total loss time

Actual green time can be now found out as,

 (3.6)

where, Gi = actual green time

gi = the effective green time

Yi = the yellow time

** = the lost time for phase i

Actual cycle length can be now found out as,

Actual cycle length, Cact = Gi + Yi + Ri (3.7)

G1

Y1

All Red

R1

Phase 1

Y1

All Red

G2

R2

Phase 2

Figure 3.9. Signal Timing of Phase-Plan

3.5. **Arduino Nano**

Arduino Nano is a small, compatible, flexible and breadboard friendly Microcontroller board, developed by Arduino.cc in Italy, based on ATmega328p ( Arduino Nano V3.x)  / Atmega168 ( Arduino Nano V3.x).It comes with exactly the same functionality as in Arduino UNO but quite in small size. It comes with an operating voltage of 5V, however, the input voltage can vary from 7 to 12V.

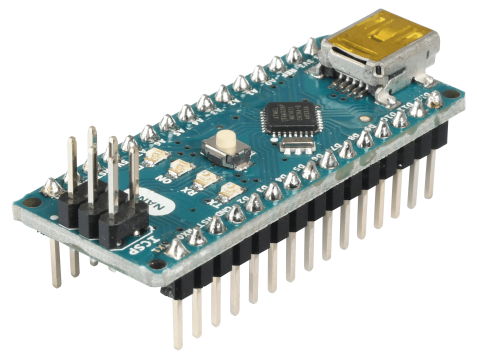


Figure 3.10. Arduino Nano

**3.5.1. Arduino Nano Pinout**

Arduino Nano pinout contains 14 digital pins, 8 analog Pins, 2 Reset Pins & 6 Power Pins.

1. Each of these Digital & Analog Pins are assigned with multiple functions but their main function is to be configured as input or output.
2. They are acted as input pins when they are interfaced with sensors, but if you are driving some load then use them as output.
3. Functions like pinMode() and digitalWrite()  are used to control the operations of digital pins while analogRead() is used to control analog pins.
4. The analog pins come with a total resolution of 10bits which measure the value from zero to 5V.
5. Arduino Nano comes with a crystal oscillator of frequency 16 MHz. It is used to produce a clock of precise frequency using constant voltage.
6. There is one limitation using Arduino Nano i.e. it doesn’t come with DC power jack, means you cannot supply external power source through a battery.
7. This board doesn’t use standard USB for connection with a computer, instead, it comes with Mini USB support.
8. Tiny size and breadboard friendly nature make this device an ideal choice for most of the applications where sizes of the electronic components are of great concern.
9. Flash memory is 16KB or 32KB that all depends on the Atmega board i.e Atmega168 comes with 16KB of flash memory while Atmega328 comes with a flash memory of 32KB. Flash memory is used for storing code. The 2KB of memory out of total flash memory is used for a bootloader.
10. The SRAM can vary from 1KB or 2KB and EEPROM is 512 bytes or 1KB for Atmega168 and [Atmega328](https://www.theengineeringprojects.com/2017/08/introduction-to-atmega328.html) respectively.
11. This board is quite similar to other Arduino boards available in the market, but the small size makes this board stand out from others. It is programmed using Arduino IDE which is an Integrated Development Environment that runs both offline and online.
12. No prior arrangements are required to run the board. All you need is board, mini USB cable and Arduino IDE software installed on the computer. USB cable is used to transfer the program from computer to the board.
13. No separate burner is required to compile and burn the program as this board comes with a built-in boot-loader.

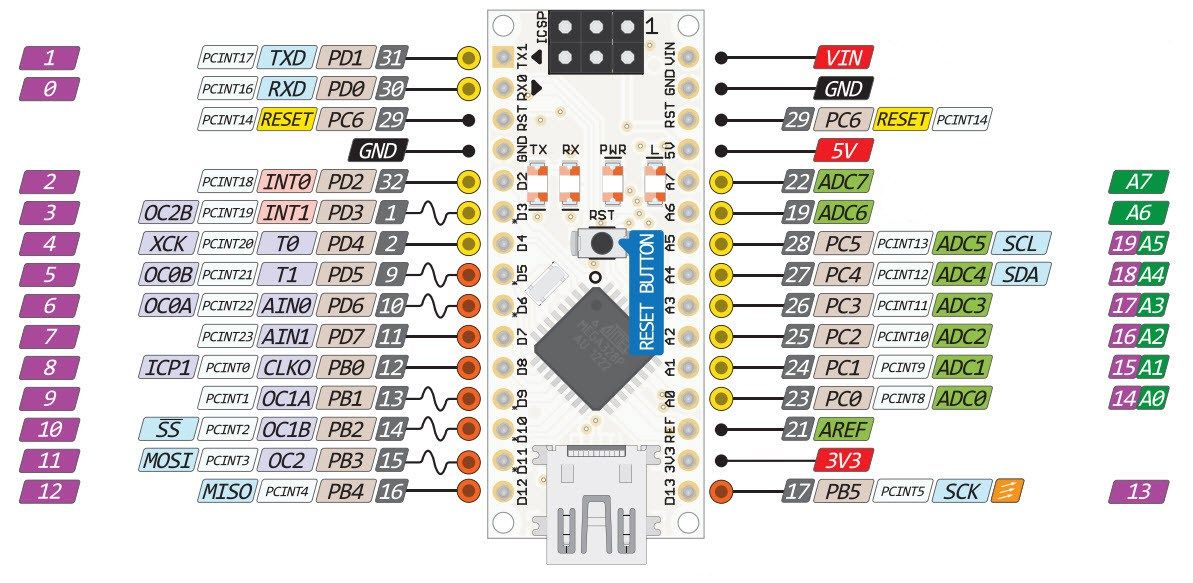


Figure 3.11. Arduino Nano pinouts

3.6. **Shift Register (74HC595)**

The 74HC595 is an 8-bit serial-in / serial or parallel-out shift register with a storage register and 3-state outputs. Both the shift and storage register have separate clocks. The device features a serial input (DS) and a serial output (Q 7S) to enable cascading and an asynchronous reset MR input. A LOW on MR will reset the shift register. Data is shifted on the LOW to HIGH transitions of the SHCP input. The data in the shift register is transferred to the storage register. Data in the storage register appear at the output whenever the output enable input (OE) is LOW. A HIGH on OE causes the outputs to assume a high-impedance OFF-state. Operation of the OE input does not affect the state of registers. Inputs include clamp diodes. This enables the use of current limiting resistors to interface inputs to voltages in excess of VCC.

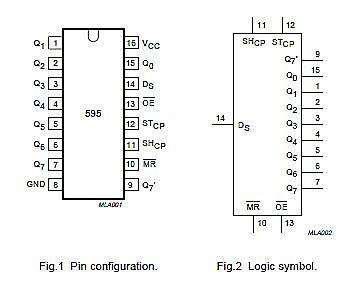


Figure 3.12. Pin Configurations and Logic Symbol for 74HC595 8-bit Shift Register

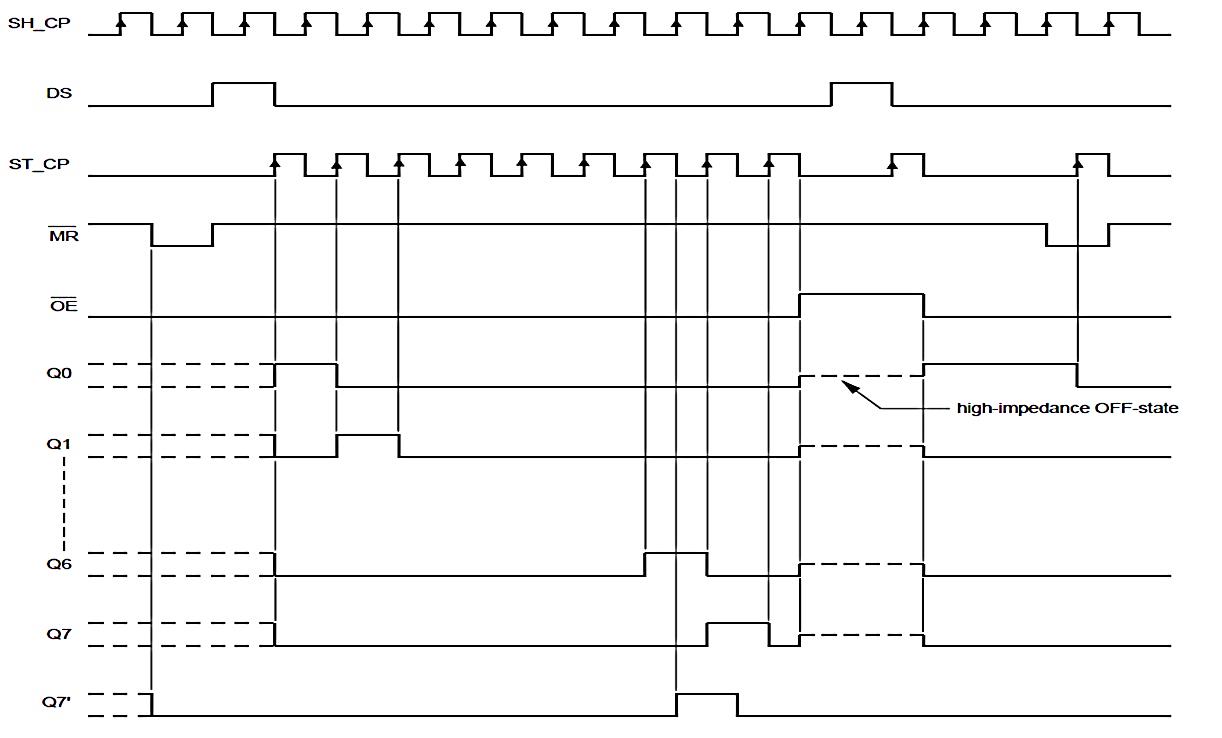


Figure 3.13. Timing Diagram for 74HC595 8-bit Shift Register

Table 3.2. Pins Diagram Description for 74HC595 8-Bit Shift Register

|  |  |  |
| --- | --- | --- |
| No | Symbol | Description |
| 1 | Q1 | Parallel data output ( bit – 1) |
| 2 | Q2 | Parallel data output ( bit – 2) |
| 3 | Q3 | Parallel data output ( bit – 3) |
| 4 | Q4 | Parallel data output ( bit – 4) |
| 5 | Q5 | Parallel data output ( bit – 5) |
| 6 | Q6 | Parallel data output ( bit – 6) |
| 7 | Q7 | Parallel data output ( bit – 7) |
| 8 | GND | Ground (0 V) |
| 9 | Q7’ | Serial data output |
| 10 | MR | Master reset (Active Low) |
| 11 | SH\_CP | Shift register clock input |
| 12 | ST\_CP | Storage register clock input |
| 13 | OE | Output enable (Active Low) |
| 14 | DS | Serial data input |
| 15 | Q0 | Parallel data output (8-bit) |
| 16 | Vcc | Positive Supply Voltage |

3.7. **Seven-segment Display Construction**

A seven-segment display is commonly used in electronic display device for decimal numbers from 0 to 9 and in some cases, basic characters. Use of light emitting diodes (LEDs) in seven segment displays made it more popular, whereas of late liquid crystal displays (LCD) displays have also come into use. Electronic devices like microwave ovens, calculators, washing machines, radios, digital clocks etc. to display numeric information are the most common applications.

A seven-segment display is constructed with tri colour common cathode led and transistor. These LEDs will glow when they are forward biased. The intensity of the LEDs depends on forward current. So, sufficient forward current has to be provided to these LEDs to glow with full intensity. This is provided by the driver and is applied to the seven segments. Colour of seven-segment display can be changed by enable to corresponding input power pin.

Table 3.3. Display numbers on a seven segment display configuration

|  |  |  |
| --- | --- | --- |
| **Number** | **g f e d c b a** | **Hex Code** |
| 0 | 0111111 | 3F |
| 1 | 0000110 | 06 |
| 2 | 1011011 | 5B |
| 3 | 1001111 | 4F |
| 4 | 1100110 | 66 |
| 5 | 1101101 | 6D |
| 6 | 1111101 | 7D |
| 7 | 0000111 | 07 |
| 8 | 1111111 | 7F |
| 9 | 1001111 | 4F |

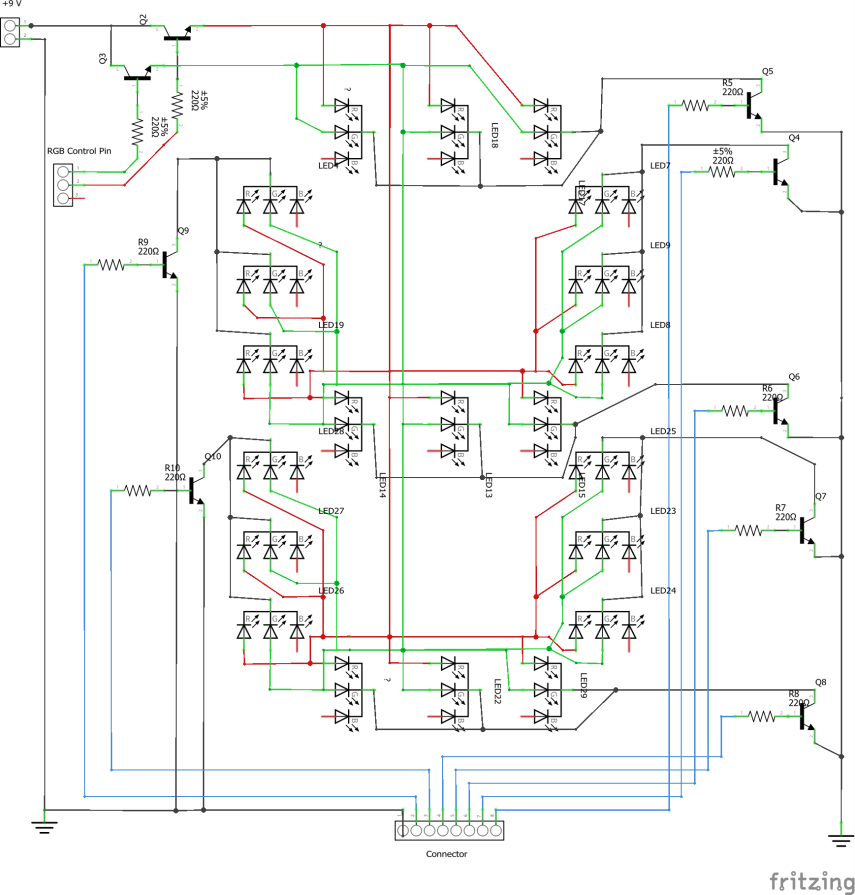


Figure 3.14. Circuit Diagram of Tri Colour Seven-segments Display

3.8. **C945 NPN Transistor**

C-945 transistor is basically a Negative Positive Negative (NPN) bipolar junction transistor. It has three regions emitter, base and collector. Its base is doped with P type semi-conductor material. Emitter and collector are doped with N type semiconducting material. It’s a low cost device and is very popular in the market due to its large range of applications. Its applications include fast switching, amplification, low power electronic circuits and many more.

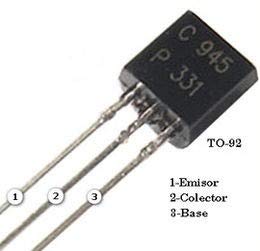


Figure 3.14. C945 NPN Transistor

Solid state switches are one of the main applications for the use of transistor to switch a DC output “ON” or “OFF”. Some output devices, such as LED’s only require a few milliamps at logic level DC voltages and can therefore be driven directly by the output of a logic gate. However, high power devices such as motors, solenoids or lamps, often require more power than that supplied by an ordinary logic gate so transistor switches are used.

If the circuit uses the Bipolar Transistor as a Switch, then the biasing of the transistor, either NPN or PNP is arranged to operate the transistor at both sides of the “ I-V ” characteristics curves have seen. The areas of operation for a transistor switch are known as the **Saturation Region** and the **Cut-off Region**. This means then that we can ignore the operating Q-point biasing and voltage divider circuitries required for amplification, and use the transistor as a switch by driving it back and forth between its “fully-OFF” (cut-off) and “fully-ON” (saturation) regions.

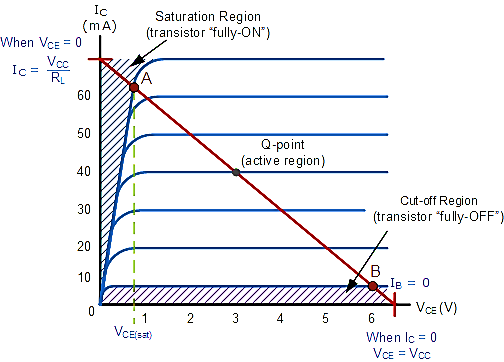


Figure 3.15. Characteristic curve of transistor

3.8.1. **Cut-off Region**

The operating conditions of the transistor are zero input base current ( IB ), zero output collector current ( IC ) and maximum collector voltage ( VCE ) which results in a large depletion layer and no current flowing through the device. Therefore the transistor is switched “Fully-OFF”. The Cut-off characteristics are

1. The input and Base are grounded ( 0v )
2. Base-Emitter voltage VBE < 0.7v
3. Base-Emitter junction is reverse biased
4. Base-Collector junction is reverse biased
5. Transistor is “fully-OFF” ( Cut-off region )
6. No Collector current flows ( IC = 0 )
7. VOUT = VCE = VCC = ”1″
8. Transistor operates as an “open switch”

3.8.2. **Saturation Region**

The transistor will be biased so that the maximum amount of base current is applied, resulting in maximum collector current resulting in the minimum collector emitter voltage drop which results in the depletion layer being as small as possible and maximum current flowing through the transistor. Therefore the transistor is switched “Fully-ON”. The characteristics of saturation region are

1. The input and Base are connected to VCC
2. Base-Emitter voltage VBE > 0.7v
3. Base-Emitter junction is forward biased
4. Base-Collector junction is forward biased
5. Transistor is “fully-ON” ( saturation region )
6. Max Collector current flows ( IC = Vcc / RL )
7. VCE = 0 ( ideal saturation )
8. VOUT = VCE = ”0″
9. Transistor operates as a “closed switch”

The “saturation region” or “ON mode” can defined when using a bipolar transistor as a switch as being, both junctions forward biased, VB > 0.7v and IC = Maximum.

3.9. **Ethernet Communication with UDP**

The User Datagram Protocol, or UDP for short, is a protocol that allows datagrams to be sent without connection in IP-based networks.. To achieve the desired services on the target hosts, it uses ports that are listed as one of the core components in the UDP header. Like many other network protocols, UDP belongs to the internet protocol family, where it is classified as a mediator between the network layer and the application layer at the transport level. UDP packet's called as user datagrams with 8 bytes header. In the user datagrams first 8 bytes contains header information and the remaining bytes contain data.

Source Port number 16 bits

Destination Port number 16 bits

Check Sum

16 bits

Total Length 16 bits

UDP Header (8 bytes)

UDP Data

Figure 3.16. User Datagram Header format

1. Source port number - This is a port number used by source host, who is transferring data. It is 16 bit longs. So port numbers range between 0 to 65,535.
2. Destination port number: This is a port number used by Destination host, who is getting data. It is also 16 bits long and also same number of port range like source host.
3. Length: Length field is a 16 bits field. It contains the total length of the user datagram, header and data.
4. Checksum: The UDP checksum is optional. It is used to detect error for the data. If the field is zero then checksum is not calculated. And true calculated then field contains 1.

3.9.1. **Characteristics of UDP**

The characteristics of UDP are given below.

1. End-to-end. UDP can identify a specific process running on a computer.
2. Unreliable, connectionless delivery
3. UDP uses a connectionless communication setup. In this UDP does not need to establish a connection before sending data. Communication consists only of the data segments themselves
4. Same best effort semantics as IP
5. No data, no sequence, no flow control
6. Subject to loss, duplication, delay, out-of-order, or loss of connection
7. Fast, low overhead
8. Suit for reliable, local network
9. 2.RTP(Real-Time Transport Protocol)

3.10. **Power System**

The main control unit is powered by 220V 60Hz power supply. Traffic control unit need for 12V DC power supply. Power consumption of Traffic control unit is calculated by;

Power for Arduino Controller = 3.15 W

Power for LED = 3.75 W

Total Require Power = 3.15 + 3.75 = 7 W