**ACKNOWLEDGMENT**

We have taken lot of efforts to make this project. However, it would not have been possible without the kind support and help of our project guide. We would like to extend my sincere thanks to all of them.

We are highly indebted to Dr. (prof) L.K.Bandhopadhyay (former Head of the Department, ECE) who is also our project guide, for his guidance and constant supervision in completing the project. At last we would also thank my group, my team members for constant support and efficient team work which made the final year project to be a successful one. The role played by each and every members of group were excellent. Co-operation, hard work and team management leads to a successful outcome.

**ABSTRACTS**

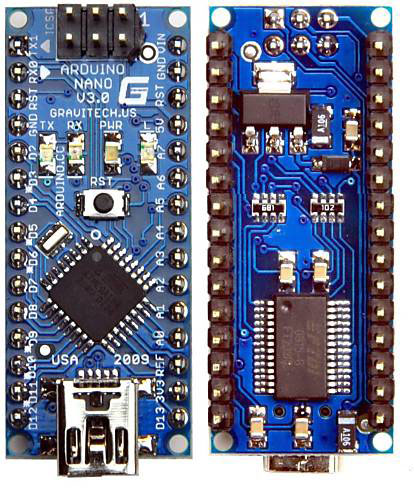
The Anti–Collision device is a detection device meant to be incorporated into cars for the purpose of safety. As opposed to the anti–collision devices present in the market today, this system is not designed to control the vehicle. Instead, it serves as an alert in the face of imminent collision. The device is intended to find a way to implement a minimum spacing for cars in traffic in an affordable way. It would also achieve safety for the passengers of a moving car. The device is made up of an infrared transmitter and receiver. Also incorporated into it is an audio visual alarm to work in with the receiver and effectively alert the driver and/or the passengers. The device would still sound an alarm even though it is not receiving infrared beams from the oncoming vehicle. This is due to reflection of its own infrared beams. At the end of the design and testing process, overall system was implemented with a constructed work, tested working and perfectly functional. The system which is the design and construction of an anti-collision system for vehicles was designed considering some factors such as economy, availability of components and research materials, efficiency, compatibility, portability and also durability. The performance of the system after test met design specifications.

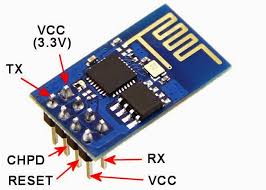
**CONTENTS**

|  |  |
| --- | --- |
| **NAME** | **PAGE NO.** |
| 1. ACKNOWLEDGEMENT | 1 |
| 1. ABSTRACTS | 2 |
| 1. INTRODUCTION | 3 |
| 1. OBJECTIVE OF PROJECT | 4 |
| 1. ADOPTED, SYSTEM IMPLEMENTATION & DETAILS OF HARDWARE & SOFTWARE USED | 5-6 |
| 1. UART COMMUNICATION WITH Wi-Fi (ESP 8266) |  |
| 1. DEFINITION OF PROBLEM |  |
| 1. Wi-Fi AT COMMUNICATION |  |
| 1. Wi-Fi P2P COMMUNICATION |  |
| 1. I2C COMMUNICATION WITH DIGITAL ACCELEROMETER |  |
| 1. COST ANALYSIS |  |
| 1. SYSTEM MAINTENANCE & EVALUATION |  |
| 1. FUTURE SCOPE |  |
| 1. CONCLUSION |  |
| 1. REFERENCES |  |

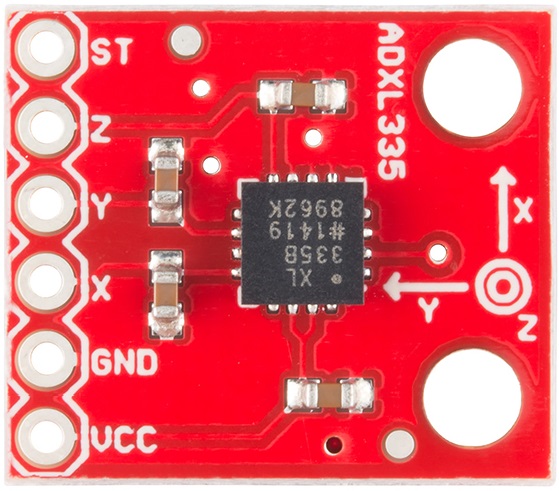
1. **INTRODUCTION**

The application consists of three main modules:





1. Wi-Fi ESP 8266
2. ATMEGA328P Arduino nano microcontroller



1. 3-Axis Analog accelerometer
2. 3-Axis digital accelerometer

Each of the modules consists of several processes which provide various functionalities.

Safety is a necessary part of man’s life. Due to the accident cases reported daily on the major roads in all parts of the developed and developing countries, more attention is needed for research in the designing an efficient car driving aiding system. It is expected that if such a device is designed and incorporated into our cars as a road safety device, it will reduce the incidence of accidents on our roads and various premises, with subsequent reduction in loss of life and property.

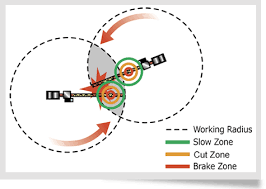
However, a major area of concern of an engineer should be safety, as it concerns the use of his/her inventions and the accompanying dangers due to human limitations. When it comes to the use of a motor vehicle, accidents that have occurred over the years tell us that something needs to be done about them from an engineering point of view. According to the 2007 edition of the Small-M report on the road accident statistic in Malaysia, a total of 6,035 people were killed in 2000 and the fatality spring up to 6,287 in 2006 from accident cases reported in 250,429 and 341,252 cases of accident for 2000 and 2006 respectively. In India according to 2016 data analysis, at every 60 sec there were 26 people loss their life in accidents. Suffice to say that the implementation of certain highway safety means such as speed restrictions, among others, has done a lot in reducing the rates of these accidents. The issue here is that policies of safe driving alone would not eradicate this, the engineer has a role to play, after all the main issue is an engineering product (the motor vehicle). Many motorists have had to travel through areas with little light under much fatigue, yet compelled to undertake the journey out of necessity. It is not always irresponsible to do this. A lot of cases reported is as a result of drivers sleeping off while driving, and when he/she eventually woke up, a head-on collision might have taken place. Not many have had the fortune to quickly avert this.

****



1. **OBJECTIVE OF PROJECT**

Safety is a necessary part of every living body’s life. Due to the accident cases reported daily on the major roads in all parts of the developed and developing countries, more attention is needed for research in the designing an efficient vehicle driving aiding system. It is expected that if such a device is designed and incorporated into our vehicle as a road safety device, it will reduce the incidence of accidents on our roads and various premises, with subsequent reduction in loss of life and property. A lot of cases reported is as a result of drivers sleeping off while driving, and when he/she eventually woke up, a head-on collision might have taken place. It is therefore important to consider the advantages of an early warning system where the driver is alerted of a possible collision with some considerable amount of time before it occurs.



1. **JUSTIFICATION OF PROJECT**

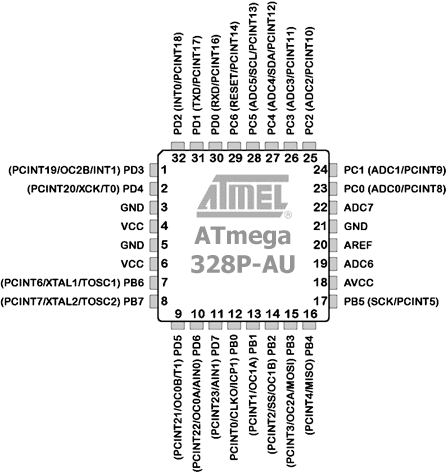
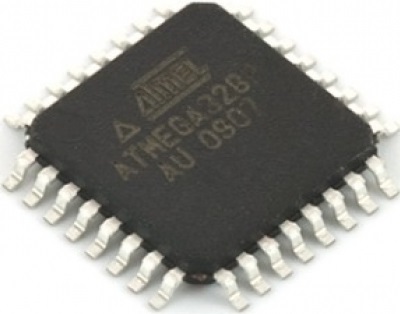
In our project, we are going to make a system as described below: -

1. Using A-GPS we can get current location of vehicle, living body or devices.
2. According to the dimension of object our system will create a protective virtual wall.
3. In our system, every device will be programmed to give them artificial intelligence.
4. In our project, every device will have intelligent feature to communicate with each other to get information of moving or static vehicle/devices.
5. Our project can prevent collision between vehicle, human body and vehicle etc.
6. This project can be installed easily.
7. **METHODOLOGY**
8. A-GPS will track the location of devices in every second, and will send data to microcontroller.
9. Using that A-GPS microcontroller will be programmed to useful data like speed of moving vehicle, direction of motion, Distance etc.
10. A microcontroller will be programmed to create protective virtual boundary.
11. A microcontroller will try to communicate with another microcontroller using peer to peer multipoint connection to get data of another microcontroller/devices.
12. This connection will be established by using Wi-Fi local host communication.
13. Since connection is peer to peer and multipoint so the range of system will increase n-times.
14. Using this data every device/vehicle will try to prevent that virtual boundary from collision, and hence no one will collide.
15. **ADOPTED, SYSTEM IMPLEMENTATION & DETAILS**

**OF**

**HARDWARE & SOFTWARE USED**

* 1. **HARDWARE USED:**
     1. **ATMEGA328P microcontroller:-**



* + 1. **Arduino Nano (ATMEGA328P module):-**

The Arduino Nano is a small, complete, and breadboard-friendly board based on the ATmega328. The Arduino Nano is programmed using the Arduino IDE (Integrated Development Environment).

The ATmega328 has 32 KB, (also with 2 KB used for the boot loader). The ATmega328 has 2 KB of SRAM and 1 KB of EEPROM. The Arduino Nano can be powered via the Mini-B USB connection, 6-20V unregulated external power supply (pin 30), or 5V regulated external power supply (pin 27). The power source is automatically selected to the highest voltage source.

Each of the 14 digital pins on the Nano can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 KOhms. In addition, some pins have specialized functions:

* Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the FTDI USB-to-TTL Serial chip.
* External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.
* PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analogWrite() function.
* SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.
* LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Nano has 8 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the analogReference() function. Additionally, some pins have specialized functionality:

* I2C: A4 (SDA) and A5 (SCL). Support I2C (TWI) communication using the Wire library (documentation on the Wiring website).

There are a couple of other pins on the board:

* AREF. Reference voltage for the analog inputs. Used with analogReference().
* Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

The Arduino Nano has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provide UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An FTDI FT232RL on the board channels this serial communication over USB and the FTDI drivers (included with the Arduino software) provide a virtual com port to software on the computer. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the FTDI chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A SoftwareSerial.h library allows for serial communication on any of the Nano's digital pins.

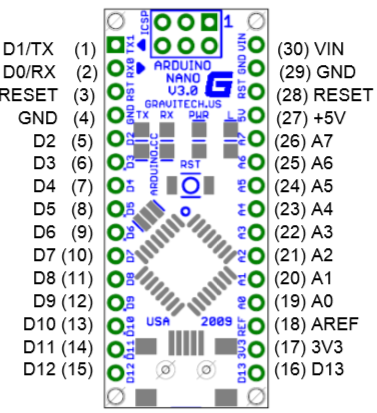
The ATmega168 and ATmega328 also support I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the documentation for details. To use the SPI communication.

**Power Source:**

The Arduino Nano can be powered via the mini-B USB connection, 6-20V unregulated external power supply (pin 30), or 5V regulated external power supply (pin 27). The power source is automatically selected to the highest voltage source.

**Features:**

|  |  |
| --- | --- |
| Operating Voltage (logic level) | 5 V |
| Input Voltage (recommended) | 7-12 V |
| Input Voltage (limits) | 6-20 V |
| Digital I/O Pins | 14 (of which 6 provide PWM output) |
| Analog Input Pins | 8 |
| DC Current per I/O Pin | 40 mA |
| Flash Memory | 32 KB (of which 2KB used by bootloader) |
| SRAM | 2 KB |
| EEPROM | 1 KB |
| Clock Speed | 16 MHz |



**Pin Description:-**

|  |  |  |  |
| --- | --- | --- | --- |
| **Pin No.** | **Name** | **Type** | **Description** |
| 1-2, 5-16 | D0-D13 | I/O | Digital input/output port 0 to 13 |
| 3, 28 | RESET | Input | Reset (active low) |
| 4, 29 | GND | PWR | Supply ground |
| 17 | 3V3 | Output | +3.3V output (from FTDI) |
| 18 | AREF | Input | ADC reference |
| 19-26 | A0-A7 | Input | Analog input channel 0 to 7 |
| 27 | +5V | Output or Input | +5V output (from on-board regulator)  or +5V (input from external power supply) |
| 30 | VIN | PWR | Supply voltage |

* + 1. **Accelerometer:**

An accelerometer is a device that measures proper acceleration; proper acceleration is not the same as coordinate acceleration (rate of change of velocity). They measure in meters per second squared (m/s2) or in G-forces (g). A single G-force for us here on planet Earth is equivalent to 9.8 m/s2, but this does vary slightly with elevation (and will be a different value on different planets due to variations in gravitational pull). Accelerometers are useful for sensing vibrations in systems or for orientation applications.

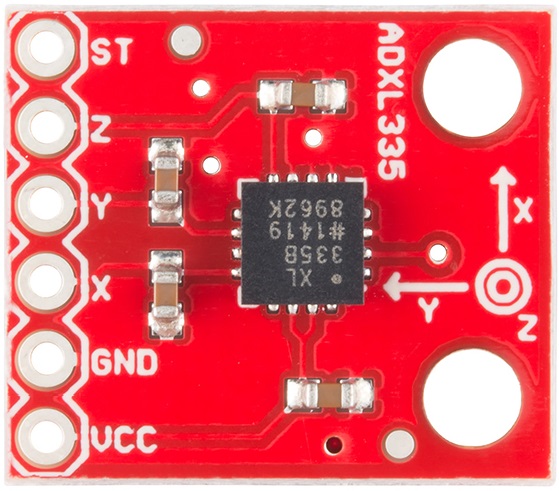
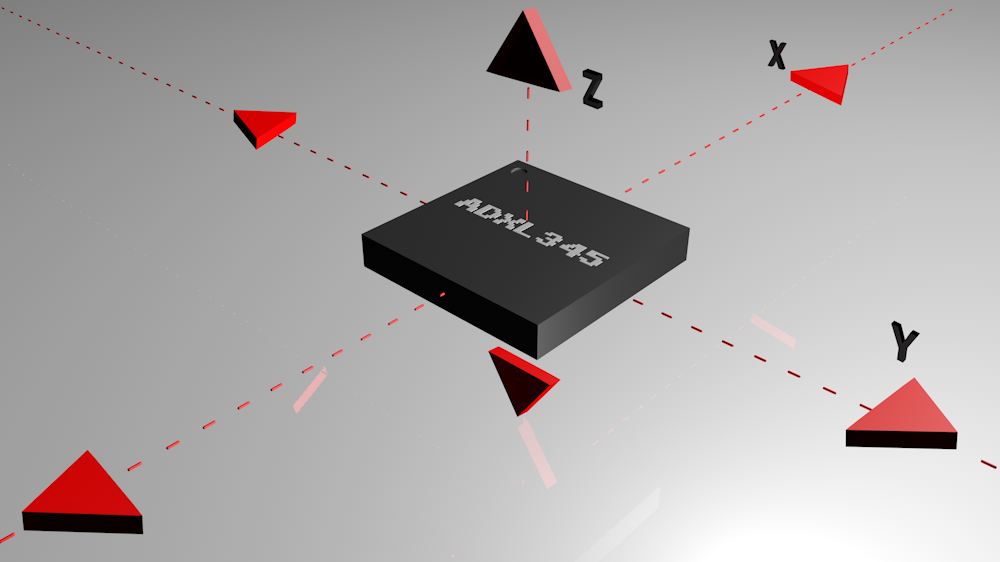


Fig:- 3-Axis analog Accelerometer Fig:- 3-Axis digital Accelerometer

**Range of Accelerometer:**

Most accelerometers will have a selectable range of forces they can measure. These ranges can vary from ±1g up to ±250g. Typically, the smaller the range, the more sensitive the readings will be from the accelerometer. For example, to measure small vibrations on a table top, using a small-range accelerometer will provide more detailed data than using a 250g range (which is more suited for rockets).



**Communication Interface:**

Accelerometers will communicate over an analog, digital, or pulse-width modulated connection interface.

* Accelerometers with an analog interface show accelerations through varying voltage levels. These values generally fluctuate between ground and the supply voltage level. An ADC on a microcontroller can then be used to read this value. These are generally less expensive than digital accelerometers.
* Accelerometers with a digital interface can either communicate over SPI or I2C communication protocols. These tend to have more functionality and be less susceptible to noise than analog accelerometers.
* Accelerometers that output data over pulse-width modulation (PWM) output square waves with a known period, but a duty cycle that varies with changes in acceleration.

**Power:**

Accelerometers are generally low-power devices. The required current typically falls in the micro (µ) or milli-amp range, with a supply voltage of 5V or less. The current consumption can vary depending on the settings (e.g., power saving mode versus standard operating mode). These different modes can make accelerometers well suited for battery powered applications.

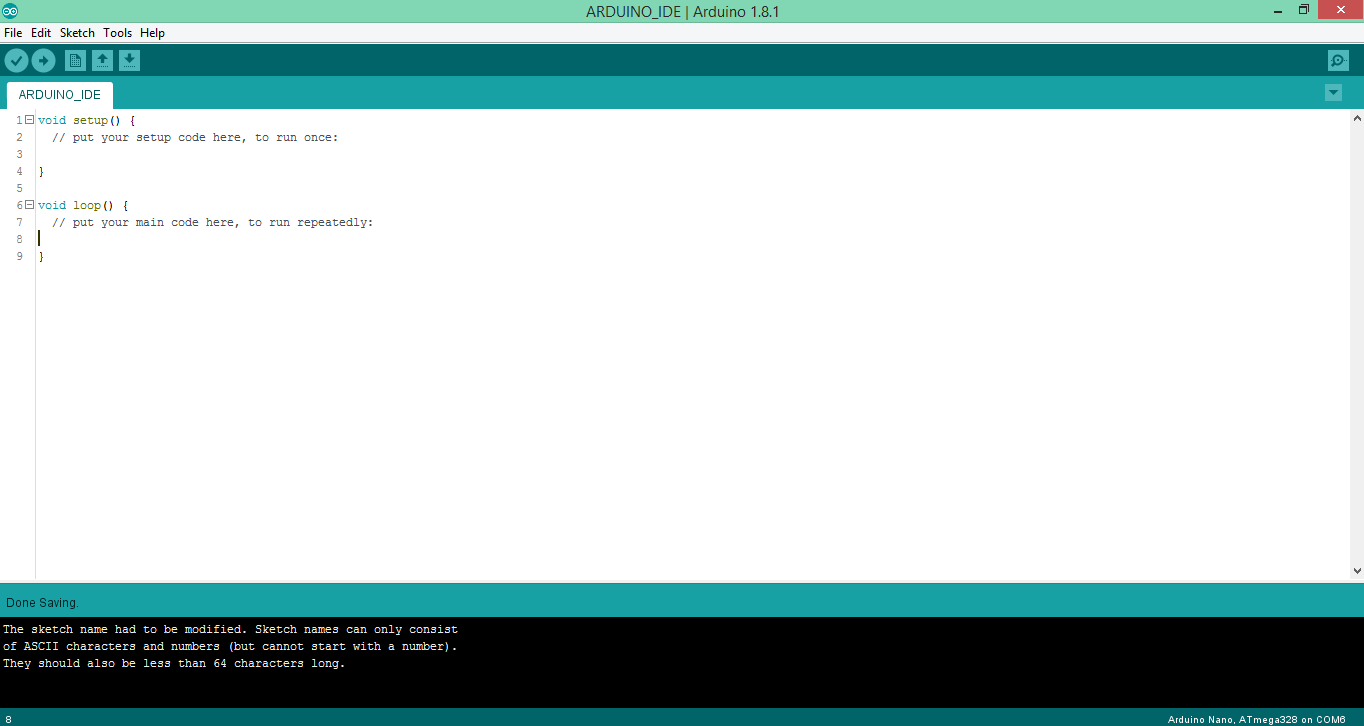
* + 1. **Wi-Fi module ESP8266:-**

The ESP8266 Wi-Fi Module is a self-contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to your Wi-Fi network. The ESP8266 is capable of either hosting an application or offloading all Wi-Fi networking functions from another application processor. Each ESP8266 module comes pre-programmed with an AT command set firmware, meaning, you can simply hook this up to your Arduino device and get about as much Wi-Fi-ability as a Wi-Fi Shield offers (and that’s just out of the box)! The ESP8266 module is an extremely cost effective board with a huge, and ever growing, community.

This module has a powerful enough on-board processing and storage capability that allows it to be integrated with the sensors and other application specific devices through its GPIOs with minimal development up-front and minimal loading during runtime. Its high degree of on-chip integration allows for minimal external circuitry, including the front-end module, is designed to occupy minimal PCB area. The ESP8266 supports APSD for VoIP applications and Bluetooth co-existence interfaces, it contains a self-calibrated RF allowing it to work under all operating conditions, and requires no external RF parts.

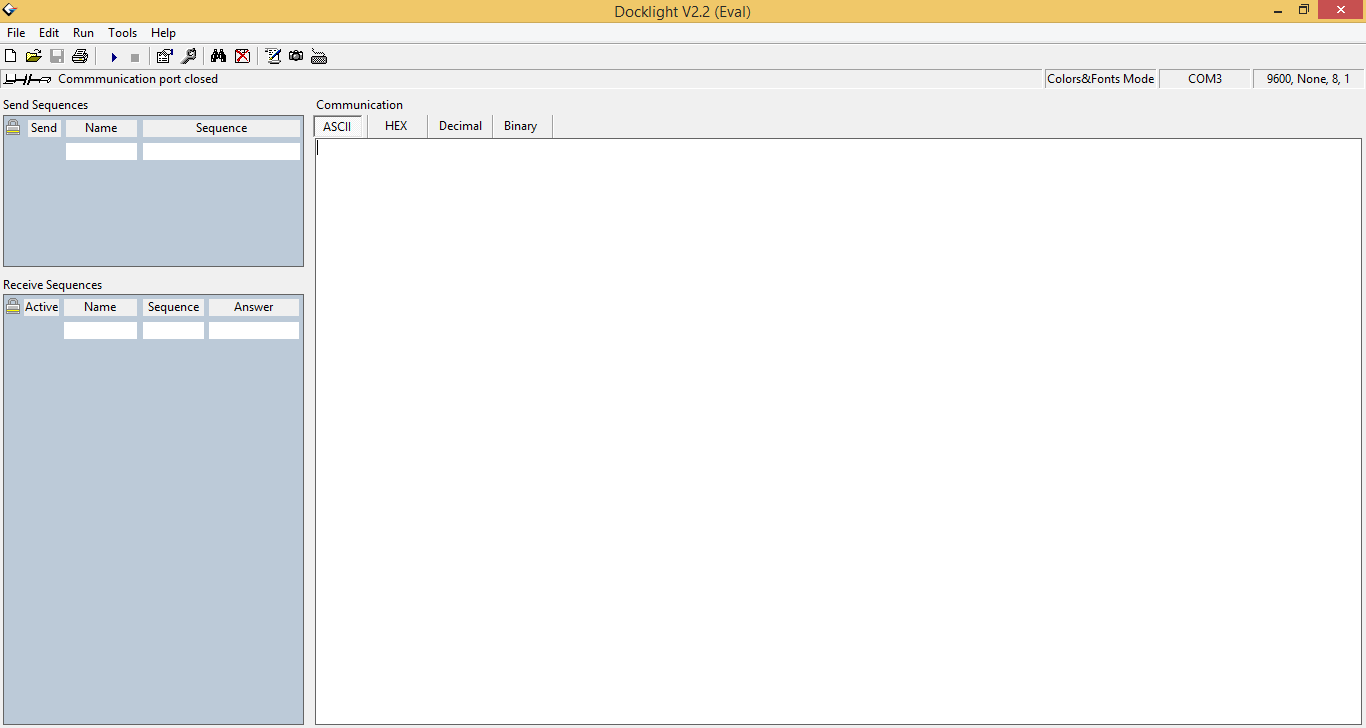
There is an almost limitless fountain of information available for the ESP8266, all of which has been provided by amazing community support. In the Documents section below you will find many resources to aid you in using the ESP8266, even instructions on how to transforming this module into an IoT (Internet of Things) solution, but the ESP8266 Module is not capable of 5-3V logic shifting and will require an external Logic Level Converter. Please do not power it directly from 5V.

* 1. **SOFTWARE USED:**
     1. **Arduino IDE (Integrated Development Environment):**



The Arduino Software (IDE) allows us to write programs and upload them to the board. The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software. This software can be used with any Arduino board.

* + 1. **Docklight Serial monitor**



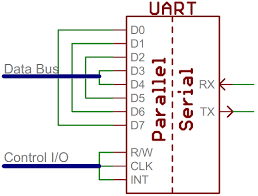
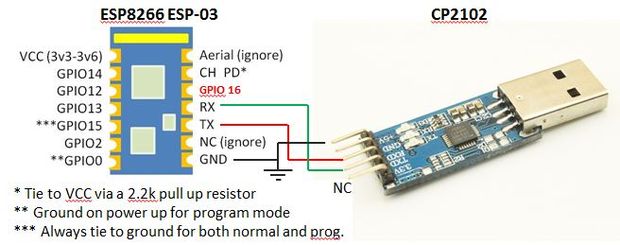
1. **UART COMMUNICATION WITH Wi-Fi ESP 8266**

The universal asynchronous receiver/transmitter (UART) takes bytes of data and transmits the individual bits in a sequential fashion. At the destination, a second UART re-assembles the bits into complete bytes. Each UART contains a shift register which is the fundamental method of conversion between serial and parallel forms. Serial transmission of digital information (bits) through a single wire or other medium is less costly than parallel transmission through multiple wires. UARTs are commonly used in conjunction with communication standards such as TIA. UART is usually an individual (or part of an) integrated circuit (IC) used for serial communication over a computer or peripheral device serial port. UARTs are now commonly included in microcontrollers. A dual UART, or DUART, combines two UARTs into a single chip.

The ESP8266 is a low cost Serial-to-Wi-Fi module that interfaces nicely to any microcontroller. However, a word of caution -- it is highly undocumented (primary reason for writing this document), and more importantly, it is frequently updated and not backward compatible. A good example is how newer versions use 9600 baud rate, while older versions used 57600-115200 baud rates.

**Usage**

First, it is important to understand how the board works. The ESP8266 has a full TCP/UDP stack support. It can also be easily configured as a web server. The module accepts commands via a simple serial interface. It then responds back with the operation's outcome (assuming everything is running correctly). Also, once the device is connected and is set to accept connections, it will send unsolicited messages whenever a new connection or a new request is issued.



**WIFI AT COMMUNICATION**

Wi-Fi AT command ESP8266 offers a complete and self-contained Wi-Fi networking solution, allowing it to either host the application or to offload all Wi-Fi networking functions from another application processor. When ESP8266 hosts the application, and when it is the only application processor in the device, it is able to boot up directly from an external flash. It has integrated cache to improve the performance of the system in such applications, and to minimize the memory requirements. Alternately, serving as a Wi-Fi adapter, wireless internet access can be added to any microcontroller-based design with simple connectivity through UART interface or the CPU AHB bridge interface.ESP8266 on-board processing and storage capabilities allow it to be integrated with the sensors and other application specific devices through its GPIOs with minimal development up-front and minimal loading during runtime. With its high degree of on-chip integration, which includes the antenna switch, power management converters, it requires minimal external circuitry, and the entire solution, including front-end module, is designed to occupy minimal PCB area.

Sophisticated system-level features include fast sleep/wake context switching for energy-efficient VoIP, adaptive radio biasing for low-power operation, advance signal processing, and spur cancellation and radio co-existence features for common cellular, Bluetooth, DDR, LVDS, LCD interference mitigation. A really cheap and easy way to connect any small microcontroller platform (for example Arduino) wirelessly to Internet. The ESP8266 is now one of the leading platforms for the Internet of Things. It’s super cheap, and super easy to work with. This is a serial module with a built-in TCP/IP stack, so you can use it standalone. We can use AT commands to connect with Wi-Fi networks and open TCP connections without need to have TCP/IP stack running in your own microcontroller: You can simply connect any microcontroller to ESP module and start pushing data up to internet.



FUNCTION AT COMMAND RESPONSE

Working AT OK

Restart AT+RST OK

Firmware version AT+GMR AT+GMR 0018000902 OK

WiFi Mode AT+CWMODE Query STA AP

**WIFI P2P (PEER TO PEER) COMMUNICATION**

Wi-Fi peer-to-peer (P2P) allows Android 4.0 (API level 14) or later devices with the appropriate hardware to connect directly to each other via Wi-Fi without an intermediate access point (Android's Wi-Fi P2P framework complies with the Wi-Fi Alliance's Wi-Fi Direct™ certification program). Using these APIs, you can discover and connect to other devices when each device supports Wi-Fi P2P, then communicate over a speedy connection across distances much longer than a Bluetooth connection. This is useful for applications that share data among users, such as a multiplayer game or a photo sharing application.

The Wi-Fi P2P APIs consist of the following main parts:

* Methods that allow you to discover, request, and connect to peers are defined in the Wi-Fi P2P class.
* Listeners that allow you to be notified of the success or failure of Wi-Fi P2P method calls. When calling Wi-Fi P2P methods, each method can receive a specific listener passed in as a parameter.
* Intents that notify you of specific events detected by the Wi-Fi P2P framework, such as a dropped connection or a newly discovered peer.

Wi-Fi Direct negotiates the link with a system that assigns each device a limited wireless access point. The "pairing" of Wi-Fi Direct devices can be set up to require the proximity of a near field communication signal, or a button press on one or all the devices.

**COST ANALYSIS**

**FUTURE SCOPE**

1. Driver's safety warning system
2. Deceleration indicating system
3. Vehicle longitudinal control and collision avoidance system for an automated highway system
4. Method, apparatus and system for transmitting and receiving data in a moving linear chain
5. Deceleration magnitude detecting and signalling device
6. Method and apparatus for automatic vehicle event detection, characterization and reporting
7. Process and device for indicating braking power or delay in cars
8. Motor vehicle early warning system
9. Panic stop, deceleration warning system
10. Vehicle collision warning system
11. Automatic following travel system
12. Obstruction detection method for vehicle
13. Inter vehicle communication system
14. Systems and methods for insurance based on monitored characteristics of an autonomous drive mode selection system

**CONCLUSION**

The system which is the design and construction of an anti-collision system for vehicles was designed considering some factors such as economy, availability of components and research materials, efficiency, compatibility, portability and also durability. The performance of the system after test met design specifications. The general operation of the system and performance is dependent on the presence of two moving cars as they get closer to each other. However, it should be stated here that the system was aimed at fabricating prototype, a replica of the actual thing. It is economically viable to undertake certain system this way since testing would not cost so much. Any desire to implement this design into a vehicle would require a laser detector. The problem of power supply would not arise due to the amount of battery power from the car battery. Also the operation of the system is dependent on how well the soldering is done, and the positioning of the components on the Vero board. The Wi-Fi P2P were make away from the power supply stage to prevent heat radiation which, might occur and affect the performance of the entire system. The construction was done in such a way that it makes maintenance and repairs an easy task and affordable for the user should there be any system breakdown. All components were soldered on one Vero-board which makes troubleshooting easier. In general, the system was designed, and the real time implementation done with a photo-type of the model. It leads to decreasing of accident rate.

**REFRENCES**

1. Zungeru, A. M. et al., (2012). Design and Implementation of a Low Cost Digital Bus Passenger Counter. Innovative Systems Design and Engineering,
2. Zungeru, A. M. et al., (2012). Design and Implementation of a Short Message Service Based Remote Controller. Computer Engineering and Intelligent systems,
3. Theraja, B.L., & Theraja, A.K. (1999). A textbook of electrical technology, S. Chand and company, New delhi, India.
4. Electronics, (2012). How stuffs work: How LIDAR Work. Online available at: http://electronics.howstuffworks.com/lidar.htm