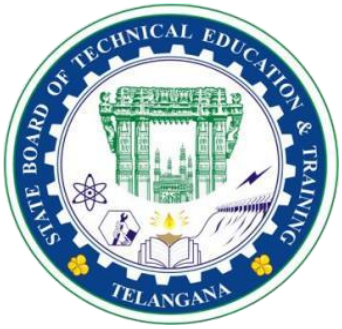
PROJECT REPORT

ON

SMART POLE - BE SMART AND SAFE



SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARDS OF

DIPLOMA IN COMPUTER ENGINEERING

Submitted by

K.GANGADHAR 21004-CS-100

Under the guidance of

SRI. V. RAMA KRISHNA, M.Tech.

DEPARTMENT OF COMPUTER ENGINEERING

GOVERMNMENT POLYTECHNIC

WARANGAL - 506007 2021-2024

DEPARTMENT OF COMPUTER ENGINEERING

GOVERMNMENT POLYTECHNIC, WARANGAL

TELANGANA STATE - 506007

CERTIFICATE

This is to certify that this is a bonafide record of the project work,entitled "**SMART POLE**" carried out by

K.GANGADHAR 21004-CS-100

Final year DCSE submitted in partial fulfillment of the requirement for the award of *Diploma in Computer Engineering* of Telangana State Board of Technical Education and Training during the academic session 2021-2024

PROJECT GUIDE

Sri. V. Rama krishna, M.Tech.

EXTERNAL EXAMINER

HEAD OF DEPARTMENT

Sri. A. Rajeshwar Rao, M.Tech.

PRINCIPAL

Dr. Byri Prabhakar

M.Com,M.Phil,LL.B,M.A,Ph.D.

ACKNOWLEDGEMENT

The success accomplishnment in this project would not have been possible,by timely help and guidence rendered by many people.We/I wish to express our/my sincere and heartful gratitude to all those who have healped and guided us/me for the completion of this project.

We/I earnestly think Dr. BYRI PRABHAKAR SIR PRINCIPAL OF GPT WARANGAL, for giving timely cooperation and taking necessary action throughout of our/my project.

We/I express our/my sincere thanks and gratitude to SRI.A RAJESHWAR RAO SIR, MISTER HOD,DEPARTMENT OF CME,GPT WARANGAL, for this valuable help and encouragement throughout the project.

We/I express our/my sincere thanks to the SRI.V.RAMA KRISHNA Senior Lecturer in Computer Engineering for this valuable guidance, provoking discussions, suggestions and sharing his valuable experties throughout the project work.

We would also thank all the staff of the department of CSE who have helped us/me directly or indirectly for the successful completion of the project.

Submitted by

K.GANGADHAR **21004-CS-100**

INDEX

1.INTRODUCTION

PROJECT PROFILE AND SCOPE

This project is introduces a novel in approach to enhancing women's safety through a smart technology known as SMART POLE.

Smart Poles are upgraded street lamps equipped with advanced features like responsive lighting, surveillance cameras, and improved connectivity. They enhance urban living by contributing to traffic management, providing parking assistance, and offering public information through digital displays. Overall, Smart Poles aim to create safer, more connected, and environmentally conscious urban environments through the integration of smart technologies.

AIM OF THE PROJECT

The "Smart Pole" project aims to create a comprehensive system that leverages smart poles equipped with advanced technology to enhance public safety, particularly for individuals facing threats or emergencies. The project integrates various features, including distress button activation, voice commands, and audio/video confirmation, to provide a holistic solution for rapid response and assistance.

BENEFITS OF PROJECT

- Enhanced public safety through quick and efficient distress alert mechanisms.
- Empowerment of users to seek help in threatening situations.
- Real-time communication facilitates faster response from emergency services.
- Utilization of renewable energy sources contributes to sustainability.

SMART POLE

The Smart Pole Project is a groundbreaking initiative aiming to redefine public safety through the integration of intelligent technology into conventional street lights. Each street light is equipped with a user-friendly button that, when pressed, activates a loud siren and simultaneously sends the precise location of the pole to the nearest police station. This innovative approach seeks to revolutionize emergency response systems, providing a swift and effective means of assistance in critical situations.

IMPORTANCE OF SMART POLE

A smart pole with a buzzer, siren, and light is important because it helps people quickly get help if they're in trouble. If someone is attacked, they can activate the alarm, making a loud noise and turning on a bright light to attract attention. At the same time, a message is sent to the nearby police station, so they know there's a problem and can respond quickly. This makes public spaces safer by giving individuals an easy and visible way to call for help in emergencies.

2. PROJECT ANALYSIS

2.1. Introduction

The Smartpole project represents an innovative venture at the intersection of urban infrastructure and technology. Aimed at enhancing city living, this analysis delves into the key facets of the Smartpole initiative. From its technological features to its potential impact on urban environments, we will explore the intricacies that make this project a noteworthy advancement. Join us in unraveling the future implications and benefits woven into the fabric of the Smartpole project.

2.2. System Design:

Components Used:

1. Buzzer:

A buzzer is integrated into the smart pole to create a loud and attention-grabbing sound when activated.

2. Siren:

The siren is employed to enhance the alert signal, ensuring it's audible over a distance.

3. Lighting System:

Bright RED LED lights are incorporated to illuminate the area when the alarm is triggered, attracting attention.

4. Microcontroller (Arduino):

An Arduino microcontroller serves as the brain of the system, managing the activation of the buzzer, siren, and lighting components.

5. GPS Module:

A GPS module is connected to the Arduino, providing real-time location data when the emergency button is pressed.

6. Communication Module:

A communication module, possibly GSM or similar technology, facilitates the transmission of emergency messages to the nearby police station.

2.3. System Workflow:

1.Emergency Button Press:

- When the emergency button is pressed, it triggers the Arduino microcontroller to initiate the emergency response sequence.

2. Buzzer Activation:

- The buzzer is activated immediately, creating a loud noise to attract attention and alert people in the vicinity.

3. Siren and Light Activation:

- Simultaneously, the siren is activated to enhance the alert, and bright LED lights are turned on to increase visibility in the surrounding area.

4. GPS Location Retrieval:

- The Arduino communicates with the GPS module to retrieve the real-time location of the smart pole.

5. Emergency Message Generation:

- A message is generated, including the location information, indicating an emergency.

6. Communication with Police Station:

- The communication module sends the emergency message to the nearby police station, alerting them to the situation.

7.Emergency Response:

- With the received information, the police station can quickly dispatch assistance to the precise location of the smart pole.

2.4. Modules Used in the Project

The system after careful analysis has been identified to the presented with the following modules:

- **GPS**
- **GSM**

1. GPS Module:

GPS is everywhere! You have probably used and benefitted from GPS. They are found in most of the smartphones, many new automobiles, and now even in smartwatches. It helps you to get where you want to go. These tiny devices can give your position and time simultaneously almost anywhere on the planet.

What is GPS:-

GPS (Global Positioning System) is a satellite-based navigation system. It provides time and location-based information to a GPS receiver,located anywhere on or near the earth's surface. GPS works in all weather conditions.

The satellite system consists of a constellation of 24 satellites in six Earth-centered orbital planes, each with four satellites, orbiting at 13,000 miles (20,000 km) above Earth and traveling at a speed of 8,700 mph (14,000 km/h).

While we only need three satellites to produce a location on earth’s surface, a fourth satellite is often used to validate the information from the other three.

The GPS modules have become small over the years. These modules have tiny processors and antenna that receive data sent by the satellites and compute your position and time.

Working of GPS

GPS works through a technique called trilateration. Trilateration is the process of determining your position based on the intersection of spheres. When a receiver receives a signal from one of the satellites, it calculates its distance from the satellite considering a 3-D sphere with the satellite located at the center of the sphere. Once the receiver does the same with 3 other GPS satellites, the receiver then proceeds to find the intersection point of the 3 spheres to calculate it's location. Used to calculate location, velocity, and elevation, **trilateration** collects signals from satellites to output location information.

The GPS module receives a timestamp from each of the visible satellites, along with data on where in the sky each one is located (among other pieces of data). From this information, the

GPS receiver now knows the distance to each satellite in view. **If the GPS receiver's antenna can see at least 4 satellites, it can accurately calculate its position and time.**

These are the main uses of GPS:

1.Location — Determining a position.

2.Navigation — Getting from one location to another.

3.Tracking — Monitoring object or personal movement.

4.Mapping — Creating maps of the world.

Parameters for Selecting the Right GPS Module:

1.GPS Pulse Accuracy

- This is the accuracy of the 1Hz pulse received from the GPS module or chipset.

2. GPS Sensitivity

- The minimum level of the signal that allows a GPS receiver to acquire or track the GPS signal.

3. Acquisition sensitivity

- Minimum level to successfully get GPS signal information while the system powers up the first time.

4. Tracking sensitivity

- Minimum level to maintain location fix once it has been attained

5. Interface Type

- When we use any GPS chipset or GPS module, then over some interface, we need to get the GPS packet information. That interface could be UART, I2C, SPI, etc.

6. Operating Supply Voltage

- This is the voltage over which the GPS module or chipset works.

7. Operating Temperature Range

- If using a GPS module for outdoor applications, then the minimum and maximum operating temperature range shall be checked.

2. GSM Module:

SIM900A Modem is built with Dual Band GSM/GPRS based SIM900A modem from SIMCOM. It works on frequencies 900/ 1800 MHz. SIM900A can search these two bands automatically. The frequency bands can also be set by AT Commands. The baud rate is configurable from 1200-115200 through AT command. The GSM/GPRS Modem is having internal TCP/IP stack to enable you to connect with internet via GPRS. SIM900A is an ultra compact and reliable wireless module. This is a complete GSM/GPRS module in a SMT type and designed with a very powerful single-chip processor integrating AMR926EJ-S core, allowing you to benefit from small dimensions and cost-effective solutions.

Specification

- Dual-Band 900/ 1800 MHz
- GPRS multi-slot class 10/8GPRS mobile station class B
- Compliant to GSM phase 2/2+
- Dimensions: 24*24*3 mm
- Weight: 3.4g
- Control via AT commands (GSM 07.07 ,07.05 and SIMCOM enhanced AT Commands)
- Supply voltage range : 5V

- Low power consumption: 1.5mA (sleep mode)
- Operation temperature: -40°C to +85 °

SIM900A GSM Module Pin Configuration Descripton:

GPIO Pins

The GPIO pins help to perform the simple and advance I/O function. All pins give the maximum output equal to the power supply which is useable to control most of the devices like sensors and other modules. All GPIO pins in SIM900A are:

- GPIO1 – Pin40
- GPIO2 – Pin41
- GPIO3 – Pin42
- GPIO4 – Pin43
- GPIO5 – Pin44
- GPIO6 – Pin47
- GPIO7 – Pin48
- GPIO8 – Pin49
- GPIO9 – Pin50
- GPIO10 – Pin51
- GPIO11 – Pin67
- GPIO12 – Pin68

Status Pins

The module has two status pins which help to indicate two different kinds of status. The first one is the working status of the module and the second for communication status. Net status means either the module is connecting to the network or other network functions, etc. Both these pins can't operate LED directly. They always act with a combination of a transistor.

- STATUS – Pin52
- NIGHTLIGHT – Pin66

SIM900A Display Interface Pins

The device offers a 4 pin display interface with itself. The display isn't necessary, it is only in case of requirement. The use of interface helps to get the visualization with the module and make it an application. All display pins are:

- DISP_DATA – Pin12 – For Display Data
- DISP_CLK – Pin11 – For Clock Input
- DISP_CS – Pin14 – To enable the display
- DISP_D/C – Pin13 – To select between data and command

Serial Port

The UART serial interface uses the two pins for proper data communication, which are RX and TX. Both pins have no independence on any other pins or modules. In SIM900A these pins are available but it also has some other pins for status/indication of data. By

combining these pins, the serial port helps to generate the RS232 connector too. All the serial pins are:

- RXD – Pin10 – To receive the data
- TXD – Pin 9- To send the data
- RTS – Pin8 – To send the request of data transmission
- CTS – Pin7 – To clear the send request
- RI – Pin4 – Ring indicator
- DSR – Pin6 – To indicate that data set ready
- DCD – Pin5 – To indicate data carry detect
- DTR – Pin3 – To indicate data terminal ready

Debug Interface

Debugging helps the developers to debug the module and update its firmware. In this module, there are sperate serial interface pins for debugging. Both pins are:

- DBG_TXD – Pin27 – For Data Transmission
- DBG_RXD – Pin28 – For Data receiving

SIM Interface

As we know that module SIM900A is a GPRS/GSM module. The module is dependent on some devices for some of its features. The most important one is the SIM. The SIM needs to connect with the module for GPRS/GSM functions to fully operate. All the sim interface of the module is:

- SIM_VDD – Pin30 – Power Supply of the SIM
- SIM_DATA – Pin31 – For data output
- SIM_CLK – Pin32 – For clock pulse
- SIM_RST – Pin33 – For reset
- SIM_PRESENCE – Pin34 – To detect the SIM

SIM900A Analog to Digital converter Pins

The module has only a single pin to detect and convert the analog signal to digital for SIM900A. The voltage range on the ADC pin is from 0 to 3 only.

- ADC – Pin25

Audio Interface

The audio interface will help to connect the mic and speaker with SIM900A. The connection of Line, Audio and Speaker will help to make the calls through the modules.

- MIC_P – Pin19
- MIC_N – Pin20
- SPK_P – Pin21
- SPK_N – Pin22
- LINEIN_R – Pin23
- LINE_L – Pin24

Control Pin

There is power on pins on the device, which helps to turn it on using external signals. There is two power on pins. The first one is PWRKEY which requires a LOW signal to power on/off the system. To do that, the pins require an input signal for a little bit long time. The second pin is PWRKEY_OUT, which gets short with the PWRKEY pin and turn on/off the device.

- PWRKEY – Pin1
- PWRKEY_OUT – Pin2

Reset pins

The device has an external LOW input signal reset pin to reset the device with the use of an external signal.

- NRESET – Pin16

SIM900A GSM Module RF Antenna

To extend the range of the SIM900A the antenna pin needs to connect with an external wire.

The official antenna is also available for the module.

- **RF_ANT – Pin60**

Power Pins

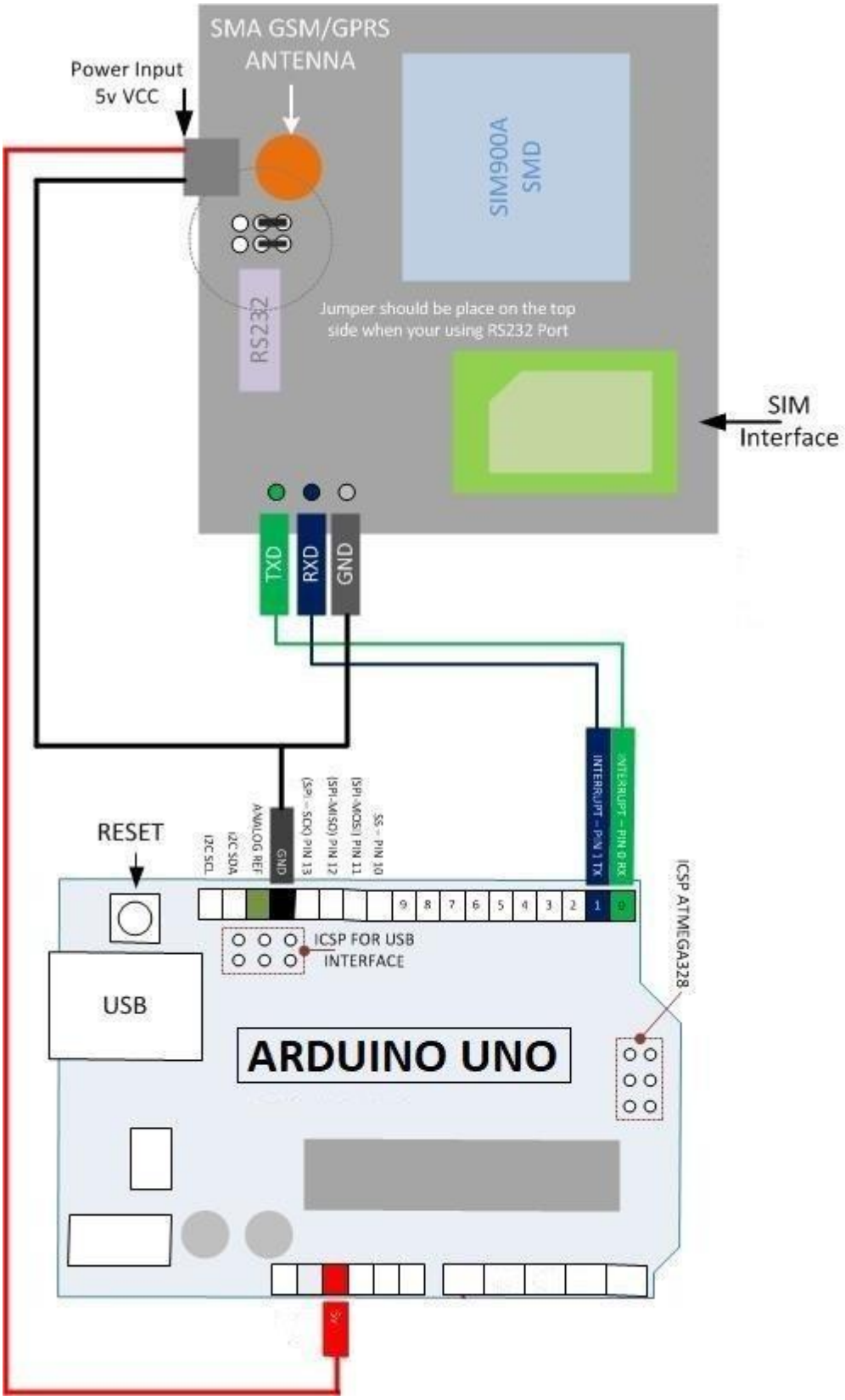
The module SIM900A has multiple types of power pin. Some works as input and some as output. The most important one to understand is VRTC, which acts as a backup for the internal RTC of the device. All power and ground pins of the module are:

- VBAT(Input) – Pin55, Pin56, Pin57
- VRTC (Input/Output) – Pin26
- VDD_EXT(OUTPUT) – Pin15
- GND – Pin17, Pin18, Pin29, Pin39, Pin45, Pin46, Pin53, Pin54, Pin58, Pin59, Pin61, Pin62, Pin63, Pin64, Pin65

Applications

- The module is the best application to design a graphic for Voice call and SMS application.
- Some IoT applications, mostly in an emergency have the module.
- The location trancing system also uses SIM900A.
- SIM900A can use for mobile communication.

Arduino Interfacing Example



2.5. System Specification

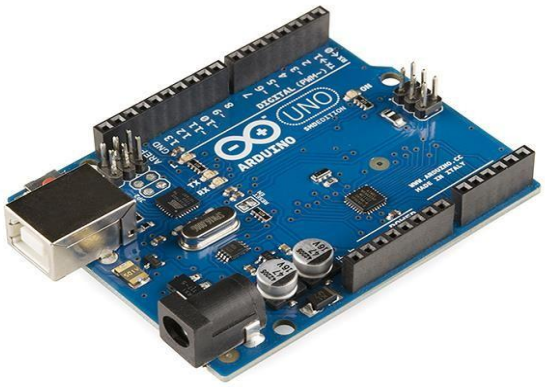
2.4.1. SOFTWARE COMPONENTS:

- Arduino IDE
- Internet Explorer 6.0
- Google Chrome
- MS Word

2.4.2. HARDWARE COMPONETS:

- Arduino Uno
- Data transferring cabel
- GSM 900a
- 2amp.12V Adapter
- GPS Module
- Button
- Buzzer
- Led

2.6. Arduino Uno



The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller (MCU) and developed by Arduino.cc and initially released in 2010. The microcontroller board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits. The board has 14 digital I/O pins (six capable of PWM output), 6 analog I/O pins, and is programmable with the Arduino IDE (Integrated Development Environment), via a type B USB cable. It can be powered by a USB cable or a barrel connector that accepts voltages between 7 and 20 volts, such as a rectangular 9-volt battery. It has the same microcontroller as the Arduino Nano board, and the same headers as the Leonardo board.The hardware reference design is distributed under a Creative

Commons Attribution Share-Alike 2.5 license and is available on the Arduino website. Layout and production files for some versions of the hardware are also available.

The word "uno" means "one" in Italian and was chosen to mark a major redesign of the Arduino hardware and software. The Uno board was the successor of the Duemilanove release and was the 9th version in a series of USB-based Arduino boards. Version 1.0 of the Arduino IDE for the Arduino Uno board has now evolved to newer releases.The ATmega328 on the board comes preprogrammed with a bootloader that allows uploading new code to it without the use of an external hardware programmer.

History

Arduino RS232 Serial board - a predecessor with an ATmega8

The Arduino project started at the Interaction Design Institute Ivrea (IDII) in Ivrea, Italy. At that time, the students used a BASIC Stamp microcontroller, at a cost that was a considerable expense for many students. In 2003, Hernando Barragán created the development platform Wiring as a Master's thesis project at IDII, under the supervision of Massimo Banzi and Casey Reas, who are known for work on the Processing language. The project goal was to create simple, low-cost tools for creating digital projects by nonengineers. The Wiring platform consisted of a printed circuit board (PCB) with an ATmega168 microcontroller, an IDE based on Processing, and library functions to easily program the microcontroller. In 2003, Massimo Banzi, with David Mellis, another IDII student, and David Cuartielles, added support for the cheaper ATmega8 microcontroller to Wiring. But instead of continuing the work on Wiring, they forked the project and renamed it Arduino. Early arduino boards used the FTDI USB-to-UART serial chip and an ATmega168.The Uno differed from all preceding boards by featuring the ATmega328P microcontroller and an ATmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

In June 2023, Arduino released two new flavors of the Uno; R4 Minima and R4 Wifi. These mark a departure from previous boards as they use Renesas RA4M1 ARM Cortex M4 microcontroller, and the R4 Wifi a Espressif ESP32-S3-MINI co-processor. These versions are form factor, pin and power compatible with version R1 to R3, so should be largely be able to be drop in replacements.

Technical specifications

Arduino Uno R3 board with ATmega328P MCU in DIP-28 package

Uno R1 to R3
Microcontroller (MCU):
IC: Microchip ATmega328P (8-bit AVR core)
Clock Speed: 16 MHz on Uno board, though IC is capable of 20 MHz maximum at 5 Volts

Flash Memory: 32 KB, of which 0.5 KB used by the bootloader
SRAM: 2 KB
EEPROM: 1 KB
USART peripherals: 1 (Arduino software default configures USART as a 8N1 UART)
SPI peripherals: 1
I²C peripherals: 1
Operating Voltage: 5 Volts
Digital I/O Pins: 14
PWM Pins: 6 (Pin # 3, 5, 6, 9, 10 and 11)
Analog Input Pins: 6
DC Current per I/O Pin: 20 mA
DC Current for 3.3V Pin: 50 mA
Size: 68.6 mm x 53.4 mm
Weight: 25 g ICSP Header: Yes Power Sources:

USB connector. USB bus specification has a voltage range of 4.75 to 5.25 volts. The official Uno boards have a USB-B connector, but 3rd party boards may have a miniUSB / microUSB / USB-C connector.
5.5mm/2.1mm barrel jack connector. Official Uno boards support 6 to 20 volts, though 7 to 12 volts is recommended. The maximum voltage for 3rd party Uno boards varies between board manufactures because various voltage regulators are used, each having a different maximum input rating. Power into this connector is routed through a series diode before connecting to VIN to protect against accidental reverse voltage situations.
VIN pin on shield header. It has a similar voltage range of the barrel jack. Since this pin doesn't have reverse voltage protection, power can be injected or pulled from this pin. When supplying power into VIN pin, an external series diode is required in case barrel jack is used. When board is powered by barrel jack, power can be pulled out of this pin.

3. PROJECT DESCRIPTION

What is smart pole?

A "smart pole" is a high-tech streetlight or utility pole that goes beyond just providing light. It has features like energy-efficient LED lights, wireless connectivity (like Wi-Fi and 4G/5G), cameras for surveillance, environmental sensors, and sometimes even electric vehicle charging. These poles are designed to make cities more efficient, connected, and safer for residents.

In addition to its primary functions, This smart pole serves as a reliable beacon for emergency situations. With a built-in button for help, individuals can easily trigger assistance when needed. The inclusion of a buzzer ensures that the alert doesn't go unnoticed, drawing immediate attention to the situation. The integration of a powerful light serves as a visual indicator, signaling distress to those in the vicinity. Simultaneously, the smart pole takes prompt action by sending an alert message to the nearest police station, providing vital information about the location and nature of the emergency. This multifaceted approach not only enhances public safety but also streamlines the emergency response process, turning every smart pole into a valuable ally in times of need.

GIRL SAFETY SMART POLES WORKING

- **smart poles for women's and children**

'Smart poles' with an alarm button, which will send a message to the nearest police station and emit a loud sound to alert people around when activated by a woman or child in distress'

- **The smart poles would have surveillance cameras and cameras with fisheye lenses.**

Fisheye lenses: A fisheye lens is an ultra-wide-angle lens that is designed for shooting wide angles generally 180 degrees

Once the alarm button is pressed, the cameras fish-eye lenses will startclicking pictures. "These photos will help the police to identify the culprit by matching with their database as well.

- As soon as the alarm is raised, a police patrol vehicle would also be sent to the spot.

Existing System:

In the current landscape, emergency situations often prompt individualsdialemergency services like 100 for assistance. While street lights are a staple in urban environments, their traditional role is limited to providing illumination. Unfortunately, this setup lacks the immediacy required to rescue people in distress. Calling emergency services relies on

verbal communication, causing delays in response times, and the static nature of street lights does little to actively intervene in emergencies.

Disadvantages of the existing system include delayed response times, dependence on verbal communication, and a lack of proactive measures to address emergencies. The Smart Pole Project emerges as a transformative solution, addressing these limitations by integrating realtime communication and location-sharing capabilities directly into the urban infrastructure.

Example of Uses:

1. **Health Emergency Response:**

   Imagine an elderly person experiencing a sudden heart attack. With the Smart Pole nearby, pressing the button not only activates the siren but also sends the location to emergency services. This swift response facilitates timely medical assistance, potentially saving a life.

2. **Women's Safety Enhancement:**

   In a scenario where a woman perceives a threat, a quick press of the Smart Pole button triggers the siren and alerts the police to her exact location. The integration of a camera in future iterations could also capture evidence crucial for law enforcement.

PRO's of Smart Pole with salient features:

1. **Enhanced Public Safety: The primary benefit is the immediate and accessible means for individuals to seek help in emergency situations, contributing to a safer urban environment.**
2. Quick Response Times: Automated alert systems ensure that emergency services, such as the police, receive timely information about the incident, allowing for quicker response times and potentially saving lives.
3. Improved Visibility: The use of lights and audible signals enhances the visibility of emergency situations, alerting both passersby and law enforcement to the location of the incident.
4. Versatility: Smart poles are versatile, adaptable to various urban needs. They can integrate additional features like environmental sensors, surveillance cameras, and digital displays, making them valuable components of smart city infrastructure.
5. Efficient Resource Allocation: By automating the alert process, emergency services can allocate resources more efficiently, responding to incidents with the right personnel and equipment.

6. Community Engagement: Smart poles can foster a sense of community engagement by providing residents with tools to actively contribute to public safety. The ease of access to help buttons empowers individuals to take action when necessary.
7. Technological Integration: These poles contribute to the overall integration of technology into urban planning, supporting the development of smart cities and advancing the use of data for better decision-making.
8. Energy Efficiency: Many smart poles use energy-efficient LED lighting, contributing to reduced energy consumption and environmental impact compared to traditional streetlights.
9. Support for Law Enforcement: The integration of surveillance cameras aids law enforcement in monitoring public spaces, investigating incidents, and maintaining overall security.
10. Emergency Preparedness: The presence of smart poles enhances a city's overall emergency preparedness and resilience, providing a distributed network of assistance points across urban areas.

SOURCE CODE:

```
#include <SoftwareSerial.h>
#include <TinyGPS.h>

int state = 1;
const int btnPin = 9;
const int ledPin = 10;
float gpslat, gpslon;

TinyGPS gps;
SoftwareSerial sgps(4, 5);
SoftwareSerial sgsm(2, 3);

void setup()
{
  Serial.begin(9600);
  sgsm.begin(9600);
  sgps.begin(9600);
  pinMode(10,OUTPUT);
  pinMode(9, INPUT);
}
void loop()
{
  sgps.listen();
  while (sgps.available())
  {
    int c = sgps.read();
    if(gps.encode(c))
    {
      gps.f_get_position(&gpslat, &gpslon);
    }
  }
  if (digitalRead(btnPin) == HIGH)
  {
    digitalWrite(ledPin,HIGH);
    delay(3000);
    digitalWrite(ledPin,LOW);
    sgsm.listen();
    sgsm.print("\r");
    delay(1000);
    sgsm.print("AT+CMGF=1\r");
    delay(1000);
    /*Replace XXXXXXXXXX to 10 digit mobile number & ZZ to 2
digit country code*/
```

```
    sgsm.print("AT+CMGS=\"+ZZXXXXXXXXXX\"\r");
    delay(1000);
    sgsm.print("https://www.google.com/maps/?q=");
    sgsm.print("ALERT HELP NEEDED");
    sgsm.print(gpslat, 6);
    sgsm.print(",");
    sgsm.print(gpslon, 6);
    delay(1000);
    sgsm.write(0x1A);
    delay(1000);
    state=1;
    Serial.println(gpslat);
    Serial.println(gpslon);
    digitalWrite(btnPin, LOW);
  }
  if (digitalRead(btnPin) == LOW)
  {
    state = 0;
  }
  delay(100);
}
```

4. SELECTED SOFTWARE

C Language

C is a general-purpose computer programming language. It was created in the 1970s by Dennis Ritchie, and remains very widely used and influential. By design, C's features cleanly reflect the capabilities of the targeted CPUs. It has found lasting use in operating systems, device drivers, and protocol stacks, but its use in application software has been decreasing.C is commonly used on computer architectures that range from the largest supercomputers to the smallest microcontrollers and embedded systems.

A successor to the programming language B, C was originally developed at Bell Labs by Ritchie between 1972 and 1973 to construct utilities running on Unix. It was applied to reimplementing the kernel of the Unix operating system.During the 1980s, C gradually gained popularity. It has become one of the most widely used programming languages,[9][10] with C compilers available for practically all modern computer architectures and operating systems. The book *The C Programming Language*, co-authored by the original language designer, served for many years as the *de facto* standard for the language. C has been standardized since 1989 by the American National Standards Institute (ANSI) and the International Organization for Standardization (ISO).

C is an imperative procedural language, supporting structured programming, lexical variable scope, and recursion, with a static type system. It was designed to be compiled to provide low-level access to memory and language constructs that map efficiently to machine instructions, all with minimal runtime support. Despite its low-level capabilities, the language was designed to encourage cross-platform programming. A standards-compliant C program written with portability in mind can be compiled for a wide variety of computer platforms and operating systems with few changes to its source code.

Since 2000, C has consistently ranked among the top two languages in the TIOBE index, a measure of the popularity of programming languages.

HISTORY OF C:

C is a general-purpose, procedural programming language. It was developed by Dennis Ritchie at Bell Labs in the early 1970s as an augmented version of Ken Thompson's B. Another Bell Labs employee, Brian Kernighan, had written the first C tutorial, and he persuaded Ritchie to coauthor a book on the language. Kernighan would write most of the book's "expository" material, and Ritchie's reference manual became its appendices.

The first edition, published February 22, 1978, was the first widely available book on the C programming language.

C was originally developed to be used in the Unix operating system, but it has since become one of the most widely used programming languages in the world. It is used in a wide variety of applications, including operating systems, embedded systems, compilers, and highperformance computing.

C is a powerful and flexible language, but it can also be difficult to learn and use. It is important to have a good understanding of the basics of computer science before attempting to learn C.

Here is a timeline of some of the key events in the history of the C language:

- 1972: Dennis Ritchie begins development of C at Bell Labs.
- 1973: C is used to reimplement the Unix kernel.
- 1978: The first edition of The C Programming Language is published.
- 1983: The American National Standards Institute (ANSI) sets up a committee to standardize the C language.
- 1989: The first ANSI C standard is published.
- 1999: The second ANSI C standard is published.
- 2011: The third ANSI C standard is published.

USAGE OF C:

- Operating systems: C is the language of choice for many operating systems, including Linux, macOS, and Unix.
- Embedded systems: C is often used to develop embedded systems, which are small, specialized devices that are typically used in control systems and automation.
- Device drivers: C is also used to develop device drivers, which are software programs that allow devices to communicate with the operating system.
- High-performance applications: C is a good choice for high-performance applications, such as scientific computing and video games.
- Teaching programming: C is a popular language for teaching programming, as it is relatively simple to learn and understand.

C is a powerful and versatile language that can be used for a wide variety of applications.

If you are interested in learning to program, C is a good language to start with.

What can you do with C:

- Develop software
- Create and maintain IoT applications
- Develop compilers
- Develop operating systems

- Develop databases
- Develop internet browsers and their extensions

Develop Software:
C is commonly used for developing system software, application software, and even embedded software. It provides low-level access to memory and hardware, making it suitable for performance-critical applications.

Create and Maintain IoT Applications:
The Internet of Things (IoT) involves connecting devices to the internet to collect and exchange data. C is used in IoT development for its efficiency and ability to interact with hardware at a low level.

Develop Compilers:
C itself is often used to write compilers for other programming languages. Many programming languages, including C++, were initially implemented in C.

Develop Operating Systems:
C has been a primary language for developing operating systems. The UNIX operating system and its variants, like Linux, are written in C. The low-level nature of C allows for close interaction with hardware and efficient system-level programming.

Develop Databases:
While higher-level languages like SQL are commonly used for database interactions, the core engines of some databases may be implemented in C or a mix of C and other languages for performance reasons. C can be used to optimize critical components of database systems.

Develop Internet Browsers and Their Extensions:
Web browsers and extensions often have performance-critical components that are implemented in C for efficiency. C can be used to develop the core rendering engine, networking components, and other critical parts of a browser.

C is also used in applications that require a high degree of calculations, such as MATLAB and Mathematica. This is because C implements algorithms and data structures quickly, which allows for faster computations.

Some features of C include:

- Enumeration: A user-defined data type that consists of constant integers or integers given names by the user.
- Bool: A fundamental data type that can hold one of two values: true or false.
- Arrays: Make code more optimized and cleanby storing multiple elements in a single array.
- Dynamic memory allocation: The process of assigning memory space during execution time.

WHY IS C SO POPULAR?

1. **Simple Code Structure:**

It uses blocks to organize code, making it easy to understand, especially for beginners.

2. **Embedded Programming:**

Widely used in embedded systems, controlling micro-controllers in robotics, computers, and automation.

3. **Foundation for C++ and Java:**

Serves as the basis for C++ and Java, making it easier to learn these widely used languages.

4. **Fast Compilation and Execution:**

Code compiles and executes quickly, without unnecessary processing overheads, making it efficient.

5. **Game Development:**

Used in PC game development for coding small and large tasks.

6. **Mid-Level Language:**

Strikes a balance between low-level and high-level languages, combining simplicity with fast program processing.

Features of C:

C programming language is known for its simplicity, efficiency, and versatility.
Here are some key features of C:

- **Procedural Language:**

C is a procedural programming language, focusing on functions and procedures to structure code.

- **Portable and Cross-Platform:**

Code written in C is often portable and can run on different platforms with minimal or no modification.

- **Efficient and Fast:**

C provides low-level access to memory and hardware, allowing for efficient manipulation and fast execution of programs.

- **Structured Language:**

It supports structured programming, allowing the use of blocks, functions, and modules to organize code logically.

- **Rich Standard Library:**

C comes with a rich standard library that provides a wide range of functions for tasks like I/O operations, string manipulation, and mathematical calculations.

- **Dynamic Memory Allocation:**

C allows dynamic memory allocation and deallocation through functions like malloc() and free(), providing flexibility in memory usage.

- **Pointer Support:**

C supports pointers, allowing direct manipulation of memory addresses and facilitating efficient data structures.

- **Low-Level Features:**

It provides features like bit manipulation, which is useful for tasks requiring precise control over individual bits in data.

- **Extensible:**

C is often used as a foundation for other languages, and it allows the development of libraries and extensions for various purposes.

- **Easy to Learn:**

With a simple syntax and a relatively small set of keywords, C is considered accessible for beginners in programming.

- **Structured Input/Output:**

C supports standard input and output operations through functions like printf() and scanf(), providing a standardized way to interact with users.

LIBRARIES IN C: stdio.h(Standard Input/Output):

- **printf(): Print formatted output to the console.**
- **scanf(): Read formatted input from the console.**
- **getchar(): Read a single character from the console.**
- **putchar(): Write a single character to the console.**

stdlib.h (Standard Library):

- **malloc(): Allocate memory dynamically.**
- **calloc(): Allocate memory for an array and initialize it to zero.**
- **free(): Deallocate dynamically allocated memory.**
- **rand(): Generate a pseudo-random number.**
- **exit(): Terminate the program.**

string.h (String Handling):

- **strlen(): Get the length of a string.**
- **strcpy(): Copy one string to another.**
- **strcat(): Concatenate two strings.**
- **strcmp(): Compare two strings.** **math.h (Mathematics):**
- **sqrt(): Calculate the square root.**
- **sin(), cos(), tan(): Trigonometric functions.**
- **abs(): Calculate the absolute value.**

ctype.h (Character Handling):

- **isalpha(), isdigit(): Check if a character is alphabetic or numeric.**
- **tolower(), toupper(): Convert a character to lowercase or uppercase.**

time.h (Time and Date):

- **time(): Get the current time.**
- **asctime(): Convert time to a string representation.**
- **ctime(): Convert time to a string representation with newline.**
- **strftime(): Format time as per the specified format.**

5. OUTPUT OF PROJECT

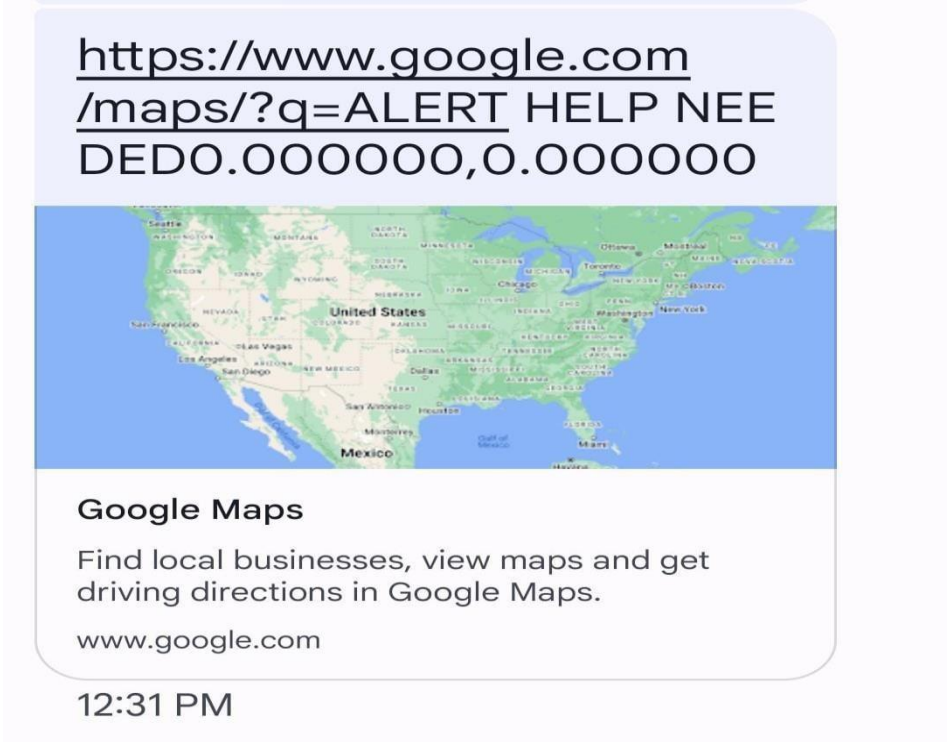


Fig 1.1. Message alert to nearest police station

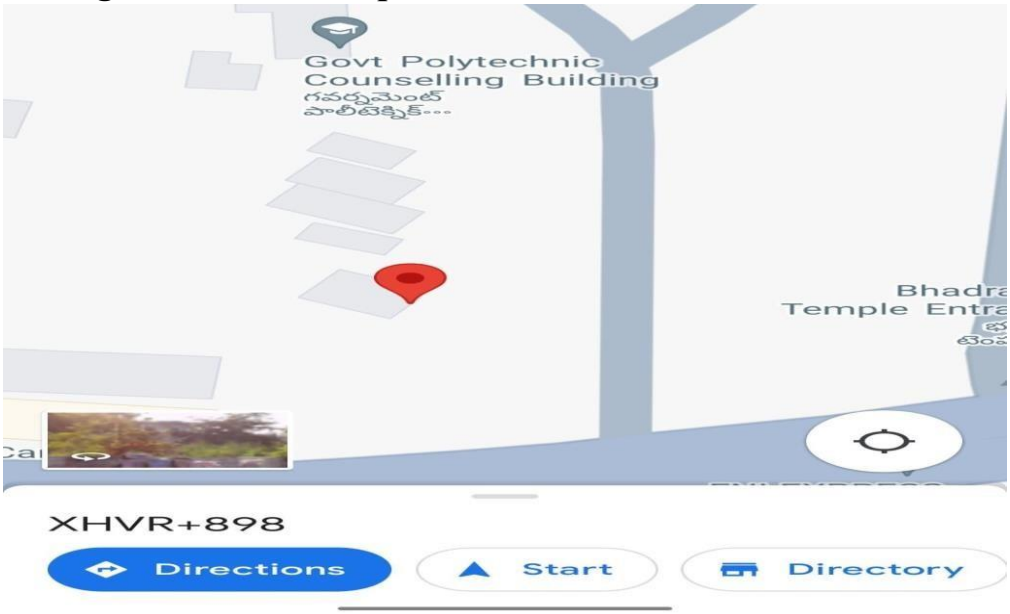


Fig 1.2. Location of the pole

Explanation of output project:

The output screen which are presented in the previous page will give you the emergency

SMS with message indicating "ALERT HELP NEEDED" and also with the coordinates of the location of the pole where the emergency situation occurs.

In Fig 1.1 the location is sent in the form of SMS to the nearest police control room when the emergency button is pressed near the pole.

In Fig 1.2 The location will give you the exact place at which pole the emergency button is pressed and the police team will take charge.

6. CONCLUSION

The Smart Pole Project not only addresses the shortcomings of the existing emergency response system but also pioneers a new era of urban safety. By turning street lights into active participants in emergency interventions, we envision creating a safer and more responsive environment for everyone. As we move forward, the Smart Pole Project stands as a testament to the fusion of technology and public welfare, proving that even the most ordinary elements of our urban landscape can become extraordinary agents of safety and protection.

It's like giving our city lights a superhero upgrade, turning them into active participants in keeping us safe. This project is proof that even everyday things, like street lights, can become powerful tools for our protection. *It's not just about technology; it's about making our cities safer and more responsive for everyone.*

This Smart Pole Project is a small idea with a big impact, showing that even the simplest elements in our city can play a crucial role in our safety and well-being.

7. REFERENCES

1. www.google.com
2. www.w3schools.com