

DESIGN OF

REAL TIME WOMEN PROTECTION DEVICE

**TEAM MEMBERS: UNDER THE GUIDANCE OF :**

**1.G. VIJAYA DURGA (EE) M. PRAVEEN**

**2.CH. NAGA JAYA LAKSHMI (EE ) (FOUNDER AND CEO OF ROBOCOUPLER)**

**3.G. MEGHANA (EE)**

**4.T.J. S. MEGHANA (EE)**

**5.P. JYOTSHNA (EC)**

**6.A. RAMYA (EC)**

* **ABSTRACT:**

Women safety is a very important issue due to rising crimes against women these days. In our country, even though it has super power and an economic development, but still there are many crimes against women. The device is mainly for women who lives in residential homes or hostels. The main purpose of this device is to act as an emergency device for women who are in potential danger of being attacked or any medical related problem. This device also includes the hostel management section for any general help.

* **INTRODUCTION:**

Introduction of our *“Women Protection Device”* is a security device specially designed for women in Hostels especially in emergency situations. It is simple and easy to use which is installed in the hostels.

It is a safety product designed to keep you and your friends safe 24/7. The device is mainly for the hostels who won’t allow the usage of mobile phones in the hostel premises and the institutes who won’t allow the mobiles.

It is packed with features for everyday necessities and real emergency situations making it an ultimate tool for all. This user-friendly device can be accessed by anyone who has installed it in their hostels as well as who had our device. Our intention is to provide you with fastest and simplest way to contact your nearest help. The basic approach is to press the required button which is mentioned on the device for our need. After that a message is sent to the respected authorities of that particular domain of the button pressed.

* **CIRCUIT DIAGRAM:**

**NODE MCU**

**LCD DISPLAY**

**SCL**

**GSM MODULE**

**D1 SDA**

**VCC**

**GND**

**D2**

**D3 VCC**

**GND**

**TX RX**

**TX**

**RX**

* **BLOCK DIAGRAM:**

**POWER**

**HOSTEL**

**NODE MCU**

**GSM MODULE**

**LCD DISPLAY**

**POLICE**

**STATION**

**HOSPITAL**

* **WORKING OPERATION:**

The device is introduced with three push buttons for the user interface. The NodeMCU is one of the main parts in this device and it is the connecting part of the input we give and the output we get. The three push buttons are named as HOSTEL, MEDICAL and POLICE STATION respectively. These three push buttons are connected to pins D1, D2, D3 of NodeMCU board. The output is engaged to the 16\*2 LCD display from NodeMCU. If the hostel button is pressed, the information related to that is displayed on LCD. The remaining two buttons will also follow the same process. The RX,TX pins of NodeMCU is connected to the TX,RX pins of GSM module respectively. The VCC and GROUND of the NodeMCU is connected to the VCC and GROUND of GSM module. The controls of is NodeMCU is transmitted to the GSM module where a sim card is installed in that, with the help of the AT commands the message or call is sent to the number we mentioned in the program. The communication between the GSM module and the receiver is done with the help of ANTENNA. The message from the GSM module is send to the respective phone numbers of hostel, medical and police station based on the control we issued.

“**AT+CMGF=1” 🡪**  This command is used for send the message the receiver.

**“ATA”** 🡪 Pick up the call

**“AT+CHUP”🡪** Hangs up the call

* List of components used:
* NodeMCU:



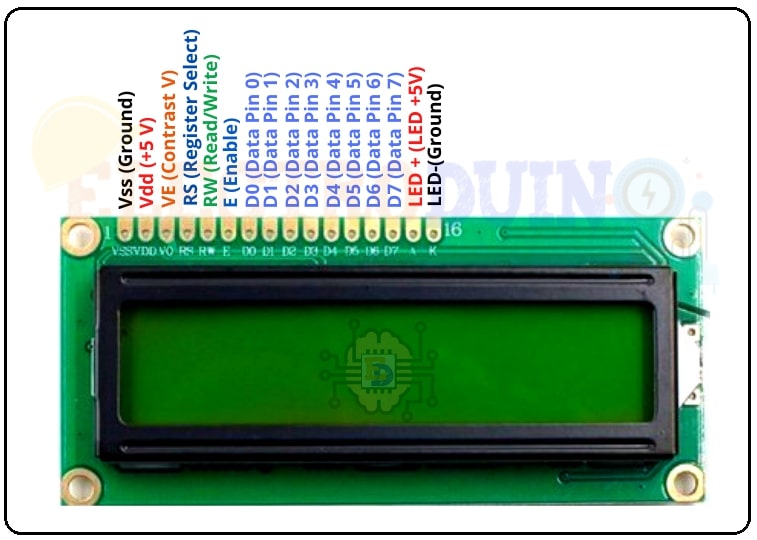
NodeMCU is an open source firmware for which open source prototyping board designs are available. The name “NodeMCU” combines “node” and “MCU” (micro-controller unit). Strictly speaking, the term “NodeMCU” refers to the firmware rather than the associated development kits.

Both the firmware and prototyping board designs are open sources.

The firmware uses the Lua scripting language. The firmware is based on the eLua project and built on the Espressif Non-OS SDK for ESP8266.It uses many open source projects, such as lua-cjson and SPIFFS. Due to resource constraints, users need to select the modules relevant for their project and build a firmware tailored to their needs. Support for the 32-bit ESP32 has also been implemented.

The prototyping hardware typically used is a circuit board functioning as dual in line package (DIP) which integrates a USB controller with a smaller surface mounted board containing the MCU and antenna. The choice of the DIP format allows for easy prototyping on breadboards. The design was initially based on the ESP-12 module of the ESP8266,which is a Wi-Fi SoC integrated with a Tensilica Xtensa LX106 core ,widely used in IoT applications.

* LCD DISPLAY:



16×2 LCD is one kind of electronic device used to display the message and data. The term LCD full form is **Liquid Crystal Display**.  The display is named 16×2 LCD because it has 16 Columns and 2 Rows. it can be displayed (16×2=32) 32 characters in total and each character will be made of 5×8 Pixel Dots. These displays are mainly based on multi-segment light-emitting diodes.  There are a lot of combinations of display available in the market like 8×1, 8×2, 10×2, 16×1, etc. but the 16×2 LCD is widely used. These LCD modules are low cost, and programmer-friendly, therefore, is used in various DIY circuits, devices, and embedded projects.

These 16 x 2 LCD display modules are constant of 16 Columns and 2 Rows. The 1st row of this module has a total of 16 columns 0 to 15 and the position of the first row is 0. Also, the 2nd row has a total of 16 columns 0 to 15 and the position of the second row is position is 1. So the total numbers of the column are 16 x 2 = 32. Its means 16 x 2 LCD module can display 32 characters at the same time.

It will be a very complicated task to handle everything with the help of a microcontroller. So an Interface IC like HD44780 is used, which is mounted on the backside of the LCD Module. The function of this IC is to get the Commands and Data from the microcontroller and process them to display meaningful information onto the LCD Screen.

* GSM Module:



The **G**lobal **S**ystem for **M**obile Communications (GSM) is a standard developed by the European Telecommunications Standards Institute (ETSI)to describe the protocols for second generation digital cellular networks used by mobile devices such as mobile phones and tablets. GSM is also a trade mark owned by the GSM Association. GSM may also refer to the Full Rate voice codec.

It was first implemented in Finland in December 1991. By the mid-2010s, it became a global standard for mobile communications achieving over 90%market share, and operating in over 193 countries and territories. 2G networks developed as a replacement for first generation analog cellular networks. The GSM standard originally described a digital, circuit switched network optimized for full duplex voice telephony. This expanded over time to include data communications, first by circuit-switched transport, then by packet data transport via General Pack Radio Service (GPRS), and Enhanced Data Rates for GSM Evolution (EDGE).

Subsequently, the 3GPP developed third generation(3G) UMTS standards, followed by the fourth(4G) LTE advanced and the fifth generation 5G standards, which do not form part of the ETSI GSM standard. Beginning in the late 2010s, various carriers worldwide standard to shunt down their GSM networks. Nevertheless, as a result of the networks widespread use, the acronym “GSM” is still used as a generic term for the plethora of <n>G mobile phone technologies evolved from it.

* Push Button:



A **Push-button** (also spelled **Pushbutton**) or simply **button** is simple switch mechanism to control some aspect of a machine or a process. Buttons are typically made out of hard material, usually plastic or metal. The surface is usually flat or shaped to accommodate the human finger or hand, so as to be easily depressed or pushed. Buttons are most often biased switches, although many un-biased buttons (due to their physical nature) still require a spring to return to their un-pushed state.

Terms for the “pushing” of button include **pressing**, **depressing**, **mashing**, **slapping**, **hitting** and **punching**.

* Prototype:

