A PROJECT REPORT

ON

**GSM BASED INDUSTRIAL FIRE DETECTION AND FIRE CONTROL**

Dissertation submitted to the TS SBTET in

Partial fulfillment of the requirement for the award of

**DIPLOMA IN**

**ELECTRONICS AND COMMUNICATION ENGINEERING**



**SUBMITTED BY**

**BAYYARAM HARSHATH KUMAR 18189-EC-005**

**BURRA SHIVA 18189-EC-009**

**CHINDAM DHEERAJ 18189-EC- 010**

**CHUNCHU RAHUL 18189-EC-011**

Under the guidance of

**DR. AVARU VENKATARAMANA**

**( M.E (ECE), Ph.D, MISTE, MISCA, MIEI, Fellow, IETE )**

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**

**GOVERNMENT POLYTECHNIC, SIDDIPET**

**SIDDIPET DIST, TELANGANA**

**(Affiliated to SBTET-TS, Hyderabad)**

**2018-2021**

**GOVERNMENT POLYTECHNIC,SIDDIPET**

(Affiliated to SBTET-TS; Hyderabad)

(Vill: Rajgopalpet, Mdl: Nangunoor, Dist: Siddipet)



**CERTIFICATE**

This is to certify that the dissertation work entitled **“REAL TIME GSM BASED FIRE ACCIDENT ALERT SYSTEM THROUGH MESSAGE WITH LOCATION AND CALL ALERT”** Are the work doneby

**BAYYARAM HARSHATH KUMAR 18189-EC-005**

**BURRA SHIVA 18189-EC-009**

**CHINDAM DHEERAJ 18189-EC- 010**

**CHUNCHU RAHUL 18189-EC-011**

Submitted in partial fulfillment for the award of **DIPLOMA IN ELECTRONICS AND COMMUNICATION** and submitted to the State Board of TechnicalEducation And Training (TS), Hyderabad is record of Bonafide work carried out by them under our guidance and supervision. The results embodied in this project report have not been submitted to any other institutions for the award of Diploma.

**GUIDE HOD PRINCIPAL**

**DR. AVARU VENKATARAMANA SARANGAPANI G.SURYANARAYANA**

**External Examiner**

**­­­ACKNOWLEDGEMENT­­**

We wish to express my healthy gratitude to MR SRI **SURYA NARAYANA**, Principal of **GOVERNMENT POLYTECHNIC**, Rajgopalpet, Siddipet for providing all the facilities required for the seminar presentation.

We wish to take this opportunity to express our sincere thanks to **SARANGAPANI,** HOD of Department for the interesttechnical support and suggestions during the seminar leading to our success.

We are obliged and great full to my guide DR.**AVARU VENKATARAMANA,** of ECEDepartment for his valuable suggestions and sagacious guidance in allrespects during the technical of our training.

We here by express my sincere thanks to all teaching staff, parents and friends who extended their help hand in the accomplished of the project.

**BAYYARAM HARSHATH KUMAR 18189-EC-005**

**BURRA SHIVA 18189-EC-009**

**CHINDAM DHEERAJ 18189-EC- 010**

**CHUNCHU RAHUL 18189-EC-011**

**KAMUNI KAVITHA 18189-EC-021**

**KARIKE PRAVEEN 18189-EC-022**

**MERUPULA JAYANTH 18189-EC-031**

**SANDA. JAYAPRAKASH 18189-EC-044**

**VANTHADUPULA BAVITHA 18189-EC-052**

**INDEX**

|  |  |  |
| --- | --- | --- |
| **S/NO** | **CONTENT** | **PAGE NO** |
| **1** | **INTRODUCTION** | **7** |
| **2** | **LITERATURE SURVEY** | **8** |
| **3** | **HARDWARE IMPLEMENTATION** | **34** |
| **4** | **SOFTWARE IMPLEMENTATION** | **37** |
| **5** | **CODING PHASE** | **43** |
| **6** | **IMPLEMENTATION OF THE PROJECT** | **45** |
| **7** | **CONCLUSION AND FUTURE SCOPE** | **47** |
| **8** | **REFERENCES** | **48** |

**GSM BASED INDUSTRIAL FIRE DETECTION AND FIRE CONTROL**

**ABSTRACT**

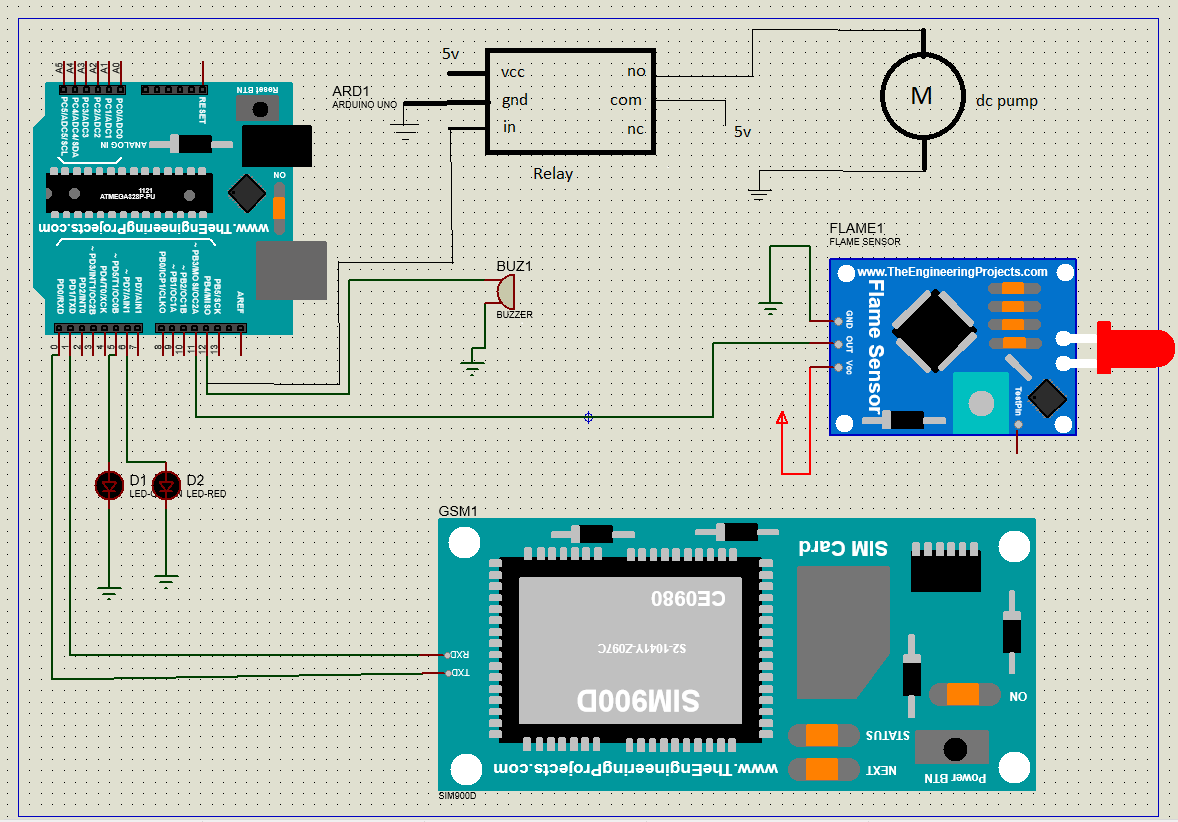
Now a days, every system is automated in order to face new challenges in the present day situation. Automated systems have less manual operations, so that the flexibility, reliabilities are high and accurate. Hence every field prefers automated control system. Especially in the field of electronics automated systems are doing better performance increasingly.

The project itself indicates that the system checks the fire in the industry, based on that we get a message alert and a call alert to the registered mobile number and automatically motor star pumping water to stop the fire. Here we use one fire sensor.

When fire is detected by the flame sensor then it gives high signal to Arduino input. Arduino is programmed in such a way that it gives the message and call alert to the user/owner through the gsm 900A.

**GSM BASED FIRE DETECTION AND FIRE CONTROL**

SCHEMATIC DIAGRAM :

****

GREEN LED

RED LED

**CHAPTER 1**

**INTRODUCTION**

Cultural property management is entrusted with the responsibility of protecting and preserving an institution's buildings, collections, operations and occupants. Constant attention is required to minimize adverse impact due to climate, pollution, theft, vandalism, insects, mold and fire.

Because of the speed and totality of the destructive forces of fire, it constitutes one of the more serious threats. Vandalized or environmentally damaged structures can be repaired and stolen objects recovered. Items destroyed by fire, however, are gone forever. An uncontrolled fire can obliterate an entire room's contents within a few minutes and completely burn out a building in a couple hours.

Actually our project deals with the fire control and alert through the message and call to the owner by the GSM. It can also alert the nearest fire station / police / owner.

**CHAPTER 2**

**LITERATURE SURVEY**

**ARDUINO :**

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc. The 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 the USB cable or by an external 9-volt battery, though it accepts voltages between 7 and 20 volts. It is similar to the Arduino Nano and Leonardo.

The word “uno” means “one” in Italian and was chosen to mark the initial release of Arduino Software. The Uno board is the first in a series of USB-based Arduino boards; it and version 1.0 of the Arduino IDE were the reference versions of Arduino, which have 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.

While the Uno communicates using the original STK500 protocol, it differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it uses the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

The Arduino/Genuino Uno has a number of facilities for communicating with a computer, another Arduino/Genuino board, or other microcontrollers. The Atmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX).

An Atmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The 16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, a .inf file is required. Arduino Software (IDE) includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1). A SoftwareSerial library allows serial communication on any of the Uno’s digital pins.

Automatic (software) reset

Rather than requiring a physical press of the reset button before an upload, the Arduino/Genuino Uno board is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the Atmega8U2/16U2 is connected to the reset line of the Atmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip.

This setup has other implications. When the Uno is connected to a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened.

**History of Arduino :**

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 Barragan 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 non-engineers. 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.

**General pin functions**

**LED:** There is a built-in LED driven by digital pin 13. When the pin is high value, the LED is on, when the pin is low, it is off.

**VIN:** The input voltage to the Arduino/Genuino board when it is using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

**5V:** This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 – 20V), the USB connector (5V), or the VIN pin of the board (7-20V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage the board.

**3V3**: A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 Ma.

**GND**: Ground pins.

**IOREF**: This pin on the Arduino/Genuino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source, or enable voltage translators on the outputs to work with the 5V or 3.3V.

**Reset:** Typically used to add a reset button to shields that block the one on the board.

**Special pin functions**

Each of the 14 digital pins and 6 analog pins on the Uno can be used as an input or output, under software control (using pinMode, digitalWrite, and digitalRead functions).

They operate at 5 volts. Each pin can provide or receive 20 Ma as the recommended operating condition and has an internal pull-up resistor (disconnected by default) of 20-50K ohm. A maximum of 40Ma must not be exceeded on any I/O pin to avoid permanent damage to the microcontroller. The Uno has 6 analog inputs, labeled A0 through A5; each provides 10 bits of resolution (i.e. 1024 different values). By default, they measure from ground to 5 volts, though it is possible to change the upper end of the range using the AREF pin and the analogReference function.

**In addition, some pins have specialized functions:**

**Serial / UART:** pins 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 Atmega8U2 USB-to-TTL serial chip.

**External interrupts:** pins 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.

**PWM (pulse-width modulation):** pins 3, 5, 6, 9, 10, and 11. Can provide 8-bit PWM output with the analogWrite() function.

**SPI (Serial Peripheral Interface):** pins 10 (SS), 11 (MOSI), 12 (MISO), and 13 (SCK). These pins support SPI communication using the SPI library.

**TWI (two-wire interface) / I²C:** pin SDA (A4) and pin SCL (A5). Support TWI communication using the Wire library.

**AREF (analog reference):** Reference voltage for the analog inputs.

Communication

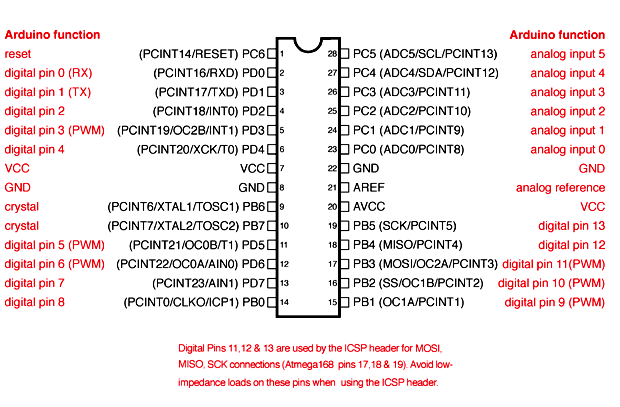
**TECHNICAL SPECIFICATIONS :**

|  |  |
| --- | --- |
| Microcontroller | Atmega328p – 8 bit AVR family microcontroller |
| Operating Voltage | 5V |
| Recommended Input Voltage | 7-12V |
| Input Voltage Limits | 6-20V |
| Analog Input Pins | 6 (A0 – A5) |
| Digital I/O Pins | 14 (Out of which 6 provide PWM output) |
| DC Current on I/O Pins | 40 Ma |
| DC Current on 3.3V Pin | 50 Ma |
| Flash Memory | 32 KB (0.5 KB is used for Bootloader) |
| SRAM | 2 KB |
| EEPROM | 1 KB |
| Frequency (Clock Speed) | 16 MHz |

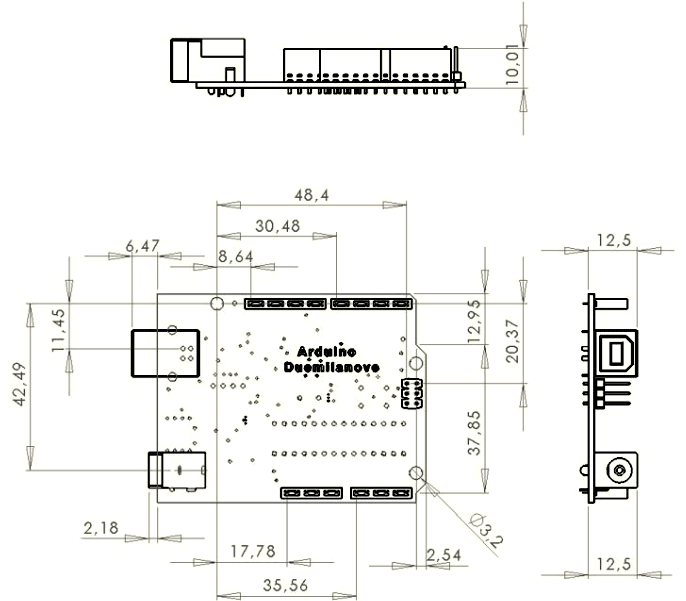
**PIN DESCRIPTION:**

|  |  |  |
| --- | --- | --- |
| **Pin Category** | **Pin Name** | **Details** |
| Power | Vin, 3.3V, 5V, GND | Vin: Input voltage to Arduino when using an external power source.  5V: Regulated power supply used to power microcontroller and other components on the board.  3.3V: 3.3V supply generated by on-board voltage regulator. Maximum current draw is 50Ma.  GND: ground pins. |
| Reset | Reset | Resets the microcontroller. |
| Analog Pins | A0 – A5 | Used to provide analog input in the range of 0-5V |
| Input/Output Pins | Digital Pins 0 – 13 | Can be used as input or output pins. |
| Serial | 0(Rx), 1(Tx) | Used to receive and transmit TTL serial data. |
| External Interrupts | 2, 3 | To trigger an interrupt. |
| PWM | 3, 5, 6, 9, 11 | Provides 8-bit PWM output. |
| SPI | 10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK) | Used for SPI communication. |
| Inbuilt LED | 13 | To turn on the inbuilt LED. |
| TWI | A4 (SDA), A5 (SCA) | Used for TWI communication. |
| AREF | AREF | To provide reference voltage for input voltage. |

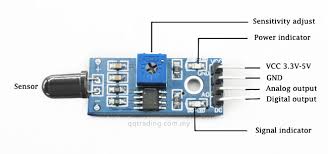
**Arduino Uno PIN mapping :**



**Arduino UNO 2d model :**



**LM393 FLAME SENSOR**



**Flame Sensor Working and Its Applications**

A sensor which is most sensitive to a normal light is known as a flame sensor. That’s why this sensor module is used in flame alarms. This sensor detects flame otherwise wavelength within the range of 760 nm – 1100 nm from the light source. This sensor can be easily damaged to high temperature. So this sensor can be placed at a certain distance from the flame. The flame detection can be done from a 100cm distance and the detection angle will be 600. The output of this sensor is an analog signal or digital signal. These sensors are used in fire fighting robots like as a flame alarm.

**What is a Flame Sensor?**

A flame-sensor is one kind of detector which is mainly designed for detecting as well as responding to the occurrence of a fire or flame. The flame detection response can depend on its fitting. It includes an alarm system, a natural gas line, propane & a fire suppression system. This sensor is used in industrial boilers. The main function of this is to give authentication whether the boiler is properly working or not. The response of these sensors is faster as well as more accurate compare with a heat/smoke detector because of its mechanism while detecting the flame.

**Working Principle**

This sensor/detector can be built with an electronic circuit using a receiver like electromagnetic radiation. This sensor uses the infrared flame flash method, which allows the sensor to work through a coating of oil, dust, water vapor, otherwise ice.

**Flame Sensor Module**

The pin configuration of this sensor is shown below. It includes four pins which include the following. When this module works with a microcontroller unit then the pins are

Pin1 (VCC pin): Voltage supply rages from 3.3V to 5.3V

Pin2 (GND): This is a ground pin

Pin3 (AOUT): This is an analog output pin (MCU.IO)

Pin4 (DOUT): This is a digital output pin (MCU.IO)

**Features & Specifications**

* Photosensitivity is high
* Response time is fast
* Simple to use
* Sensitivity is adjustable
* Detection angle is 600,
* It is responsive to the flame range.
* Accuracy can be adjustable

If the flame intensity is lighter within 0.8m then the flame test can be activated, if the flame intensity is high, then the detection of distance will be improved.

**Applications**

These sensors are used in several dangerous situations which include the following.

* Hydrogen stations
* Industrial heating
* Fire detection
* Fire alarm
* Fire fighting robot
* Drying systems
* Industrial gas turbines
* Domestic heating systems
* Gas-powered cooking devices

Thus, this is all about a flame sensor. From the above information finally, we can conclude that the main purpose of this sensor is to reduce the risks which are associated with the ignition. These sensors respond frequently than a smoke or heat detector. Here is a question for you, what are the advantages of the flame-sensor?

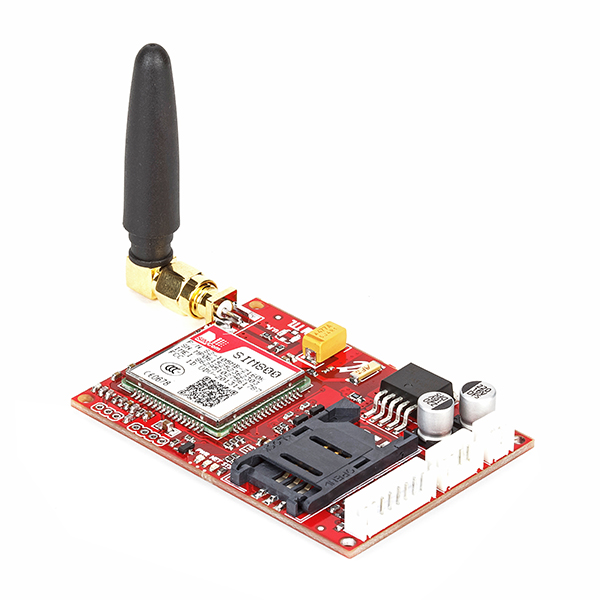
**Specifications**

* Operating Voltage: 3.3V to 5V DC
* Operating Current: 15ma
* Output Digital – 0V to 5V, Adjustable trigger level from preset
* Output Analog – 0V to 5V based on infrared radiation from fire flame falling on the sensor
* LEDs indicating output and power
* PCB Size: 3.2cm x 1.4cm
* LM393 based design

**Pin details**

* VCC = 3.3V to 5V DC
* GND = Ground
* DO = Digital Output
* AO = Analog Output

**GSM**



**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 °

## **What is GSM?**

GSM stands for **G**lobal **S**ystem for **M**obile Communication. It is a digital cellular technology used for transmitting mobile voice and data services. Important facts about the GSM are given below −

* The concept of GSM emerged from a cell-based mobile radio system at Bell Laboratories in the early 1970s.
* GSM is the name of a standardization group established in 1982 to create a common European mobile telephone standard.
* GSM is the most widely accepted standard in telecommunications and it is implemented globally.
* GSM is a circuit-switched system that divides each 200 kHz channel into eight 25 kHz time-slots. GSM operates on the mobile communication bands 900 MHz and 1800 MHz in most parts of the world. In the US, GSM operates in the bands 850 MHz and 1900 MHz.
* GSM owns a market share of more than 70 percent of the world's digital cellular subscribers.
* GSM makes use of narrowband Time Division Multiple Access (TDMA) technique for transmitting signals.
* GSM was developed using digital technology. It has an ability to carry 64 kbps to 120 Mbps of data rates.
* Presently GSM supports more than one billion mobile subscribers in more than 210 countries throughout the world.
* GSM provides basic to advanced voice and data services including roaming service. Roaming is the ability to use your GSM phone number in another GSM network.

GSM digitizes and compresses data, then sends it down through a channel with two other streams of user data, each in its own timeslot.

## **Why GSM?**

Listed below are the features of GSM that account for its popularity and wide acceptance.

* Improved spectrum efficiency
* International roaming
* Low-cost mobile sets and base stations (BSs)
* High-quality speech
* Compatibility with Integrated Services Digital Network (ISDN) and other telephone company services
* Support for new services

A GSM network comprises of many functional units. These functions and interfaces are explained in this chapter. The GSM network can be broadly divided into −

* The Mobile Station (MS)
* The Base Station Subsystem (BSS)
* The Network Switching Subsystem (NSS)
* The Operation Support Subsystem (OSS)

## **GSM - The Mobile Station**

The MS consists of the physical equipment, such as the radio transceiver, display and digital signal processors, and the SIM card. It provides the air interface to the user in GSM networks. As such, other services are also provided, which include −

* Voice teleservices
* Data bearer services
* The features' supplementary services



The MS also provides the receptor for SMS messages, enabling the user to toggle between the voice and data use. Moreover, the mobile facilitates access to voice messaging systems. The MS also provides access to the various data services available in a GSM network. These data services include −

* X.25 packet switching through a synchronous or asynchronous dial-up connection to the PAD at speeds typically at 9.6 Kbps.
* General Packet Radio Services (GPRSs) using either an X.25 or IP based data transfer method at the speed up to 115 Kbps.
* High speed, circuit switched data at speeds up to 64 Kbps.

We will discuss more about GMS services in **GSM - User Services**.

### What is SIM?

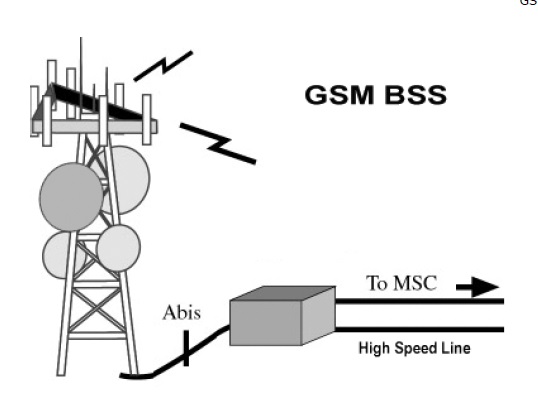
The SIM provides personal mobility so that the user can have access to all subscribed services irrespective of both the location of the terminal and the use of a specific terminal. You need to insert the SIM card into another GSM cellular phone to receive calls at that phone, make calls from that phone, or receive other subscribed services.

## **GSM - The Base Station Subsystem (BSS)**

The BSS is composed of two parts −

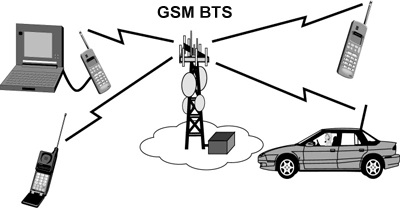
* The Base Transceiver Station (BTS)
* The Base Station Controller (BSC)

The BTS and the BSC communicate across the specified Abis interface, enabling operations between components that are made by different suppliers. The radio components of a BSS may consist of four to seven or nine cells. A BSS may have one or more base stations. The BSS uses the Abis interface between the BTS and the BSC. A separate high-speed line (T1 or E1) is then connected from the BSS to the Mobile MSC.



### The Base Transceiver Station (BTS)

The BTS houses the radio transceivers that define a cell and handles the radio link protocols with the MS. In a large urban area, a large number of BTSs may be deployed.



The BTS corresponds to the transceivers and antennas used in each cell of the network. A BTS is usually placed in the center of a cell. Its transmitting power defines the size of a cell. Each BTS has between 1 and 16 transceivers, depending on the density of users in the cell. Each BTS serves as a single cell. It also includes the following functions −

* Encoding, encrypting, multiplexing, modulating, and feeding the RF signals to the antenna
* Transcoding and rate adaptation
* Time and frequency synchronizing
* Voice through full- or half-rate services
* Decoding, decrypting, and equalizing received signals
* Random access detection
* Timing advances
* Uplink channel measurements

### The Base Station Controller (BSC)

The BSC manages the radio resources for one or more BTSs. It handles radio channel setup, frequency hopping, and handovers. The BSC is the connection between the mobile and the MSC. The BSC also translates the 13 Kbps voice channel used over the radio link to the standard 64 Kbps channel used by the Public Switched Telephone Network (PSDN) or ISDN.

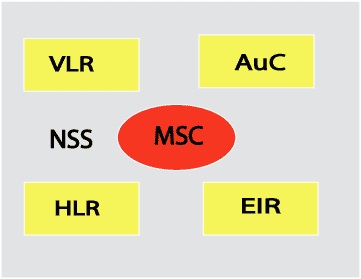
It assigns and releases frequencies and time slots for the MS. The BSC also handles intercell handover. It controls the power transmission of the BSS and MS in its area. The function of the BSC is to allocate the necessary time slots between the BTS and the MSC. It is a switching device that handles the radio resources.

The additional functions include−

* Control of frequency hopping
* Performing traffic concentration to reduce the number of lines from the MSC
* Providing an interface to the Operations and Maintenance Center for the BSS
* Reallocation of frequencies among BTSs
* Time and frequency synchronization
* Power management
* Time-delay measurements of received signals from the MS

## **GSM - The Network Switching Subsystem (NSS)**

The Network switching system (NSS), the main part of which is the Mobile Switching Center (MSC), performs the switching of calls between the mobile and other fixed or mobile network users, as well as the management of mobile services such as authentication.



The switching system includes the following functional elements −

### Home Location Register (HLR)

The HLR is a database used for storage and management of subscriptions. The HLR is considered the most important database, as it stores permanent data about subscribers, including a subscriber's service profile, location information, and activity status. When an individual buys a subscription in the form of SIM, then all the information about this subscription is registered in the HLR of that operator.

### Mobile Services Switching Center (MSC)

The central component of the Network Subsystem is the MSC. The MSC performs the switching of calls between the mobile and other fixed or mobile network users, as well as the management of mobile services such as registration, authentication, location updating, handovers, and call routing to a roaming subscriber. It also performs such functions as toll ticketing, network interfacing, common channel signaling, and others. Every MSC is identified by a unique ID.

### Visitor Location Register (VLR)

The VLR is a database that contains temporary information about subscribers that is needed by the MSC in order to service visiting subscribers. The VLR is always integrated with the MSC. When a mobile station roams into a new MSC area, the VLR connected to that MSC will request data about the mobile station from the HLR. Later, if the mobile station makes a call, the VLR will have the information needed for call setup without having to interrogate the HLR each time.

### Authentication Center (AUC)

The Authentication Center is a protected database that stores a copy of the secret key stored in each subscriber's SIM card, which is used for authentication and ciphering of the radio channel. The AUC protects network operators from different types of fraud found in today's cellular world.

### Equipment Identity Register (EIR)

The Equipment Identity Register (EIR) is a database that contains a list of all valid mobile equipment on the network, where its International Mobile Equipment Identity (IMEI) identifies each MS. An IMEI is marked as invalid if it has been reported stolen or is not type approved.

## **GSM - The Operation Support Subsystem (OSS)**

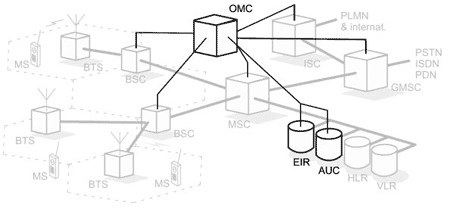
The operations and maintenance center (OMC) is connected to all equipment in the switching system and to the BSC. The implementation of OMC is called the operation and support system (OSS).

Here are some of the OMC functions−

* Administration and commercial operation (subscription, end terminals, charging, and statistics).
* Security Management.
* Network configuration, Operation, and Performance Management.
* Maintenance Tasks.

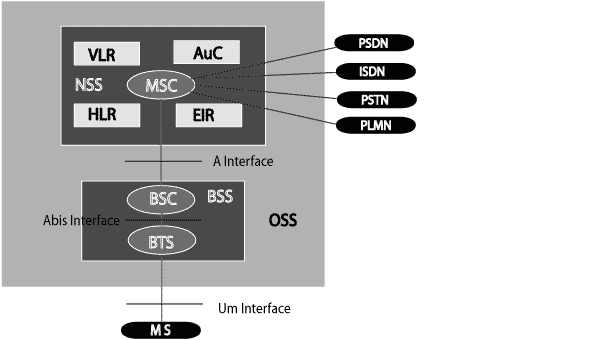
The operation and Maintenance functions are based on the concepts of the Telecommunication Management Network (TMN), which is standardized in the ITU-T series M.30.

Following is the figure, which shows how OMC system covers all the GSM elements.



The OSS is the functional entity from which the network operator monitors and controls the system. The purpose of OSS is to offer the customer cost-effective support for centralized, regional, and local operational and maintenance activities that are required for a GSM network. An important function of OSS is to provide a network overview and support the maintenance activities of different operation and maintenance organizations.

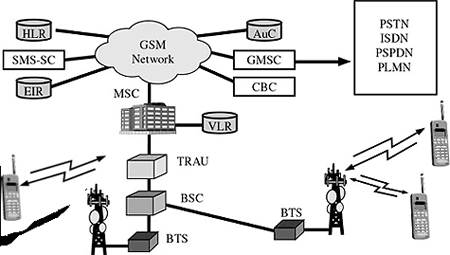
A simple pictorial view of the GSM architecture is given below −



The additional components of the GSM architecture comprise of databases and messaging systems functions −

* Home Location Register (HLR)
* Visitor Location Register (VLR)
* Equipment Identity Register (EIR)
* Authentication Center (AuC)
* SMS Serving Center (SMS SC)
* Gateway MSC (GMSC)
* Chargeback Center (CBC)
* Transcoder and Adaptation Unit (TRAU)

The following diagram shows the GSM network along with the added elements –



The MS and the BSS communicate across the Um interface. It is also known as the *air interface* or the *radio link*. The BSS communicates with the Network Service Switching (NSS) center across the *A* interface.

## **GSM network areas**

In a GSM network, the following areas are defined −

* **Cell** − Cell is the basic service area; one BTS covers one cell. Each cell is given a Cell Global Identity (CGI), a number that uniquely identifies the cell.
* **Location Area** − A group of cells form a Location Area (LA). This is the area that is paged when a subscriber gets an incoming call. Each LA is assigned a Location Area Identity (LAI). Each LA is served by one or more BSCs.
* **MSC/VLR Service Area** − The area covered by one MSC is called the MSC/VLR service area.
* **PLMN** − The area covered by one network operator is called the Public Land Mobile Network (PLMN). A PLMN can contain one or more MSCs.

**RELAY**

|  |  |  |
| --- | --- | --- |
| **Pin Number** | **Pin Name** | **Description** |
| 1 | Coil End 1 | Used to trigger(On/Off) the Relay, Normally one end is connected to 5V and the other end to ground |
| 2 | Coil End 2 | Used to trigger(On/Off) the Relay, Normally one end is connected to 5V and the other end to ground |
| 3 | Common (COM) | Common is connected to one End of the Load that is to be controlled |
| 4 | Normally Close (NC) | The other end of the load is either connected to NO or NC. If connected to NC the load remains connected before trigger |
| 5 | Normally Open (NO) | The other end of the load is either connected to NO or NC. If connected to NO the load remains disconnected before trigger |

### ****Features of 5-Pin 5V Relay****

* Trigger Voltage (Voltage across coil) : 5V DC
* Trigger Current (Nominal current) : 70mA
* Maximum AC load current: 10A @ 250/125V AC
* Maximum DC load current: 10A @ 30/28V DC
* Compact 5-pin configuration with plastic moulding
* Operating time: 10msec Release time: 5msec
* Maximum switching: 300 operating/minute (mechanically)

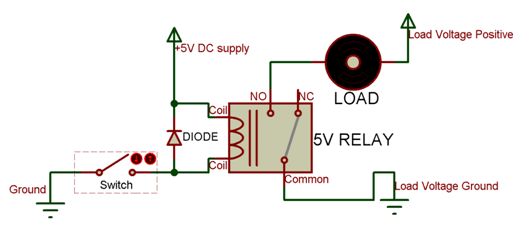
**Equivalent Relays**

3V Relay, [12V Relay](https://components101.com/switches/12v-relay), 1-channel Relay module, 4-channel Relay Module.

**How to use a Relay**

Relays are most commonly used switching device in electronics. Let us learn how to use one in our circuits based on the requirement of our project.

Before we proceed with the circuit to drive the relay we have to consider two important parameter of the relay. Once is the **Trigger Voltage**, this is the voltage required to turn on the relay that is to change the contact from Common->NC to Common->NO. Our relay here has 5V trigger voltage, but you can also find relays of values 3V, 6V and even 12V so select one based on the available voltage in your project. The other parameter is your **Load Voltage & Current**, this is the amount of voltage or current that the NC,NO or Common terminal of the relay could withstand, in our case for DC it is maximum of 30V and 10A. Make sure the load you are using falls into this range.



The above circuit shows a bare-minimum concept for a relay to operate. Since the relay has 5V trigger voltage we have used a +5V DC supply to one end of the coil and the other end to ground through a switch. This **switch**can be anything from a small transistor to a microcontroller or a microprocessor which can perform switching operating. You can also notice a diode connected across the coil of the relay, this diode is called the **Fly back** **Diode**. The purpose of the diode is to protect the switch from high voltage spike that can produced by the relay coil. As shown one end of the load can be connected to the Common pin and the other end is either connected to NO or NC. If connected to NO the load remains disconnected before trigger and if connected to NC the load remains connected before trigger.

**Applications of Relay**

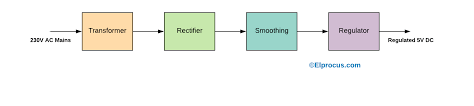
* Commonly used in switching circuits.
* For Home Automation projects to switch AC loads
* To Control (On/Off) Heavy loads at a pre-determined time/condition
* Used in safety circuits to disconnect the load from supply in event of failure
* Used in Automobiles electronics for controlling indicators glass motors etc.

**CHAPTER 3**

**HARDWARE IMPLEMENTATION**

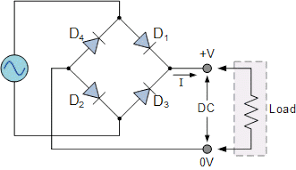
**POWER SUPPLY**

The power supply unit used to provide a constant 5V of dc supply from 230V of AC supply. These 5V dc will acts as power to different standard circuits. It mainly consists of following blocks.

**BRIDGE WAVE RECTIFIER**

A rectifier is an electrical device that converts alternating current to direct current, a process known as rectification. The term rectifier describes a diode that is being used to convert AC to DC.

A bridge rectifier converts the whole of the input waveform to one of constant polarity as its output. Bridge- wave rectifier converts both polarities of the input waveform to DC, and is more efficient. However, in a circuit with a center tapped transformer(9-0-9) is used.



For a single-phase AC, if the transformer is center -tapped, then two diodes back- to back (i.e. anodes to anode or cathode to cathode) can form a full wave rectifier. Many windings are required on the transformer secondary to obtain the same output voltage.

In this only two diodes are activated at a time i.e. D1 and D3 activate for positive cycle and D2 and D4 activates for the negative half cycle. D2 and D4 convert negative to positive cycle as it as negative supply and negative cycle as positive cycle at its output.

**VOLTAGE REGULATOR**

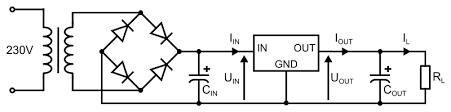
This is most common voltage regulator that is still used in embedded designs. LM7805 voltage regulator is a linear regulator. With proper heat sink this LM78XX types can handle even more than one amp current they also have thermal over load protection, short circuit protection

This will connect at the output of the rectifier to get constant DC supply insead of ripple voltages. It mainly consists of three pins :

1. Input voltage
2. Ground
3. Output voltage

For some devices we require 12V/9V/4V DC supply at that time we go for 7812/7809/7804 regulated instead of 7805 regulator. It also has some feature and pins has 7805 regulator except output is of 12V/9V/4V instead of 5V.

The general circuit for total power supply to any embedded device is as shown below.



**BUZZER**

We interface the buzzer to the Arduino, if fire is detected thebuzzer will make the sound along the message and call alert to user



**Water motor**

Here we use water motor for fire control in the industry.



**CHAPTER 4**

**SOFTWARE IMPLEMENTATION**

The Arduino Integrated Development Environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in functions from C and C++. It is used to write and upload programs to Arduino compatible boards, but also, with the help of third-party cores, other vendor development boards.

The source code for the IDE is released under the GNU General Public License, version 2. The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board’s firmware. By default, avrdude is used as the uploading tool to flash the user code onto official Arduino boards.

Arduino IDE is a derivative of the Processing IDE,[10] however as of version 2.0, the Processing IDE will be replaced with the Visual Studio Code-based Eclipse Theia IDE framework.

Written in C, C++

Operating system Windows, macOS, Linux

Platform IA-32, x86-64, ARM

Type Integrated development environment

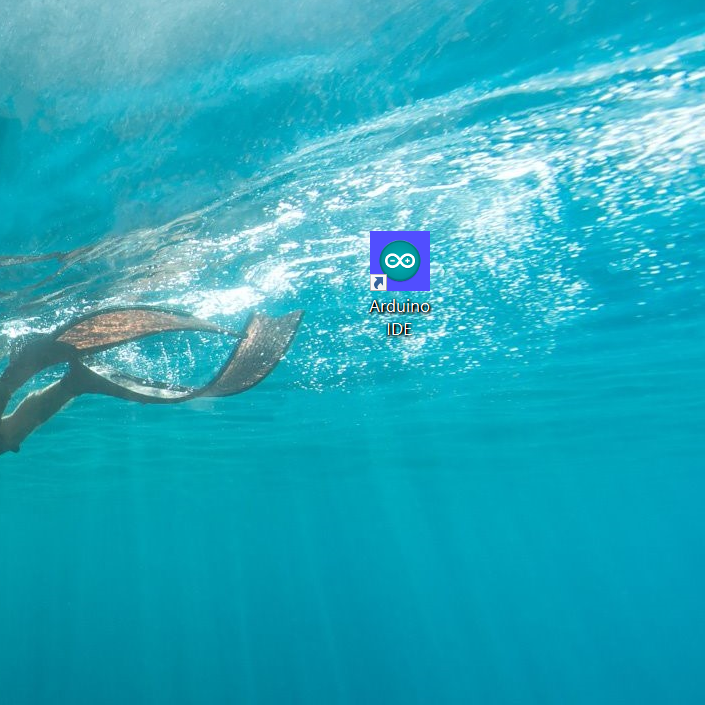
License LGPL or GPL license

Website blog.arduino.cc/2020/08/24/cli-and-ide-get-better-together/

With the rising popularity of Arduino as a software platform, other vendors started to implement custom open source compilers and tools (cores) that can build and upload sketches to other microcontrollers that are not supported by Arduino’s official line of microcontrollers.

In October 2019 the Arduino organization began providing early access to a new Arduino Pro IDE with debugging and other advanced features.

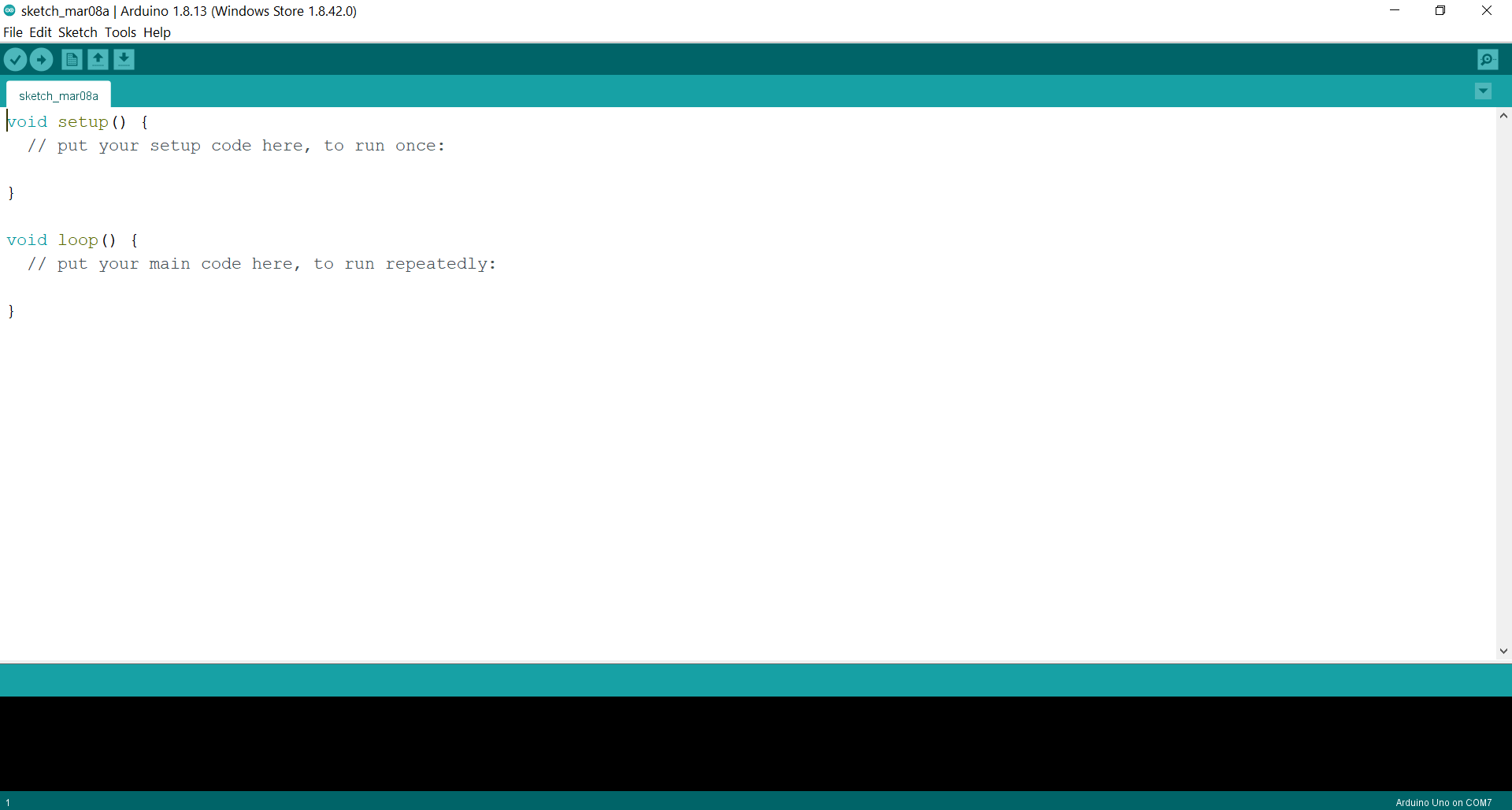
1. Click on Arduino IDE software.



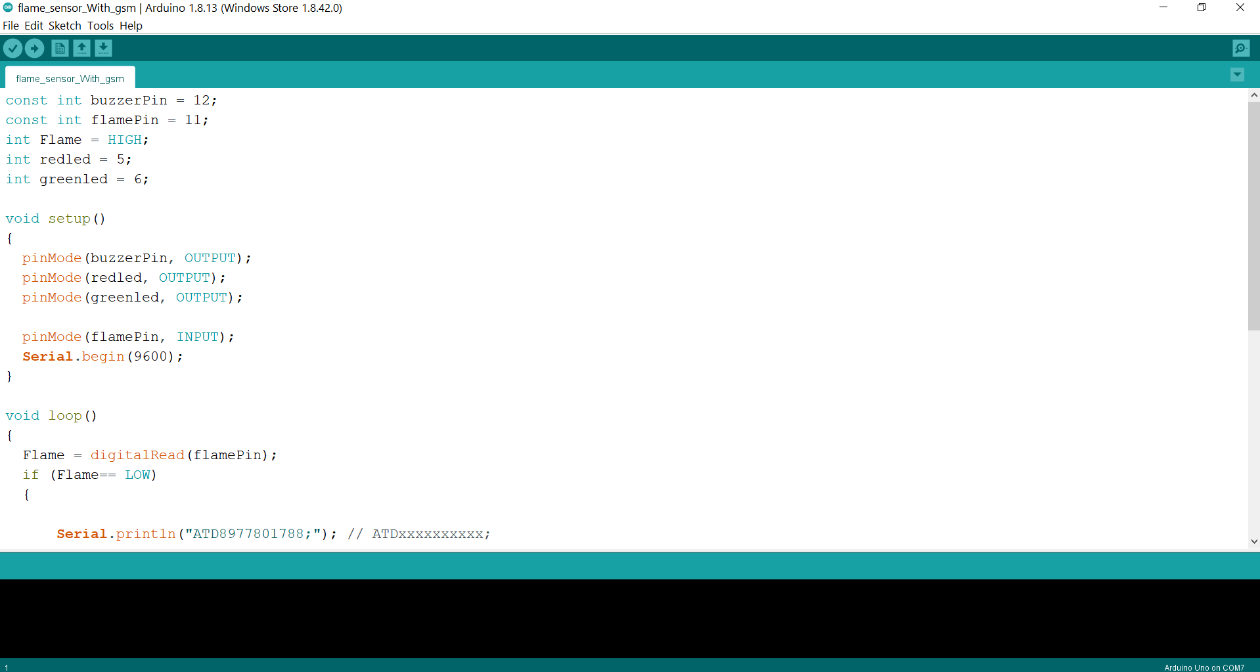
**2.** Click on the file from the title bar.

3. Click new (or) ctrl + N.

4. The following figure will appear.

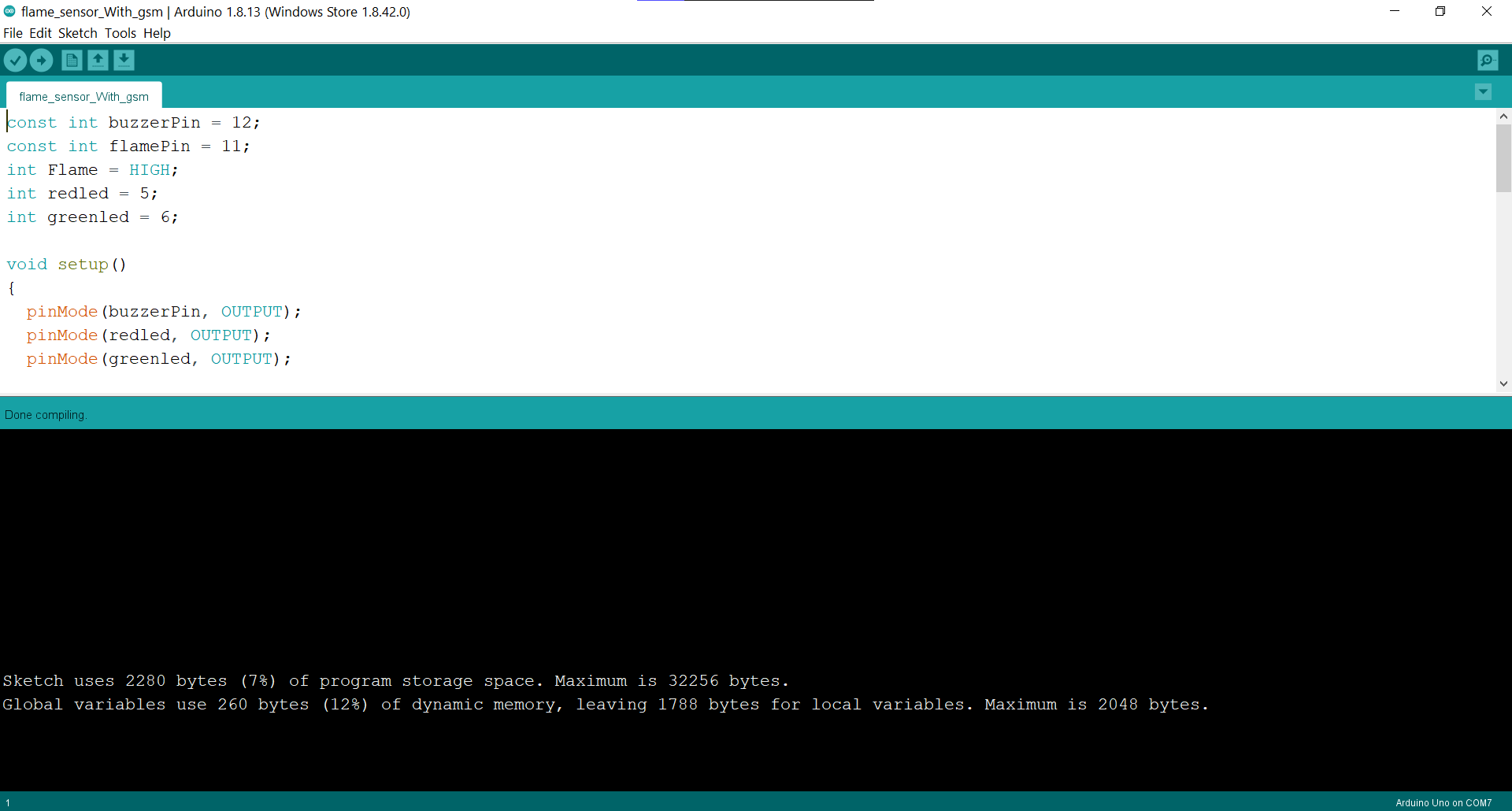
****

5. Enter the program.

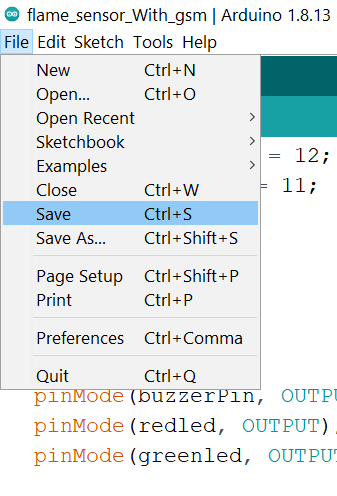


**6.** Compile the program



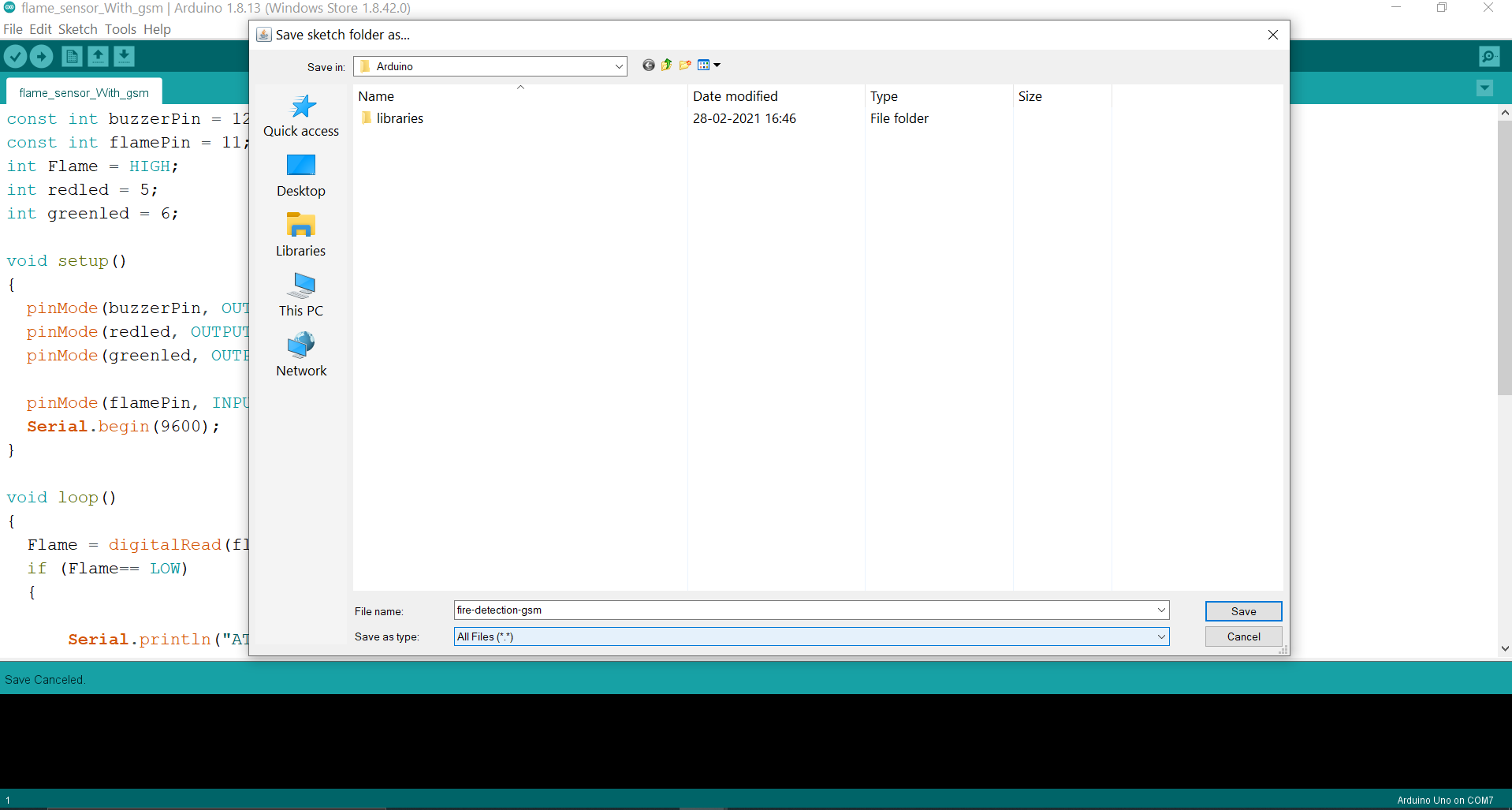


**7.** Click on file from the title bar.



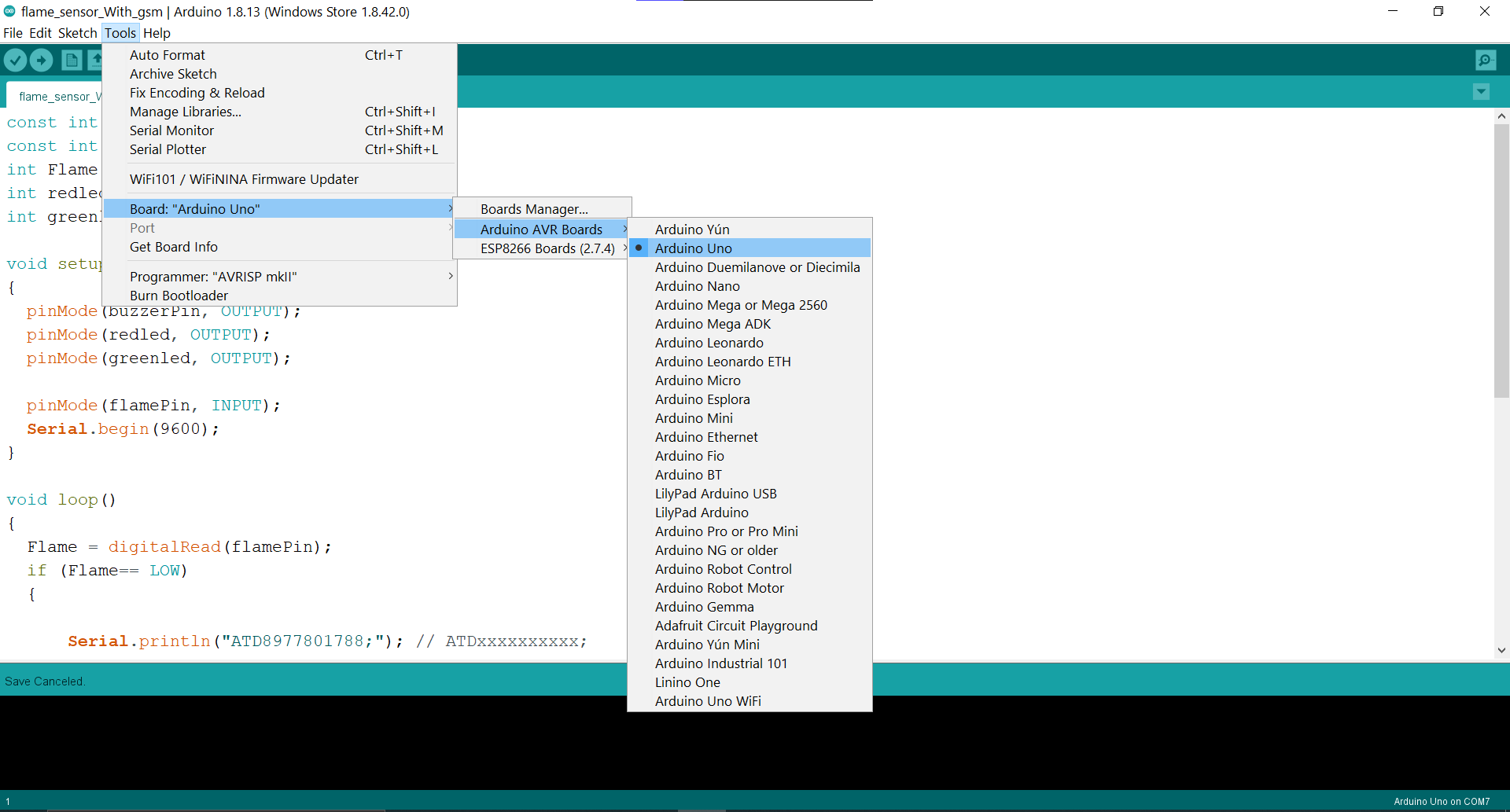
8. Click on save option to save the program.

9. Enter the file name for code.

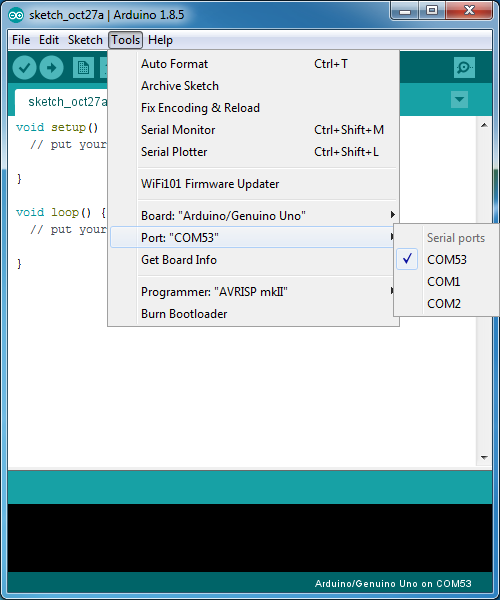


10. connect your Arduino to computer via USB cable.

11. select Arduino UNO board from the tools section in the titlebar 🡪 boards.

****

**12.** select the respective port from title bar tools 🡪 port.



13. Upload the code to Arduino.



**CHAPTER 5**

**CODING PHASE**

**ARDUINO CODE :**

const int buzzerPin = 12;

const int flamePin = 11;

int Flame = HIGH;

int redled = 5;

int greenled = 6;

void setup()

{

pinMode(buzzerPin, OUTPUT);

pinMode(redled, OUTPUT);

pinMode(greenled, OUTPUT);

pinMode(flamePin, INPUT);

Serial.begin(9600);

}

void loop()

{

Flame = digitalRead(flamePin);

if (Flame== LOW)

{

Serial.println(“ATD8977801788;”);

digitalWrite(buzzerPin, HIGH);

digitalWrite(redled, HIGH);

digitalWrite(greenled, LOW);

Serial.println(“AT+CMGF=1”); //Sets GSM in text mode

delay(1000); // Delay of 1 second

Serial.println(“AT+CMGS=\”+918977801788\”\r”);

Serial.println(“FIRE IN THE industry!”);

Serial.println((char)26);

//delay(100);

}

else

{

digitalWrite(buzzerPin, LOW);

digitalWrite(greenled, HIGH);

digitalWrite(redled, LOW);

}

}

**CHAPTER 6**

**IMPLEMENTATION OF THE PROJECT**

**WORKING**

We had connected the GSM module to the Arduino and the flame sensor as input to the Arduino. Also we had connected buzzer for local alert and also connected a water motor for fire control.

When the fire is detected police/owner will get the call and the message alert.and also a buzzer sound is also produced

**AT commands to GSM module**

**SMS Text mode :**

|  |  |
| --- | --- |
| **Command** | **Description** |
| AT+CSMS | Select message service |
| AT+CPMS | Preferred message storage |
| AT+CMGF | Message format |
| AT+CSCA | Service centre address |
| AT+CSMP | Set text mode parameters |
| AT+CSDH | Show text mode parameters |
| AT+CSCB | Select cell broadcast message types |
| AT+CSAS | Save settings |
| AT+CRES | Restore settings |
| AT+CNMI | New message indications to TE |
| AT+CMGL | List messages |
| AT+CMGR | Read message |
| AT+CMGS | Send message |
| AT+CMSS | Send message from storage |
| AT+CMGW | Write message to memory |
| AT+CMGD | Delete message |

**Testing :**

|  |  |
| --- | --- |
| **Command** | **Description** |
| AT | Checking communication between the module and computer. |

**Call control :**

|  |  |
| --- | --- |
| **Command** | **Description** |
| ATA | Answer command |
| ATD | Dial command |
| ATH | Hang up call |
| ATL | Monitor speaker loudness |
| ATM | Monitor speaker mode |
| ATO | Go on-line |
| ATP | Set pulse dial as default |
| ATT | Set tone dial as default |
| AT+CSTA | Select type of address |
| AT+CRC | Cellular result codes |

**CHAPTER 7**

**CONCLUSION AND FUTURE SCOPE**

**CONCLUSION :**

The prototype of the proposed system was implemented and it proved to be efficient fire detection and controlling the fire also alerting the owner and police with message and call through GSM.

This prototype will monitor the industry 24/7 and alerts the user when fire is detected through GSM.

**FUTURE SCOPE :**

The project can be extended in adding sensors like gas sensor, temperature sensor.

We can make use of ESP8266 Nodemcu and monitor the changes through the internet. And control the damage to the industry.

**CHAPTER 8**

**REFERENCES AND BIBLIOGRAPHY**

Websites:

* <http://web.science.mq.edu.au/>
* <https://www.arduino.cc/>
* http://en.wikipedia.org/wiki/
* http://www.journals.elsevier.com/
* http://pages.cs.wisc.edu/
* http://cs.stanford.edu/projects/
* http://www.Drip Irrigation Images.com
* http://gardenbot.org/howTo/soilMoisture/

**Journals & other books:**

• K.Prathyusha, M. Chaitanya Suman, “Design of Embedded System for the Automation of

Drip Irrigation”. IJAIEM (2319-4847), vol 1, Issue 2, October 2012.

• Prathyusha.K1, G. Sowmya Bala2, Dr. K. Sreenivasa Ravi, “A real time irrigation

control system for precision agriculture using WSN in Indian agricultural sectors”

International Journal of Computer Science, Engineering and Applications (IJCSEA)

Vol.3, No.4, August 20.

• Akyildig, I.F., 2005. A Survey on Sensor Networks [J].IEEE Communications Magazine,

2002(8):725-734.

• Luis Ruiz-Garcia, Loredana Lunadei 1, Pillar Barreiro 1 and Jose Ignacio Robla 2 “A

Review of Sensor Technologies and Applications in Agriculture and Food Industry: State

of the Art and Current Trends” Sensors 2009, 9, 4728-4750; doi:10.3390/s90604728.

• Klute, A. (ed.), 1986: Methods of Soil Analysis, Part 1: Physical and Mineralogical

Methods. American Society of Agronomy, Madison, Wisconsin, United States, 1188 pp