CHAPTER 1: INTRODUCTION

A sensor interface device is essential for sensor data collection of industrial sensor networks in IoT environments. Sensor Networks has been employed to collect data about physical phenomena in various applications such as habitat monitoring and ocean monitoring, and surveillance. As an emerging technology brought about rapid advances in modern wireless telecommunication, Internet of Things (IoT) has attracted a lot of attention and is expected to bring benefits to numerous application areas including industrial sensor systems and healthcare systems manufacturing. Sensor systems are well-suited for long-term industrial environmental data acquisition for IoT representation. Sensor interface device is essential for detecting various kinds of sensor data of industrial Sensor Network in IoT environments. It enables us to acquire sensor data. Thus, we can better understand the outside environment information. However, in order to meet the requirements of long-term industrial environmental data acquisition in the IoT, the acquisition interface device can collect multiple sensor data at the same time, so that more accurate and diverse data information can be collected from industrial sensor network.

Computer communication systems and the Internet are playing an important role in the daily life. Using this knowledge many applications are imaginable. Home automation, utility meters, appliances, security systems, card readers, and building controls, which can be easily controlled using either special front-end software or a standard internet browser client from anywhere around the world can access and control the sensors. Web access functionality is embedded in a device to enable low cost widely accessible and enhanced user interface functions for the device. A web server in the device provides access to the user interface functions for the device through a device web page by using internet.

A web server can be embedded into any appliance and connected to the Internet so the appliance can be monitored and controlled from remote places through the browser in a desktop by using its application with normally connected to local host. In this project the Central node is replicated using a GSM cell phone and the Management node is replicated using laptop that uses IoT platform.

1.1: Introduction to Embedded Systems

1.1.1 What is an embedded system?

An Embedded System is a combination of computer hardware and software and perhaps additional mechanical or other parts, designed to perform a specific function. An embedded system is a microcontroller-based software driven, reliable, real-time control system, autonomous, or human or network interactive, operating on diverse physical variables and in diverse environments and sold into a competitive and cost conscious market.

An embedded system is not a computer system that is used primarily for processing nor is it a software system on PC or UNIX. It is not a traditional business or scientific application. Embedded systems are classified into high-end embedded and low-end embedded systems. High-end embedded system -generally 32, 64 Bit Controllers used with operating systems. Examples are Personal Digital Assistant and Mobile phones and many more. Low-end embedded systems -generally 8, 16 Bit controllers are used with a minimal operating systems and hardware layout designed for the specific purpose.

1.1.2 Characteristics of Embedded System

The following below are the characteristics of an embedded system,

- An embedded system is software embedded into computer hardware that makes a system dedicated to be used for variety of application.
- Embedded system generally used for do specific task that provide real-time output on the basis of various characteristics of an embedded system.
- Embedded system may contain a smaller part within a larger device that used for serving the more specific application to perform variety of task using hardware-software intermixing configuration.
- It provides high reliability and real-time computation ability.
- Embedded systems range from no user interface at all, in systems dedicated only to one task, to complex graphical user interfaces that resemble modern computer desktop operating systems.
- A common array of n configuration for very-high-volume embedded systems is the system on a chip (SoC) which contains a complete system consisting of multiple processors, multipliers, caches and interfaces on a single chip. These SoCs can be implemented as an application-specific integrated circuit (ASIC) or using a field-programmable gate array (FPGA).

1.1.3 Classification of Embedded Systems

The embedded systems are classified as:

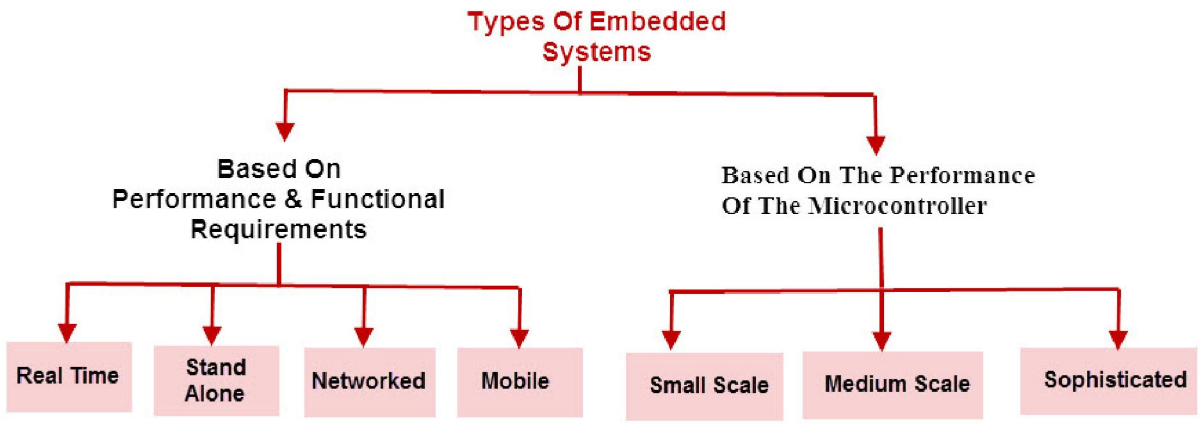


Fig 1.1.3 Classification of Embedded system

1.1.3.1 Small scale embedded system

Normally small scale embedded system is designed by using an 8 bit microcontroller that may even be activated by a battery. For developing embedded software for such system, an editor, assembler or cross assembler are used for specific microcontroller or processor used in the system.

Fig 1.1.3.1 Small scale embedded system

1.1.3.2 Medium scale embedded systems

The medium scale embedded systems are designed using single or multiple 16 bit or 32 bit microcontroller or digital signal processors (DSP's) or reduced instruction set computer (RISC's). These types of embedded systems have both hardware and software complexities. The development tools like real time operating system (RTOS), source code engineering tools, simulator, debugger and integrated development tools are required for such complex software design system. These software tools also provide the solution for the hardware complexities, so assembler is used very rarely.

Fig 1.1.3.2 Medium scale embedded systems

1.1.3.3 Sophisticated embedded system

Sophisticated embedded systems consist of large quantity of hardware and software complexities hence they may required scalable processors or configurable processors and programmable logic arrays (PLA's).They are used for cutting-edge applications that need hardware and software Co-design and components which have to assemble in the final system.

Fig 1.1.3.3 Sophisticated embedded system

1.1.3.4 Real time embedded system

Real time embedded system are designed to perform some specific work in specific time. Real time embedded systems are classified into two types such as soft and hard real time systems. Hard real systems are the one whose response time required are very low. For example, Nuclear reactor monitoring system and aircraft navigation system. Soft real time systems are ones' whose response times are not of big concern. They are used in tablets, computers and other devices.

Fig 1.1.3.4 Real time embedded system

1.1.3.5 Stand alone embedded system

These type embedded systems are works in standalone made in which input i.e. electrical signals from sensors or keyword or push button are taken, then processed and produced the desired output to drive another system such as LED or LCD display for displaying some meaningful information to user. Such stand alone embedded systems are used in mp3 players, digital cameras, video game consoles, microwave ovens and temperature measurement systems.

Fig 1.1.3.5 Stand alone embedded system

1.1.3.6 Mobile embedded system

Mobile embedded systems are used in portable embedded devices like mobile phones, tablets, digital cameras, I phones, smart phones and personal digital assistants etc. the systems has some limitation such as memory constrains, small in size and lack of good user interface such as keyboard and display.

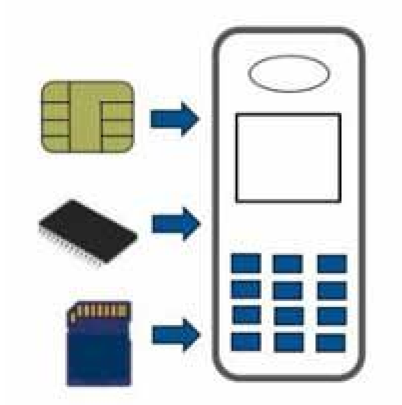


Fig 1.1.3.6 Mobile embedded system

1.1.3.7 Networked embedded systems

These types of embedded systems consists of various components such as sensors, controllers, actuators etc. interconnected through a network using TCP/IP or UDP. The networked embedded system built on ASIP's or general purpose processors.

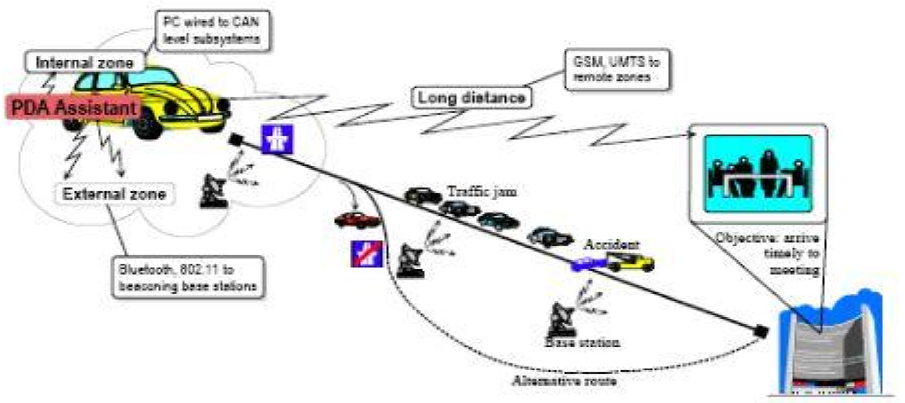


Fig 1.1.3.7 Networked embedded systems

1.1.4 Key factors while designing an embedded system.

An embedded system is any computer system hidden inside a product other than a computer. Number of difficulties is encountered when writing embedded system software in addition to those we encounter when we write applications, some of them are,

- Throughput – Our system may need to handle a lot of data in a short period of time.
- Response – Our system may need to react to events quickly.
- Testability – Setting up equipment to test embedded software can be difficult.
- Debugging ability –Without a screen or a keyboard, finding out what the software is doing wrong (other than not working) is a troublesome problem.
- Reliability – Embedded systems must be able to handle any situation without human intervention.
- Memory space – Memory is limited on embedded systems, and you must make the software and the data fit into whatever memory exists.
- Program installation – you will need special tools to get your software into embedded systems.
- Power consumption – Portable systems must run on battery power, and the software in these systems must conserve power.
- Processor hogs – computing that requires large amounts of CPU time can complicate the response problem.
- Cost – Reducing the cost of the hardware is a concern in many embedded system projects; software often operates on hardware that is barely adequate for the job.

1.1.5 Applications

- Military and aerospace embedded software applications.
- Communication Applications.
- Industrial automation and process control software.
- Mastering the complexity of applications.
- Reduction of product design time.
- Real time processing of ever increasing amounts of data.
- Intelligent, autonomous sensors.

Chapter 2- SYSTEM OVERVIEW

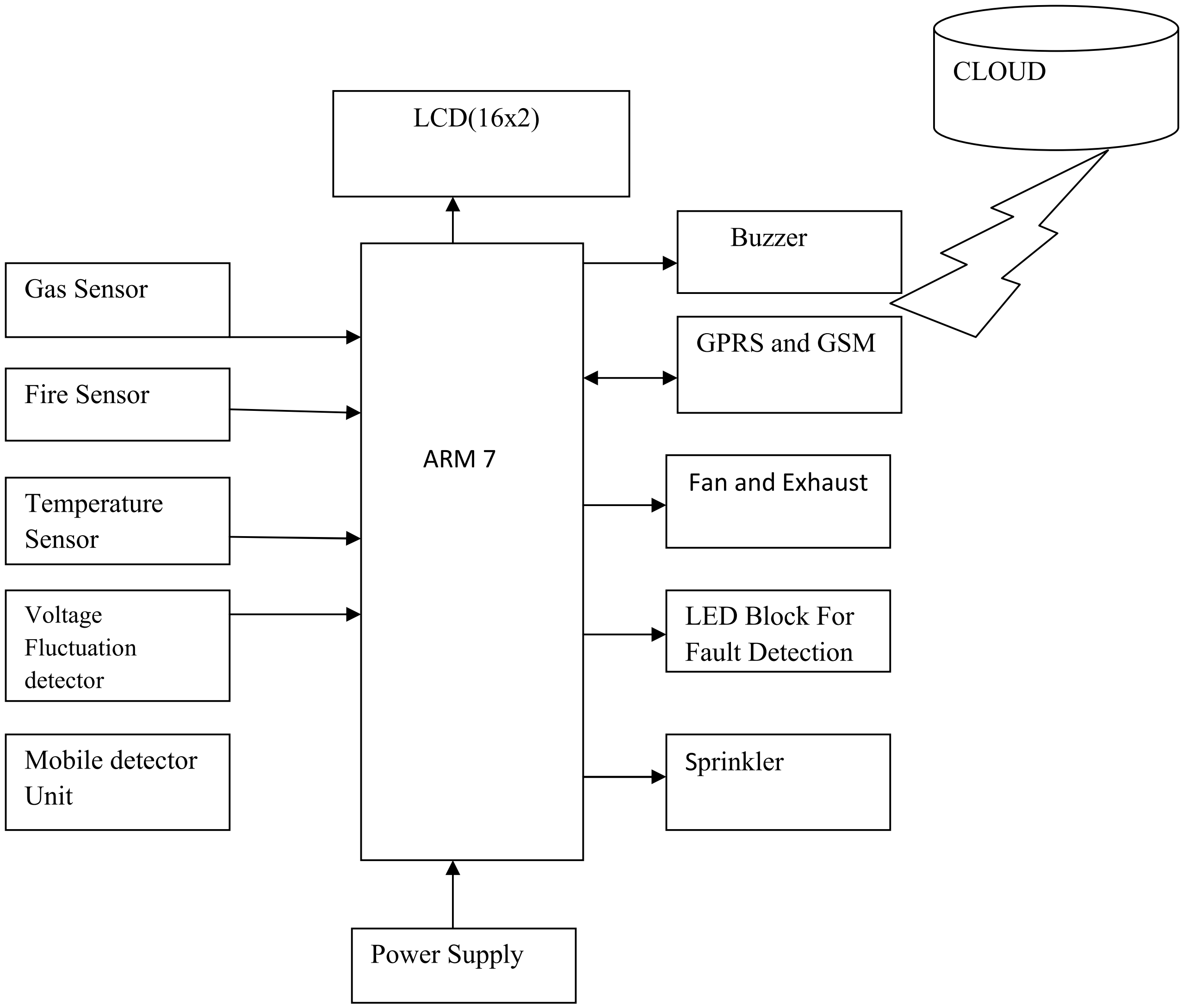
2.1 Problem statement

- To keep a real time watch for gas leakage, rise in temperature, over voltage, fire breakage and radiations from mobile in any part of the industry.
- To continuously update values of voltage, temperature, gas and fire status to the cloud.
- To send alert message to mobile via GSM in case of any mishap.
- To switch on the fan, exhaust, sprinkler in case of temperature rise, gas leakage and fire break respectively.

Each of the sensors should operate in an autonomous behaviour to monitor any mishaps such as gas leakage, rise in temperature, over voltage, fire breakage and radiations from mobile in any part of the industry. The values should be continuously updated to the cloud via IoT. An alert message to the subscribed number should be sent in case of encountering any rise in the readings compared to the permissible values. Remedies such as exhaust, fan and sprinkler should be automatically switched on, on detecting any gas leakage, temperature rise and fire break respectively.

2.2 Proposed System

The below figure shows the block diagram of the proposed system:



The system includes a ARM7-LPC2148 microcontroller acting as a base station through which all other communications take place. All the sensors are interfaced to any of the 45 usable pins on the controller board. The buzzer, fan and exhaust are connected to the controller via a driver unit. The sprinkler is connected to the controller via a relay. The mobile detection unit is an independent unit that continuously scans for RF signals. This is not connected to the base station and will have its own buzzer and LED.

2.3 Methodology

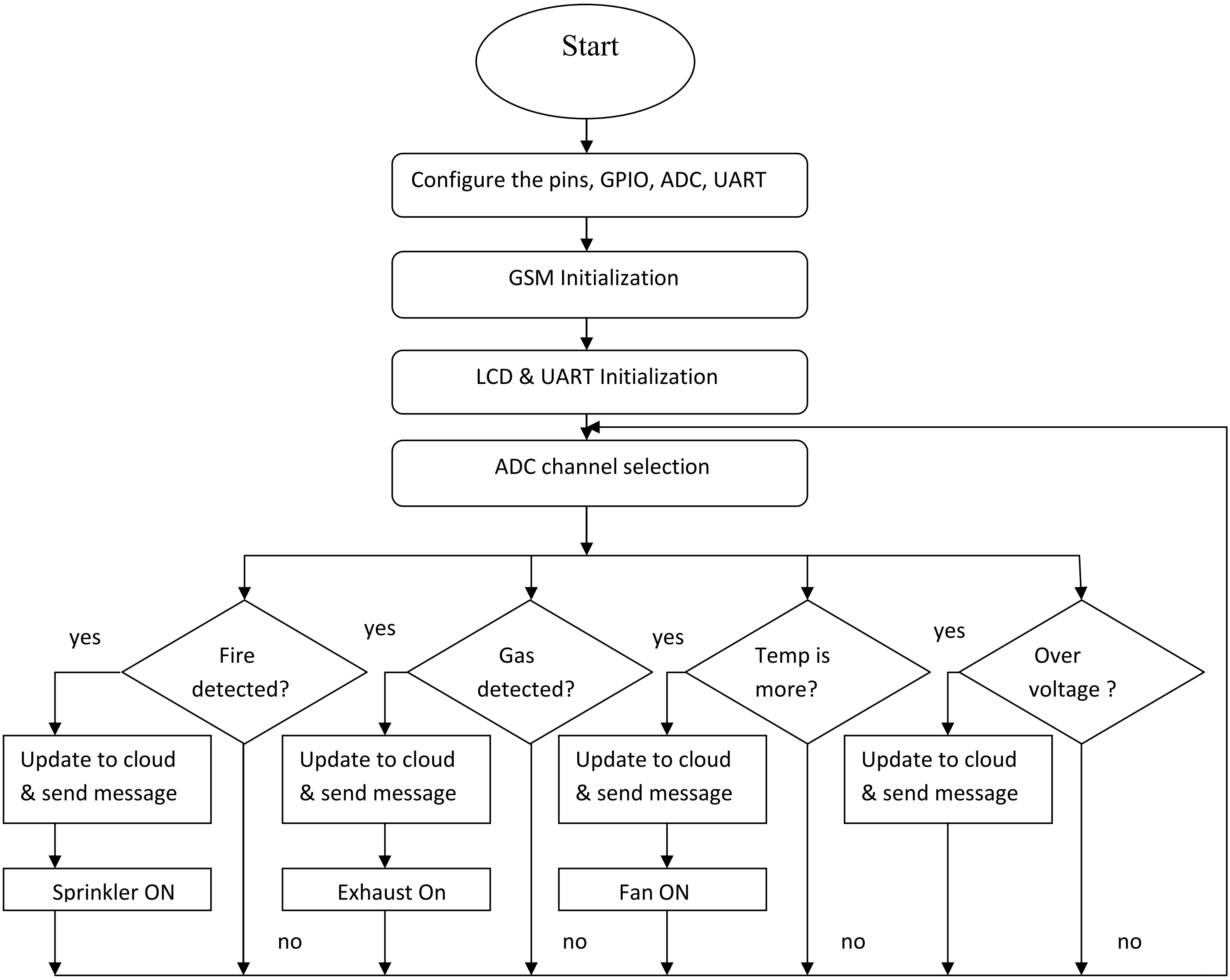


Fig 2.3a Flow chart of operation of controller

When the power supply is turned on, the GSM module gets initialised along with the GPIOs and UART. Once the initialization is done, it displays the message on the LCD and sends a message to the owner via GSM. The ADCs and the other channels continuously monitor the sensors. Threshold values will be defined before hand on the software. Whenever any of the sensors value goes beyond the threshold, the controller then takes the steps to bring down this to normal condition. The LCD will keep updating the voltage and temperature values real time and same will be updated on the IoT. If temperature goes beyond 35^{o}c, the fan is turned on. If gas leakage is detected, the exhaust is turned ON. If flame sensor detects fire, the sprinkler is automatically driven on and finally if the voltage goes beyond 12V then the buzzer will beep loudly until the condition is brought back to normal.

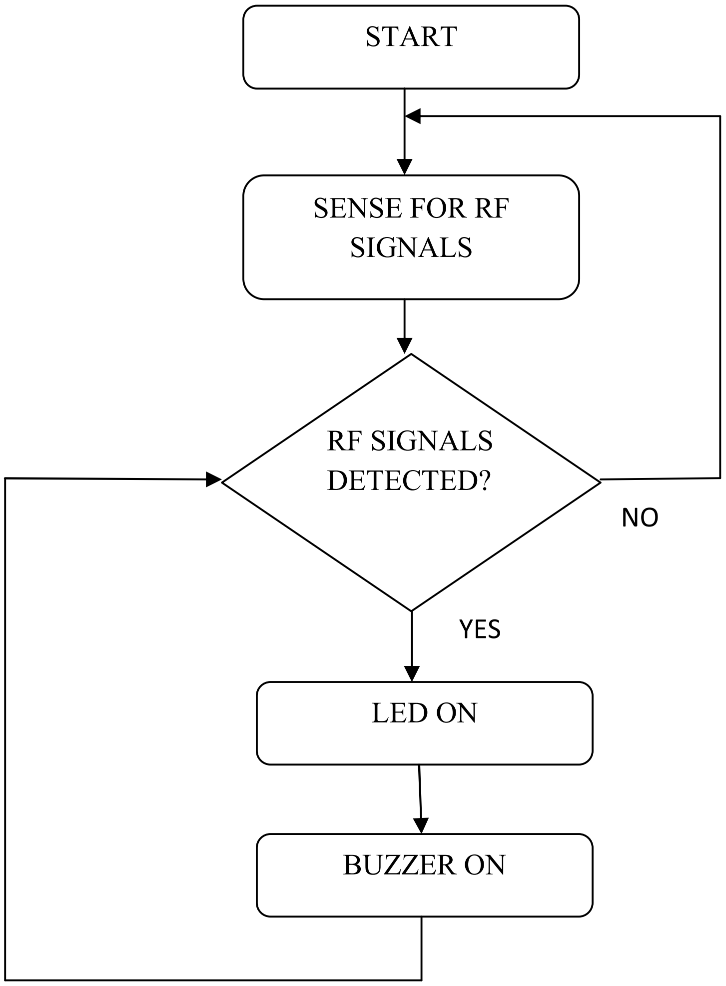


Fig 2.3b Flow diagram of Mobile Detector Circuit

The mobile detector circuit is an independent network that continuously senses for RF signals. The capacitor used is in this circuit is main component to detect the signal. An IC 555 is used to turn the buzzer on which is discussed in the following chapter. If any signal gets detected, the LED and Buzzer will turn on and continuously is ON till the signals stop coming. The capacitor is capable of sensing Calls, Messages, Video transmission and Data even if the mobile is in silent mode.

2.4 Interfacing Diagram

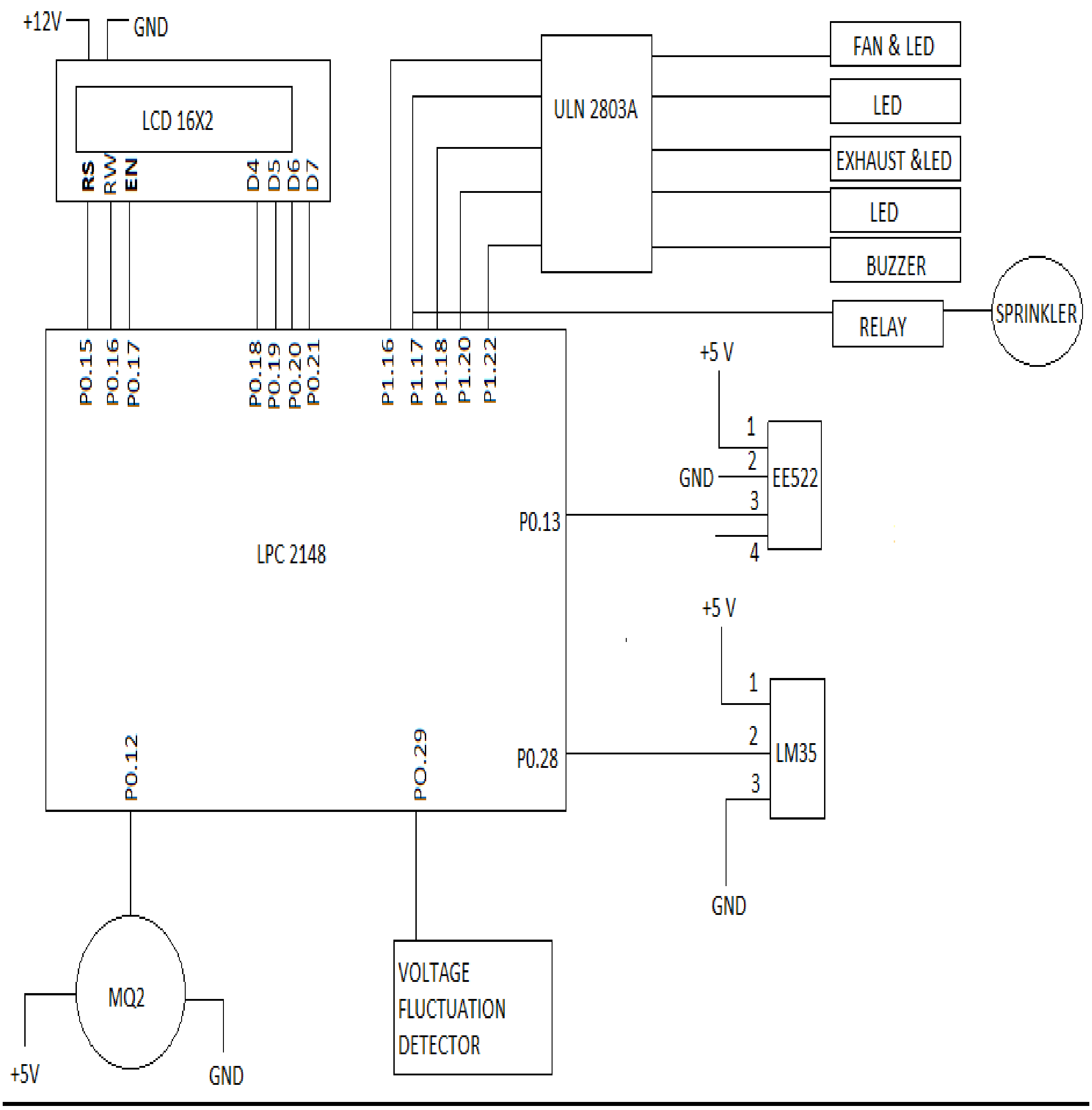


Fig 2.4 Interface Diagram

The above diagram represents the interfacing diagram with all the pin connections and sensors connected accordingly. The processor has 64 pins in which 45 usable pins. The two ports are individually used as shown. The two ADC ports and both UARTs are used in this project.

Chapter 3: Hardware Requirements

The hardware requirements for the project are:

1. ARM7-LPC2148 Microcontroller
2. Temperature Sensor
3. Fire Sensor
4. Gas Sensor
5. Mobile Detector Circuit
6. Liquid Crystal Display
7. Sprinkler and buzzer
8. GSM
9. WIFI Module

3.1 ARM 7 LPC2148 Microcontrollers

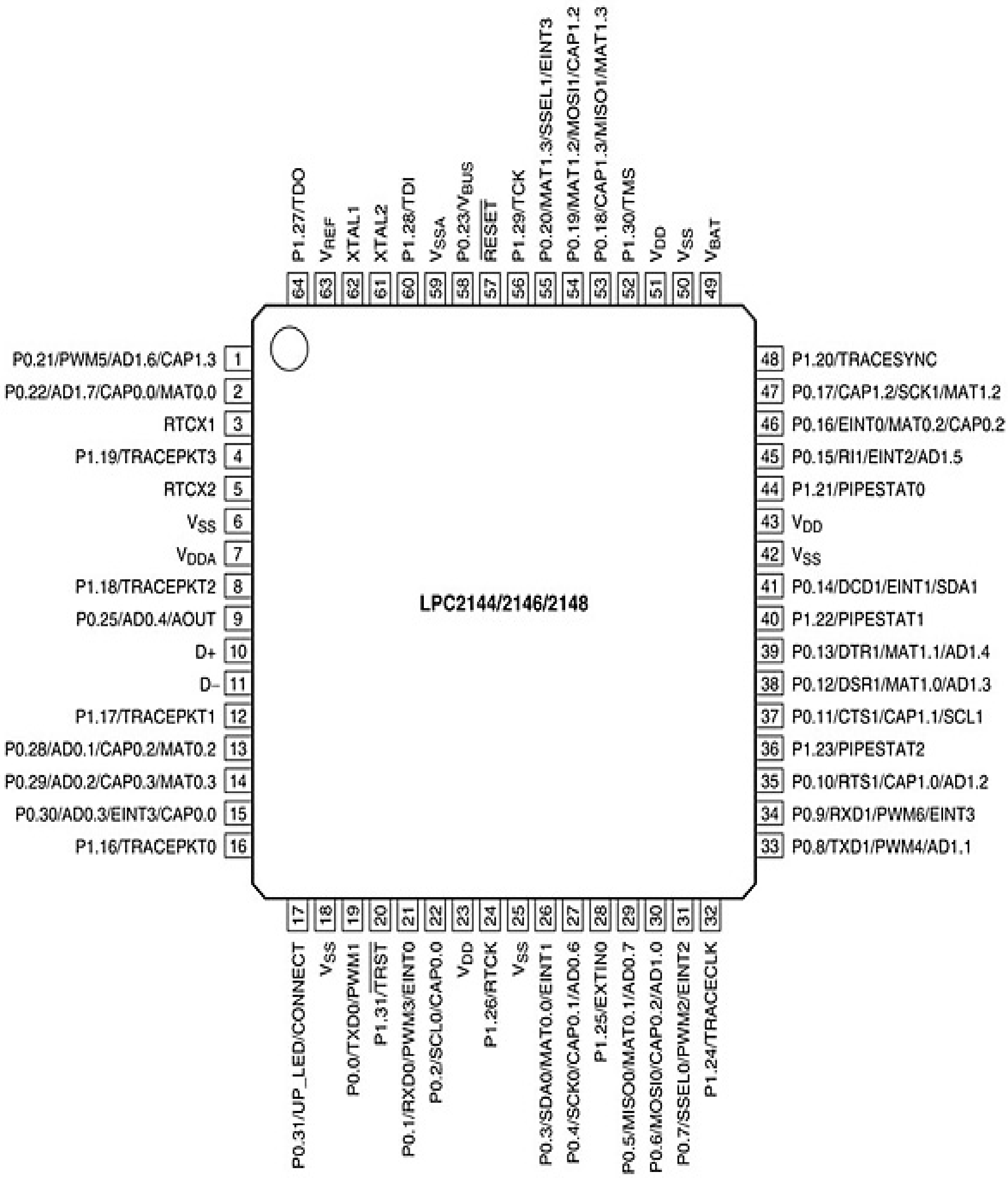
3.1.1 General Description

The LPC2148 microcontrollers are based on a 32-bit ARM7TDMI-S CPU with real-time emulation and embedded trace support, which combines microcontroller with embedded high speed flash memory ranging from 32 kB to 512 kB. A 128-bit wide memory interface and unique accelerator architecture that enables 32-bit code execution at the maximum clock rate. For critical code size applications, the alternative 16-bit Thumb mode reduces code by more than 30 % with minimal performance penalty.

Due to their tiny size and low power consumption, LPC2148 are ideal for applications where miniaturization is a key requirement, such as access control and point-of-sale. Serial communications interfaces ranging from a USB 2.0 Full-speed device, multiple UARTs, SPI, SSP to I2C-bus and on-chip SRAM of 8 kB up to 40 kB, make these devices very well suited for communication gateways and protocol converters, soft modems, voice recognition and low end imaging, providing both large buffer size and high processing power. Various

32-bit timers, single or dual 10-bit. ADC(s), 10-bit DAC, PWM channels and 45 fast GPIO lines with up to nine edge or level sensitive external interrupt pins make these microcontrollers suitable for industrial control and medical systems.

3.1.2 Pin Diagram



3.1.3 Features of ARM7-LPC2148

- 32-bit ARM7TDMI-S microcontroller in a tiny LQFP64 package.
- 4kB to 64 kB of on-chip RAM and 128kB to 512 kB of ROM. 128-bit wide interface/accelerator enables high-speed 60 MHz operation.
- In-System Programming/In-Application Programming (ISP/IAP) via on-chip boot loader software. Single flash sector or full chip erase in 400 ms and programming of 256 bytes in 1ms.
- Embedded ICE RT and Embedded Trace interfaces offer real-time debugging with the on-chip Real Monitor software and high-speed tracing of instruction execution.
- USB 2.0 Full-speed compliant device controller with 2 kB of endpoint RAM. In addition, the LPC2148 provides 8 kB of on-chip RAM accessible to USB by DMA.
- Two10-bit ADCs provide a total of 16 analog inputs, with conversion times as low as 2.44 μs per channel.
- One 10-bit DAC provides variable analog output.
- Two 32-bit timers/external event counters (with four capture and four compare channels each), PWM unit (six outputs) and watchdog.
- Low power Real-Time Clock (RTC) with independent power and 32 kHz clock input.
- Multiple serial interfaces including two UARTs (16C550), two Fast I2C-bus (400 kbit/s), SPI and SSP with buffering and variable data length capabilities.
- Vectored Interrupt Controller (VIC) with configurable priorities and vector addresses.
- Up to 45 of 5 V tolerant fast general purpose I/O pins in a tiny LQFP64 package.

- Up to 31 external interrupt pins available.
- 60 MHz maximum CPU clock available from programmable on-chip PLL with settling time of 100 μs.
- On-chip integrated oscillator operates with an external crystal from 1 MHz to 25 MHz.
- Power saving modes include idle and Power-down.
- Individual enable/disable of peripheral functions as well as peripheral clock scaling for additional power optimization.
- Processor wake-up from Power-down mode via external interrupt or BOD.
- Single power supply chip with POR and BOD circuits.
- CPU operating voltage range of 3.0 V to 3.6 V (3.3 V ± 10 %) with 5 V tolerant I/O pads.

3.3.4 Architectural Overview

The ARM7TDMI-S is a general purpose 32-bit microprocessor, which offers high performance and very low power consumption. The ARM architecture is based on Reduced Instruction Set Computer (RISC) principles and the instruction set and related decode mechanism are much simpler than those of micro programmed Complex Instruction Set Computers (CISC). This simplicity results in a high instruction throughput and impressive real-time interrupt response from a small and cost-effective processor core. Pipeline techniques are employed so that all parts of the processing and memory systems can operate continuously. Typically, while one instruction is being executed, its successor is being decoded, and a third instruction is being fetched from memory.

The ARM7TDMI-S processor also employs a unique architectural strategy known as Thumb, which makes it ideally suited to high-volume applications with memory restrictions, or applications where code density is an issue. The key idea behind Thumb is that of a super-reduced instruction set. The Thumb set's 16-bit instruction length allows it to approach twice the density of standard ARM code while retaining most of the ARM's performance advantage over a traditional 16-bit processor using 16-bit registers. This is possible because Thumb code operates on the same 32-bit register set as ARM code. Thumb code is able to provide up to 65 % of the code size of ARM, and 160 % of the performance of an equivalent ARM processor connected to a 16-bit memory system. The particular flash implementation in the LPC2148 allows for full speed execution also in ARM mode. It is recommended to program performance critical and short code sections (such as interrupt service routines and DSP algorithms) in ARM mode. The impact on the overall code size will be minimal but the speed can be increased by 30% over Thumb mode.

3.2 Temperature Sensor

Temperature sensors are vital to a variety of everyday products. For example, household ovens, refrigerators, and thermostats all rely on temperature maintenance and control in order to function properly. Temperature control also has applications in chemical engineering. Examples of this include maintaining the temperature of a chemical reactor at the ideal set-point, monitoring the temperature of a possible runaway reaction to ensure the safety of employees, and maintaining the temperature of streams released to the environment to minimize harmful environmental impact.

While temperature is generally sensed by humans as "hot", "neutral", or "cold", chemical engineering requires precise, quantitative measurements of temperature in order to accurately control a process. This is achieved through the use of temperature sensors, and temperature regulators which process the signals they receive from sensors.

From a thermodynamics perspective, temperature changes as a function of the average energy of molecular movement. As heat is added to a system, molecular motion increases and the system experiences an increase in temperature. It is difficult, however, to directly measure the energy of molecular movement, so temperature sensors are generally designed to measure a property which changes in response to temperature. The devices are then calibrated to traditional temperature scales using a standard. The sensor that we sue in the project is LM35.

3.2.1 LM35 Temperature Sensor

The LM35 is an integrated circuit sensor that can be used to measure temperature with an electrical output proportional to the temperature (Degree). The LM35 temperature sensor measure temperature more accurately than using a thermistor. The sensor circuitry is sealed and not subject to oxidation, etc. The LM35 generates higher output voltage than thermocouples and may not require that the output voltage be amplified. It has an output voltage that is proportional to the Celsius temperature. The scale factor of LM35 is 0.1 V/ ^{o}C. The LM35 draws only 60 micro amps from its supply and possesses a low self heating capability. The sensor self heating causes less than 0.1 ^{o}C temperature rise in still air.

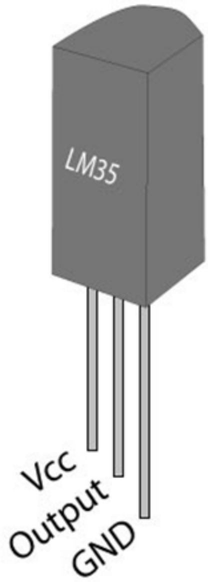


Fig 3.2.1 LM35 Sensor

The first pin is provided with supply voltage of 5V. The second pin provides an analog output which is connected to the pin P0.28 which is an ADC of the controller. The third pin is connected to the ground.

3.3 Fire sensor

Flame sensor is the most sensitive to ordinary light that is why its reaction is generally used as flame alarm purposes. This module can detect flame or wavelength in 760 nm to 1100 nm range of light source. Small plate output interface can and single-chip can be directly connected to the microcomputer IO port. The sensor and flame should keep a certain distance to avoid high temperature damage to the sensor. The shortest test distance is 20cm, if the flame is bigger, test it with farther distance. The detection angle is 60 degrees so the flame spectrum is especially sensitive. The module which we use here is a flame sensor module EE522.

3.3.1 EE522 Flame Sensor Module

This module is sensitive to the flame and radiation. It also can detect ordinary light source in the range of a wavelength 760nm-1100 nm. The detection distance is up to 100 cm. The Flame sensor can output digital or analog signal. It can be used as a flame alarm or in fire fighting robots.

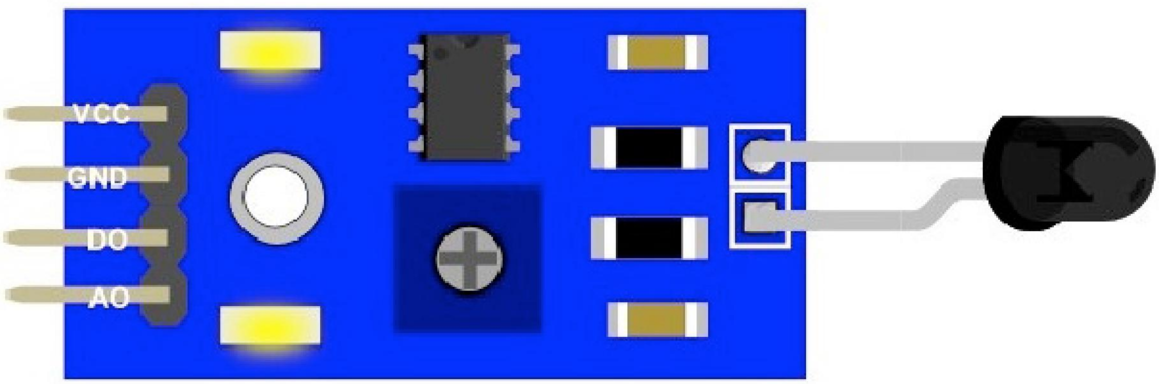


Fig 3.3.1 Flame Sensor Module

There are four pins in the flame sensor module. The first pin is input voltage of about 5V DC. The second pin connected to ground. The third pin is output pin called Digital Output and this is connected to pin P0.13 of the controller. Since the output is digital there is no need of additional ADC. The fourth pin is the Analog Out pin which provides an analog output and is not used. The IC used in the module is the comparator chip LM393. This is used to provide stable readings on the output end.

3.4 Gas Sensor

Gas leak detection is the process of identifying potentially hazardous gas leaks by sensors. These sensors usually employ an audible alarm to alert people when a dangerous gas has been detected. Exposure to toxic gases can also occur in operations such as painting, fumigation, fuel filling, construction, excavation of contaminated soils, landfill operations, entering confined spaces, etc. Common sensors include combustible gas sensors, photo ionization detectors, infrared point sensors, ultrasonic sensors, electrochemical gas sensors, and semiconductor sensors. More recently, infrared imaging sensors have come into use. All of these sensors are used for a wide range of applications and can be found in industrial plants, refineries, pharmaceutical manufacturing, fumigation facilities, paper pulp mills, aircraft and ship-building facilities, hazmat operations, waste-water treatment facilities, vehicles, indoor air quality testing and homes. The gas sensor we use in the project is MQ2.

3.4.1 MQ2 Gas Sensor

This sensor is used to sense the leakage of LPG. In normal conditions the output of this sensor is ‘high’ and it goes ‘low’, when the LPG is sensed.

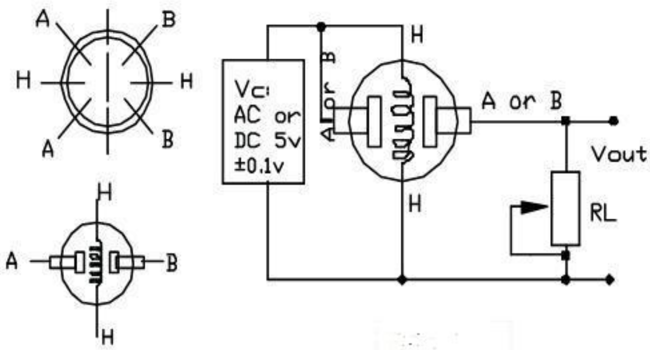


Fig 3.4.1a MQ2 Gas Sensor Fig 3.4.1b Internal Circuit of MQ2

The preferred wiring is to connect both 'A' pins together and both 'B' pins together. It is safer and it is assumed that is has more reliable output results. In the picture, the heater is for +5V and is connected to both 'A' pins. This is only possible if the heater needs a fixed +5V voltage. The variable resistor in the picture is the load-resistor and it can be used to determine a good value. A fixed resistor for the load-resistor is used in most cases. The voltage for the internal heater is very important. Some sensors use 5V for the heater, others need 2V. The heater may not be connected directly to an output-pin of the Microcontroller, since it uses too much current for that. The sensors that use 5V or 6V for the internal heater do get warm. They can easily get 50 or 60 degrees Celsius. After the "burn-in time", the heater needs to be on for about 3 minutes (tested with MQ-2) before the readings become stable.

3.4.2 List of other GAS sensors

1. MQ-2: Sensitive for Methane, Butane, LPG, smoke. This sensor is sensitive for flammable and combustible gasses. The heater uses 5V.
2. MQ-3: Sensitive for Alcohol, Ethanol, smoke. The heater uses 5V.
3. MQ-4: Sensitive for Methane, CNG Gas. The heater uses 5V.
4. MQ-5: Sensitive for Natural gas, LPG. The heater uses 5V.
5. MQ-6: Sensitive for LPG and Butane. The heater uses 5V.
6. MQ-7: Sensitive for Carbon monoxide. The heater uses alternating voltage between 5V and 1.4V.
7. MQ-8: Sensitive to hydrogen gas. The heated uses 5V.
8. MQ-9: Sensitive to Carbon monoxide and other flammable gasses. The heater uses alternating voltage between 5V and 1.5V. If only carbon monoxide is to be tested, then it can set at 1.5V.

3.5 Mobile Detector Circuit

The mobile detector can sense the presence of an activated mobile phone from a distance of 1m. The circuit can detect both the incoming calls, outgoing calls, SMS and video transmission even if the mobile is kept in the silent mode. The RF transmission signal from an activated mobile phone activates a buzzer and the LED starts blinking. The alarm continues until the signal transmission seizes. The circuit is as shown below,

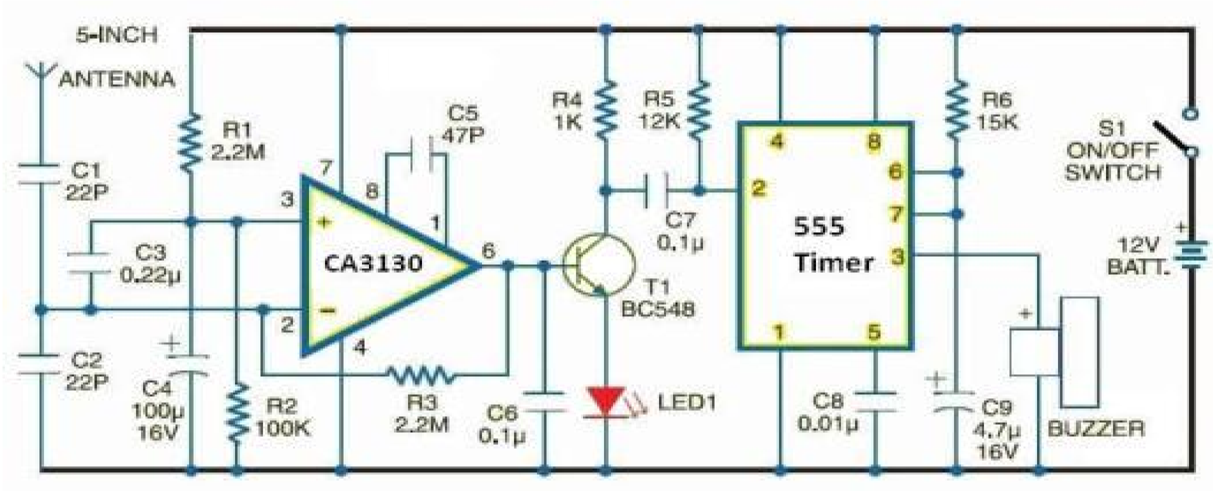


Fig 3.5 Mobile detector circuit

The transmission frequency of mobile phone ranges from 0.9GHz to 3GHz with a wavelength of 3.3cm to 10cm. A 0.22uF disk capacitor is used to capture the RF signals from the mobile phone. The lead length of the capacitor is fixed as 18mm with a spacing of 8mm between the leads to get the desired frequency. The disc capacitor along with the leads acts as a small gigahertz loop antenna to collect the RF signals from the mobile phone.

The op amp IC CA3130 is used in the circuit as a current to voltage converter with a capacitor of 0.22uF connected between its inverting and non inverting inputs. Since the input impedance is very high, very low current at very low frequency is achieved. The output from the pin 6 of the IC is fed to the base of the transistor TI BC548 to increase the voltage level. The transmitter drives the triggering input of the IC 555 used in monostable multivibrator mode. The pulse from pin 3 of IC 555 drives a piezo buzzer.

3.6 Liquid Crystal Display

A liquid-crystal display (LCD) is a flat-panel display or other electronically modulated optical device that uses the light-modulating properties of liquid crystals. Liquid crystals do not emit light directly, instead using a backlight or reflector to produce images in color or monochrome. LCDs are available to display arbitrary images or fixed images with low information content, which can be displayed or hidden, such as preset words, digits, and 7-segment displays, as in a digital clock. They use the same basic technology, except that arbitrary images are made up of a large number of small pixels, while other displays have larger elements. A 16X2 LCD is used in this project.

The 44780 standard requires 3 control lines as well as either 4 or 8 I/O lines for the data bus. The user may select whether the LCD is to operate with a 4-bit data bus or an 8-bit data bus. If a 4-bit data bus is used the LCD will require a total of 7 data lines (3 control lines plus the 4 lines for the data bus). If an 8-bit data bus is used the LCD will require a total of 11 data lines (3 control lines plus the 8 lines for the data bus).

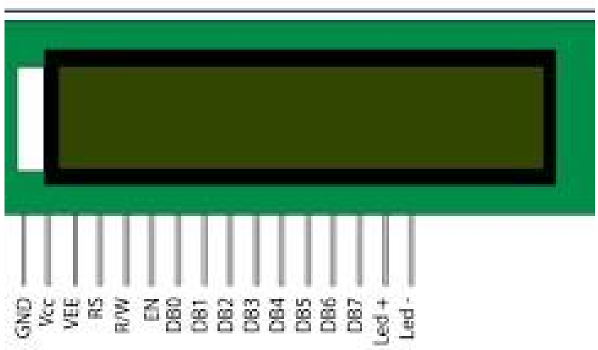


Fig 3.6a LCD 16X2

Fig 3.6b Pin Diagram

The three control lines are referred to as EN, RS, and RW. The EN line is called "Enable." This control line is used to tell the LCD that you are sending it data. To send data to the LCD, your program should make sure this line is low (0) and then set the other two control lines and/or put data on the data bus. When the other lines are completely ready, bring EN high (1) and wait for the minimum amount of time required by the LCD datasheet (this varies from LCD to LCD), and end by bringing it low (0) again.

The RS line is the "Register Select" line. When RS is low (0), the data is to be treated as a command or special instruction (such as clear screen, position cursor, etc.). When RS is high (1), the data being sent is text data which should be displayed on the screen. For example, to display the letter "T" on the screen you would set RS high.

The RW line is the "Read/Write" control line. When RW is low (0), the information on the data bus is being written to the LCD. When RW is high (1), the program is effectively querying (or reading) the LCD. Only one instruction ("Get LCD status") is a read command. All others are write commands--so RW will almost always be low .Finally, the data bus consists of 4 or 8 lines (depending on the mode of operation selected by the user). In the case of an 8-bit data bus, the lines are referred to as DB0, DB1, DB2, DB3, DB4, DB5, DB6, and DB7.

The LCD is connected to a 12V supply. The RS, R/W, EN pins are connected to the P0.15, P0.16, P0.17 of the controller. The RS is set to 1 for data and 0 for command. The LCD is operated in 4 Bit mode and hence the D4, D5, D6, D7 pins are connected to P0.18, P0.19, P0.20, P0.21 respectively. The enable pin is made low to high every 250ms to display character on the screen. The LCD is used to display all the alert messages on the screen.

3.7 Sprinkler and Buzzer

The sprinkler used in this project is mainly if a fire breaks out. The sprinkler is a motor that sucks water and then pumps it out through an outlet. The sprinkler requires draws more current and hence it is activated using a relay. It requires 12V for operation. As long as the fire is detected on the Flame sensor the sprinkler keeps pumping water out.

The buzzer on the other hand is driven by the controller only when any of the five parameters exceed the set threshold value. The buzzer unlike sprinkler works for all the five sensing devices. The mobile ring detector has a buzzer of its own which is driven by the IC 555.

3.8 Global System for Mobile communication (GSM)

Global System for Mobile communications is the most popular standard for mobile phones in the world. Its promoter, the GSM Association, estimate that 82% of the global mobile market uses the standard. GSM is used by over 2 billion people across more than 212 countries and territories. Its ubiquity makes international roaming very common between mobile phone operators, enabling subscribers to use their phones in many parts of the world.

GSM has used a variety of voice CODECs to squeeze 3.1 kHz audio into between 5.6 and 13 kbit/s. Originally, two CODECs named after the types of data channel they were allocated, data channel they used and called them Half Rate (5.6 kbit/s) and Full Rate (13 kbit/s). These used a system based upon linear predictive coding (LPC). In addition to being efficient with bit rates, these CODECs also made it easier to identify more important parts of the audio, allowing the air interface layer to prioritize and better protect these parts of the signal.

There five different cell sizes in a GSM network-macro, micro, Pico, femto and umbrella cells. The coverage area of each cell varies according to the implementation environment. Macro cells can be regarded as cells where the base station antenna is installed on a mast or a building above average roof top level. Micro cells are cells whose antenna height is under average roof top level; they are typically used in urban areas. Picocells are small cells whose coverage diameter is a few dozen meters; they are mainly used indoors. Femtocells are cells designed for use in residential or small business environments and connect to the service provider’s network via a broadband internet connection. Umbrella cells are used to cover shadowed regions of smaller cells and fill in gaps in coverage between those cells.

Cell horizontal radius varies depending on antenna height, antenna gain and propagation conditions from a couple of hundred meters to several tens of kilometers. The longest distance the GSM specification supports in practical use is 35 kilometers (22 mi). Indoor coverage is also supported by GSM and may be achieved by using an indoor picocell base station or an indoor repeater with distributed indoor antennas fed through power splitters to deliver the radio signals from an antenna outdoors to the separate indoor distributed antenna system. These are typically deployed when a lot of call capacity is needed indoors. For example, in shopping centres and airports. However, this is not a prerequisite, since indoor coverage is also provided by in-building penetration of the radio signals from nearby cells.

The modulation used in GSM is Gaussian minimum-shift keying (GMSK), a kind of continuous-phase frequency shift keying. In GMSK, the signal to be modulated onto the carrier is first smoothed with a Gaussian low-pass filter prior to being fed to a frequency modulator, which greatly reduces the interference to neighbouring channels (adjacent channel interference).

3.8.1 SUBSCRIBER IDENTITY MODULE (SIM)

One of the key features of GSM is the Subscriber Identity Module (SIM), commonly known as a SIM card. The SIM is a detachable smart card containing the user's subscription information and phonebook. This allows the user to retain his or her information after switching handsets. Alternatively, the user can also change operators while retaining the handset simply by changing the SIM. Some operators will block this by allowing the phone to use only a single SIM or only a SIM issued by them; this practice is known as SIM locking, and is illegal in some countries.

Fig 3.8.1 SIM Cards

A subscriber can usually contact the provider to remove the lock for a fee, utilize private services to remove the lock or make use of ample software and websites available on the Internet to unlock the handset themselves. While most web sites offer the unlocking for a fee, some do it for free. The locking applies to the handset, identified by its International Mobile Equipment Identity (IMEI) number, not to the account (which is identified by the SIM card). It is always possible to switch to another (non-locked) handset if such a handset is available.

3.8.2 GSM Modems

A modem is a communication device that converts binary into analog acoustic signals for transmission over telephone lines and converts these acoustics signals back into binary form at the receiving end. Conversion to analog signal is known as modulation; conversion back to binary signal is known as demodulation.

In the terminology used in the RS-232C communication standard, modems are DCEs, which mean the connected at one end to a DTE (e.g. computer) device. Low-speed modems are designed to operate asynchronously. Each data frame conforms an asynchronous transmission mechanism.

High-speed modems as well as leased-lines modems use synchronous transmission. The two modems use a common time base and operate continuously at substantially the same frequency and the phase relationship by circuit that monitors the connection.

A half-duplex modem must alternately send and receive signals. Half-duplex allows more of the channel bandwidth to be put to use but slows data communications. A full-duplex modem can simultaneously handle two signals using two carriers to transmit and receive data. Each carrier uses a half of the bandwidth available to it and its modulation.

ASK is not used for data communications because it is very susceptible to electrical noise interference. Low-speed modems use FSK, higher speed modems use PSK, and the very high speed modems use a conjunction of ASK and PSK. The SMS/MMS Gateway requires a connection to an SMSC (Short Messaging Service Centre) to interface with SMS and MMS networks.

3.8.3 WORKING OF GSM MODEM

A GSM modem is a wireless modem that works with GSM wireless networks. A wireless modem is similar to a dial-up modem. The main difference is that a wireless modem transmits data through a wireless network whereas a dial-up modem transmits data through a copper telephone line. Most mobile phones can be used as a wireless modem.

To send SMS messages, first place a valid SIM card into a GSM modem, which is then connected to microcontroller by RS232 cable. After connecting a GSM modem to a microcontroller, you can control the GSM modem by sending instructions to it. The instructions used for controlling the GSM modem are called AT commands. GSM modems support a common set of standard AT commands. In addition to this common set of standard AT commands, GSM modems support an extended set of AT commands. One use of the extended AT commands is to control the sending and receiving of SMS messages.

The following table lists the AT commands that are related to the writing and sending of SMS messages:

AT command	Meaning
+CMGS	Send message
+CMSS	Send message from storage
+CMGW	Write message to memory
+CMGD	Delete message
+CMGC	Send command

AT command	Meaning
+CMMS	More messages to send

Table 3.8.3 GSM AT Commands

It sends the characters you typed to the GSM modem. It then displays the response it receives from GSM modem on the screen.

The GSM Modem comes with a serial interface through which the modem can be controlled using AT command interface. An antenna and a power adapter are provided. The basic communication takes place as shown below,

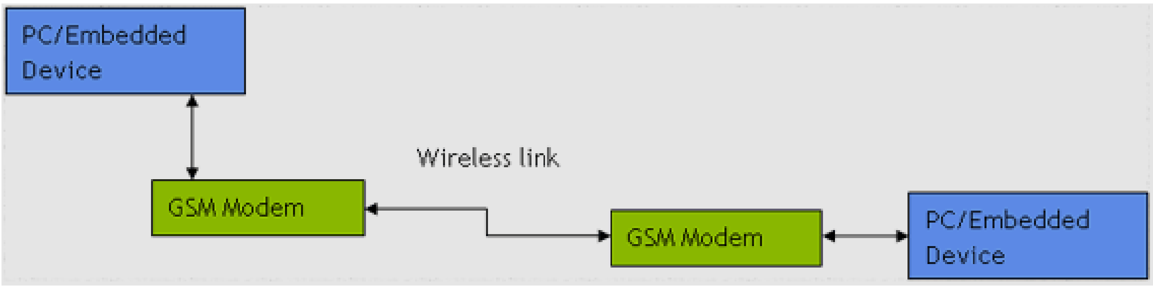


Fig 3.8.3: Basic Architecture of GSM Module

GSM/GPRS Modem-RS232 is built with Dual Band GSM/GPRS engine-SIM900A, works on frequencies 900/ 1800 MHz. The Modem is coming with RS232 interface,which allows you connect PC as well as microcontroller with RS232 Chip(MAX232). The baud rate is configurable from 9600-115200 through AT command. The GSM/GPRS Modem is having internal TCP/IP stack to enable you to connect with internet via GPRS. It is suitable for SMS, Voice as well as DATA transfer application in M2M interface. The onboard Regulated Power supply allows you to connect wide range unregulated power supply . Using this modem,you can make audio calls, SMS, Read SMS, attend the incoming calls and internet ect through simple AT commands

3.9 WIFI Module ESP 8266

ESP8266 offers a complete and self-contained Wi-Fi networking solution, allowing it to either host the application or to offload all Wi-Fi networking functions from another application processor. When ESP8266 hosts the application, and when it is the only application processor in the device, it is able to boot up directly from an external flash. It has integrated cache to improve the performance of the system in such applications, and to minimize the memory requirements. Alternately, serving as a Wi-Fi adapter, wireless internet access can be added to any microcontroller-based design with simple connectivity through UART interface or the CPU AHB bridge interface.

ESP8266 on-board processing and storage capabilities allow it to be integrated with the sensors and other application specific devices through its GPIOs with minimal development up-front and minimal loading during runtime. With its high degree of on-chip integration, which includes the antenna switch balun, power management converters, it requires minimal external circuitry, and the entire solution, including front-end module, is designed to occupy minimal PCB area. Sophisticated system-level features include fast sleep/wake context switching for energy- efficient VoIP, adaptive radio biasing for low-power operation, advance signal processing, and spur cancellation and radio co-existence features for common cellular, Bluetooth, DDR, LVDS, LCD interference mitigation.

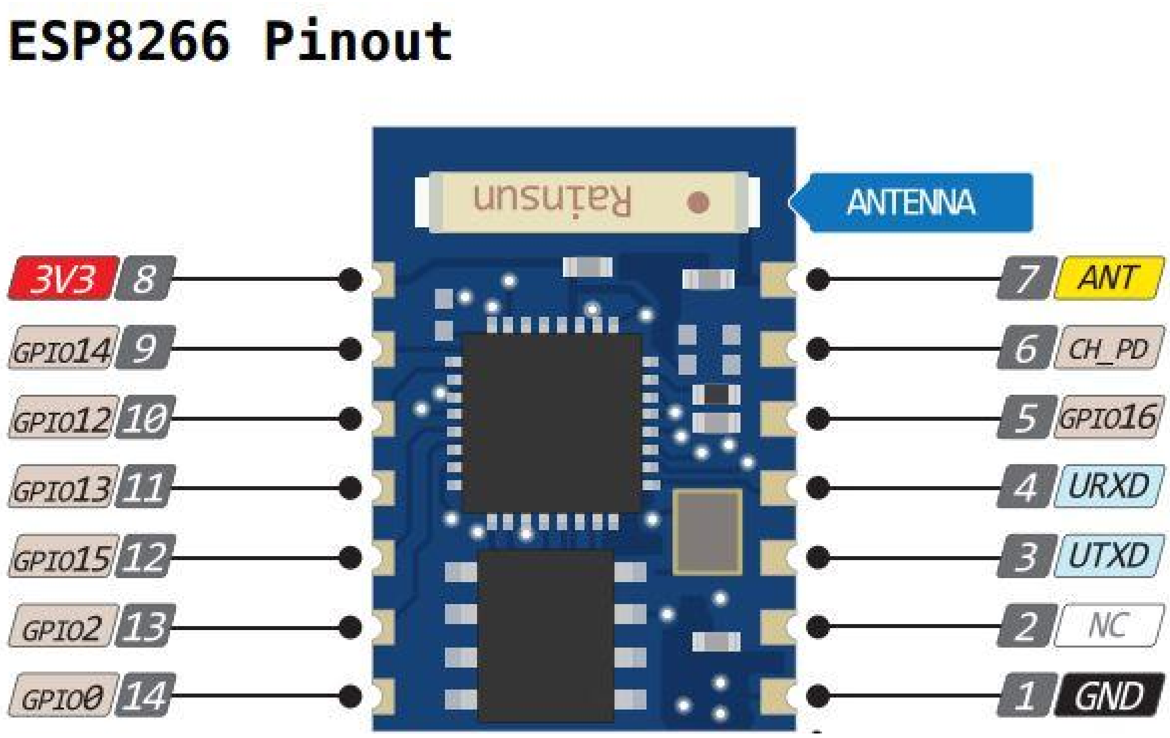


Fig 3.9 ESP822 WIFI Module

3.9.1 Features

The module has the following below mentioned features,

- 802.11 b/g/n protocol.
- Wi-Fi Direct (P2P), soft-AP.
- Integrated TCP/IP protocol stack.
- Integrated TR switch, balun, LNA, power amplifier and matching network.
- Integrated PLL, regulators, and power management units.
- +19.5dBm output power in 802.11b mode.
- Integrated temperature sensor.
- Supports antenna diversity.
- Power down leakage current of < 10uA.
- Integrated low power 32-bit CPU could be used as application processor.
- SDIO 2.0, SPI, UART.
- STBC, 1×1 MIMO, 2×1 MIMO.

- A-MPDU & A-MSDU aggregation & 0.4□s guard interval.
- Wake up and transmit packets in < 2ms
- Standby power consumption of < 1.0mW (DTIM3).

This board has all the pins broken out to the standard, so it is compatible with berg headers, breadboards and dot matrix board. It has a 5V AMS Voltage regulator on board, which means that you can plug in your 5V supply directly and start working with the board. Easy to Access thru PC using USB to TTL converter Board

For IoT space, ESP8266 ESP-12 is a highly integrated chip designed for the needs of a new connected world. It offers a complete and self-contained Wi-Fi networking solution, allowing it to either host the application or to offload all Wi-Fi networking functions from another application processor. ESP8266 ESP-12 has powerful on-board processing and storage capabilities that allow it to be integrated with the sensors and other application specific devices through its GPIOs with minimal development up-front and minimal loading during runtime. Its high degree of on-chip integration allows for minimal external circuitry, and the entire solution, including front-end module, is designed to occupy minimal PCB area.

The VCC and GND pins are connected to 5V and GND respectively. The TX pin of the module is connected to the RX pin i.e P0.8 of the controller and the RX pin of the module is connected to TX pin i.e P0.9 pin of the controller. The WIFI module is programmed to connect to a WIFI hotspot “SMILE” with a password “SMILE123456789”. Once connected the module will continuously send the information on to the cloud and real time monitoring of the system is enabled.

3.10 7805 Voltage Regulator

7805 is a voltage regulator integrated circuit. It is a member of 78xx series of fixed linear voltage regulator ICs. The voltage source in a circuit may have fluctuations and would not give the fixed voltage output. The voltage regulator IC maintains the output voltage at a constant value. The xx in 78xx indicates the fixed output voltage it is designed to provide. 7805 provides +5V regulated power supply. Capacitors of suitable values can be connected at input and output pins depending upon the respective voltage levels

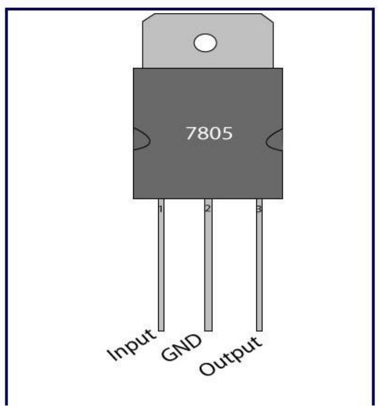


Fig 3.10 IC 7805

The main purpose of the IC 7805 is to provide 5 volt regulated supply. The mains provide an input of 12V, 2A. This will be provided as input to IC 7805 and a regulated 5V, 500mA is obtained. This can be used to provide to the sensors which require the voltage around this range and also the LED block.

3.11 LM317 Variable Voltage Supply Module

The LM317 device is an adjustable three-terminal positive-voltage regulator capable of supplying more than 1.5A over an output-voltage range of 1.25V to 37V. It requires only two external resistors to set the output voltage. The device features a typical line regulation of 0.01% and typical load regulation of 0.1%. It includes current limiting, thermal overload protection, and safe operating area protection.

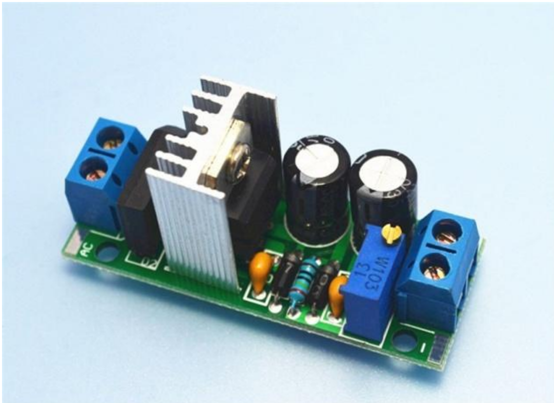


Fig 3.11 LM317 module

The main feature of the module is that it provides an output voltage range, adjustable from 1.25V to 37V. The output current greater than 1.5A. An internal Short-Circuit Current Limiting circuitry. Thermal overload protection and output safe area compensation.

CHAPTER 4 : SOFTWARE REQUIREMENTS

4.1 INTRODUCTION TO KEIL MICRO VISION (IDE)

Keil an ARM Company makes C compilers, macro assemblers, real-time kernels, debuggers, simulators, integrated environments, evaluation boards, and emulators for ARM7/ARM9/Cortex-M3, XC16x/C16x/ST10, 251, and 8051 MCU families.

Keil development tools for the 8051 Microcontroller Architecture support every level of software developer from the professional applications engineer to the student just learning about embedded software development. When starting a new project, simply select the microcontroller you use from the Device Database and the µVision IDE sets all compiler, assembler, linker, and memory options for you.

Keil is a cross compiler. So first we have to understand the concept of compilers and cross compilers. After then we shall learn how to work with keil.

4.2 CONCEPT OF COMPILER

Compilers are programs used to convert a High Level Language to object code. Desktop compilers produce an output object code for the underlying microprocessor, but not for other microprocessors. I.E the programs written in one of the HLL like 'C' will compile the code to run on the system for a particular processor like x86 (underlying microprocessor in the computer). For example compilers for Dos platform is different from the Compilers for Unix platform So if one wants to define a compiler then compiler is a program that translates source code into object code.

The compiler derives its name from the way it works, looking at the entire piece of source code and collecting and reorganizing the instruction. See there is a bit little difference between compiler and an interpreter. Interpreter just interprets whole program at a time while compiler analyses and execute each line of source code in succession, without looking at the entire program.
The advantage of interpreters is that they can execute a program immediately. Secondly programs produced by compilers run much faster than the same programs executed by an interpreter. However compilers require some time before an executable program emerges. Now as compilers translate source code into object code, which is unique for each type of computer, many compilers are available for the same language.

4.3 CONCEPT OF CROSS COMPILER

A cross compiler is similar to the compilers but we write a program for the target processor (like 8051 and its derivatives) on the host processors (like computer of x86). It means being in one environment you are writing a code for another environment is called cross development. And the compiler used for cross development is called cross compiler. So the definition of cross compiler is a compiler that runs on one computer but produces object code for a different type of computer.

4.4 KEIL C CROSS COMPILER

Keil is a German based Software development company. It provides several development tools like

- IDE (Integrated Development environment)
- Project Manager
- Simulator
- Debugger
- C Cross Compiler, Cross Assembler, Locator/Linker

The Keil ARM tool kit includes three main tools, assembler, compiler and linker. An assembler is used to assemble the ARM assembly program. A compiler is used to compile the C source code into an object file. A linker is used to create an absolute object module suitable for our in-circuit emulator.

4.5 Building an Application in µVision2

To build (compile, assemble, and link) an application in µVision2, you must:

1. Select Project -(forexample,166\EXAMPLES\HELLO\HELLO.UV2).
2. Select Project - Rebuild all target files or Build target.µVision2 compiles, assembles, and links the files in your project.

4.6 Creating Your Own Application in µVision2

To create a new project in µVision2, you must:

1. Select Project - New Project.
2. Select a directory and enter the name of the project file.
3. Select Project - Select Device and select an 8051, 251, or C16x/ST10 device from the Device Database™.
4. Create source files to add to the project.

5. Select Project - Targets, Groups, Files. Add/Files, select Source Group1, and add the source files to the project.
6. Select Project - Options and set the tool options. Note when you select the target device from the Device Database™ all special options are set automatically. You typically only need to configure the memory map of your target hardware. Default memory model settings are optimal for most applications.
7. Select Project - Rebuild all target files or Build target.

4.7 Debugging an Application in µVision2

To debug an application created using µVision2, you must:

1. Select Debug - Start/Stop Debug Session.
2. Use the Step toolbar buttons to single-step through your program. You may enter G, main in the Output Window to execute to the main C function.
3. Open the Serial Window using the Serial #1 button on the toolbar.

Debug your program using standard options like Step, Go, Break, and so on.

4.8 Starting µVision2 and Creating a Project

µVision2 is a standard Windows application and started by clicking on the program icon. To create a new project file select from the µVision2 menu Project – New Project…. This opens a standard Windows dialog that asks you for the new project file name. We suggest that you use a separate folder for each project. You can simply use the icon Create New Folder in this dialog to get a new empty folder. Then select this folder and enter the file name for the new project, i.e. Project1. µVision2 creates a new project file with the name

PROJECT1.UV2 which contains a default target and file group name. You can see these names in the Project.

4.9 Window – Files.

Now use from the menu Project – Select Device for Target and select a CPU for your project. The Select Device dialog box shows the µVision2 device data base. Just select the microcontroller you use. We are using for our examples the Philips 80C51RD+ CPU. This selection sets necessary tool Options for the 80C51RD+ device and simplifies in this way the tool Configuration.

4.10 Building Projects and Creating a HEX Files

Typical, the tool settings under Options – Target are all you need to start a new application. You may translate all source files and line the application with a click on the Build Target toolbar icon. When you build an application with syntax errors, µVision2 will display errors and warning messages in the Output Window – Build page. A double click on a message line opens the source file on the correct location in a µVision2 editor window. Once you have successfully generated your application you can start debugging.

After you have tested your application, it is required to create an Intel HEX file to download the software into an EPROM programmer or simulator. µVision2 creates HEX files with each build process when Create HEX files under Options for Target – Output is enabled. You may start your PROM programming utility after the make process when you specify the program under the option Run User Program #1.

4.11 CPU Simulation

µVision2 simulates up to 16 Mbytes of memory from which areas can be mapped for read, write, or code execution access. The µVision2 simulator traps and reports illegal memory accesses. In addition to memory mapping, the simulator also provides support for the integrated peripherals of the various 8051 derivatives. The on-chip peripherals of the CPU you have selected are configured from the Device.

4.12 Database selection

You have made when you create your project target. Refer to page 58 for more Information about selecting a device. You may select and display the on-chip peripheral components using the Debug menu. You can also change the aspects of each peripheral using the controls in the dialog boxes.

4.13 EMBEDDED C

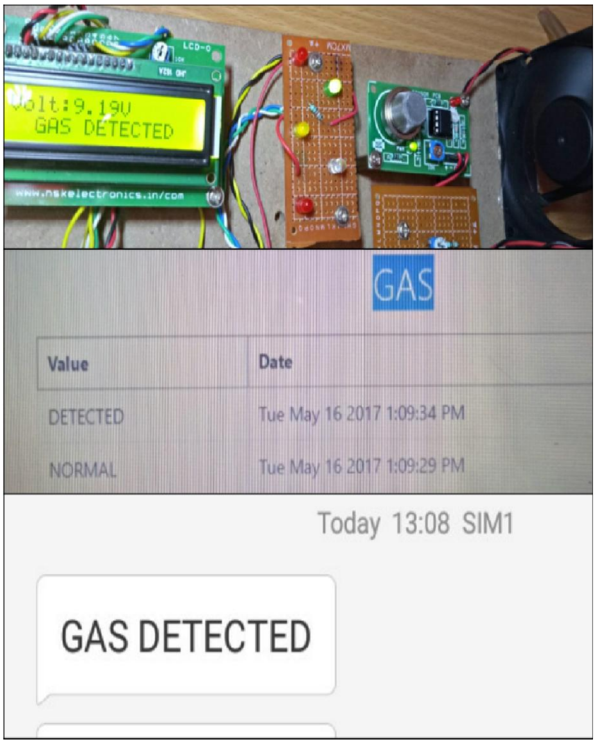
Use of embedded processors in passenger cars, mobile phones, medical equipment, aerospace systems and defense systems is widespread, and even everyday domestic appliances such as dish washers, televisions, washing machines and video recorders now include at least one such device.

Because most embedded projects have severe cost constraints, they tend to use low-cost processors like the 8051 family of devices considered in this book. These popular chips have very limited resources available most such devices have around 256 bytes (not megabytes!) of RAM, and the available processor power is around 1000 times less than that of a desktop processor. As a result, developing embedded software presents significant new challenges, even for experienced desktop programmers. If you have some programming experience - in C, C++ or Java - then this book and its accompanying CD will help make your move to the embedded world as quick and painless as possible.

Chapter 5: Results

The proposed system was tested for its working and all the sensors were found to work efficiently and the following results were recorded and are shown below.

5.1 Gas leakage

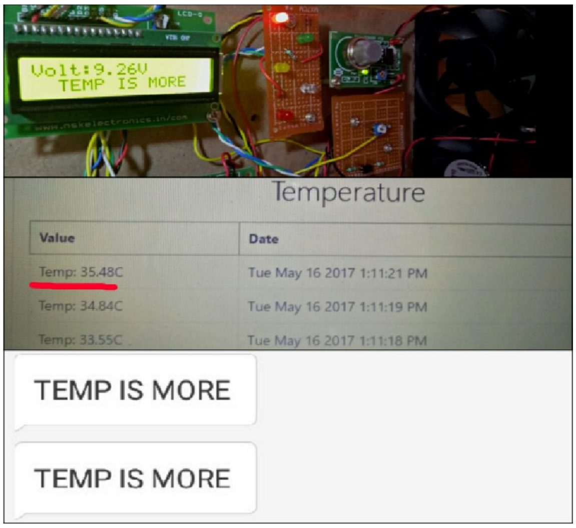


1. Once the gas leakage is sensed by the MQ-2 sensor, the message "GAS DETECTED" is displayed on the LCD screen and the green LED glows. Simultaneously the exhaust fan turns ON.
2. The status of the gas leakage is continuously updated to the cloud via IoT and displays the message "DETECTED" on sensing the gas.
3. An alert message "GAS DETECTED" is sent to the subscribed mobile number.

5.2 Fire Detection

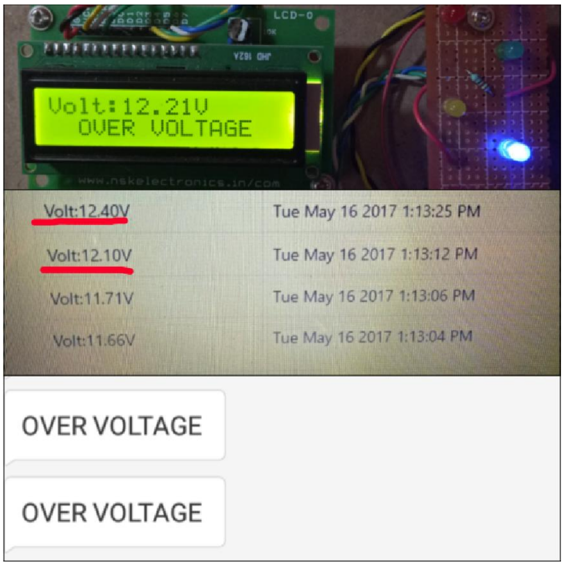
1. Whenever a fire mishap is sensed by the EE522 sensor, the message "FIRE DETECTED" is displayed on the LCD screen and yellow LED glows. Simultaneously the sprinkler turns ON.
2. The status of the fire sensor is continuously updated to the cloud via IoT and displays the message "FIRE DETECTED" on sensing the fire.

5.3 Temperature



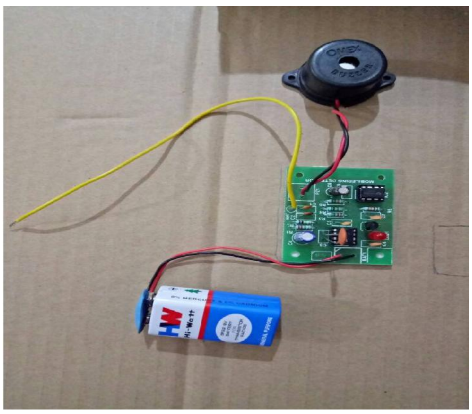
1. Whenever temperature crosses 35^0 C threshold, it is sensed by the LM35 sensor, the message "TEMP IS MORE" is displayed on the LCD screen and red LED glows. Simultaneously the fan turns ON.
2. The real time value of the temperature is continuously updated to the cloud via IoT.
3. An alert message "TEMP IS MORE" is sent to the subscribed mobile number once the threshold is crossed.

5.4 Over voltage



1. Whenever voltage crosses 12V threshold, it is indicated by the message "OVER VOLTAGE" on the LCD screen and blue LED glows.
2. The real time value of the voltage is continuously updated to the cloud via IoT.
3. An alert message "OVER VOLTAGE" is sent to the subscribed mobile number once the threshold is crossed.

5.5 Mobile Ring detector



1. Once the detector circuit senses radiation from mobile anywhere in its surrounding, the buzzer starts beeping and the red LED glows .
2. The LED and buzzer keep going until the radiations have completely seized.

Chapter 6: Conclusion

Industrial Safety Automation using IOT has been implemented successfully. We have successfully integrated the various part of the industry to the digital world. The project is cost effective and can be easily implemented for other real time applications. It can be realised with the usage of less power. This project is secure and user friendly and can be employed by the government in large scale to help home automations too. By automating an industry, a safe working environment can be created with the available advanced mechanism and the entire system integrated into one network. The industry is substantially safe from fire accidents, voltage fluctuations and gas leakage.

6.1 Scope for future

This project has tremendous scope in the future. It can be developed and made more user friendly with additional features such as:

- 3-D projector can be implemented with smart phones.
- It can be implemented for household applications and home automations.
- Hologram system can be implemented to create an augmented reality

Chapter 7: Bibliography

- International Journal of Computer Science and Mobile Computing A Monthly Journal of Computer Science and Information Technology

- Hemlata Tomer Int. Journal of Engineering Research and Application ISSN : 2248-9622, Vol. 5, Issue 7, (Part -2) July 2015, pp.115-120

- Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), 2011 2nd International Conference

- International Journal of Modern Engineering Research (IJMER)

- International Journal of Science, Engineering and Technology Research (IJSETR), Volume 4, Issue 10, October 2015

- IRE Transactions on Electron Devices (Volume: 7, Issue: 4, Oct. 1960)

- IEEE TRANSACTIONS ON POWER ELECTRONICS, VOL. 20, NO. 3, MAY 2005

- International Journal of Scientific and Research Publications, Volume 4, Issue 6, June 2014

- Proceedings of National Conference on New Horizons in IT - NCNHIT 2013

- International Research Journal of Engineering and Technology (IRJET)

TEXT BOOKS REFERED:

1. "The 8051 Microcontroller and Embedded systems" by Muhammad Ali Mazidi and Janice Gillispie Mazidi , Pearson Education.