A Project Report

On

INTELLIGENT BABY SWING CRIB

UDP PROJECT REPORT
GROUP CODE – 130001142

Submitted by

RONAK SHARMA	(110513111008)
PRASHANT LOHIYA	(110513111011)
KRUPALI VAISHNAV	(110513111013)
SNEHA CHAVDA	(110513111015)

In partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

ELECTRONICS AND COMMUNICATION DEPARTMENT

ALPHA COLLEGE OF ENGGINEERING & TECHNOLOGY
AT. KHATRAJ, TA. KALOL, DIST. GANDHINAGAR-38272

Gujarat Technological University, Ahmedabad

April , 2014

ALPHA COLLEGE OF ENGGINEERING & TECHNOLOGY

ELECTRONICS AND COMMUNICATION DEPARTMENT 2014

CERTIFICATE

Date: 28/4/2014

This is to certify that the dissertation entitled "INTELLIGENT BABY SWING CRIB" has been carried out by SHARMA RONAKKUMAR BHAILALBHAI under my guidance in fulfillment of the degree of Bachelor of Engineering in ELECTRONICS AND COMMUNICATION ENGINEERING (8th Semester) of Gujarat Technological University, Ahmedabad during the academic year 2013-14.

Guide:	Head of the Department
PROF. RONAK GADARIA (INTERNAL GUIDE)	PROF. RONAK GADARIA

ACKNOWLEDGEMENT

I would like to place on record my deep sense of gratitude to Prof. Ronak Gadaria, HOD of Electronics & Communication Engineering, Alpha College of Engineering & Technology, Khatraj, Gandhinagar, for his generous guidance, help and useful suggestions.

I express my sincere gratitude to Prof. Ronak Gadaria(Asst. professor and In charge head of Dept. of Electronics & Communication Engineering, Alpha College of Engineering & Technology, Khatraj, Gandhinagar). for his stimulating guidance, continuous encouragement and supervision throughout the course of present work.

I also wish to extend my thanks to Prof. Ronak Gadaria & Prof. Devashish Raval and other colleagues for attending my seminars and for their insightful comments and constructive suggestions to improve the quality of this project work.

Signature(s) of Students

Ronak Sharma ________________

Prashant Lohiya ________________

Krupali Vaishnav ________________

Sneha Chavda ________________

ABSTRACT

This project is about swinging crib for baby. This project is not a new invention but sure that it is a new creation. In this project we have used different devices of electronics and try to make it fully automatic. We also called this project a new system or a machine because after all the system which can reduce a man power is called a machine.

Now, how the machine will work? In this system when a baby will cry then the noise detector or a range microphone which is placed in the specific area in crib, will send signal to micro controller and that microcontroller will pass signal to servo motor to start and it will start swinging. So, this system will work automatically. Now in this system we can also adjust manual time setting and crib will continue to swing. By default we adjust 10 minutes for swinging, but if baby still continue to cry then it will continue to swing. Now if baby still keep crying then the signal from servo motor will send signal to the microcontroller and that will active GSM module to send a message to baby's guardian(to whom we want to send).This message is stored in the controller. In this, user has to give a call to the number which is in the gsm module , that call be terminated automatically and after that , that number will be stored as guardian number whom we want to send a message. That gsm module will give a call back to that number for confirmation. For every start of crib , it will ask for the call to that number. This full task is done by the controller and GSM module by inter facing them with each other. Extra feature of this system is lcd & keyboard that is used for showing the swinging time and battery life if we use external battery for power supply and the key board for manual time setting. That is also noticeable that when message is sending then there will b also buzzer which give sound for home , which useful for person who is working in home or near the baby , there will b some musical instrument for entertainment for baby and some little lighting for attractive purpose and for concentration for baby. We have put a fan for air circulation, if temperature will increase above 35 degree Celsius the fan will b automatically start.

TABLE OF CONTENT

LIST OF FIGURE:-

1.1 INSPIRATION:-

We want to share our idea which we have applied to our project . First of all tell me that ,we show many children around us, there is any one person allotted to care of that child. So, we thought that ,we can help the guardian who is far from family and who is from busy family. By this project we can reduce the man power and attention from the child. We show around us that there is always one person behind a bay to look after it. Many a time we also have to take a charge of that situation, and it feels like boring to sit near the crib and have to swing the crib by string, so to solve such a problem we try for it. This full idea is ours, we four friends sit for lunch in college and by talking such matter we got idea , and after that we have decided to do so.

For more guidance we took help of Google and we got a crib detail, that was of CHINA but that crib was electric crib not automatic, and we want to make it automatic and start our work. By starting to work on project we have come extra ideas to add to the crib , which make our project intelligent, and finally we keep our project name as INTELLIGENT BABY SWING CRIB.

Now talk about the system, When a baby will sleep in this crib then guardian can do its work, guardian has not to worry about a baby when it wake up from the sleep. When it happens then crib will swing automatically, when guardian will go out of the home and at that time if baby will cry more than 10 minutes then system will send a message to the guardian. Now for more study go onwards.

1.2 ABOUT PROJECT

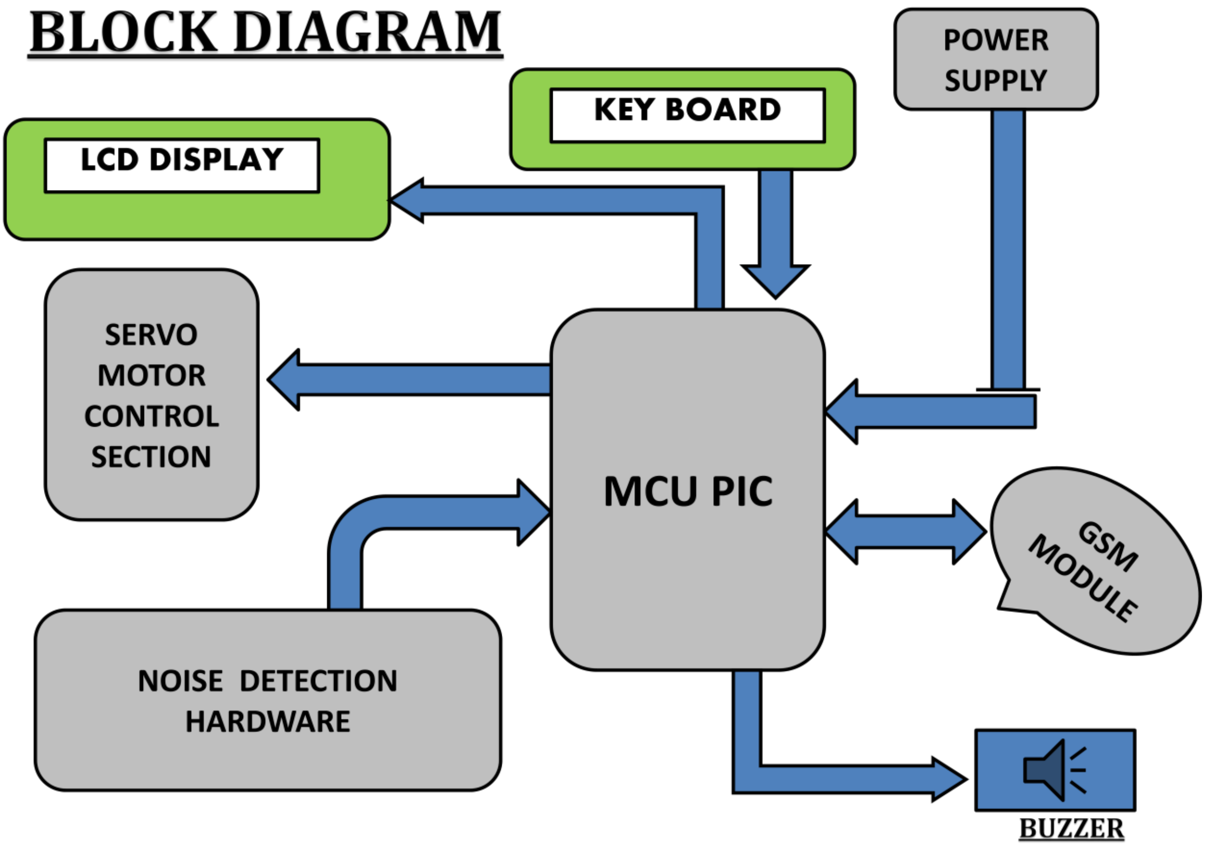


Figure 1: block diagram

COMPONENT

- PIC Controller .
- Servo motor .
- GSM Module .
- Noise detection hardware .(Sensor)
- Baby crib .
- Buzzer .
- LCD .

1.3 FEATURES

- Automatic start swinging.
- GSM Module.
- Mannual time setting .
- Buzzer system .
- EEPROM use for storing mobile number .
- External battery .
- Display battery life.
- Easy to operate .

Chapter-2 : PIC CONTROLLER

2.1 ABOUT PIC CONTROLLER :-

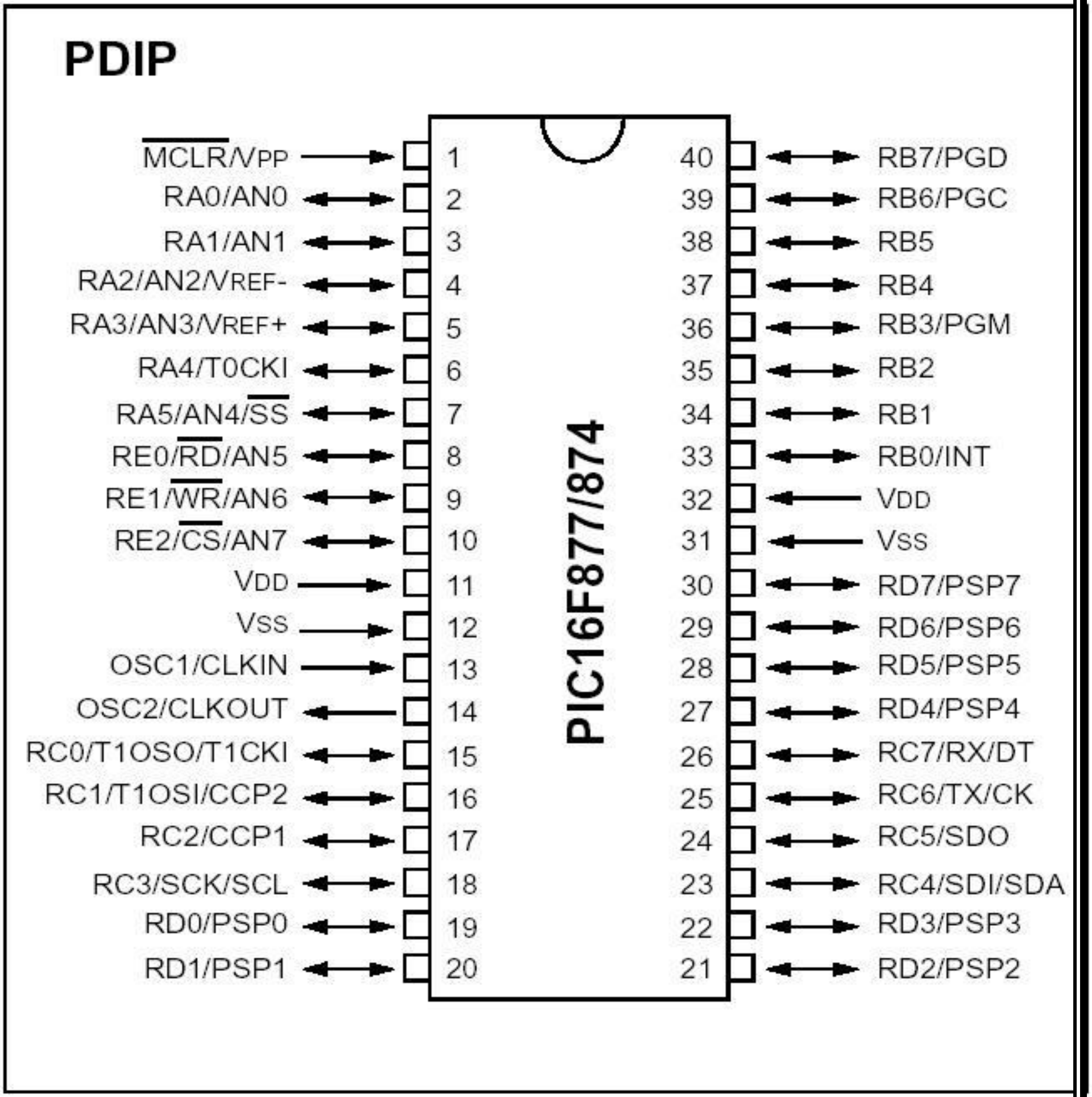


Figure 2: Pin Diagram Of PIC

2.2 BLOCK DIAGRAM OF PIC:-

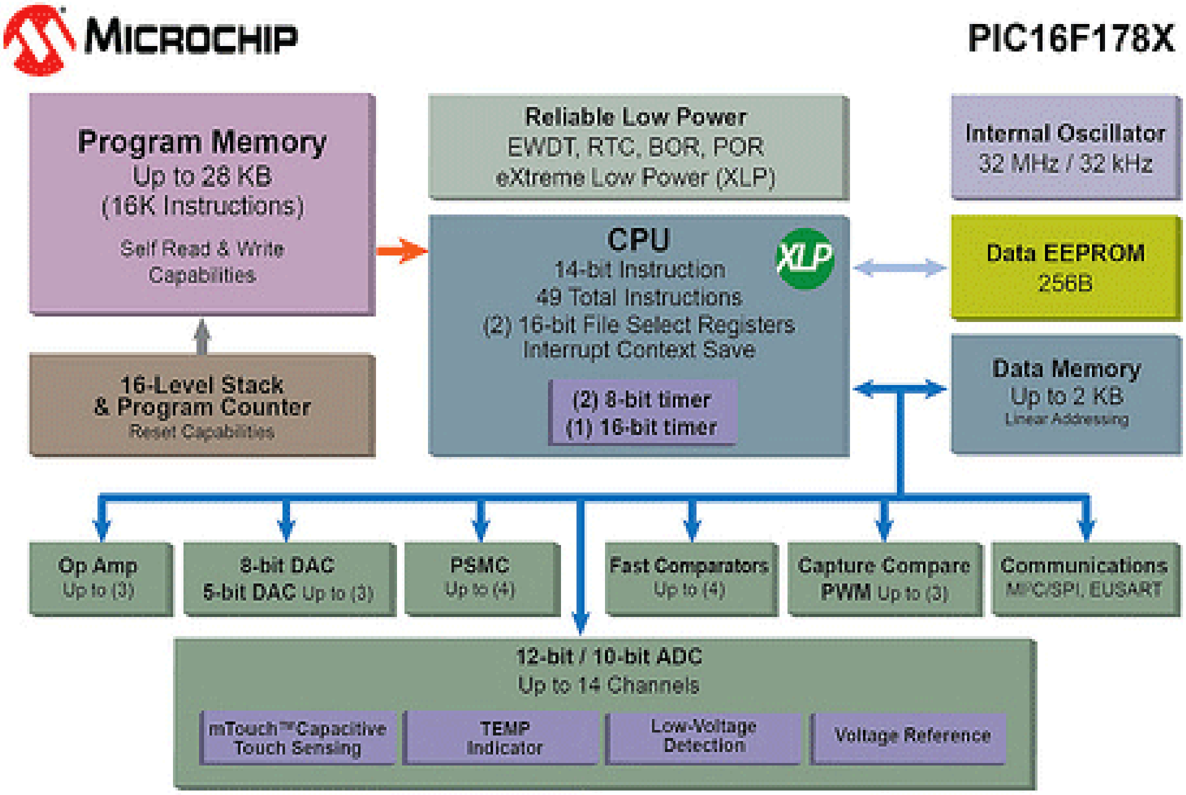
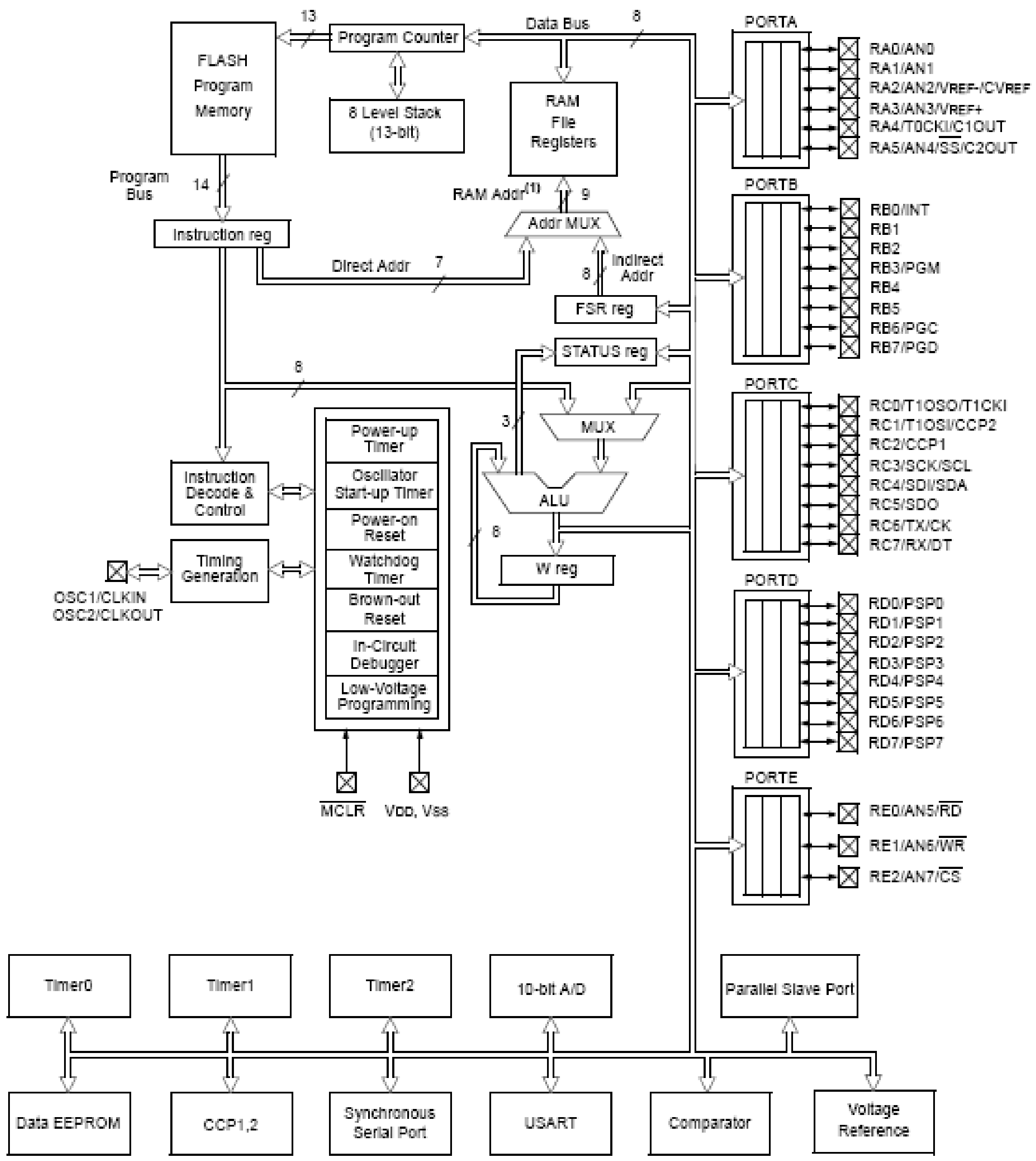


Figure 3: Block Diagram Of PIC Courtesy of Google image



Device	Program FLASH	Data Memory	Data EEPROM
PIC16F874A	4K words	192 Bytes	128 Bytes
PIC16F877A	8K words	368 Bytes	256 Bytes

Note 1: Higher order bits are from the STATUS register.

Figure 4: Descriptive Block Diagram of PIC

High-Performance RISC CPU:

- Only 35 single-word instructions to learn
- All single-cycle instructions except for program branches, which are two-cycle

- Operating speed: DC – 20 MHz clock input DC – 200 ns instruction cycle

- Up to 8K x 14 words of Flash Program Memory,
- Up to 368 x 8 bytes of Data Memory (RAM),
- Up to 256 x 8 bytes of EEPROM Data Memory

- Pin out compatible to other 28-pin or 40/44-pin
- PIC16CXXX and PIC16FXXX microcontrollers

2.3 Peripheral Features:

• Timer0: 8-bit timer/counter with 8-bit prescaler

• Timer1: 16-bit timer/counter with prescaler, can be incremented during Sleep via external crystal/clock

• Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler

• Two Capture, Compare, PWM modules
- Capture is 16-bit, max. resolution is 12.5 ns
- Compare is 16-bit, max. resolution is 200 ns
- PWM max. resolution is 10-bit

• Synchronous Serial Port (SSP) with SPI™ (Master mode) and I2C™ (Master/Slave)

• Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address Detection

• Parallel Slave Port (PSP) – 8 bits wide with external RD, WR and CS controls (40/44-pin only)

Analog Features:

- 10-bit, up to 8-channel Analog-to-Digital Converter (A/D)

- Brown-out Reset (BOR)

- Analog Comparator module with:

- Two analog comparators

- Programmable on-chip voltage reference (VREF) module

- Programmable input multiplexing from device inputs and internal voltage reference

- Comparator outputs are externally accessible

Special Microcontroller Features:

- 100,000 erase/write cycle Enhanced Flash program memory typical

- 1,000,000 erase/write cycle Data EEPROM memory typical

- Data EEPROM Retention > 40 years

- Self-reprogrammable under software control

- In-Circuit Serial Programming™ (ICSP™) via two pins

- Single-supply 5V In-Circuit Serial Programming

- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation

- Programmable code protection

- Power saving Sleep mode

- Selectable oscillator options
- In-Circuit Debug (ICD) via two pins

2.5 DEVICE OVERVIEW:-

This document contains device specific information
about the following devices:

- PIC16F873A
- PIC16F874A
- PIC16F876A
- PIC16F877A

PIC16F873A/876A devices are available only in 28-pin packages, while PIC16F874A/877A devices are available in 40-pin and 44-pin packages. All devices in the PIC16F87XA family share common architecture with the following differences:

- The PIC16F873A and PIC16F874A have one-half of the total on-chip memory of the PIC16F876A and PIC16F877A

- The 28-pin devices have three I/O ports, while the 40/44-pin devices have five

- The 28-pin devices have fourteen interrupts, while the 40/44-pin devices have fifteen

- The 28-pin devices have five A/D input channels, while the 40/44-pin devices have eight

- The Parallel Slave Port is implemented only on the 40/44-pin devices The available features are summarized in Table 1-1.Block diagrams of the PIC16F873A/876A andPIC16F874A/877A devices are provided in Figure 1-1and Figure 1-2, respectively. The pin outs for these device families are listed in Table 1-2 and Table 1-3.Additional information may be found in the PIC micro ®Mid-Range Reference Manual (DS33023), which may be obtained from your local Microchip Sales Representative or downloaded from the microchip web site. The Reference Manual should be considered a complementary document to this data sheet and is highly recommended reading for a better understanding of the device architecture and operation of the peripheral modules.

MEMORY ORGANIZATION :-

There are three memory blocks in each of thePIC16F87XA devices. The program memory and data memory have separate buses so that concurrent access can occur and is detailed in this section. The EEPROM data memory block is detailed in **Section 3.0"Data EEPROM and Flash Program Memory"**. Additional information on device memory may be found in the PIC micro® Mid-Range MCU Family Reference Manual (DS33023).

Program Memory Organization :-

The PIC16F87XA devices have a 13-bit program counter capable of addressing an 8K word x 14 bit program memory space. The PIC16F876A/877A devices have 8K words x 14 bits of Flash program memory, while PIC16F873A/874A devices have 4K words x 14 bits. Accessing a location above the physically implemented address will cause a wraparound. The Reset vector is at 0000h and the interrupt vector is at 0004h.

DATA EEPROM AND FLASH PROGRAM MEMORY:-

The data EEPROM and Flash program memory is readable and writable during normal operation (over the full VDD range). This memory is not directly mapped in the register file space. Instead, it is indirectly addressed through the Special Function Registers. There are six SFRs used to read and write this memory:

- EECON1
- EECON2
- EEDATA
- EEDATH
- EEADR
- EEADRH

When interfacing to the data memory block, EEDATA holds the 8-bit data for read/write and EEADR holds the address of the EEPROM location being accessed. These devices have 128 or 256 bytes of data EEPROM(depending on the device), with an address range from 00h to FF h. On devices with 128 bytes, addresses from80h to FF h are unimplemented and will wraparound to the beginning of data EEPROM memory. When writing to unimplemented locations, the on-chip charge pump will be turned off. When interfacing the program memory block, the EEDATA and EEDATH registers form a two-byte word that holds the 14-bit data for read/write and the EEADR and EEADRH registers form a two-byte word that holds the 13-bit address of the program memory location being accessed. These devices have 4 or 8K words of program Flash, with an address range from 0000h to0FFFh for the PIC16F873A/874A and 0000h to 1FFFh for the PIC16F876A/877A. Addresses above the range of the respective device will wraparound to the beginning of program memory. The EEPROM data memory allows single-byte read and write. The Flash program memory allows single-word reads and four-word block writes. Program memory write operations automatically perform an erase-before write on blocks of four words. A byte write in data EEPROM memory automatically erases the location and writes the new data (erase-before-write). The write time is controlled by an on-chip timer. The write/erase voltages are generated by an on-chip charge pump, rated to operate over the voltage range of the device for byte or word operations. When the device is code-protected, the CPU may continue to read and write the data EEPROM memory. Depending on the settings of the write-protect bits, the device may or may not be able to write certain blocks of the program memory; however, reads of the program memory are allowed. When code-protected, the device programmer can no longer access data or program memory; this does NOT inhibit internal reads or writes.

Reading Data EEPROM Memory :-

To read a data memory location, the user must write the address to the EEADR register, clear the EEPGD control bit (EECON1<7>) and then set control bit RD (EECON1<0>). The data is available in the very next cycle in the EEDATA register; therefore, it can be read in the next instruction (see Example 3-1). EEDATA will
hold this value until another read or until it is written to by the user (during a write operation).
The steps to reading the EEPROM data memory are:

1. Write the address to EEADR. Make sure that the address is not larger than the memory size of the device.

2. Clear the EEPGD bit to point to EEPROM data memory.

3. Set the RD bit to start the read operation.

The steps to write to EEPROM data memory are:

1. If step 10 is not implemented, check the WR bit to see if a write is in progress.

2. Write the address to EEADR. Make sure that the address is not larger than the
memory size of the device.

3. Write the 8-bit data value to be programmed in the EEDATA register.

4. Clear the EEPGD bit to point to EEPROM data memory.

5. Set the WREN bit to enable program operations.

6. Disable interrupts (if enabled).
7. Execute the special five instruction sequence:

   - Write 55h to EECON2 in two steps (first to W, then to EECON2)

   - Write AA h to EECON2 in two steps (first to W, then to EECON2)

   - Set the WR bit

8. Enable interrupts (if using interrupts).

9. Clear the WREN bit to disable program operations.

10. At the completion of the write cycle, the WR bit is cleared and the EEIF interrupt
flag bit is set.

Data EEPROM Writing Memory :-

To write an EEPROM data location, the user must first write the address to the EEADR register and the data to the EEDATA register. Then the user must follow a specific write sequence to initiate the write for each byte. The write will not initiate if the write sequence is not exactly followed (write 55h to EECON2, write A Ah to
EECON2, then set WR bit) for each byte. We strongly recommend that interrupts be disabled during this code segment (see Example 3-2). Additionally, the WREN bit in EECON1 must be set to enable write. This mechanism prevents accidental writes to data EEPROM due to errant (unexpected)code execution (i.e., lost programs). The user should keep the WREN bit clear at all times, except when updating EEPROM. The WREN bit is not cleared by hardware After a write sequence has been initiated, clearing the WREN bit will not affect this write cycle. The WR bit will be inhibited from being set unless the WREN bit is set. At the completion of the write cycle, the WR bit is cleared in hardware and the EE Write Complete Interrupt Flag bit (EEIF) is set. The user can either enable this interrupt or poll this bit. EEIF must be cleared by software.

3.6 Writing to Flash Program Memory

Flash program memory may only be written to if the destination address is in a segment of memory that is not write-protected, as defined in bits WRT1:WRT0 of the device configuration word (Register 14-1). Flash program memory must be written in four-word blocks. A block consists of four words with sequential addresses, with a lower boundary defined by an address, where EEADR<1:0> = 00. At the same time, all block writes to program memory are done as erase and write operations. The write operation is edge-aligned and cannot occur across boundaries. To write program data, it must first be loaded into the buffer registers (see Figure 3-1). This is accomplished by first writing the destination address to EEADR and EEADRH and then writing the data to EEDATA and EEDATH. After the address and data have been set up, then the following sequence of events must be executed:

1. Set the EEPGD control bit (EECON1<7>).

2. Write 55h, then AA h, to EECON2 (Flash programming sequence).

3. Set the WR control bit (EECON1<1>). All four buffer register locations **MUST** be written to with correct data. If only one, two or three words are being written to in the block of four words, then a read from the program memory location(s) not being written to must be performed. This takes the data from the program location(s) not being written and loads it into the EEDATA and EEDATH registers. Then the sequence of events to transfer data to the buffer registers must be
executed.

To transfer data from the buffer registers to the program memory, the EEADR and EEADRH must point to the last location in the four-word block (EEADR<1:0> = 11).

Then the following sequence of events must be
executed:

1. Set the EEPGD control bit (EECON1<7>).

2. Write 55h, then AA h, to EECON2 (Flash programming sequence).

3. Set control bit WR (EECON1<1>) to begin the write operation. The user must follow the same specific sequence to initiate the write for each word in the program block, writing each program word in sequence (00,01,10,11). When the write is performed on the last word (EEADR<1:0> = 11), the block of four words are automatically erased and the contents of the buffer registers are written into the program memory. After the “BSF EECON1,WR” instruction, the processor requires two cycles to set up the erase/write operation. The user must place two NOP instructions after the WR bit is set. Since data is being written to buffer registers, the writing of the first three words of the block appears to occur immediately. The processor will halt internal operations for the typical 4 ms, only during the cycle in which the erase takes place (i.e., the last word of the four-word block). This is not Sleep mode as the clocks
and peripherals will continue to run. After the write cycle, the processor will resume operation with the third
instruction after the EECON1 write instruction. If the
sequence is performed to any other location, the action
is ignored.

CHAPTER-3 :- SERVO MOTOR

3.1 Description:-

Servo refers to an error sensing feedback control which is used to correct the performance of a system. Servo or RC **Servo Motors** are DC motors equipped with a servo mechanism for precise control of angular position. The RC servo motors usually have a rotation limit from 90° to 180°. Some **servos** also have rotation limit of 360° or more. But servos do not rotate continually. Their rotation is restricted in between the fixed angles. Servo systems use the error sensing negative feedback method to provide precise angular motion. Servo Motors are used where precise control on angular motion is needed. Servo motors are widely used in the field of Robotics to design robotic arms, palms, legs and so on. They are also used in RC toys like RC helicopter, airplanes and cars. The interfacing of servo motor using PIC microcontroller has been explained here. Readers are advised to go through the article on Servo Motors to learn basic mechanism and control of servo motor.

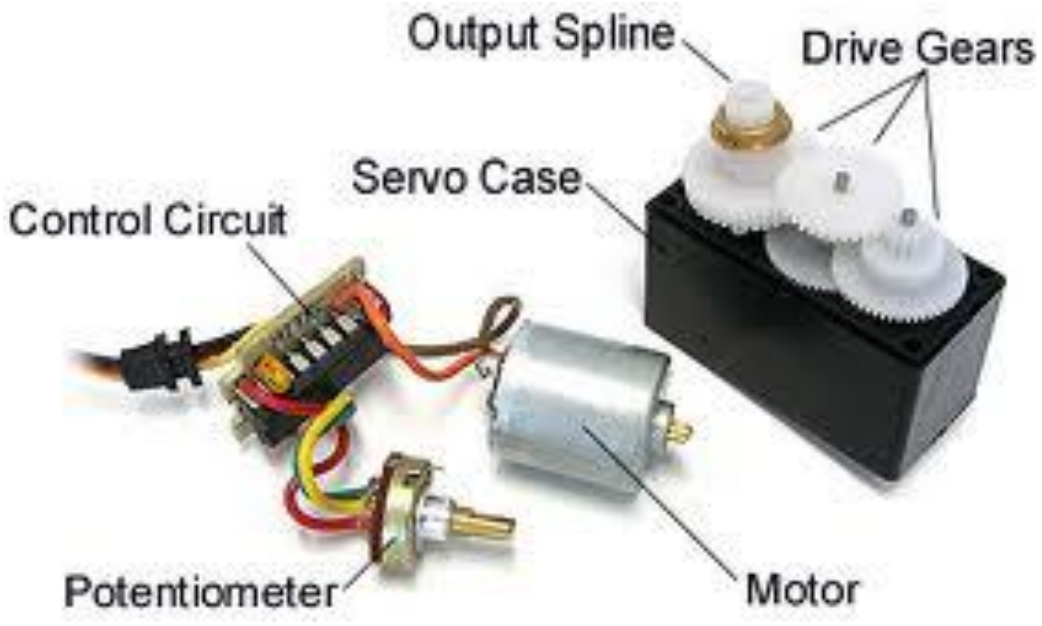


Figure 5: Component of Servo motor

3.2 INSTRUCTION

Servo Motor wiring and plugs :-

The Servo Motors come with three wires or leads. Two of these wires are to provide ground and positive supply to the servo DC motor. The third wire is for the control signal. These wires of a servo motor are color coded. The red wire is the DC supply lead and must be connected to a DC voltage supply in the range of 4.8 V to 6V. The black wire is to provide ground. The color for the third wire (to provide control signal) varies for different manufacturers. It can be yellow (in case of Hitec), white (in case of Futaba), brown etc.
Futaba provides a J-type plug with an extra flange for proper connection of the servo. Hitec has an S-type connector. A Futaba connector can be used with a Hitec servo by clipping of the extra flange. Also a Hitec connector can be used with a Futaba servo just by filing off the extra width so that it fits in well.

Hitec splines have 24 teeth while Futaba splines are of 25 teeth. Therefore splines made for one servo type cannot be used with another. Spline is the place where a servo arm is connected. It is analogous to the shaft of a common DC motor.
Unlike DC motors, reversing the ground and positive supply connections does not change the direction (of rotation) of a servo. This may, in fact, damage the servo motor. That is why it is important to properly account for the order of wires in a servo motor.

A Servo motor has three wire terminals : two of these wires are to provide ground and positive supply to the Servo DC motor, while the third wire is for the control signal. These wires of a servo motor are color coded. The servo motor can be driven only when PWM (pulse width modulated) signals are provided to the control terminal The total pulse duration for a typical servo motor should be of 20 milliseconds. The on-time duration of the control signal varies from 1ms to 2ms. This on-time variation provides angular variation from 0 to 180 degree..

. For example, if the servo motor should be positioned at 45° angle, the desired output control pulse can be obtained as follows:

180° angular displacement is achieved by the pulse duration = 1 ms

1° angular displacement is achieved by the pulse duration of = 1 /180 ms

45° angular displacement is achieved by the pulse duration of = (1/180) x 45 = **0.25 ms**

So total on-time pulse will be = 1ms + 0.25ms =**1.25 ms**

3.3 WORKING

Servo Motor Control :-

The servo motor can be moved to a desired angular position by sending PWM (pulse width modulated) signals on the control wire. The servo understands the language of pulse position modulation. A pulse of width varying from 1 millisecond to 2 milliseconds in a repeated time frame is sent to the servo for around 50 times in a second. The width of the pulse determines the angular position.

For example, a pulse of 1 millisecond moves the servo towards 0°, while a 2 milliseconds wide pulse would take it to 180°. The pulse width for in between angular positions can be interpolated accordingly. Thus a pulse of width 1.5 milliseconds will shift the servo to 90°.

It must be noted that these values are only the approximations. The actual behaviour of the servos differs based on their manufacturer.
A sequence of such pulses (50 in one second) is required to be passed to the servo to sustain a particular angular position. When the servo receives a pulse, it can retain the corresponding angular position for next 20 milliseconds. So a pulse in every 20 millisecond time frame must be fed to the servo.

Power supply for Servo :-

The servo requires a DC supply of 4.8 V to 6 V. For a specific servo, its voltage rating is given as one of its specification by the manufacturer. The DC supply can be given through a battery or a regulator. The battery voltage must be closer to the operating voltage of the servo. This will reduce the wastage of power as thermal radiation. A switched regulator can be used as the supply for better power efficiency. Learn more about working of a servo motor through exclusive images at the Insight about servo motor.

Selection of a Servo :-

The typical specifications of servo motors are torque, speed, weight, dimensions, motor type and bearing type. The motor type can be of 3 poles or 5 poles. The pole refers to the permanent magnets that are attached with the electromagnets. 5 pole servos are better than 3 pole motor because they provide better torque.
The servos are manufactured with different torque and speed ratings. The torque is the force applied by the motor to drive the servo arm. Speed is the measure that gives the estimate that how fast the servo attains a position. A manufacturer may compromise torque over speed or speed over torque in different models. The servos with better torque must be preferred.
The weight and dimensions are directly proportional to the torque. Obviously, the servo having more torque will also have larger dimensions and weight. The selection of a servo can be made according to the torque and speed requirements of the application. The weight and dimension may also play a vital role in optimizing the selection such as when a servo is needed for making an RC airplane or helicopter.

3.4 FIGURE LAY OUT

Interference and Noise Signal :-

The PWM signal is given to the servo by the control wire. The noise or interference signals from the surrounding electronics or other servos can cause positional errors. To eliminate this problem the control signals are supplied after amplification. This will suppress the noise and interference signals.

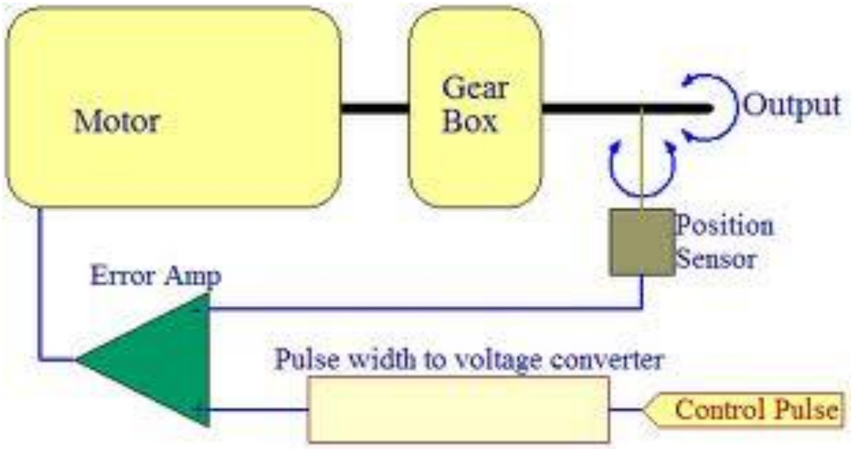


Figure 6: Block Diagram of Servomotor

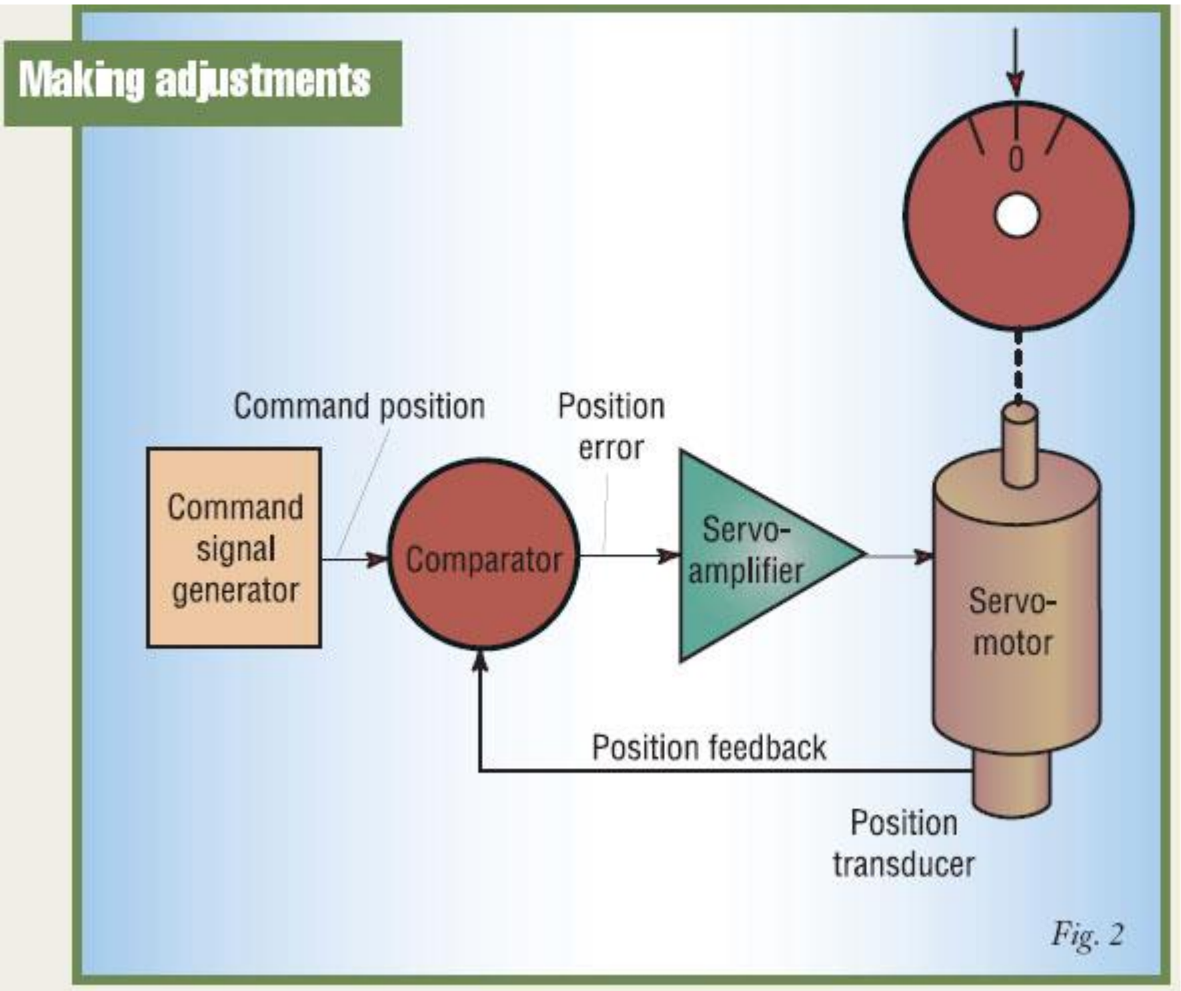


Fig. 2

The comparator compares the 0° command with the position feedback; the result is called position error.

Figure 7: Adjustment of Servo motor

A more complex application of a closed-loop control system is a motion control system that has a positioning loop and a velocity loop. ill. 1 (below) shows an example of this type of motion control loop used to drive a servomotor that moves a welding index table. The servomotor in this application has an encoder attached to its shaft that's used to provide velocity data as well as positioning data. The pulse train from the encoder is sent to the frequency-to-voltage converter, and the output voltage signal from the converter is sent to the summing junction. A position set point is set in the up/down counter, which is converted from a digital value to an analog voltage. The voltage is sent to the summing junction as a reference

signal. The summing junction compares the position reference signal to the velocity feedback signal and produces an error signal that's sent to the op amp as the inverting input signal. If the error is negative, it means the motor hasn't moved the table into the correct position, and additional voltage will be sent to the motor from the output signal from the op amp. The output signal from the op amp must go through a pulse-width modulator (PWM) circuit, a control logic circuit, and a motor control circuit.

When the motor shaft moves the table, the encoder will indicate that the table is at a new position. The position loop compares the set point position (the desired position) in the up/down counter buffer with the actual position that's reported by the encoder. The amount of difference is converted from a digital signal to an analog voltage and sent to the summing junction. The velocity loop compares the amount of actual error to the amount of possible error and determines how close the indexing table is to its destination. The closer the table gets to its destination position, the smaller the amount of signal voltage is sent to the op amp and to the motor and it will slow the table. When the encoder indicates that the indexing table is at the correct position, the set point signal and the feedback signal will cancel each other and the resulting signal value will be zero, which means that no voltage is sent to the motor so it will stop at that position.

CHAPTER:-4 GSM MODULE

4.1 ABOUT GSM

SIM 900

GSM Modem

This GSM Modem can accept any GSM network operator SIM card and act just like a mobile phone with its own unique phone number. Advantage of using this modem will be that you can use its RS232 port to communicate and develop embedded applications. Applications like SMS Control, data transfer, remote control and logging
can be developed easily. The modem can either be connected to PC serial port directly or to any microcontroller. It can be used to send and receive SMS or make/receive voice calls. It can also be used in GPRS mode to connect to internet and do many applications for data logging and control. In GPRS mode you can also
connect to any remote FTP server and upload files for data logging. This GSM modem is a highly flexible plug and play quad band GSM modem for direct and easy integration to RS232 applications. Supports features like Voice, SMS, Data/Fax, GPRS and integrated TCP/IP stack.

4.2 Applications

- SMS based Remote Control & Alerts
- Security Applications
- Sensor Monitoring
- GPRS Mode Remote Data Logging

4.3 Features

- Highly Reliable for 24x7 operation with SMPS power supply adapter / Matched Antenna
- Status of Modem Indicated by LED
- Simple to Use & Low Cost
- Quad Band Modem supports all GSM operator SIM cards

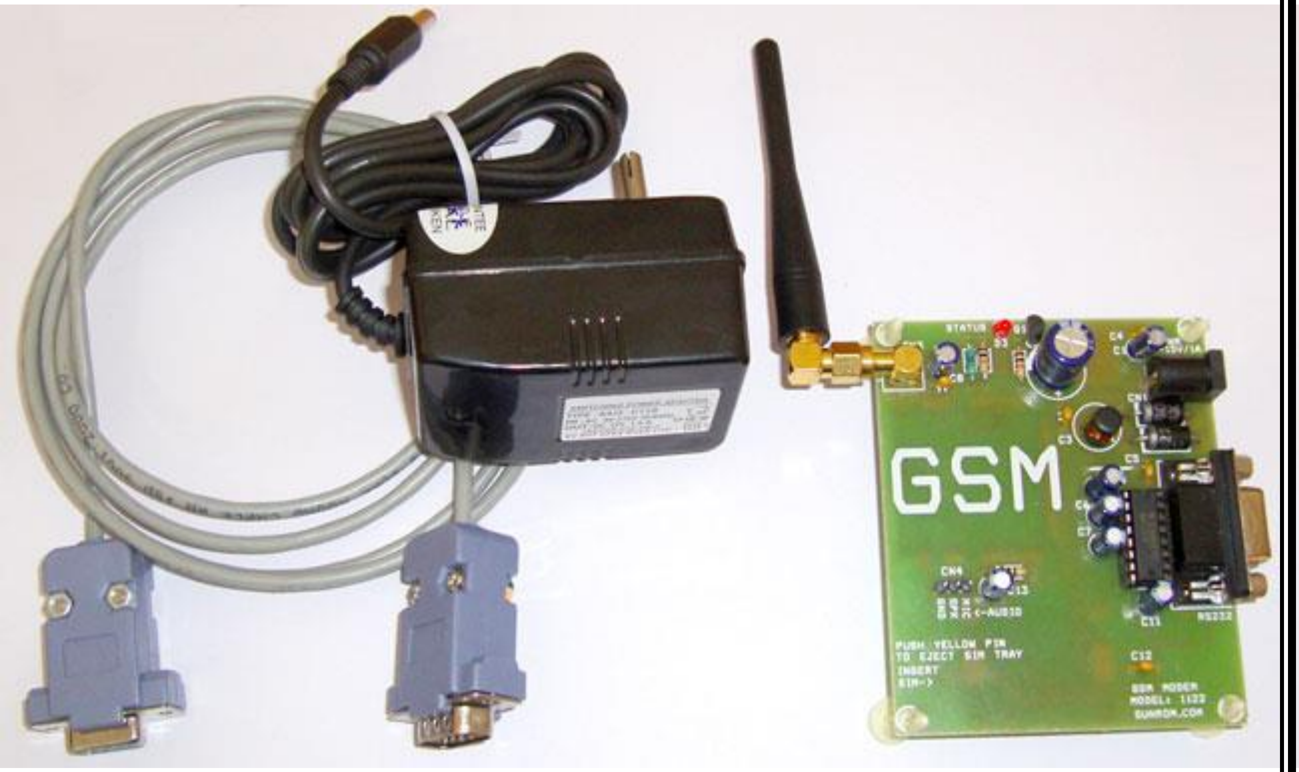

GSM
STATUS
RS232
PUSH YELLOW PIN TO EJECT SIM TRAY
INSERT SIM->
GSM MODEM MODEL: 1122 SUNROM.COM

Figure 8: Component and Kit of GSM module

INTRODUCTION :-

This user's manual describes the operation of the RTD GSM900 integrated global GSM wireless modem designed for mobile, marine, and automotive applications.

- SIM 900 – IT IS A GSM MODULE WHICH WE USE
- To interface SIM900 with controller , there are different software are used.
- Using that software we send AT-COMMANDS to module and operate it or can say interface with it
- There are some AT-COMMANDS which we are going to use are listed below

Features

Some of the key features of the GSM900 include:

- Low power Dual band Siemens TC35 cellular engine, GSM900/1800Mh
- 9,6/14,4 k bit/s data rate, group 3 faxes, SMS and SMS cell broadcast
- Onboard SIM-card socket for 3V standard cards
- 16C550 UART interfaces to host computer
- Supports COM1,COM2,COM3,COM4 or COM x
- Available IRQ's 2,5,6,7,10,11,12,14,15
- Status LED indicating GSM activity and status
- 16 TTL I/O's 8 outputs 8 inputs
- 5V only operation, 2.3W typical
- Wide operating temperature range –20 to + 70C guaranteed
- Onboard temperature sensor
- Fully PC/104 compliant, IDAN versions available
- The following paragraphs briefly describe the major features of the GSM35. A more detailed discussion is included in Chapter 4 (Hardware description) The boards installation is described in Chapter 2 (Board Installation).

4.4 GSM cellular modem :-

The Real Time Devices GSM900 wireless GSM modem unit provides a direct and reliable GSM connection to stationary or GSM 900/1800 mobile fields around the world. GSM connectivity is achieved using the SiemensTC35 engine. This unit works in the 900/1800MHz band supporting GSM02.22 network and service provider personalisation. Connect any standard GSM antenna directly to the OSX connector of the GSM900. The antenna should be connected to the TC35 using a flexible50-Ohm antenna cable. In IDAN installations the antenna connection is brought to the front side of the IDAN-frame. The antenna used should meet the following specifications: Frequency 890-910MHz (TX), 935- 960MHz (RX); Impedance 50 Ohms; VSWR 1,7:1 (TX) 1,9:1 (RX); Gain <1,5dB references to 1/2-dipole; 1W power (c w) max 2W peak at 55 degrees Centigrade.

GSM900 RTD Finland O y A SIM-card socket is located on the solder side of the module. The card can only be removed while the TC35 has been placed in shutdown mode. The GPRS35 is also available using the MC35 GPRS Modem. It supports all the features of the GSM35 and, on top, the advantages of the fast GPRS technology. The MC35 based GPRS modem GPRS35 is available now. During the 3Q 2002 RTD Finland will also release a triple band GPRS module that will operate in the 1900MHz band. The part number will be GPRS45. Engineering samples will be available in August 2002.

16C550 compatible UART :-

Communication to the GSM35 board is performed through a standard UART channel. This onboard serial port leaves the other system serial ports free for the user. All operating systems will recognize and support this 16C550 standard UART, and therefore no special communication drivers are needed to receive data from your GSM35 board. The address and interrupt of your serial channels can be individually set with the onboard jumper fields.

I/O interfaces :-

The GSM35 can be controlled and monitored from the software through two dedicated I/O registers. A special I/O connector is available for the user to connect to the general-purpose TTL level digital I/O.

Mechanical description :-

The GSM35 is designed on a PC/104 form factor. An easy mechanical interface to both PC/104 and RTD IDAN systems can be achieved. Stack your GSM35 directly on a PC/104 compatible CPU module using the onboard mounting holes and standoffs.

Connector description :-

The GPS and GSM antenna interfaces use an OSX type miniature coaxial connector. Connect your antenna directly to the GSM35 antenna connectors, or use a short cable inside your enclosure to connect to a feed through connector to allow connection of the antennas to the wall of your enclosure. All I/O connections are made using header type terminals.

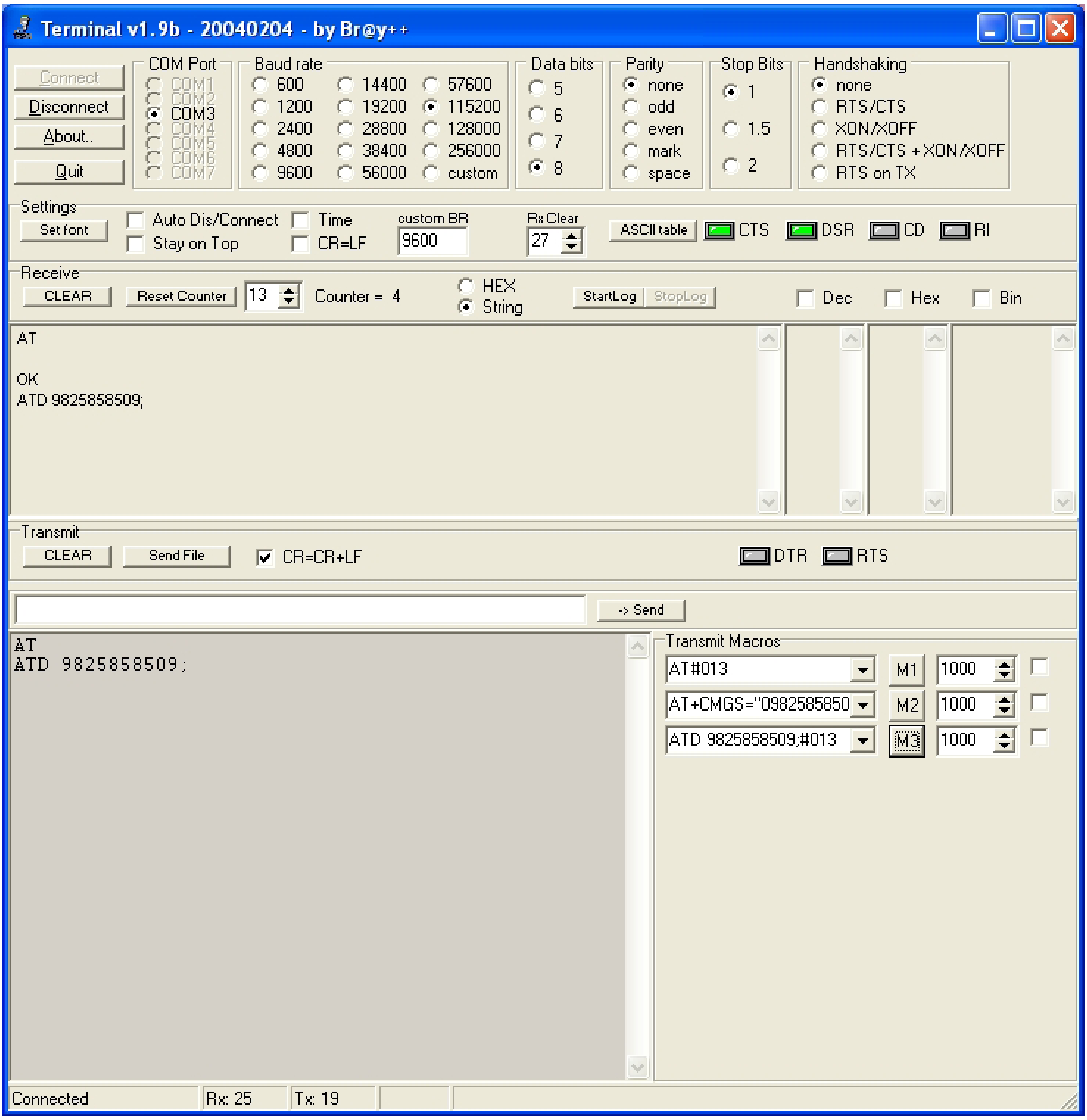
APPLICATION NOTE DETAILS

GSM DEVICE	SIM900
FIRMWARE	R11.00
ADDITIONAL NOTE	AT COMMANDS MUST BE UPPER CASE
COMPILED BY	MARK COXEN
EMAIL	MARK@OTTO.CO.ZA
MOBILE	+27 82 888 6275

Using GSM Modem with PC

If you wish to use GSM modem with PC then connect serial cable to PC and power it on. Use the supplied serial cable to connect to PC’s serial port. Use Hyper terminal software which

comes with Windows XP or use any other Terminal software with following settings.



COM Port: ○ COM1 ○ COM2 ● COM3 ○ COM4 ○ COM5 ○ COM6 ○ COM7

Baud rate: ○ 600 ○ 1200 ○ 2400 ○ 4800 ○ 9600 ○ 14400 ○ 19200 ○ 28800 ○ 38400 ○ 56000 ○ 57600 ● 115200 ○ 128000 ○ 256000 ○ custom

Data bits: ○ 5 ○ 6 ○ 7 ● 8

Parity: ● none ○ odd ○ even ○ mark ○ space

Stop Bits: ● 1 ○ 1.5 ○ 2

Handshaking: ● none ○ RTS/CTS ○ XON/XOFF ○ RTS/CTS+XON/XOFF ○ RTS on TX

□ Auto Dis/Connect □ Stay on Top □ Time □ CR=LF

custom BR 9600

Rx Clear 27

Receive: 13 Counter = 4 ○ HEX ● String □ Dec □ Hex □ Bin

```
AT

OK
ATD 9825858509;
```

☑ CR=CR+LF

```
AT
ATD 9825858509;
```

Transmit Macros

AT#013	M1	1000	□
AT+CMGS="0982585850	M2	1000	□
ATD 9825858509;#013	M3	1000	□

Connected | Rx: 25 | Tx: 19

4.5 AT-COMMANDS

COMMAND	**FUNCTION**
AT+CMGF=1	sets text mode (0 sets Packet Data Mode)
AT+CSCS="GSM"	sets GSM-character text mode
AT+CNMI=2,1,0,0	new message indications - Default is set
AT+CPBS="ON"	sets the phone book memory storage
AT+CPBW=1,"+27*********",145,"Module No"	set the sim card phone number into memory location 1 on SIM-Card
AT+CSCS?	Check that the operator SMS centre number is set
AT+CSCA="+27.....",145	check that operator service centre number is set
AT+CSMP=49,255,0,241&W	sets the parameter for SMS delivery reports and saves this set-ting
AT+CNMI=2,1,0,1,0&W	second parameter to set the SMS delivery deport and saves the setting
AT+CSAS=0	save SMS settings TO PROFILE "0" OR "1"
SAVE A MESSAGE TO MEMORY	EXPLANATION
AT+CMGW	Saves message to SIM memory
>your message < ctrl-z>	Type the info of the message to be stored
+CMGW: 1	The location of the stored message is returned
SENDING AND SMS FROM MEMORY	EXPLANATION
AT+CMSS=1,"+27…",145	Location of stored message to send, destination number must be set
DELETE MESSAGES	EXPLANATION
AT+CMGDA="DEL READ"	Deletes all read messages stored on the SIM card
AT+CMGDA="DEL UNREAD"	Deletes all unread messages stored on the SIM card
AT+CMGDA="DEL SENT"	Deletes all sent messages stored on the SIM card
AT+CMGDA="DEL UNSENT"	Deletes all unsent messages stored on the SIM card
AT+CMGDA="DEL INBOX"	Deletes all received messages stored on the SIM card
AT+CMGDA="DEL ALL"	Deletes all messages stored on the SIM card

SAVE A MESSAGE TO MEMORY	***EXPLANATION***
AT+CMGW	Saves message to SIM memory
>your message < ctrl-z>	Type the info of the message to be stored
+CMGW: 1	The location of the stored message is returned

SENDING AND SMS FROM MEMORY	***EXPLANATION***
AT+CMSS=1,"+27…",145	Location of stored message to send, destination

number must be set

DELETE MESSAGES	***EXPLANATION***
AT+CMGDA="DEL READ"	Deletes all read messages stored on the SIM card
AT+CMGDA="DEL UNREAD"	Deletes all unread messages stored on the SIM card
AT+CMGDA="DEL SENT"	Deletes all sent messages stored on the SIM card
AT+CMGDA="DEL UNSENT"	Deletes all unsent messages stored on the SIM card
AT+CMGDA="DEL INBOX"	Deletes all received messages stored on the SIM card
AT+CMGDA="DEL ALL"	Deletes all messages stored on the SIM card

Sample Code for Interfacing with microcontroller for sending SMS

Connect MCU TXD/RXD through MAX232 so your MCU can communicate with GSM Modem.

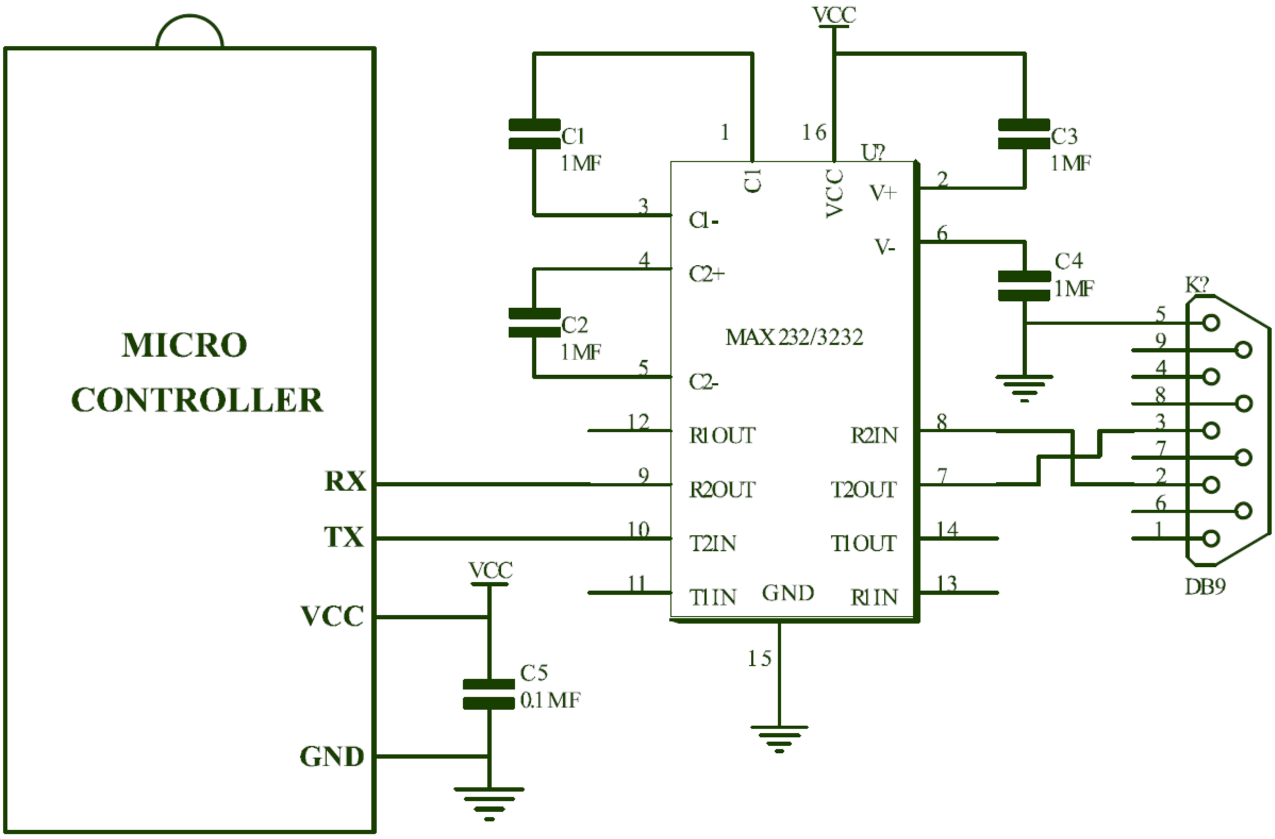


Figure 9: Interface of GSM with PIC

Then use following reference code to send SMS. Before you go ahead with MCU interfacing, it is best practice to first try all these commands with PC with the use of Terminal software. Then make a note of all AT commands and then proceed with

development.

```
void main()
```

```
{
initADC(); // setup ADC
serialInit(); // setup 9600 serial communication
while(1)
{
printf("AT\n");
delayms(2000); // 2 sec delay
printf("AT+CMGF=1\n");
delayms(2000); // 2 sec delay
printf("AT+CMGS=\"09825858509\"\n");
delayms(2000); // 2 sec delay
printf ("CH#1=%bu ", getADC(1)); // sends ADC value as SMS
putchar(26); // Ctrl-Z indicates end of SMS and transmit
message
delayms(2000); // 2 sec delay
}
```

CHAPTER :- 5 NOISE DETECTOR

5.1 INTRODUCTION

Voice Activity Detector (VAD) is used for detection silence intervals.
Principle of functioning of detector is based on methods of adaptive filtrations, allowing tracing dynamically appearance of fragments of noise signal. Depending on conditions of the channel, mips and quality detection, it’s possible different realizations of variants Voice Activity detectors. It allows using them effectively on the background of broadband noises and channel distortion. Such detectors can be effectively used in coders as well as in other applications (IP telephony, Call service centers), where it is important to reduce the rate of bit transmission and save processing resources during the temporary intervals to corresponding silence.

Features :-

ƒ Fast adaptation to changing of channel distortion and external noises;
ƒ Operation with low SNR;

ƒ Easy integration with target applications.

Signal requirement :-

ƒ Signal format: PCM 16-bits;
ƒ 8 kHz, ..., 32 kHz sampling rate.

5.2 WORKING PRINCIPLE

BLOCK DIAGRAM OF VAD (voice activity detection) :-

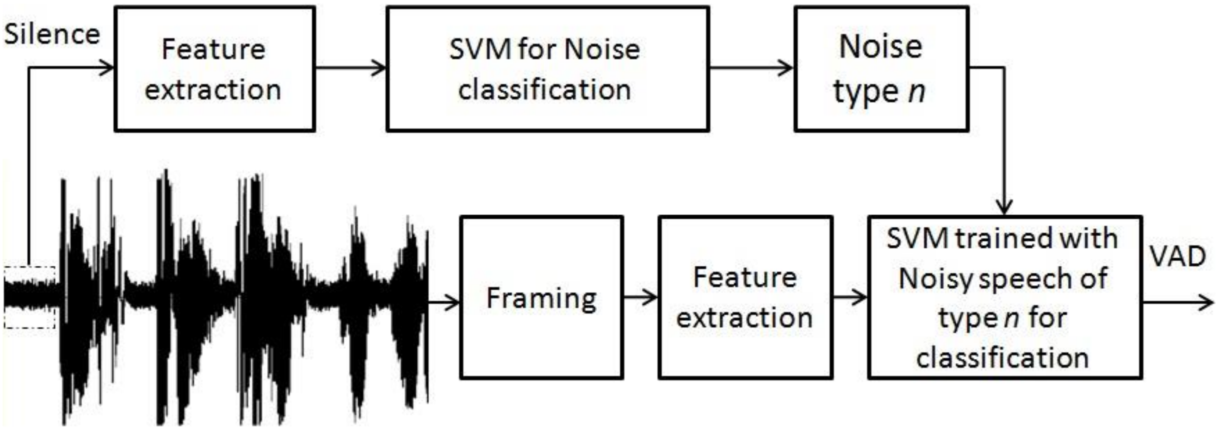
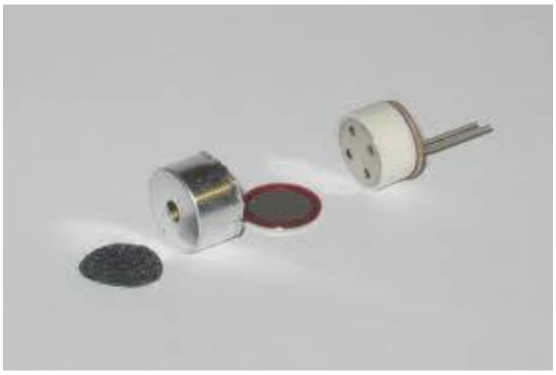

Silence
Feature extraction
SVM for Noise classification
Noise type *n*
Framing
Feature extraction
SVM trained with Noisy speech of type *n* for classification
VAD

Figure 10: Block Diagram of VAD

IMAGES OF SOME SMALL MICRO PHONE :-



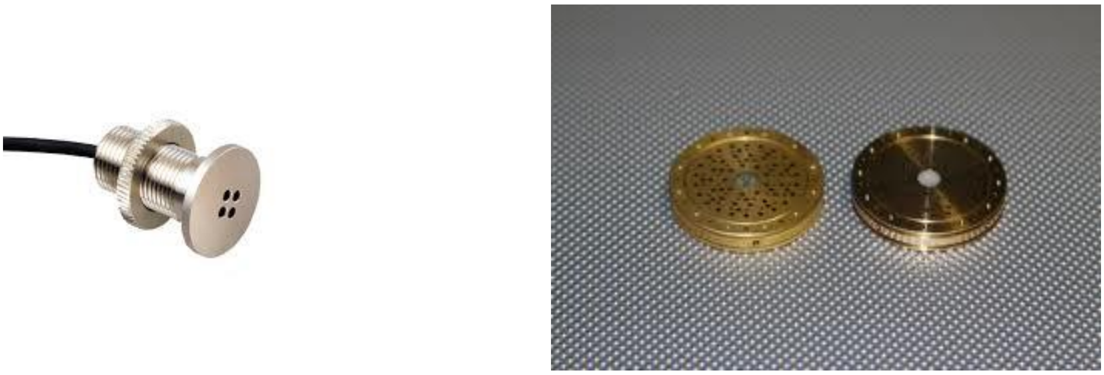


Figure 11: Model of Microphone

CHAPTER :- 6 DC MOTOR

A **DC motor** relies on the fact that like magnet poles repel and unlike magnetic poles attract each other. A coil of wire with a current running through it generates an electromagnetic field aligned with the center of the coil. By switching the current on or off in a coil its magnet field can be switched on or off or by switching the direction of the current in the coil the direction of the generated magnetic field can be switched 180°. A simple *DC motor* typically has a stationary set of magnets in the stator and an armature with a series of two or more windings of wire wrapped in insulated stack slots around iron pole pieces (called stack teeth) with the ends of the wires terminating on a commutator. The armature includes the mounting bearings that keep it in the center of the motor and the power shaft of the motor and the commutator connections. The winding in the armature continues to loop all the way around the armature and uses either single or parallel conductors (wires), and can circle several times around the stack teeth. The total amount of current sent to the coil, the coil's size and what it's wrapped around dictate the strength of the electromagnetic field created. The sequence of turning a particular coil on or off dictates what direction the effective electromagnetic fields are pointed. By turning on and off coils in sequence a rotating magnetic field can be created. These rotating magnetic fields interact with the magnetic fields of the magnets (permanent orelectromagnets) in the stationary part of the motor (stator) to create a force on the armature which causes it to rotate. In some DC motor designs the stator fields use electromagnets to create their magnetic fields which allow greater control over the motor. At high power levels, DC motors are almost always cooled using forced air.

The commutator allows each armature coil to be activated in turn. The current in the coil is typically supplied via two brushes that make moving contact with the commutator. Now, some brushless DC motors have electronics that switch the DC current to each coil on and off and have no brushes to wear out or create sparks.

Different number of stator and armature fields as well as how they are connected provide different inherent speed/torque regulation characteristics. The speed of a DC motor can be controlled by changing the voltage applied to the armature. The introduction of variable resistance in the armature circuit or field circuit allowed speed control. Modern DC motors are often controlled by power electronics systems which adjust the voltage by "chopping" the DC current into on and off cycles which have an effective lower voltage.

Since the series-wound DC motor develops its highest torque at low speed, it is often used in traction applications such as electric locomotives, and trams. The DC motor was the mainstay of electric traction drives on both electric and diesel-electric locomotives, street-cars/trams and diesel electric drilling rigs for many years. The introduction of DC motors and an electrical grid system to run machinery starting in the 1870s started a new second Industrial Revolution. DC motors can operate directly from rechargeable batteries, providing the motive power for the first electric vehicles and today's hybrid cars and electric cars as well as driving a host of cordless tools. Today DC motors are still found in applications as small as toys and disk drives, or in large sizes to operate steel rolling mills and paper machines.

If external power is applied to a DC motor it acts as a DC generator, a dynamo. This feature is used to slow down and recharge batteries on hybrid car and electric cars or to return electricity back to the electric grid used on a street car or electric powered train line when they slow down. This process is called regenerative braking on hybrid and electric cars. In diesel electric locomotives they also use their DC motors as generators to slow down but dissipate the energy in resistor stacks. Newer designs are adding large battery packs to recapture some of this energy.

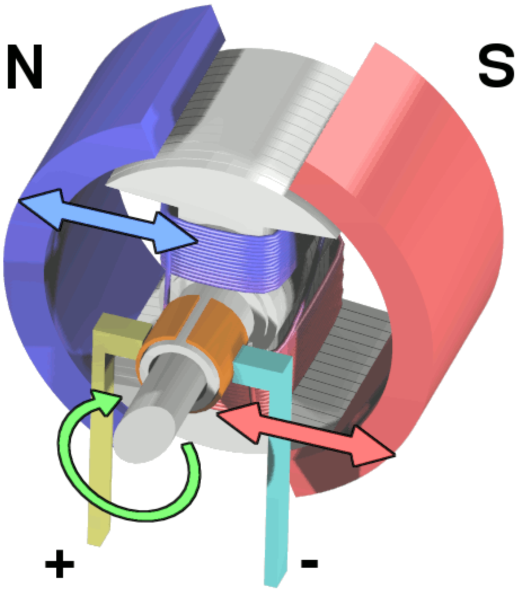


Figure 11: Internal construction of DC motor

REFERENCE :-

www.wikipedia.org

www.solorbotics.net

www.globalspec.com

www.diametriks.com

www.skitronics.co.in

www.electronicsforu.com

www.circuiteasy.com

8051 mazidi

APPENDIX :-

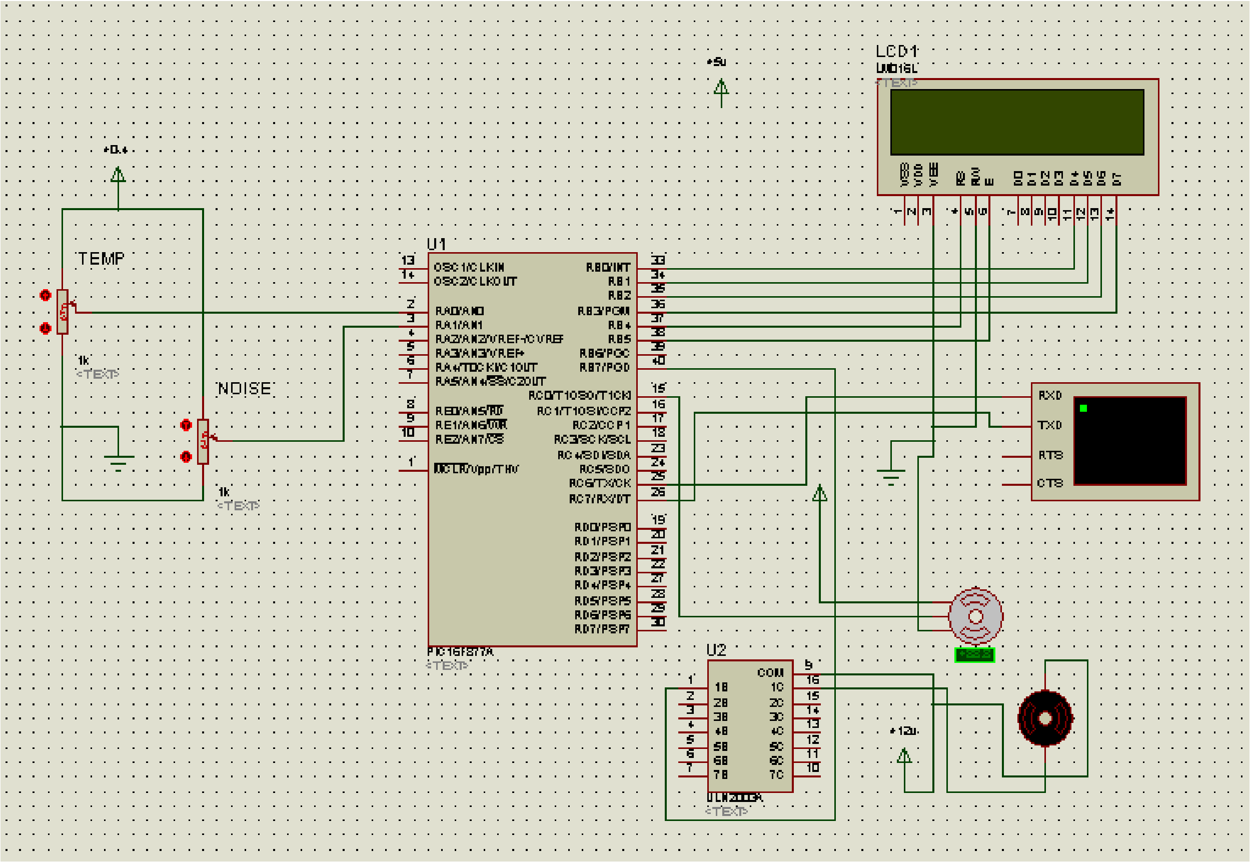
SOFT WARE :-

Mikro-C pro for 8051

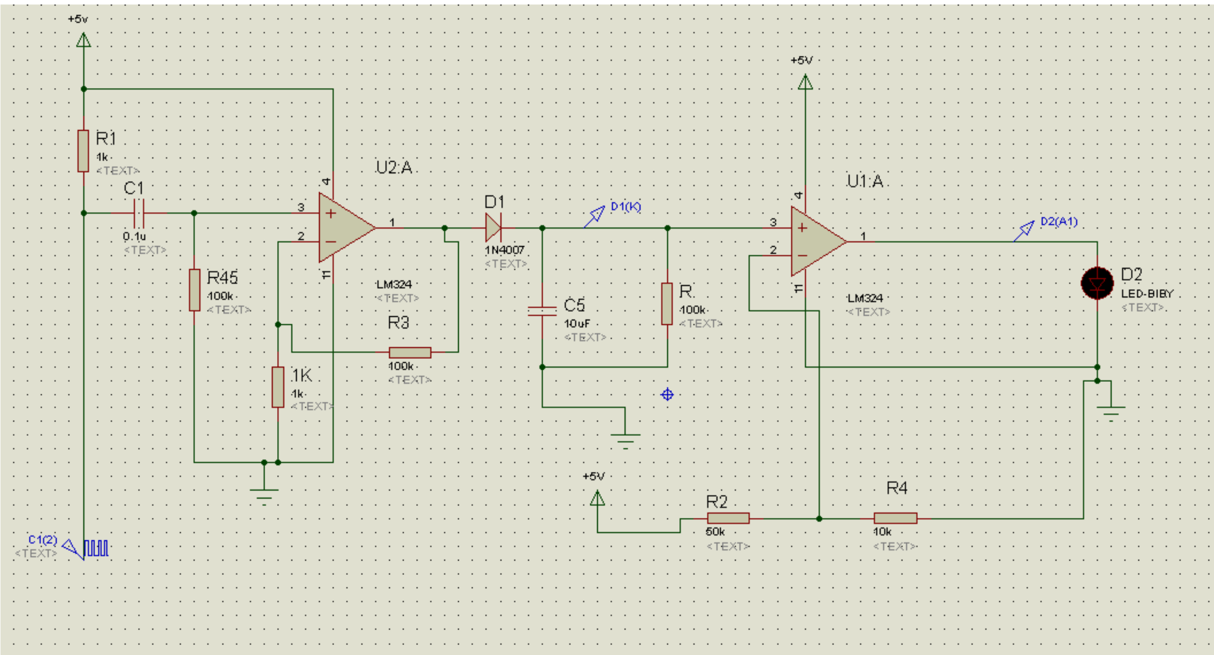
Mikro-C pro for PIC

Proteus for Design

Keil



TEMP
1k
<TEXT>
NOISE
1k
<TEXT>
U1
OSC1/CLKIN
OSC2/CLKOUT
RA0/AN0
RA1/AN1
RA2/AN2/VREF-/CVREF
RA3/AN3/VREF+
RA4/T0CKI/C1OUT
RA5/AN4/SS/C2OUT
RE0/AN5/RD
RE1/AN6/WR
RE2/AN7/CS
MCLR/Vpp/THV
RB0/INT
RB1
RB2
RB3/PGM
RB4
RB5
RB6/PGC
RB7/PGD
RC0/T1OSO/T1CKI
RC1/T1OSI/CCP2
RC2/CCP1
RC3/SCK/SCL
RC4/SDI/SDA
RC5/SDO
RC6/TX/CK
RC7/RX/DT
RD0/PSP0
RD1/PSP1
RD2/PSP2
RD3/PSP3
RD4/PSP4
RD5/PSP5
RD6/PSP6
RD7/PSP7
PIC16F877A
<TEXT>
+5v
LCD1
LM016L
<TEXT>
VSS
VDD
VEE
RS
RW
E
D0
D1
D2
D3
D4
D5
D6
D7
RXD
TXD
RTS
CTS
U2
COM
1B
2B
3B
4B
5B
6B
7B
1C
2C
3C
4C
5C
6C
7C
ULN2003A
<TEXT>
+12v


+5v
R1
1k
<TEXT>
C1
0.1u
<TEXT>
U2:A
3
2
4
11
1
LM324
<TEXT>
R45
100k
<TEXT>
R3
100k
<TEXT>
1K
1k
<TEXT>
D1
1N4007
<TEXT>
D1(K)
C5
10uF
<TEXT>
R
100k
<TEXT>
+5V
U1:A
3
2
4
11
1
LM324
<TEXT>
D2(A1)
D2
LED-BIBY
<TEXT>
+5V
R2
50k
<TEXT>
R4
10k
<TEXT>
C1(2)
<TEXT>