V.S.R. GOVT.
DEGREE & P.G. COLLEGE

A

PROJECT REPORT ON

AUTOMATIC FAULT DETECTION AND LOCATION IN POWER TRANSMISSION LINES

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT

FOR THE AWARD OF DEGREE OF

BACHELOR OF SCIENCE IN ELECTRONICS

Submitted to the **KRISHNA UNIVERSITY**

By

S. Anil Kumar

Y163223049

Under The Guidance of

K.KRANTHI M.Sc,M.Phil

(Lecturer in Electronics)

DEPARTMENT OF ELECTRONICS

V.S.R GOVT DEGREE & P.G COLLEGE,MOVVA

(6)

V.S.R GOVT DEGREE & P.G COLLEGE,MOVVA

DEPARTMENT OF ELECTRONICS

V.S.R. GOVT.
DEGREE & P.G. COLLEGE

CERTIFICATE

This is to Certified that the Project Work Entitled "AUTOMATIC FAULT DETECTION AND LOCATION IN POWER TRANSMISSION LINES" is a Bonafide work Carried out by---S. Anil kumar---------Regd No-Y163223049

Year-2018--2019----, in partial fulfilment for the Award of Undergraduation in ELECTRONICS, of KRISHNA UNIVERSITY, Machilipatnam ,during the Year 2018-19.

It is certified that all Corrections/ Suggestions indicated for Internal Assesement have been incorporated in the report .This project report have been approved as it satisfies the academic requirements in respect of project work prescribed for the Bachelor Degree in ELECTRONICS.

Signature of the Guide

Signature of the HOD

Signature of the Principal

Signature of the Examiners

1)

2)

ACKNOWLEDGEMENT

We are the Students of III B.SC(M.E.Cs) Electronics Cluster in V.S.R Govt Degree & P.G College,Movva are preparing a final year project **"AUTOMATIC FAULT DETECTION AND LOCATION IN POWER TRANSMISSION LINES". We Whole heartedly express our Sincere Gratitude to Smt. K.KRANTHI who guided us for the completion of this final year project.We are also thankful to our principal Smt V.Rama Jyotshna Kumari garu for giving us an opportunity to do this project.**

1)K.V.Sravan Kumar(Y163223040)
2)M.Balaji(Y163223043)
3) P.Rupak(Y163223047)
4)S.Durga Prasad(Y163223048)
5)S.Anil Kumar(Y163223049)
6)V.Teja Salu Kumar(Y163223051)

DECLARATION

We K.V.Sravan Kumar(Y163223040), M.Balaji(Y163223043),

P.Rupak(Y163223047), S.Durga Prasad(Y163223048),

S.Anil Kumar(Y163223049), V.Teja Salu Kumar(Y163223051) students of

6 Semester B.SC in ELECTRONICS, V.S.R Govt Degree & P.G College,Movva here by declare that the project work Entitled **"AUTOMATIC FAULT DETECTION AND LOCATION IN POWER TRANSMISSION LINES"**

Submitted to the Krishna University,Machilipatnam during the academic year 2018-19 ,is aRecord of original work done by us under the Guidance of Smt K.Kranthi ,Lecturer in Department Electronics,V.S.R Govt Degree & p.g College,Movva .This project work is SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT for the Award of Degree of **BACHELOR OF SCIENCE IN ELECTRONICS. This results and works Embodied in this Thesis have not been submitted to any other University (or) Institute for the award of any Degree.**

Govt. Degree College

S. Anil Kumar

STUDENT NAME

Date: 30-04-2019

Place: movva.

Lecturer, Department of Electronics

DECLARATION

I here by declare that the entire work for project contained in this dissertation entitled

"AUTOMATIC FAULT DETECTION AND LOCATION IN POWER TRANSMISSION LINES"

submitted in the partial fulfilment of B.SC-(ELECTRONICS CLUSTER), of V.S.R GOVT DEGREE&P.G COLLEGE,MOVVA was done by me under the guidance of

Mrs..K.KRANTHI,M.SC,M.PHIL,Lecturer in Electronics.

Department of Electronics
V.S.R GOVT DEGREE& P.G COLLEGE
MOVVA.

Signature of the students

1. S. Anil kumar
2.
3.
4.
5.

Lec.incharge HOD of the department External Examiner

INDEX

INDEX
(Flow charts)

AUTOMATIC FAULT DETECTION AND LOCATION IN POWER TRANSMISSION LINES

ABSTRACT

Many electricity transmission companies across the world and Ghana in particular are continuously looking for ways to utilise modern technologies, in order to improve reliability of power supply to consumers. These transmission companies manly relies on circuit indicators (FCIs) to assist in locating specific spots within their transmission lines where power fault had occured. This will ensure a shorter response time for technical crew to rectify these faults and thus help save transformers from damage and disasters. The system uses a current transformer, a voltage transformer, 8051 Microcontroller, RS-232 connector, and LCD. The system automatically detects faults, analyses and classifies these faults and then, calculates the fault distance from the control room using an impedance-based algorithm method. Finally the fault information is transmitted to the control room. In conclusion, the time required to locate a fault is drastically reduced, as the system automatically and accurately provides accurate fault location information.

BLOCK DIAGRAM:

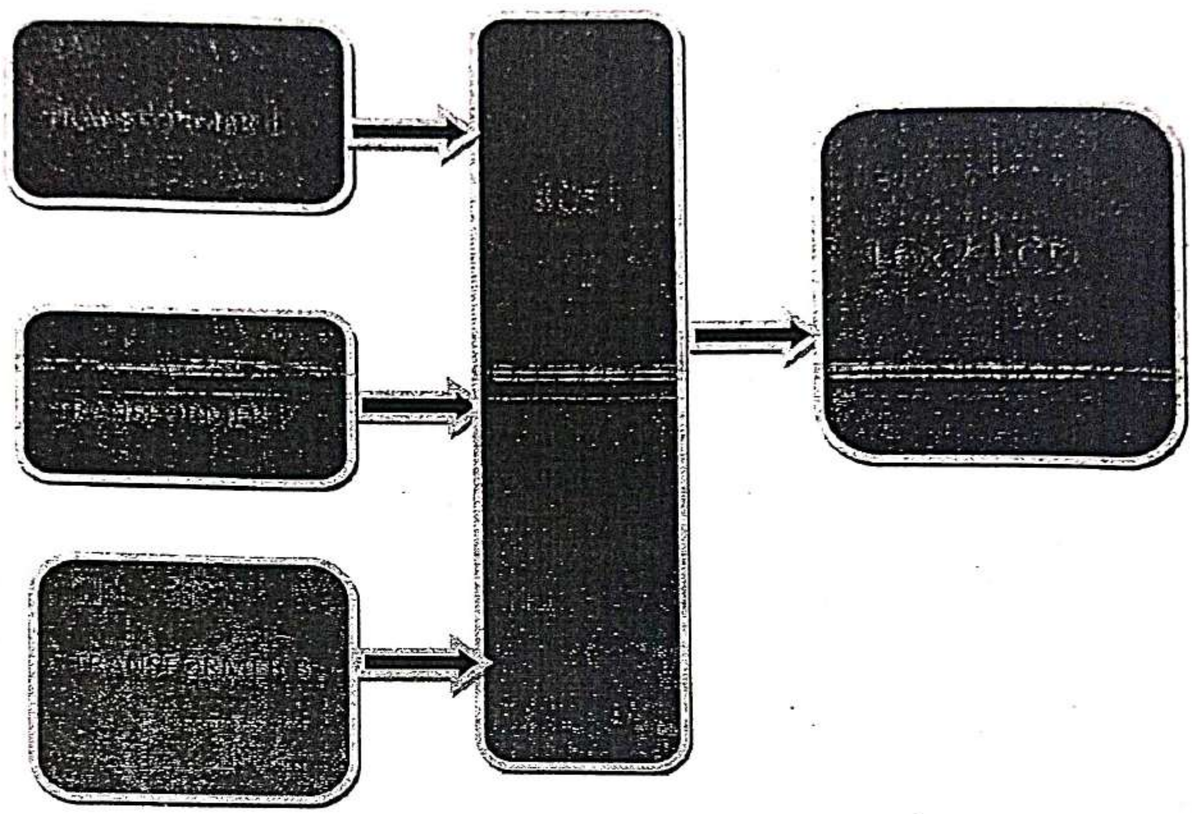


Fig :1: Block diagram of the System

CIRCUIT:

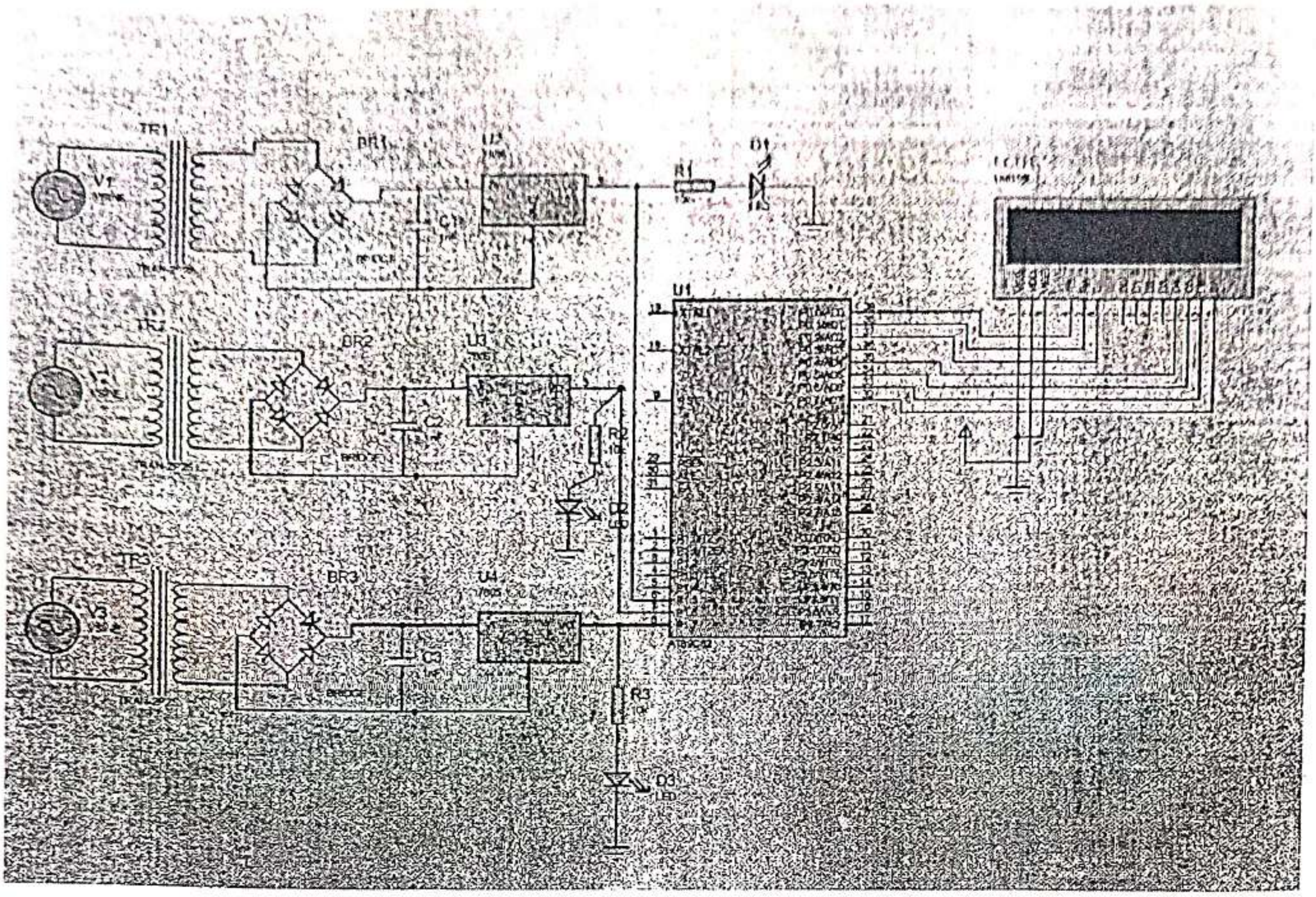


Fig :2: Block diagram of the system

THE MICROCONTROLLER:

A microcontroller is a general purpose device, but that is meant to read data, perform limited calculations on that data and control its environment based on those calculations. The prime use of a microcontroller is to control the operation of a machine using a fixed program that is stored in ROM and that does not change over the lifetime of the system.

The microcontroller design uses a much more limited set of single and double byte instructions that are used to move data and code from internal memory to the ALU. The microcontroller is concerned with getting data from and to its own pins; the architecture and instruction set are optimized to handle data in bit and byte size.

The AT89C51 is a low-power, high-performance CMOS 8-bit microcontroller with 4k bytes of Flash Programmable and erasable read only memory (EROM). The device is manufactured using Atmel's high-density nonvolatile memory technology and is functionally compatible with the industry-standard 80C51 microcontroller instruction set and pin out. By combining versatile 8-bit CPU with Flash on a monolithic chip, the Atmel's AT89c51 is a powerful microcomputer, which provides a high flexible and cost-effective solution to many embedded control applications.

AT89C51 MICROCONTROLLER:

FEATURES:

- 80C51 based architecture
- 4-Kbytes of on-chip Reprogrammable Flash Memory
- 128 x 8 RAM
- Two 16-bit Timer/Counters
- Full duplex serial channel
- Boolean processor
- Four 8-bit I/O ports, 32 I/O lines
- Memory addressing capability

- 64K ROM and 64K RAM

- Power save modes:
  - Idle and power-down
- Six interrupt sources
- Most instructions execute in 0.3 us
- CMOS and TTL compatible
- Maximum speed: 40 MHz @ Vcc = 5V
- Industrial temperature available
- **Packages available:**
  - 40-pin DIP
  - 44-pin PLCC
  - 44-pin PQFP

Pin configuration:

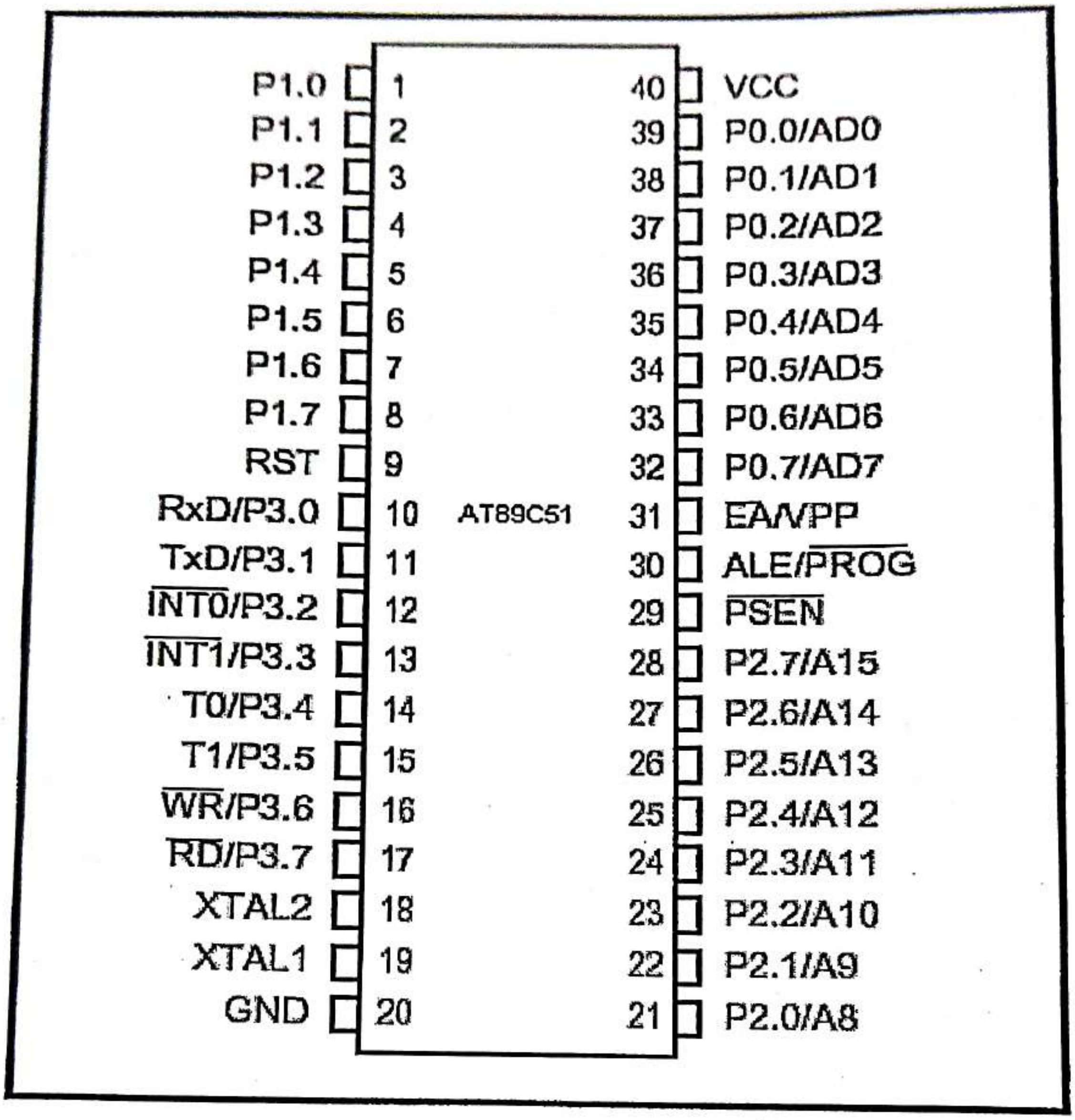


Fig :3: Pin configuration of 8051

AT89C51 Block Diagram:

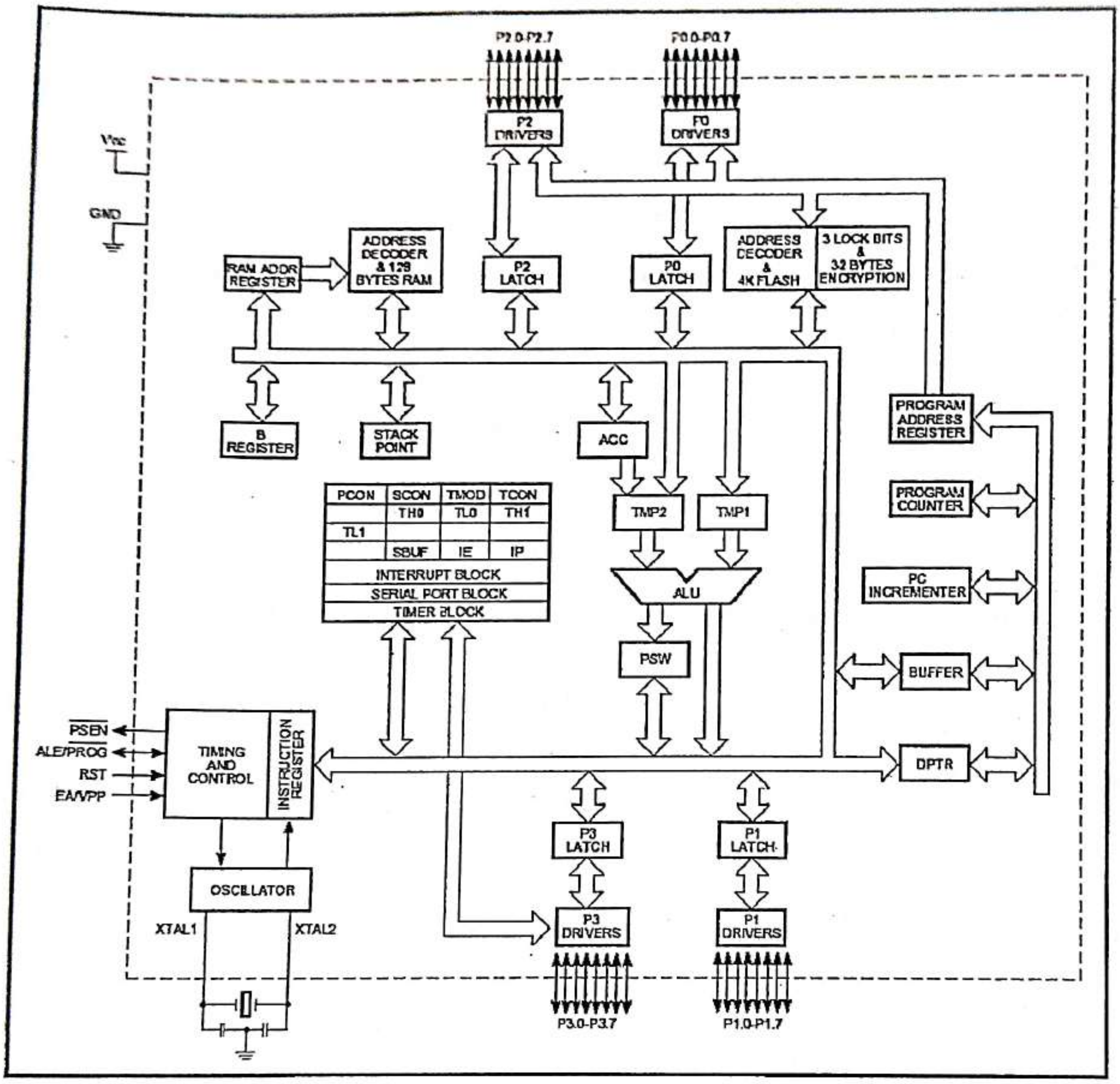


Fig :4: Block diagram of the Microcontroller 8051

PIN DESCRIPTION:

VCC:

Supply voltage

GND:

Ground

Port 0:

Port 0 is an 8-bit open drain bi-directional I/O port. As an output port, each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high impedance inputs.

Port 0 can also be configured to be the multiplexed low order address/data bus during access to external program and data memory. In this mode, P 0 has internal pull-ups. Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. External pull-ups are required during program verification.

Port 1:

Port 1 is an 8-bit bi-directional I/O port with internal pull-ups. The port 1output buffers can sink/source four TTL inputs. When 1s are written to port 1 pins, they are pulled high by the internal pull-ups can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current (1) because of the internal pull-ups.

Port 2:

Port 2 is an 8-bit bi-directional I/O port with internal pull-ups. The port 2 output buffers can sink/source four TTL inputs. When 1s are written to port 2 pins, they are pulled high by the internal pull-ups can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current because of the internal pull-ups.

Port 2 emits the high-order address byte during fetches from external program memory and during access to DPTR. In this application Port 2 uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit data address (MOVX@R1), Port 2 emits the contents of the P2 Special Function Register. Port 2 also receives the high-order address bits and some control signals during Flash programming and verification.

Port 3:

Port 3 is an 8-bit bi-directional I/O port with internal pull-ups. The port 3 output buffers can sink/source four TTL inputs. When 1s are written to port 3 pins, they are pulled high by the internal pull-ups can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current because of the internal pull-ups.

Port 3 also receives some control signals for Flash Programming and verification.

Port pin	Alternate Functions
P3.0	RXD(serial input port)
P3.1	TXD(serial input port)
P3.2	INT0(external interrupt 0)
P3.3	INT1(external interrupt 1)
P3.4	T0(timer 0 external input)
P3.5	T1(timer 1 external input)
P3.6	WR(external data memory write strobe)
P3.7	RD(external data memory read strobe)

RST:

Rest input A on this pin for two machine cycles while the oscillator is running resets the device.

ALE/PROG:

Address Latch Enable is an output pulse for latching the low byte of the address during access to external memory. This pin is also the program pulse input (PROG) during Flash programming.

In normal operation ALE is emitted at a constant rate of 1/16 the oscillator frequency and may be used for external timing or clocking purpose. Note, however, that one ALE pulse is skipped during each access to external Data memory.

PSEN:

Program Store Enable is the read strobe to external program memory when the AT89c51 is executing code from external program memory PSEN is activated twice each machine cycle, except that two PSEN activations are skipped during each access to external data memory.

EA /VPP:

External Access Enable (EA) must be strapped to GND in order to enable the device to fetch code from external program memory locations starting at 0000h up to FFFFH. Note, however, that if lock bit 1 is programmed EA will be internally latched on reset. EA should be strapped to Vcc for internal program executions. This pin also receives the 12-volt programming enable voltage (Vpp) during Flash programming when 12-volt programming is selected.

XTAL1:

Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

XTAL 2:

Output from the inverting oscillator amplifier.

OPERATING DESCRIPTION:

The detail description of the AT89C51 included in this description is:

- Memory Map and Registers
- Timer/Counters
- Interrupt System

MEMORY MAP AND REGISTERS:

Memory:

The AT89C51 has separate address spaces for program and data memory. The program and data memory can be up to 64K bytes long. The lower 4K program memory can reside on-chip. The AT89C51 has 128 bytes of on-chip RAM.

The lower 128 bytes can be accessed either by direct addressing or by indirect addressing. The lower 128 bytes of RAM can be divided into 3 segments as listed below

1. **Register Banks 0-3:** locations 00H through 1FH (32 bytes). The device after reset defaults to register bank 0. To use the other register banks, the user must select them in software. Each register bank contains eight 1-byte registers R0-R7. Reset initializes the stack point to location 07H, and is incremented once to start from 08H, which is the first register of the second register bank.

2. **Bit Addressable Area:** 16 bytes have been assigned for this segment 20H-2FH. Each one of the 128 bits of this segment can be directly addressed (0-7FH). Each of the 16 bytes in this segment can also be addressed as a byte.

3. **Scratch Pad Area:** 30H-7FH are available to the user as data RAM. However, if the data pointer has been initialized to this area, enough bytes should be left aside to prevent SP data destruction.

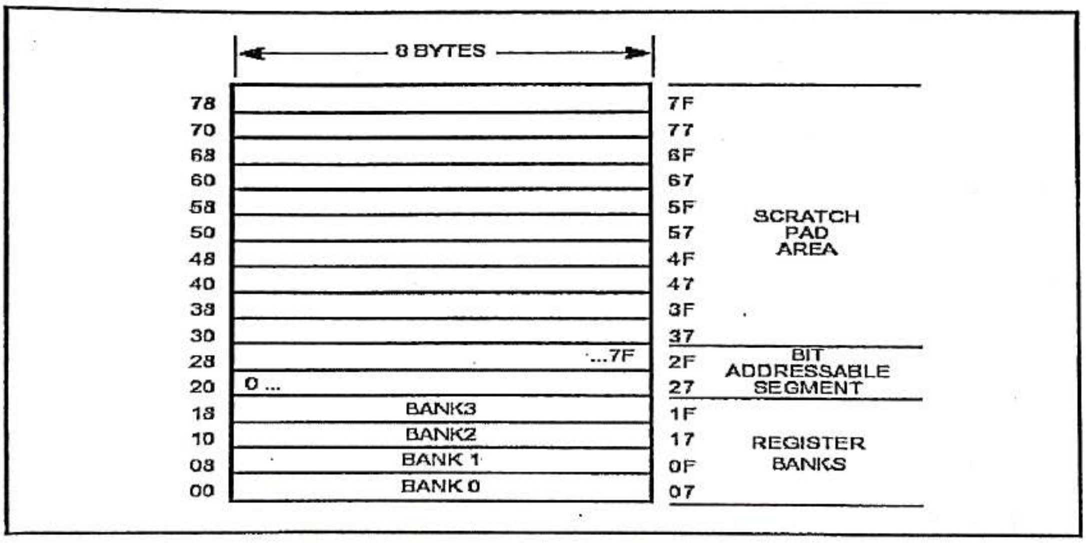


Fig : 5: Memory of the Microcontroller 8051

SPECIAL FUNCTION REGISTERS(SFR):

The Special Function Registers (SFR's) are located in upper 128 Bytes direct addressing area. The SFR Memory Map in shows that.

Not all of the addresses are occupied. Unoccupied addresses are not implemented on the chip. Read accesses to these addresses in general return random data, and write accesses have no effect. User software should not write 1s to these unimplemented locations, since they may be used in future microcontrollers to invoke new features. In that case, the reset or inactive values of the new bits will always be 0, and their active values will be 1.

The functions of the SFR's are outlined in the following sections.

Accumulator (ACC):

ACC is the Accumulator register. The mnemonics for Accumulator-specific instructions, however, refer to the Accumulator simply as A.

B Register (B):

The B register is used during multiply and divide operations. For other instructions it can be treated as another scratch pad register.

Program Status Word (PSW):

The PSW register contains program status information.

Stack Pointer (SP):

The Stack Pointer Register is eight bits wide. It is incremented before data is stored during PUSH and CALL executions. While the stack may reside anywhere in on chip RAM, the Stack Pointer is initialized to 07H after a reset. This causes the stack to begin at location 08H.

Data Pointer (DPTR):

The Data Pointer consists of a high byte (DPH) and a low byte (DPL). Its function is to hold a 16-bit address. It may be manipulated as a 16-bit register or as two independent 8-bit registers.

Serial Data Buffer (SBUF):

The Serial Data Buffer is actually two separate registers, a transmit buffer and a receive buffer register. When data is moved to SBUF, it goes to the transmit buffer, where it is held for serial transmission. (Moving a byte to SBUF initiates the transmission.) When data is moved from SBUF, it comes from the receive buffer.

Timer Registers:

Register pairs (TH0, TL0) and (TH1, TL1) are the 16-bit Counter registers for Timer/Counters 0 and 1, respectively.

Control Registers:

Special Function Registers IP, IE, TMOD, TCON, SCON, and PCON contain control and status bits for the interrupt system, the Timer/Counters, and the serial port.

TIMER/COUNTERS:

The IS89C51 has two 16-bit Timer/Counter registers: Timer 0 and Timer 1. All two can be configured to operate either as Timers or event counters. As a Timer, the register is incremented every machine cycle. Thus, the register counts machine cycles. Since a machine cycle consists of 12 oscillator periods, the count rate is 1/12 of the oscillator frequency.

As a Counter, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T0 and T1. The external input is sampled during S5P2 of every machine cycle. When the samples show a high in one cycle and a low in the next

cycle, the count is incremented. The new count value appears in the register during S3P1 of the cycle following the one in which the transition was detected. Since two machine cycles (24 oscillator periods) are required to recognize a 1-to-0 transition, the maximum count rate is 1/24 of the oscillator frequency. There are no restrictions on the duty cycle of the external input signal, but it should be held for at least one full machine cycle to ensure that a given level is sampled at least once before it changes.

In addition to the Timer or Counter functions, Timer 0 and Timer 1 have four operating modes: 13-bit timer, 16-bit timer, 8-bit auto-reload, split timer.

TIMERS:

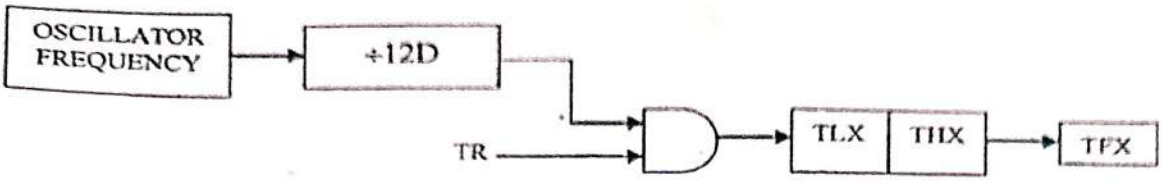


Fig :6: Microcontroller 8051 –Timer/Counter (16- bit)

SFR'S USED IN TIMERS:

The special function registers used in timers are,

- TMOD Register
- TCON Register
- Timer(T0) & timer(T1) Registers

(i) TMOD Register:

TMOD is dedicated solely to the two timers (T0 & T1).

- The timer mode SFR is used to configure the mode of operation of each of the two timers. Using this SFR your program may configure each timer to be a 16-bit timer, or 13 bit timer, 8-bit auto reload timer, or two separate timers. Additionally you may configure the timers to only count when an external pin is activated or to count "events" that are indicated on an external pin.

- It can consider as two duplicate 4-bit registers, each of which controls the action of one of the timers.

(ii) TCON Register:

- The timer control SFR is used to configure and modify the way in which the 8051's two timers operate. This SFR controls whether each of the two timers is running or stopped and contains a flag to indicate that each timer has overflowed. Additionally, some non-timer related bits are located in TCON SFR.
- These bits are used to configure the way in which the external interrupt flags are activated, which are set when an external interrupt occurs.

TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Fig :7: TCON Register

(iii) TIMER 0 (T0):

- TO (Timer 0 low/high, address 8A/8C h)

These two SFR's taken together represent timer 0. Their exact behavior depends on how the timer is configured in the TMOD SFR; however, these timers always count up. What is configurable is how and when they increment in value.

TH0	TL0

Fig :8: Timer Block diagram

(iv) TIMER 1 (T1):

- T1 (Timer 1 Low/High, address 8B/ 8D h)

These two SFR's, taken together, represent timer 1. Their exact behavior depends on how the timer is configured in the TMOD SFR; however, these timers always count up. What is Configurable is how and when they increment in value.

TH1	TL1

Fig :9: Block diagram of Timer 1

The Timer or Counter function is selected by control bits C/T in the Special Function Register TMOD. These two Timer/Counters have four operating modes, which are selected by bit pairs (M1, M0) in TMOD. Modes 0, 1, and 2 are the same for both Timer/Counters, but Mode 3 is different.

The four modes are described in the following sections.

Mode 0:

Both Timers in Mode 0 are 8-bit Counters with a divide-by-32 pre scalar. Figure 8 shows the Mode 0 operation as it applies to Timer 1. In this mode, the Timer register is configured as a 13-bit register. As the count rolls over from all 1s to all 0s, it sets the Timer interrupt flag TF1. The counted input is enabled to the Timer when TR1 = 1 and either GATE = 0 or INT1 = 1. Setting GATE = 1 allows the Timer to be controlled by external input INT1, to facilitate pulse width measurements. TR1 is a control bit in the Special Function Register TCON. Gate is in TMOD.

The 13-bit register consists of all eight bits of TH1 and the lower five bits of TL1. The upper three bits of TL1 are indeterminate and should be ignored. Setting the run flag (TR1) does not clear the registers.

Mode 0 operation is the same for Timer 0 as for Timer 1, except that TR0, TF0 and INT0 replace the corresponding Timer 1 signals. There are two different GATE bits, one for Timer 1 (TMOD.7) and one for Timer 0 (TMOD.3).

Mode 1:

Mode 1 is the same as Mode 0, except that the Timer register is run with all 16 bits. The clock is applied to the combined high and low timer registers (TL1/TH1). As clock pulses are received, the timer counts up: 0000H, 0001H, 0002H, etc. An overflow occurs on the FFFFH-to-0000H overflow flag. The timer continues to count. The overflow flag is the TF1 bit in TCON that is read or written by software

Mode 2:

Mode 2 configures the Timer register as an 8-bit Counter (TL1) with automatic reload, as shown in Figure 10. Overflow from TL1 not only sets TF1, but also reloads TL1 with the contents of TH1, which is preset by software. The reload leaves the TH1 unchanged. Mode 2 operation is the same for Timer/Counter 0.

Mode 3:

Timer 1 in Mode 3 simply holds its count. The effect is the same as setting TR1 = 0. Timer 0 in Mode 3 establishes TL0 and TH0 as two separate counters. The logic for Mode 3 on Timer 0 is shown in Figure 11. TL0 uses the Timer 0 control bits: C/T, GATE, TR0, INT0, and TF0. TH0 is locked into a timer function (counting machine cycles) and over the use of TR1 and TF1 from Timer 1. Thus, TH0 now controls the Timer 1 interrupt.

Mode 3 is for applications requiring an extra 8-bit timer or counter. With Timer 0 in Mode 3, the AT89C51 can appear to have three Timer/Counters. When Timer 0 is in Mode 3, Timer 1 can be turned on and off by switching it out of and into its own Mode 3. In this case, Timer 1 can still be used by the serial port as a baud rate geherator or in any application not requiring an interrupt.

INTERRUPT SYSTEM:

An interrupt is an external or internal event that suspends the operation of micro controller to inform it that a device needs its service. In interrupt method, whenever any device needs its service, the device notifies the micro controller by sending it an interrupt signal. Upon receiving an interrupt signal, the micro controller interrupts whatever it is doing and serves the device. The program associated with interrupt is called as **interrupt service subroutine (ISR)**.Main advantage with interrupts is that the micro controller can serve many devices.

Baud Rate:

The baud rate in Mode 0 is fixed as shown in the following equation. Mode 0 Baud Rate = Oscillator Frequency /12 the baud rate in Mode 2 depends on the value of the SMOD bit in Special Function Register PCON. If SMOD = 0 the baud rate is 1/64 of the oscillator frequency. If SMOD = 1, the baud rate is 1/32 of the oscillator frequency.

Mode 2 Baud Rate = 2SMODx (Oscillator Frequency)/64.

In the IS89C51, the Timer 1 overflow rate determines the baud rates in Modes 1 and 3.

NUMBER OF INTERRUPTS IN 89C51:

There are basically five interrupts available to the user. Reset is also considered as an interrupt. There are two interrupts for timer, two interrupts for external hardware interrupt and one interrupt for serial communication.

Memory location	Interrupt name
0000H	Reset
0003H	External interrupt 0
000BH	Timer interrupt 0
0013H	External interrupt 1
001BH	Timer interrupt 1
0023H	Serial COM interrupt

Lower the vector, higher the priority. The External Interrupts INT0 and INT1 can each be either level-activated or transition-activated, depending on bits IT0 and IT1 in Register TCON. The flags that actually generate these interrupts are the IE0 and IE1 bits in TCON. When the service routine is vectored, hardware clears the flag that generated an external interrupt only if the interrupt was transition-activated. If the interrupt was level-activated, then the external requesting source (rather than the on-chip hardware) controls the request flag.

The Timer 0 and Timer 1 Interrupts are generated by TF0and TF1, which are set by a rollover in their respective Timer/Counter registers (except for Timer 0 in Mode 3).When a timer interrupt is generated, the on-chip hardware clears the flag that is generated.

The Serial Port Interrupt is generated by the logical OR of RI and TI. The service routine normally must determine whether RI or TI generated the interrupt, and the bit must be cleared in software.

All of the bits that generate interrupts can be set or cleared by software, with the same result as though they had been set or cleared by hardware. That is, interrupts can be generated and pending interrupts can be canceled in software.

Each of these interrupt sources can be individually enabled or disabled by setting or clearing a bit in Special Function Register IE (interrupt enable) at address 0A8H. There is a global enable/disable bit that is cleared to disable all interrupts or to set the interrupts.

IE (Interrupt enable register):

Steps in enabling an interrupt:

Bit D7 of the IE register must be set to high to allow the rest of register to take effect. If EA=1, interrupts are enabled and will be responded to if their corresponding bits in IE are high. If EA=0, no interrupt will be responded to even if the associated bit in the IE register is high.

Description of each bit in IE register:

D7 bit: Disables all interrupts. If EA =0, no interrupt is acknowledged, if EA=1 each interrupt source is individually enabled or disabled by setting or clearing its enable bit.

D6 bit: Reserved.

D5 bit: Enables or disables timer 2 over flow interrupt (in 8052).

D4 bit: Enables or disables serial port interrupt.

D3 bit: Enables or disables timer 1 over flow interrupt.

D2 bit: Enables or disables external interrupt 1.

D1 bit: Enables or disables timer 0 over flow interrupt.

D0 bit: Enables or disables external interrupt 0.

Interrupt priority in 89C51:

There is one more SRF to assign priority to the interrupts which is named as interrupt priority (IP). User has given the provision to assign priority to one interrupt. Writing one to that particular bit in the IP register fulfils the task of assigning the priority.

Description of each bit in IP register:

D7 bit: Reserved.

D6 bit: Reserved.

D5 bit: Timer 2 interrupt priority bit (in 8052).

D4 bit: Serial port interrupt priority bit.

D3 bit: Timer 1 interrupt priority bit.

D2 bit: External interrupt 1 priority bit.

D1 bit: Timer 0 interrupt priority bit.

D0 bit: External interrupt 0 priority bit.

GSM :

Definition of GSM:

GSM (Global System for Mobile communications) is an open, digital cellular technology used for transmitting mobile voice and data services.

GSM (Global System for Mobile communication) is a digital mobile telephone system that is widely used in Europe and other parts of the world. GSM uses a variation of Time Division Multiple Access (TDMA) and is the most widely used of the three digital wireless telephone technologies (TDMA, GSM, and CDMA). GSM digitizes and compresses data, then sends it down a channel with two other streams of user data, each in its own time slot.

It operates at either the 900 MHz or 1,800 MHz frequency band. It supports voice calls and data transfer speeds of up to 9.6 kbit/s, together with the transmission of SMS (Short Message Service).

History:

In 1982, the European Conference of Postal and Telecommunications Administrations (CEPT) created the Group Special Mobile (GSM) to develop a standard for a mobile telephone system that could be used across Europe. In 1987, a memorandum of understanding was signed by 13 countries to develop a common cellular telephone system across Europe. Finally the system created by SINTEF lead by Torleiv Maseng was selected.

In 1989, GSM responsibility was transferred to the European Telecommunications Standards Institute (ETSI) and phase I of the GSM specifications were published in 1990. The first GSM network was launched in 1991 by Radiolinja in Finland with joint technical infrastructure maintenance from Ericsson.

By the end of 1993, over a million subscribers were using GSM phone networks being operated by 70 carriers across 48 countries. As of the end of 1997, GSM service was available in more than 100 countries and has become the *de facto* standard in Europe and Asia.

GSM Frequencies:

GSM networks operate in a number of different frequency ranges (separated into GSM frequency ranges for 2G and UMTS frequency bands for 3G). Most 2G GSM networks operate in the 900 MHz or 1800 MHz bands.

Some countries in the Americas (including Canada and the United States) use the 850 MHz and 1900 MHz bands because the 900 and 1800 MHz frequency bands were already allocated. Most 3G GSM networks in Europe operate in the 2100 MHz frequency band. The rarer 400 and 450 MHz frequency bands are assigned in some countries where these frequencies were previously used for first-generation systems.

GSM-900 uses 890–915 MHz to send information from the mobile station to the base station (uplink) and 935–960 MHz for the other direction (downlink), providing 124 RF channels (channel numbers 1 to 124) spaced at 200 kHz. Duplex spacing of 45 MHz is used.

In some countries the GSM-900 band has been extended to cover a larger frequency range. This 'extended GSM', E-GSM, uses 880–915 MHz (uplink) and 925–960 MHz (downlink), adding 50 channels (channel numbers 975 to 1023 and 0) to the original GSM-900 band.

Time division multiplexing is used to allow eight full-rate or sixteen half-rate speech channels per radio frequency channel. There are eight radio timeslots (giving eight burst periods) grouped into what is called a TDMA frame. Half rate channels use alternate frames in the same timeslot. The channel data rate for all 8 channels is 270.833 Kbit/s, and the frame duration is 4.615 ms.

The transmission power in the handset is limited to a maximum of 2 watts in GSM850/900 and 1 watt in GSM1800/1900. GSM operates in the 900MHz and 1.8GHz bands in Europe and the 1.9GHz and 850MHz bands in the US. The 850MHz band is also used for GSM and 3G in Australia, Canada and many South American countries. By having harmonized spectrum across most of the globe, GSM's international roaming capability allows users to access the same services when travelling abroad as at home. This gives consumers seamless and same number connectivity in more than 218 countries.

Terrestrial GSM networks now cover more than 80% of the world's population. GSM satellite roaming has also extended service access to areas where terrestrial coverage is not available.

Mobile Telephony Standards:

Standard	Generation	Frequency band	Throughput	
GSM	2G	Allows transfer of voice or low-volume digital data.	9.6 kbps	9.6 kbps
GPRS	2.5G	Allows transfer of voice or moderate-volume digital data.	21.4-171.2 kbps	48 kbps
EDGE	2.75G	Allows simultaneous transfer of voice and digital data.	43.2-345.6 kbps	171 kbps
UMTS	3G	Allows simultaneous transfer of voice and high-speed digital data.	0.144-2 Mbps	384 kbps

1G:

The first generation of mobile telephony (written **1G**) operated using analogue communications and portable devices that were relatively large. It used primarily the following standards:

- **AMPS (Advanced Mobile Phone System)**: which appeared in 1976 in the United States, was the first cellular network standard. It was used primarily in the Americas, Russia and Asia. This first-generation analogue network had weak security mechanisms which allowed hacking of telephones lines.

- **TACS (Total Access Communication System)**: is the Europeam version of the AMPS model. Using the 900 MHz frequency band, this system was largely used in England and then in Asia (Hong-Kong and Japan).

- **ETACS (Extended Total Access Communication System)**: is an improved version of the TACS standard developed in the United Kingdom that uses a larger number of communication channels.

The first-generation cellular networks were made obsolete by the appearance of an entirely digital second generation.

Second Generation of Mobile Networks (2G):

The second generation of mobile networks marked a break with the first generation of cellular telephones by switching from analogue to digital. The main 2G mobile telephony standards are:

- GSM (*Global System for Mobile communications*): is the most commonly used standard in Europe at the end of the 20th century and supported in the United States. This standard uses the 900 MHz and 1800 MHz frequency bands in Europe. In the United States, however, the frequency band used is the 1900 MHz band. Portable telephones that are able to operate in Europe and the United States are therefore called **tri-band**.

- **CDMA** (*Code Division Multiple Access*): uses a spread spectrum technique that allows a radio signal to be broadcast over a large frequency range.

- **TDMA** (*Time Division Multiple Access*) uses a technique of time division of communication channels to increase the volume of data transmitted simultaneously. TDMA technology is primarily used on the American continent, in New Zealand and in the Asia-Pacific region.

With the 2G networks, it is possible to transmit voice and low volume digital data, for example text messages (**SMS**, for *Short Message Service*) or multimedia messages (**MMS**, for *Multimedia Message Service*). The GSM standard allows a maximum data rate of 9.6 kbps.

Extensions have been made to the GSM standard to improve throughput. One of these is the **GPRS** (*General Packet Radio System*) service which allows theoretical data rates on the order of 114 Kbit/s but with throughput closer to 40 Kbit/s in practice. As this technology does not fit within the "3G" category, it is often referred to as **2.5G**

The **EDGE** (*Enhanced Data Rates for Global Evolution*) standard, billed as **2.75G**, quadruples the throughput improvements of GPRS with its theoretical data rate of 384 Kbps, thereby allowing the access for multimedia applications. In reality, the EDGE standard allows maximum theoretical data rates of 473 Kbit/s, but it has been limited in order to comply with the IMT-2000 (*International Mobile Telecommunications-2000*) specifications from the **ITU** (*International Telecommunications Union*).

3G:

The IMT-2000 (*International Mobile Telecommunications for the year 2000*) specifications from the International Telecommunications Union (ITU) defined the characteristics of **3G** (third generation of mobile telephony). The most important of these characteristics are:

1. High transmission data rate.
2. 144 Kbps with total coverage for mobile use.
3. 384 Kbps with medium coverage for pedestrian use.
4. 2 Mbps with reduced coverage area for stationary use.
5. World compatibility.
6. Compatibility of 3rd generation mobile services with second generation networks.

3G offers data rates of more than 144 Kbit/s, thereby allowing the access to multimedia uses such as video transmission, video-conferencing or high-speed internet access. 3G networks use different frequency bands than the previous networks: 1885-2025 MHz and 2110-2200 MHz.

The main 3G standard used in Europe is called **UMTS** (*Universal Mobile Telecommunications System*) and uses **WCDMA** (*Wideband Code Division Multiple Access*) encoding. UMTS technology uses 5 MHz bands for transferring voice and data, with data rates that can range from 384 Kbps to 2 Mbps. **HSDPA** (*High Speed Downlink Packet Access*) is a third generation mobile telephony protocol, (considered as "3.5G"), which is able to reach data rates on the order of 8 to 10 Mbps. HSDPA technology uses the 5 GHz frequency band and uses WCDMA encoding.

Introduction to the GSM Standard:

The **GSM** (*Global System for Mobile communications*) network is at the start of the 21st century, the most commonly used mobile telephony standard in Europe. It is called as Second Generation (2G) standard because communications occur in an entirely digital mode, unlike the first generation of portable telephones. When it was first standardized in 1982, it was called as

Group Special Mobile and later, it became an international standard called **"Global System for Mobile communications"** in 1991.

In Europe, the GSM standard uses the 900 MHz and 1800 MHz frequency bands. In the United States, however, the frequency band used is the 1900 MHz band.

For this reason, portable telephones that are able to operate in both Europe and the United States are called **tri-band** while those that operate only in Europe are called **bi-band**.

The GSM standard allows a maximum throughput of 9.6 kbps which allows transmission of voice and low-volume digital data like text messages (**SMS**, for *Short Message Service*) or multimedia messages (**MMS**, for *Multimedia Message Service*).

GSM Standards:

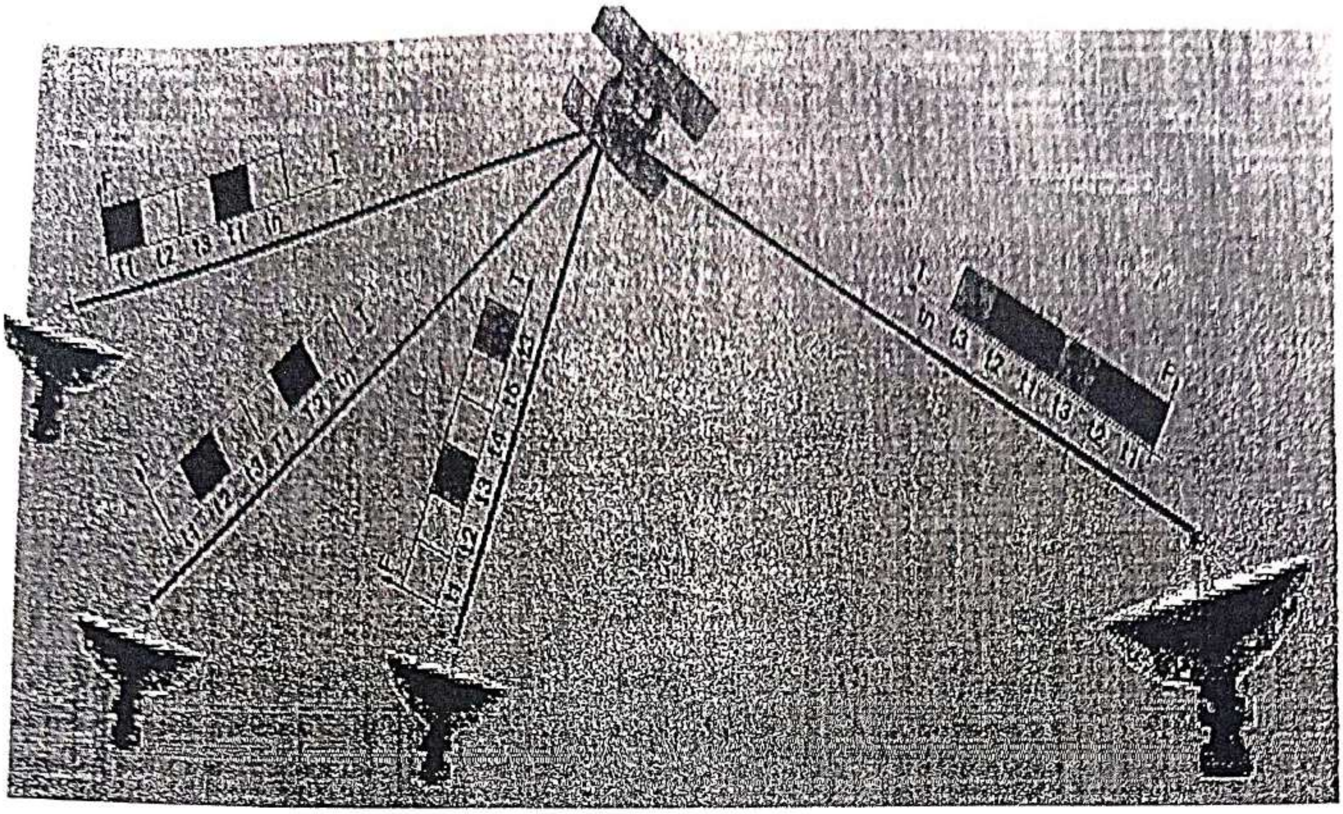
GSM uses narrowband TDMA, which allows eight simultaneous calls on the same radio frequency. There are three basic principles in multiple access, FDMA (Frequency Division Multiple Access), TDMA (Time Division Multiple Access), and CDMA (Code Division Multiple Access). All three principles allow multiple users to share the same physical channel. But the two competing technologies differ in the way user sharing the common resource.

TDMA allows the users to share the same frequency channel by dividing the signal into different time slots. Each user takes turn in a round robin fashion for transmitting and receiving over the channel. Here, users can only transmit in their respective time slot

CDMA uses a spread spectrum technology that is it spreads the information contained in a particular signal of interest over a much greater bandwidth than the original signal. Unlike TDMA, in CDMA several users can transmit over the channel at the same time.

TDMA in brief:

In late1980's, as a search to convert the existing analog network to digital as a means to improve capacity, the cellular telecommunications industry association chose TDMA over FDMA. Time Division Multiplex Access is a type of multiplexing where two or more channels of information are transmitted over the same link by allocating a different time interval for the transmission of each channel. The most complex implementation using TDMA principle is of GSM's (Global System for Mobile communication). To reduce the effect of co-channel interference, fading and multipath, the GSM technology can use frequency hopping, where a call jumps from one channel to another channel in a short interval.



Time Division Multiple Access

Fig :10: Time division multiplexing

TDMA systems still rely on switch to determine when to perform a handoff. Handoff occurs when a call is switched from one cell site to another while travelling. The TDMA handset constantly monitors the signals coming from other sites and reports it to the switch without caller's awareness. The switch then uses this information for making better choices for handoff at appropriate times. TDMA handset performs hard handoff, i.e., whenever the user moves from one site to another, it breaks the connection and then provides a new connection with the new site.

Advantages of TDMA:

There are lots of advantages of TDMA in cellular technologies.

1. It can easily adapt to transmission of data as well as voice communication.
2. It has an ability to carry 64 kbps to 120 Mbps of data rates. This allows the operator to do services like fax, voice band data and SMS as well as bandwidth intensive application such as multimedia and video conferencing.
3. Since TDMA technology separates users according to time, it ensures that there will be no interference from simultaneous transmissions.
4. It provides users with an extended battery life, since it transmits only portion of the time during conversations. Since the cell size grows smaller, it proves to save base station equipment, space and maintenance.

TDMA is the most cost effective technology to convert an analog system to digital.

Disadvantages of TDMA:

One major disadvantage using TDMA technology is that the users has a predefined time slot. When moving from one cell site to other, if all the time slots in this cell are full the user might be disconnected. Likewise, if all the time slots in the cell in which the user is currently in are already occupied, the user will not receive a dial tone.

The second problem in TDMA is that it is subjected to multipath distortion. To overcome this distortion, a time limit can be used on the system. Once the time limit is expired, the signal is ignored.

The concept of cellular network:

Mobile telephone networks are based on the concept of **cells**, circular zones that overlap to cover a geographical area.

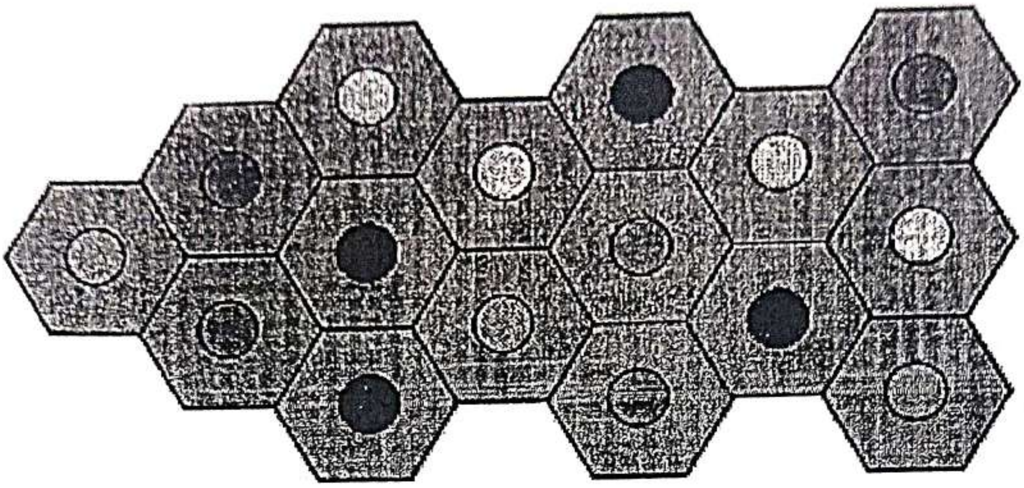


Fig :11: Cells and Zones of a Mobile telephone Network

Cellular networks are based on the use of a central transmitter-receiver in each cell, called a **"base station"** (or *Base Transceiver Station*, written **BTS**). The smaller the radius of a cell, the higher is the available bandwidth. So, in highly populated urban areas, there are cells with a radius of a few hundred meters, while huge cells of up to 30 kilometers provide coverage in rural areas.

In a cellular network, each cell is surrounded by 6 neighbouring cells (thus a cell is generally drawn as a hexagon). To avoid interference, adjacent cells cannot use the same frequency. In practice, two cells using the same frequency range must be separated by a distance of two to three times the diameter of the cell.

Architecture of the GSM Network:

In a GSM network, the user terminal is called a **mobile station.** A mobile station is made up of a **SIM** (*Subscriber Identity Module*) card allowing the user to be uniquely identified and a mobile terminal. The terminals (devices) are identified by a unique 15-digit identification number called **IMEI** (*International Mobile Equipment Identity*). Each SIM card also has a unique (and secret) identification number called **IMSI** (*International Mobile Subscriber Identity*). This code can be protected using a 4-digit key called a *PIN code*.

The SIM card therefore allows each user to be identified independently of the terminal used during communication with a base station. Communications occur through a radio link (air interface) between a mobile station and a base station.

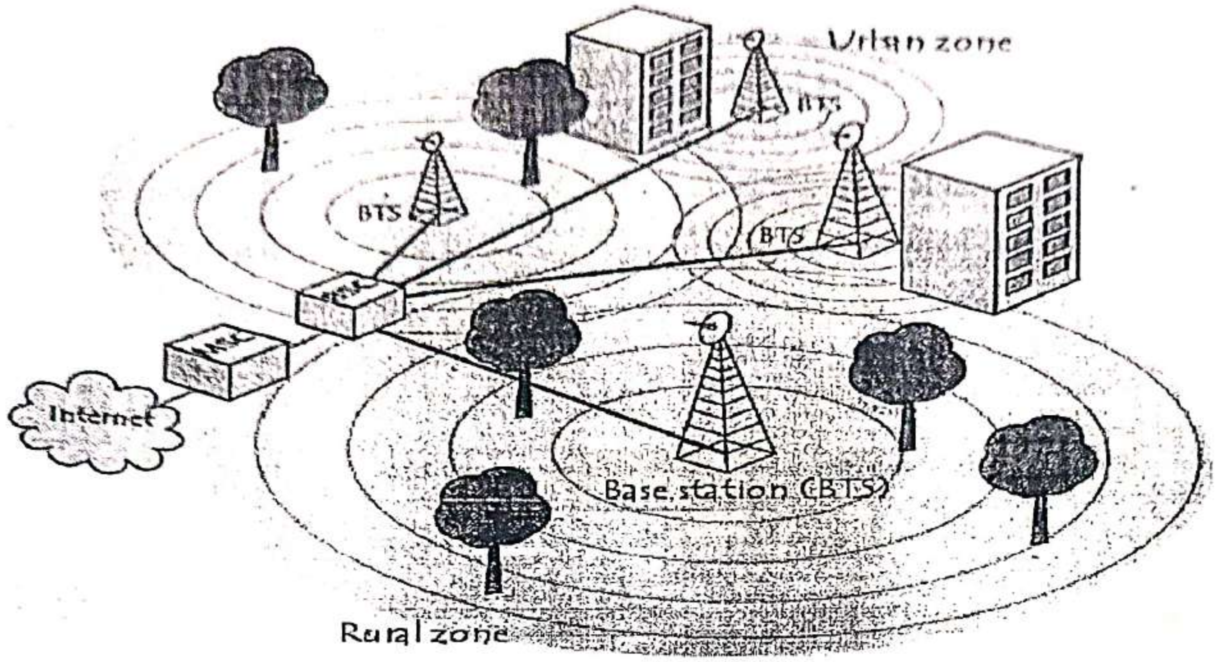


Fig :12: Architecture of a GSM Network

All the base stations of a cellular network are connected to a **base station controller (BSC)** which is responsible for managing distribution of the resources. The system consisting of the base station controller and its connected base stations is called the **Base Station Subsystem (BSS)**.

Finally, the base station controllers are themselves physically connected to the **Mobile Switching Centre** (MSC), managed by the telephone network operator, which connects them to the public telephone network and the Internet. The MSC belongs to a **Network Station Subsystem** (NSS), which is responsible for managing user identities, their location and establishment of communications with other subscribers. The MSC is generally connected to databases that provide additional functions:

1. **Home Location Register (HLR)**: is a database containing information (geographic position, administrative information etc.) of the subscribers registered in the area of the switch (MSC).

2. **Visitor Location Register (VLR):** is a database containing information of users other than the local subscribers. The VLR retrieves the data of a new user from the HLR of the user's subscriber zone. The data is maintained as long as the user is in the zone and is deleted when the user leaves or after a long period of inactivity (terminal off).

3. **Equipment Identify Register (EIR)**: is a database listing the mobile terminals.

4. **Authentication Centre (AUC):** is responsible for verifying user identities.

5. The cellular network formed in this way is designed to support mobility via management of *handovers* (movements from one cell to another).

Finally, GSM networks support the concept of **roaming** i.e., movement from one operator network to another.

Introduction to Modem:

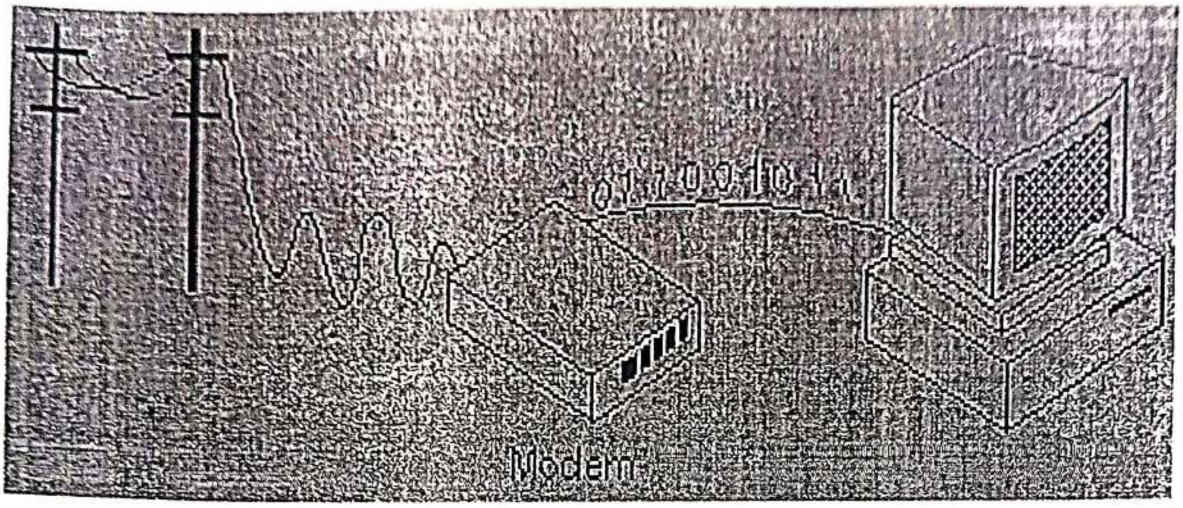


Fig :13: Diagram of Working of a MODEM

Modem stands for *modulator-demodulator:*

A modem is a device or program that enables a computer to transmit data over telephone or cable lines. Computer information is stored digitally, whereas information transmitted over telephone lines is transmitted in the form of analog waves. A modem converts between these two forms.

Fortunately, there is one standard interface for connecting external modems to computers called *RS-232*. Consequently, any external modem can be attached to any computer that has an RS-232 port, which almost all personal computers have. There are also modems that come as an expansion board that can be inserted into a vacant expansion slot. These are sometimes called *onboard* or *internal modems*.

While the modem interfaces are standardized, a number of different protocols for formatting data to be transmitted over telephone lines exist. Some, like CCITT V.34 are official standards, while others have been developed by private companies. Most modems have built-in support for the more common protocols at slow data transmission speeds at least, most modems can communicate with each other. At high transmission speeds, however, the protocols are less standardized.

Apart from the transmission protocols that they support, the following characteristics distinguish one modem from another:

- **Bps:** How fast the modem can transmit and receive data. At slow rates, modems are measured in terms of baud rates. The slowest rate is 300 baud (about 25 cps). At higher speeds, modems are measured in terms of bits per second (bps). The fastest modems run at 57,600 bps, although they can achieve even higher data transfer rates by compressing the data. Obviously, the faster the transmission rate, the faster the data can be sent and received. It should be noted that the data cannot be received at a faster rate than it is being sent.

- **Voice/data:** Many modems support a switch to change between voice and data modes. In data mode, the modem acts like a regular modem. In voice mode, the modem acts like a regular telephone. Modems that support a voice/data switch have a built-in loudspeaker and microphone for voice communication.

- **Auto-answer:** An auto-answer modem enables the computer to receive calls in the absence of the operator.

- **Data compression:** Some modems perform data compression, which enables them to send data at faster rates. However, the modem at the receiving end must be able to decompress the data using the same compression technique.

-

- **Flash memory:** Some modems come with *flash memory* rather than conventional ROM which means that the communications protocols can be easily updated if necessary.

- **Fax capability:** Most modern modems are fax modems, which mean that they can send and receive faxes.

GSM Modem:

A GSM modem is a wireless modem that works with a GSM wireless network. A wireless modem behaves like a dial-up modem. The main difference between them is that a dial-up modem sends and receives data through a fixed telephone line while a wireless modem sends and receives data through radio waves.

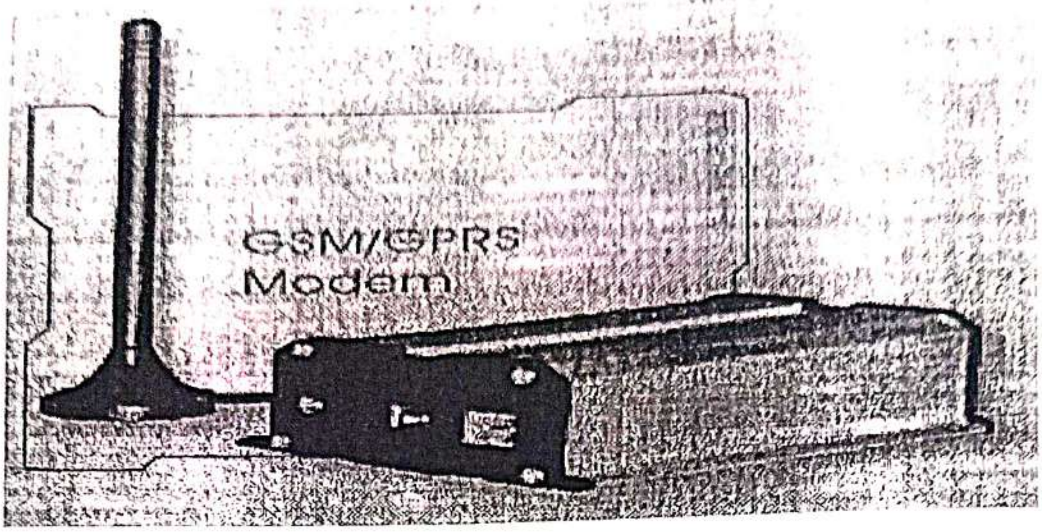


Fig :14: Configaration of a GSM MODEM

A GSM modem can be an external device or a PC Card / PCMCIA Card. Typically, an external GSM modem is connected to a computer through a serial cable or a USB cable. A GSM modem in the form of a PC Card / PCMCIA Card is designed for use with a laptop computer. It should be inserted into one of the PC Card / PCMCIA Card slots of a laptop computer. Like a GSM mobile phone, a GSM modem requires a SIM card from a wireless carrier in order to operate.

A SIM card contains the following information:

- Subscriber telephone number (MSISDN)
- International subscriber number **(IMSI, International Mobile Subscriber Identity)**
- State of the SIM card
- Service code (operator)

- Authentication key
- PIN (***Personal Identification Code***)
- PUK (***Personal Unlock Code***)

Computers use AT commands to control modems. Both GSM modems and dial-up modems support a common set of standard AT commands. In addition to the standard AT commands, GSM modems support an extended set of AT commands. These extended AT commands are defined in the GSM standards. With the extended AT commands, the following operations can be performed:

- Reading, writing and deleting SMS messages.
- Sending SMS messages.
- Monitoring the signal strength.
- Monitoring the charging status and charge level of the battery.
- Reading, writing and searching phone book entries.

Establishing connection between PC and GSM modem

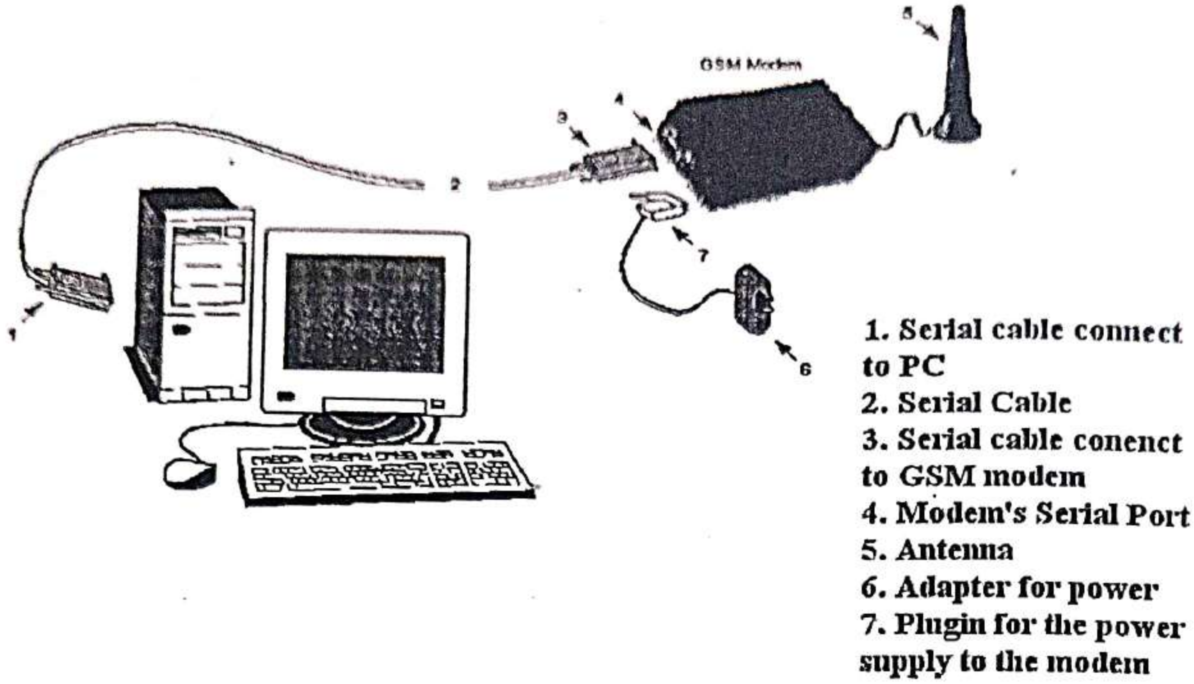


Fig :15: Connection between PC and GSM MODEM

The number of SMS messages that can be processed by a GSM modem per minute is very low i.e., about 6 to 10 SMS messages per minute.

Introduction to AT Commands:

AT commands are instructions used to control a modem. AT is the abbreviation of ATtention. Every command line starts with "AT" or "at". That's the reason, modem commands are called AT commands. Many of the commands that are used to control wired dial-up modems, such as ATD (Dial), ATA (Answer), ATH (Hook control) and ATO (Return to online data state) are also supported by GSM modems and mobile phones.

Besides this common AT command set, GSM modems and mobile phones support an AT command set that is specific to the GSM technology, which includes SMS-related commands like AT+CMGS (Send SMS message), AT+CMSS (Send SMS message from storage), AT+CMGL (List SMS messages) and AT+CMGR (Read SMS messages).

It should be noted that the starting "AT" is the prefix that informs the modem about the start of a command line. It is not part of the AT command name. For example, D is the actual AT command name in ATD and +CMGS is the actual AT command name in AT+CMGS.

Some of the tasks that can be done using AT commands with a GSM modem or mobile phone are listed below:

- Get basic information about the mobile phone or GSM modem. For example, name of manufacturer (AT+CGMI), model number (AT+CGMM), IMEI number (International Mobile Equipment Identity) (AT+CGSN) and software version (AT+CGMR).

- Get basic information about the subscriber. For example, MSISDN (AT+CNUM) and IMSI number (International Mobile Subscriber Identity) (AT+CIMI).

- Get the current status of the mobile phone or GSM/GPRS modem. For example, mobile phone activity status (AT+CPAS), mobile network registration status (AT+CREG), radio signal strength (AT+CSQ), battery charge level and battery charging status (AT+CBC).

- Establish a data connection or voice connection to a remote modem (ATD, ATA, etc).

- Send and receive fax (ATD, ATA, AT+F*).

- Send (AT+CMGS, AT+CMSS), read (AT+CMGR, AT+CMGL), write (AT+CMGW) or delete (AT+CMGD) SMS messages and obtain notifications of newly received SMS messages (AT+CNMI).

- Read (AT+CPBR), write (AT+CPBW) or search (AT+CPBF) phonebook entries.

- Perform security-related tasks, such as opening or closing facility locks (AT+CLCK), checking whether a facility is locked (AT+CLCK) and... changing passwords(AT+CPWD). (Facility lock examples: SIM lock [a password must be given to the SIM card every time the mobile phone is switched on] and PH-SIM lock [a certain SIM card is associated with the mobile phone. To use other SIM cards with the mobile phone, a password must be entered.])

- Control the presentation of result codes / error messages of AT commands. For example, the user can control whether to enable certain error messages (AT+CMEE) and whether error messages should be displayed in numeric format or verbose format (AT+CMEE=1 or AT+CMEE=2).

- Get or change the configurations of the mobile phone or GSM/GPRS modem. For example, change the GSM network (AT+COPS), bearer service type (AT+CBST), radio link protocol parameters (AT+CRLP), SMS center address (AT+CSCA) and storage of SMS messages (AT+CPMS).

- Save and restore configurations of the mobile phone or GSM/GPRS modem. For example, save (AT+CSAS) and restore (AT+CRES) settings related to SMS messaging such as the SMS center address.

It should be noted that the mobile phone manufacturers usually do not implement all AT commands, command parameters and parameter values in their mobile phones. Also, the behavior of the implemented AT commands may be different from that defined in the standard. In general, GSM modems, designed for wireless applications, have better support of AT commands than ordinary mobile phones.

Basic concepts of SMS technology:

1. ***Validity Period of an SMS Message:***

An SMS message is stored temporarily in the SMS center if the recipient mobile phone is offline. It is possible to specify the period after which the SMS message will be deleted from the SMS center so that the SMS message will not be forwarded to the recipient mobile phone when it becomes online. This period is called the validity period. A mobile phone should have a menu option that can be used to set the validity period. After setting it, the mobile phone will include the validity period in the outbound SMS messages automatically.

2. ***Message Status Reports:***

Sometimes the user may want to know whether an SMS message has reached the recipient mobile phone successfully. To get this information, you need to set a flag in the SMS message to notify the SMS center that a status report is required about the delivery of this SMS message. The status report is sent to the user mobile in the form of an SMS message.

A mobile phone should have a menu option that can be used to set whether the status report feature is on or off. After setting it, the mobile phone will set the corresponding flag in the outbound SMS messages for you automatically. The status report feature is turned off by default on most mobile phones and GSM modems.

3. ***Message Submission Reports:***

After leaving the mobile phone, an SMS message goes to the SMS center. When it reaches the SMS center, the SMS center will send back a message submission report to the mobile phone to inform whether there are any errors or failures (e.g. incorrect SMS message format, busy SMS center, etc). If there is no error or failure, the SMS center sends back a positive submission report to the mobile phone. Otherwise it sends back a negative submission report to the mobile phone. The mobile phone may then notify the user that the message submission was failed and what caused the failure.

If the mobile phone does not receive the message submission report after a period of time, it concludes that the message submission report has been lost. The mobile phone may then send the SMS message again to the SMS center. A flag will be set in the new SMS message to inform the SMS center that this SMS message has been sent before. If the previous message submission was successful, the SMS center will ignore the new SMS message but send back a message submission report to the mobile phone. This mechanism prevents the sending of the same SMS message to the recipient multiple times.

Sometimes the message submission report mechanism is not used and the acknowledgement of message submission is done in a lower layer.

4 .Message Delivery Reports:

After receiving an SMS message, the recipient mobile phone will send back a message delivery report to the SMS center to inform whether there are any errors or failures (example causes: unsupported SMS message format, not enough storage space, etc). This process is transparent to the mobile user. If there is no error or failure, the recipient mobile phone sends back a positive delivery report to the SMS center. Otherwise it sends back a negative delivery report to the SMS center.

If the sender requested a status report earlier, the SMS center sends a status report to the sender when it receives the message delivery report from the recipient. If the SMS center does not receive the message delivery report after a period of time, it concludes that the message delivery report has been lost. The SMS center then ends the SMS message to the recipient for the second time.

Sometimes the message delivery report mechanism is not used and the acknowledgement of message delivery is done in a lower layer.

LCD (Liquid Cristal Display):

A liquid crystal display (LCD) is a thin, flat display device made up of any number of color or monochrome pixels arrayed in front of a light source or reflector. Each pixel consists of a column of liquid crystal molecules suspended between two transparent electrodes, and two polarizing filters, the axes of polarity of which are perpendicular to each other. Without the liquid crystals between them, light passing through one would be blocked by the other. The liquid crystal twists the polarization of light entering one filter to allow it to pass through the other.

A program must interact with the outside world using input and output devices that communicate directly with a human being. One of the most common devices attached to an controller is an LCD display. Some of the most common LCDs connected to the contollers are 16X1, 16x2 and 20x2 displays. This means 16 characters per line by 1 line 16 characters per line by 2 lines and 20 characters per line by 2 lines, respectively.

Many microcontroller devices use 'smart LCD' displays to output visual information. LCD displays designed around LCD NT-C1611 module, are inexpensive, easy to use, and it is even possible to produce a readout using the 5X7 dots plus cursor of the display. They have a standard ASCII set of characters and mathematical symbols. For an 8-bit data bus, the display requires a +5V supply plus 10 I/O lines (RS RW D7 D6 D5 D4 D3 D2 D1 D0). For a 4-bit data bus it only requires the supply lines plus 6 extra lines(RS RW D7 D6 D5 D4). When the LCD display is not enabled, data lines are tri-state and they do not interfere with the operation of the microcontroller.

Features:

(1) Interface with either 4-bit or 8-bit microprocessor.

(2) Display data RAM

(3) 80x8 bits (80 characters).

(4) Character generator ROM

(5) 160 different 5 x7 dot-matrix character patterns.

(6) Character generator RAM

(7) 8 different user programmed 5 x7 dot-matrix patterns.

(8).Display data RAM and character generator RAM may be Accessed by the microprocessor.

(9) Numerous instructions

(10) Clear Display, Cursor Home, Display ON/OFF, Cursor ON/OFF,

Blink Character, Cursor Shift, Display Shift.

(11) Built-in reset circuit is triggered at power ON.

(12) Built-in oscillator.

Description Of 16x2:

This is the first interfacing example for the Parallel Port. We will start with something simple. This example doesn't use the Bi-directional feature found on newer ports, thus it should work with most, if no all Parallel Ports. It however doesn't show the use of the Status Port as an input. So what are we interfacing? A 16 Character x 2 Line LCD Module to the Parallel Port. These LCD Modules are very common these days, and are quite simple to work with, as all the logic required to run them is on board.

Schematic Diagram:

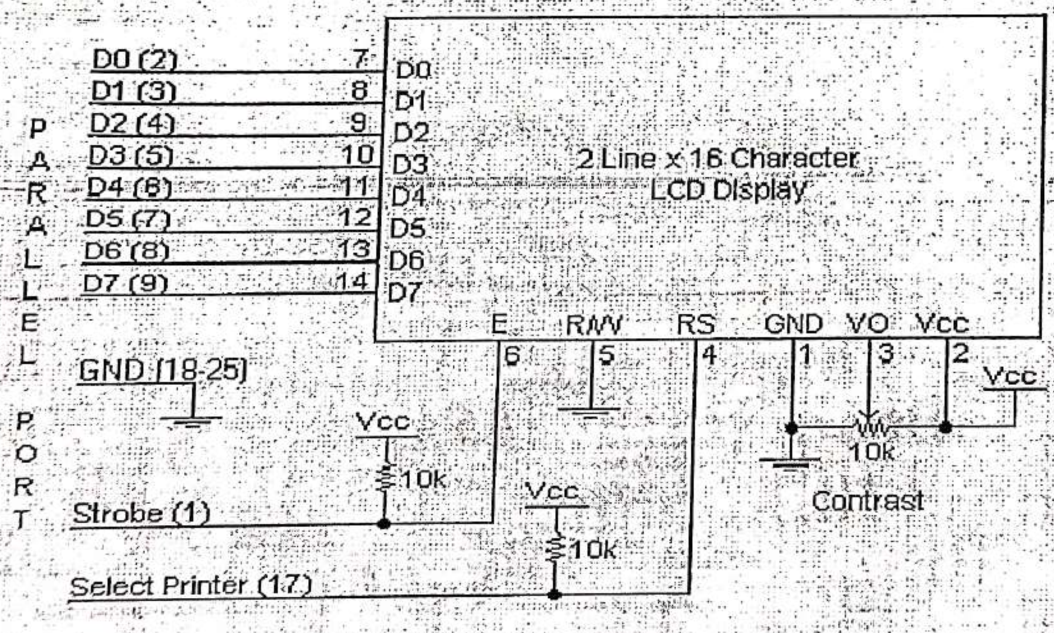


Fig :16: Schematic diagram of a 2-Line x 16 Character LCD Supply

- The LCD panel's *Enable* and *Register Select* is connected to the Control Port. The Control Port is an open collector / open drain output. While most Parallel Ports have internal pull-up resistors, there are a few which don't. Therefore by incorporating the two 10K external pull up resistors, the circuit is more portable for a wider range of computers, some of which may have no internal pull up resistors.

- We make no effort to place the Data bus into reverse direction. Therefore we hard wire the *R/W* line of the LCD panel, into write mode. This will cause no bus conflicts on the data lines. As a result we cannot read back the LCD's internal Busy Flag which tells us if the LCD has accepted and finished processing the last instruction. This problem is overcome by inserting known delays into our program.

16 x 2 Alphanumeric LCD Module Features:

- Intelligent, with built-in Hitachi HD44780 compatible LCD controller and RAM providing simple interfacing
- 61 x 15.8 mm viewing area
- 5 x 7 dot matrix format for 2.96 x 5.56 mm characters, plus cursor line
- Can display 224 different symbols
- Low power consumption (1 mA typical)
- Powerful command set and user-produced characters
- TTL and CMOS compatible

- Connector for standard 0.1-pitch pin headers

16 x 2 Alphanumeric LCD Module Specifications:

1	V_{SS}	-	Power, GND
2	V_{DD}	-	Power, 5V
3	Vo	-	Power, for LCD Drive
4	RS	H/L	Register Select Signal H: Data Input L: Instruction Input
5	R/W	H/L	H: Data Read (LCD->MPU) L: Data Write (MPU->LCD)
6	E	H,H->L	Enable
7-14	DB0-DB7	H/L	Data Bus; Software selectable 4- or 8-bit mode
15	NC	-	NOT CONNECTED
16	NC	-	NOT CONNECTED

FEATURES:

- 5 x 8 dots with cursor
- Built-in controller (KS 0066 or Equivalent)
- + 5V power supply (Also available for + 3V)
- 1/16 duty cycle
- B/L to be driven by pin 1, pin 2 or pin 15, pin 16 or A.K (LED)
- N.V. optional for + 3V power supply

Data can be placed at any location on the LCD. For 16×1 LCD, the address locations are:

POSITION		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
ADDRESS	LINE1	00	01	02	03	04	05	06	07	40	41	42	43	44	45	46	47

Fig :15: Address locations for a 1x16 line LCD

Even limited to character based modules, there is still a wide variety of shapes and sizes available. Line lengths of 8,16,20,24,32 and 40 characters are all standard, in one, two and four line versions.

Several different LC technologies exists. "supertwist" types, for example, offer Improved contrast and viewing angle over the older "twisted nematic" types. Some modules are available with back lighting, so that they can be viewed in dimly-lit conditions. The back lighting may be either "electro-luminescent", requiring a high voltage inverter circuit, or simple LED illumination.

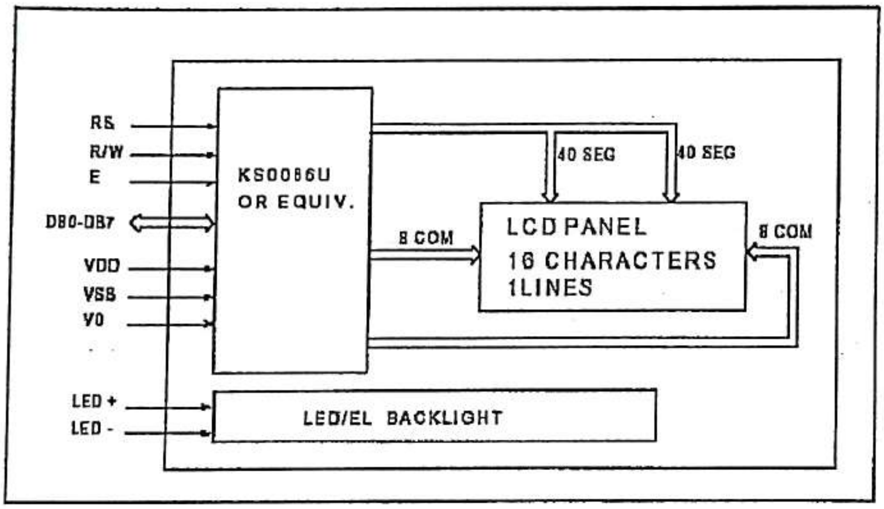


Fig :17: Connection and Control Signals of LCD

Power supply for LCD driving:

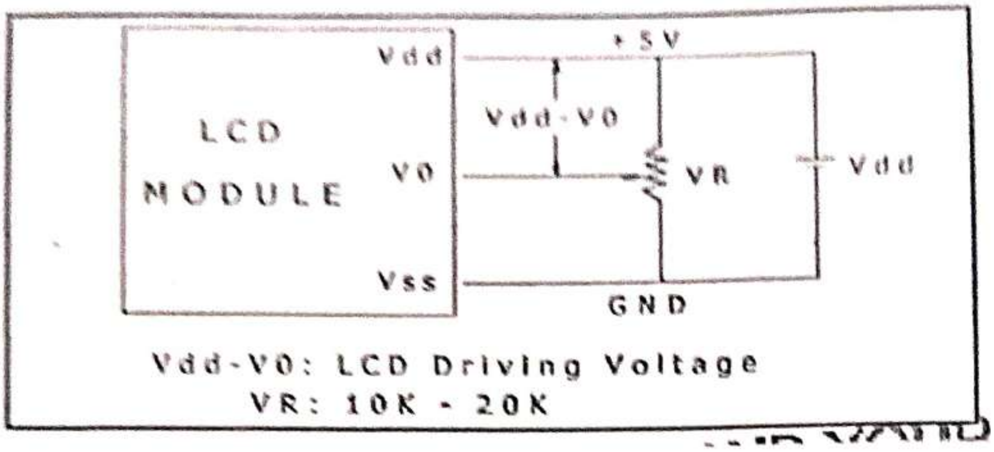


Fig: 18:power supply for LCD

PIN DESCRIPTION:

Most LCDs with 1 controller has 14 Pins and LCDs with 2 controller has 16 Pins (two pins are extra in both for back-light LED connections).

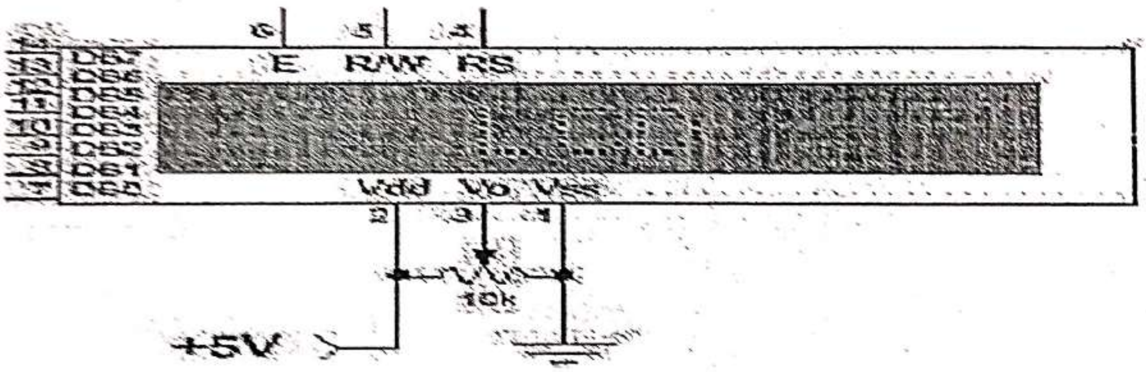


Fig 19: pin diagram of 1x16 lines LCD

Pin specifications:

PIN	SYMBOL	FUNCTION
1	Vss	Power Supply(GND)
2	Vdd	Power Supply(+5V)
3	Vo	Contrast Adjust
4	RS	Instruction/Data Register Select
5	R/W	Data Bus Line
6	E	Enable Signal
7-14	DB0-DB7	Data Bus Line
15	A	Power Supply for LED B/L(+)
16	K	Power Supply for LED B/L(-)

Fig 17: Pin specifications

CONTROL LINES:

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.

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 sould be displayed on the screen. For example, to display the letter "T" on the screen you would set RS high.

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.

Logic status on control lines:

• E - 0 Access to LCD disabled

- 1 Access to LCD enabled

• R/W - 0 Writing data to LCD

- 1 Reading data from LCD

• RS - 0 Instructions

- 1 Character

Writing data to the LCD:

1) Set R/W bit to low

2) Set RS bit to logic 0 or 1 (instruction or character)

3) Set data to data lines (if it is writing)

4) Set E line to high

5) Set E line to low

Read data from data lines (If it is reading) on LCD:

1) Set R/W bit to high

2) Set RS bit to logic 0 or 1 (instruction or character)

3) Set data to data lines (if it is writing)

4) Set E line to high

5) Set E line to low

Entering Text:

First, a little tip: it is manually a lot easier to enter characters and commands in hexadecimal rather than binary (although, of course, you will need to translate commands from binary couple of sub-miniature hexadecimal rotary switches is a simple matter, although a little bit into hex so that you know which bits you are setting). Replacing the d.i.l. switch pack with a of re-wiring is necessary.

SWITCHES:

The switches must be the type where On = 0, so that when they are turned to the zero position, all four outputs are shorted to the common pin, and in position "F", all four outputs are open circuit.

All the available characters that are built into the module are shown in Table 3. Studying the table, you will see that codes associated with the characters are quoted in binary and hexadecimal, most significant bits ("left-hand" four bits) across the top, and least significant bits ("right-hand" four bits) down the left.

Most of the characters conform to the ASCII standard, although the Japanese and Greek characters (and a few other things) are obvious exceptions. Since these intelligent modules were designed in the "Land of the Rising Sun," it seems only fair that their Katakana phonetic symbols should also be incorporated.

The more extensive Kanji character set, which the Japanese share with the Chinese, consisting of several thousand different characters, is not included! Using the switches, of whatever type, and referring to Table 3, enter a few characters onto the display, both letters and numbers. The RS switch (S10) must be "up" (logic 1) when sending the characters, and switch E (S9) must be pressed for each of them.

Thus the operational order is: set RS high, enter character, trigger E, leave RS high, enter another character, trigger E, and so on.

The first 16 codes in Table 3, 00000000 to 00001111, ($00 to $0F) refer to the CGRAM. This is the Character Generator RAM (random access memory), which can be used to hold user-defined graphics characters. This is where these modules really start to show their potential, offering such capabilities as bar graphs, flashing symbols, even animated characters. Before the user-defined characters are set up, these codes will just bring up strange looking symbols.

Codes 00010000 to 00011111 ($10 to $1F) are not used and just display blank characters. ASCII codes "proper" start at 00100000 ($20) and end with 01111111 ($7F). Codes 10000000 to 10011111 ($80 to $9F) are not used, and 10100000 to 11011111 ($A0 to $DF) are the Japanese characters.

Fig 20: character details in LCD

Initialization by Instructions:

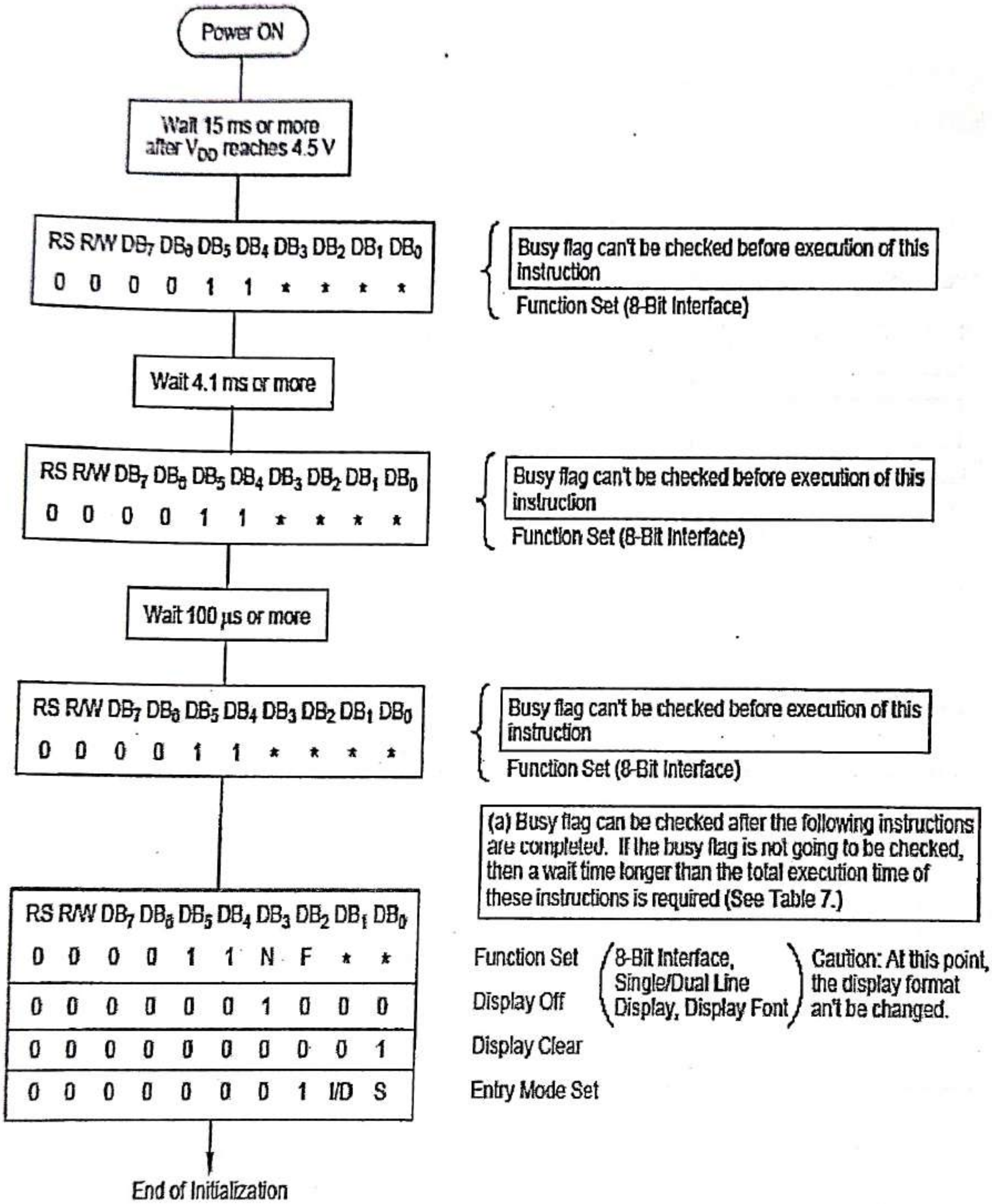
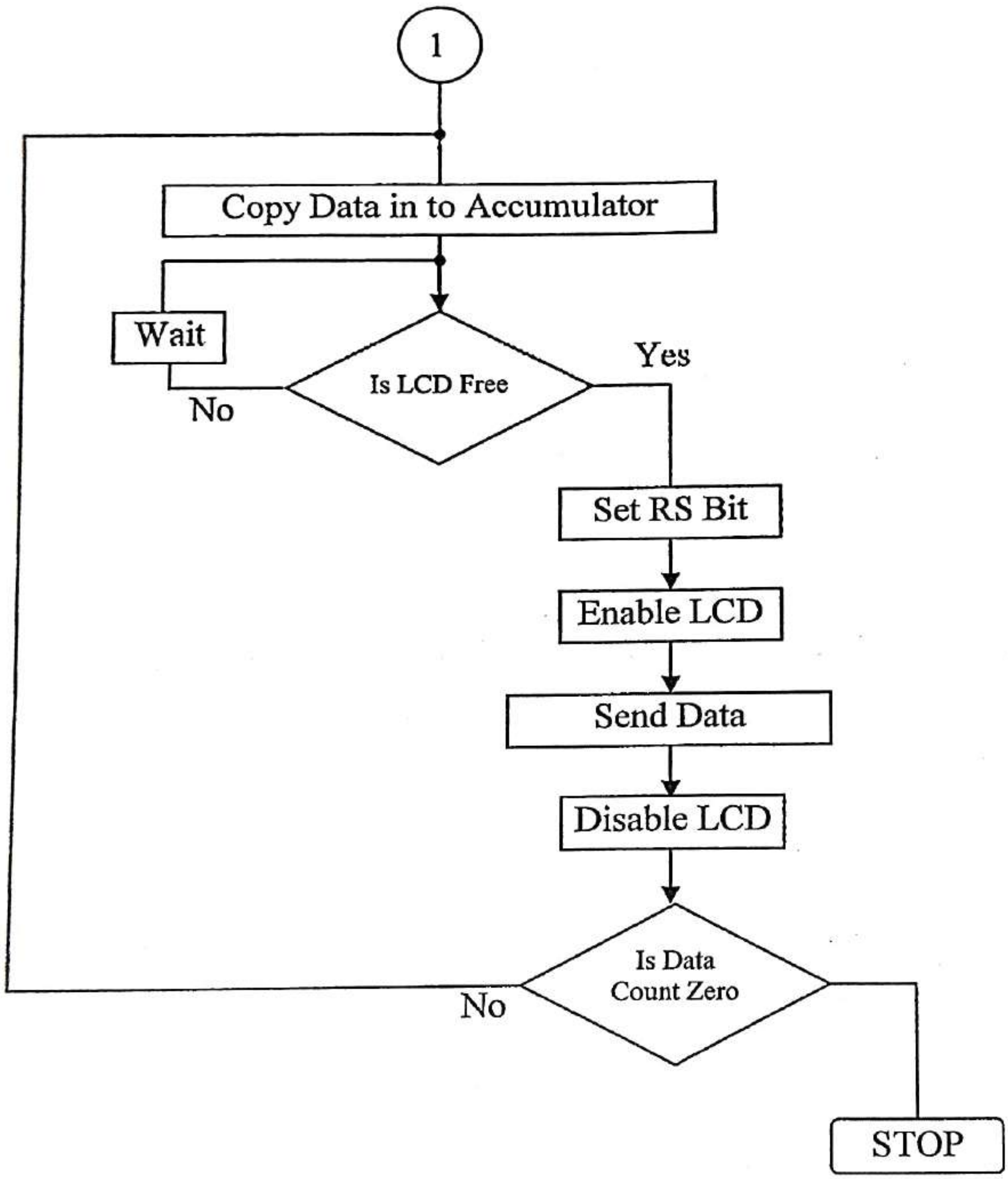


Fig 21: flow chart of LCD

If the power conditions for the normal operation of the internal reset circuit are not satisfied, then executing a series of instructions must initialize LCD unit. The procedure for this initialization process is as above show.

FLOWCHART:



POWER SUPPLY:

Block diagram:

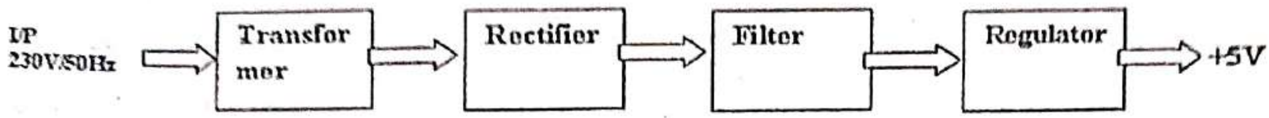


Fig :22: Block diagram of a Power Supply

Circuitdiagram:

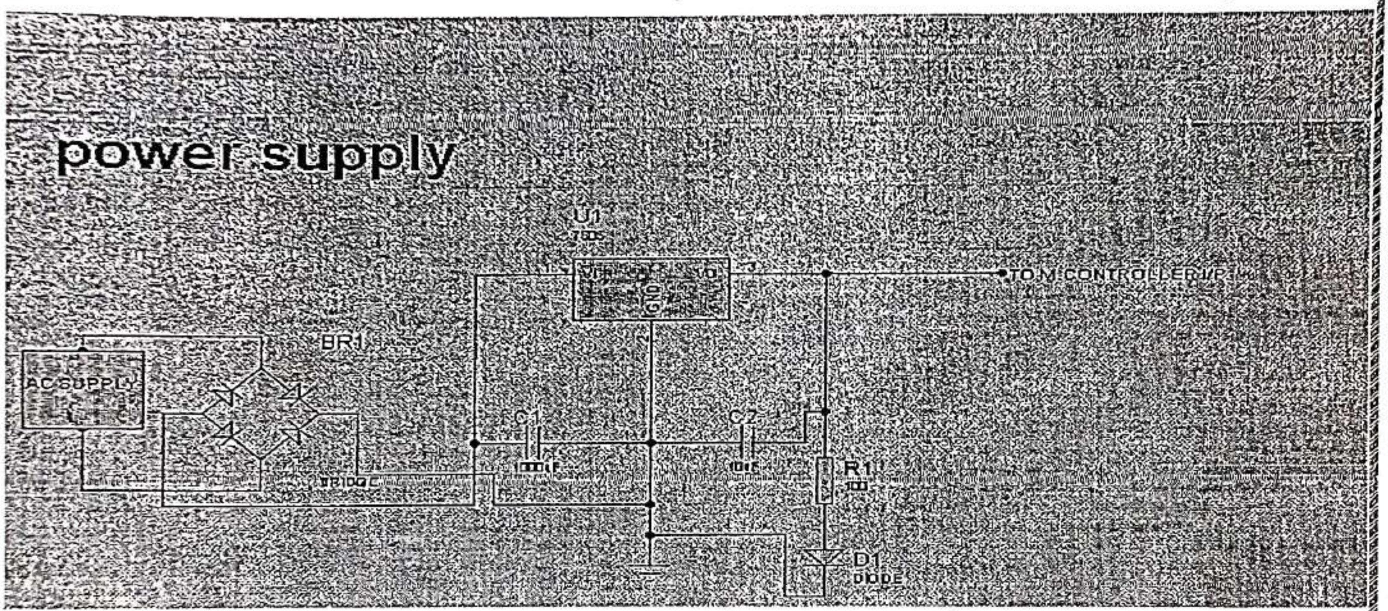


Fig :23: Circuit Diagram of a Power Supply

Description:

Transformer:

A **transformer** is a device that transfers electrical energy from one circuit to another through inductively coupled conductors—the transformer's coils. A varying current in the first or *primary* winding creates a varying magnetic flux in the transformer's core, and thus a varying magnetic field through the *secondary* winding. This varying magnetic field induces a varying electromotive force (EMF) or "voltage" in the secondary winding. This effect is called mutual induction.

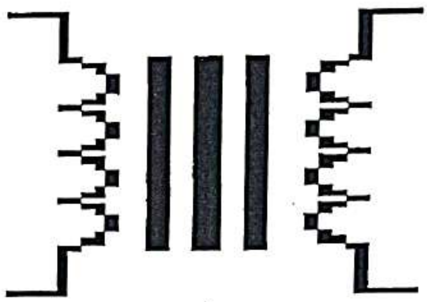


Fig :24: Transformer Symbol

Transformer is a device that converts the one form energy to another form of energy like a transducer.

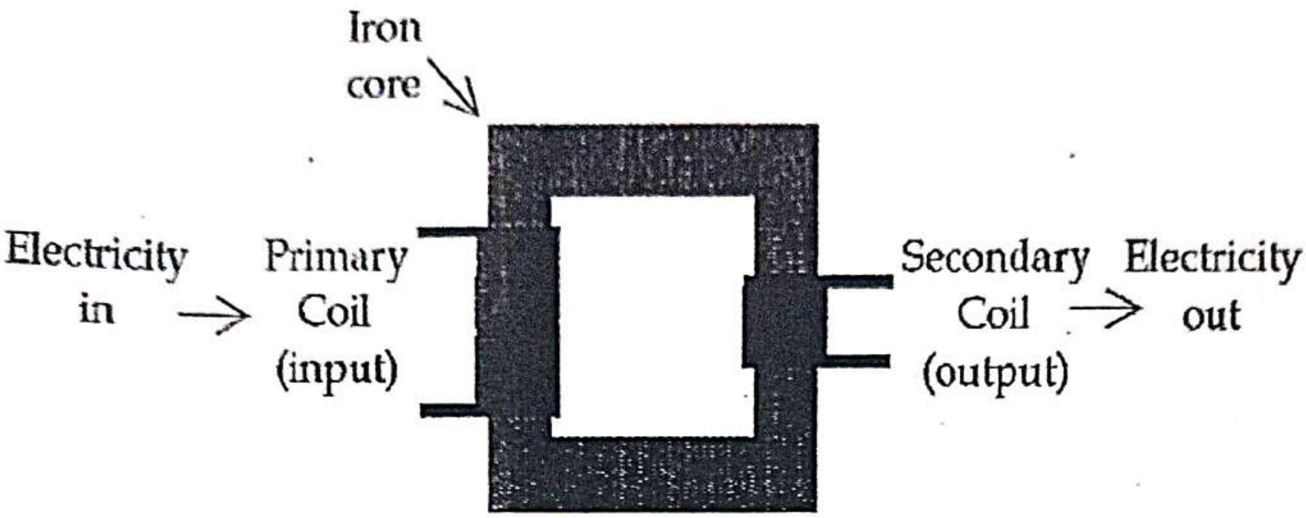


Fig :25: Construction Figure of The Transformer

Basic Principle :

A transformer makes use of Faraday's law and the ferromagnetic properties of an iron core to efficiently raise or lower AC voltages. It of course cannot increase power so that if the voltage is raised, the current is proportionally lowered and vice versa.

From Faraday's Law

$$\frac{V_s}{V_p} = \frac{N_s}{N_p}$$

For Ideal transformer

The voltage ratio is equal to the turns ratio, and power in equals power out.

From conservation of energy

$$P_p = V_p I_p = V_s I_s = P_s$$

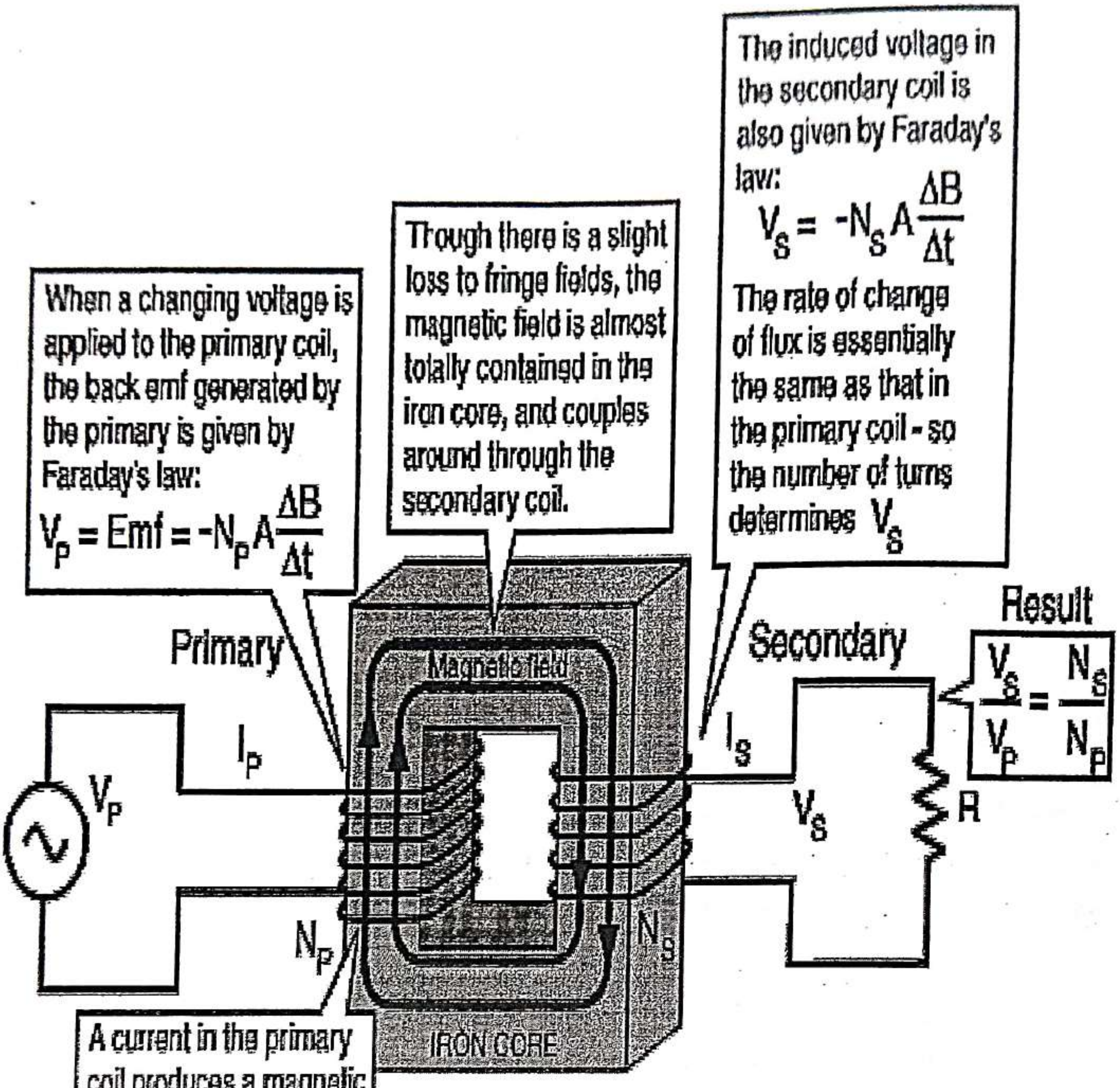


Fig :26: Basic principle of the Transformer

Transformer Working:

A transformer consists of two coils (often called 'windings') linked by an iron core, as shown in figure below. There is no electrical connection between the coils, instead they are linked by a magnetic field created in the core.

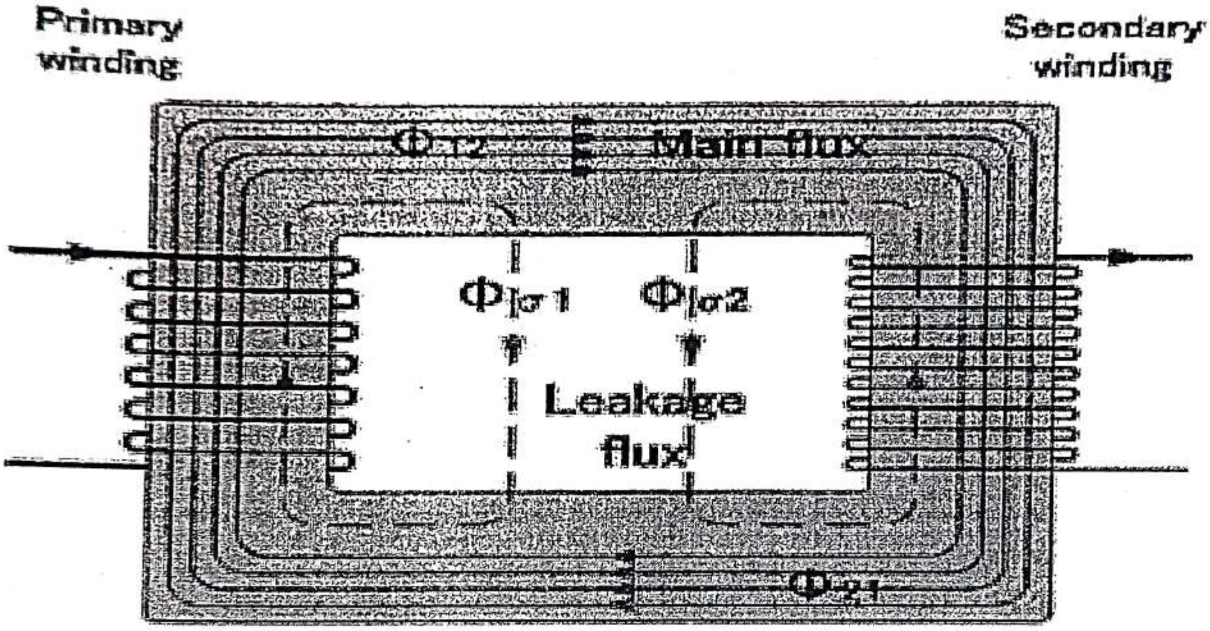


Fig :27: Basic Transformer Construction

Transformers are used to convert electricity from one voltage to another with minimal loss of power. They only work with AC (alternating current) because they require a changing magnetic field to be created in their core. Transformers can increase voltage (step-up) as well as reduce voltage (step-down).

Alternating current flowing in the primary (input) coil creates a continually changing magnetic field in the iron core. This field also passes through the secondary (output) coil and the changing strength of the magnetic field induces an alternating voltage in the secondary coil. If the secondary coil is connected to a load the induced voltage will make an induced current flow. The correct term for the induced voltage is 'induced electromotive force' which is usually abbreviated to induced e.m.f.

The iron core is laminated to prevent 'eddy currents' flowing in the core. These are currents produced by the alternating magnetic field inducing a small voltage in the core, just like that induced in the secondary coil. Eddy currents waste power by needlessly heating up the core but they are reduced to a negligible amount by laminating the iron because this increases the electrical resistance of the core without affecting its magnetic properties.

Transformers have two great advantages over other methods of changing voltage:

1. They provide total electrical isolation between the input and output, so they can be safely used to reduce the high voltage of the mains supply.
2. Almost no power is wasted in a transformer. They have a high efficiency (power out / power in) of 95% or more.

Classification of Transformer:

- Step-Up Transformer
- Step-Down Transformer

Step-Down Transformer:

Step down transformers are designed to reduce electrical voltage. Their primary voltage is greater than their secondary voltage. This kind of transformer "steps down" the voltage applied to it. For instance, a step down transformer is needed to use a 110v product in a country with a 220v supply.

Step down transformers convert electrical voltage from one level or phase configuration usually down to a lower level. They can include features for electrical isolation, power distribution, and control and instrumentation applications. Step down transformers typically rely on the principle of magnetic induction between coils to convert voltage and/or current levels. Step down transformers are made from two or more coils of insulated wire wound around a core made of iron. When voltage is applied to one coil (frequently called the primary or input) it magnetizes the iron core, which induces a voltage in the other coil, (frequently called the secondary or output). The turn's ratio of the two sets of windings determines the amount of voltage transformation.

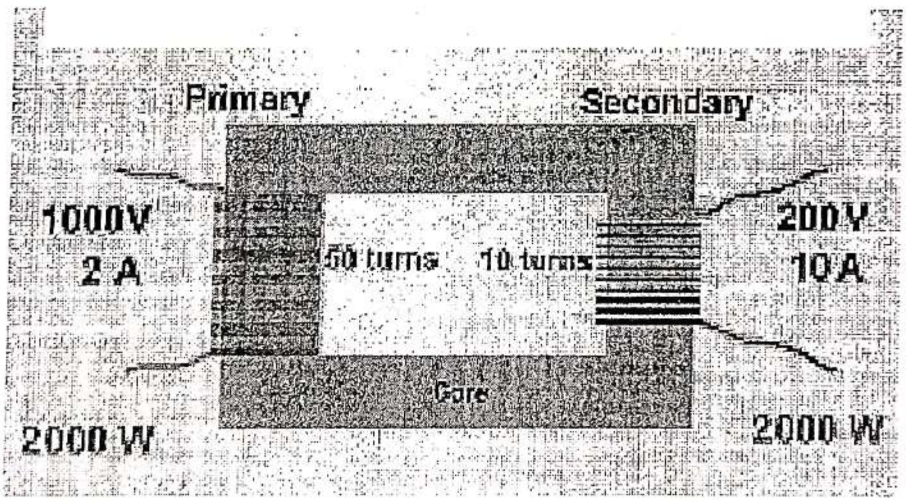


Fig :28: Step –Down Transformer

An example of this would be: 100 turns on the primary and 50 turns on the secondary, a ratio of 2 to 1.

Step down transformers can be considered nothing more than a voltage ratio device.

With step down transformers the voltage ratio between primary and secondary will mirror the "turn's ratio" (except for single phase smaller than 1 kva which have compensated secondary). A practical application of this 2 to 1 turn's ratio would be a 480 to 240 voltage step down. Note that if the input were 440 volts then the output would be 220 volts. The ratio between input and output voltage will stay constant. Transformers should not be operated at voltages higher than the nameplate rating, but may be operated at lower voltages than rated. Because of this it is possible to do some non-standard applications using standard transformers.

Single phase step down transformers 1 kva and larger may also be reverse connected to step-down or step-up voltages. (Note: single phase step up or step down transformers sized less than 1 KVA should not be reverse connected because the secondary windings have additional turns to overcome a voltage drop when the load is applied. If reverse connected, the output voltage will be less than desired.)

Step-Up Transformer:

A step up transformer has more turns of wire on the secondary coil, which makes a larger induced voltage in the secondary coil. It is called a step up transformer because the voltage output is larger than the voltage input.

Step-up transformer 110v 220v design is one whose secondary voltage is greater than its primary voltage. This kind of transformer "steps up" the voltage applied to it. For instance, a step up transformer is needed to use a 220v product in a country with a 110v supply.

A step up transformer 110v 220v converts alternating current (AC) from one voltage to another voltage. It has no moving parts and works on a magnetic induction principle; it can be designed to "step-up" or "step-down" voltage. So a step up transformer increases the voltage and a step down transformer decreases the voltage.

The primary components for voltage transformation are the step up transformer core and coil. The insulation is placed between the turns of wire to prevent shorting to one another or to ground. This is typically comprised of Mylar, nomex, Kraft paper, varnish, or other materials. As a transformer has no moving parts, it will typically have a life expectancy between 20 and 25 years.

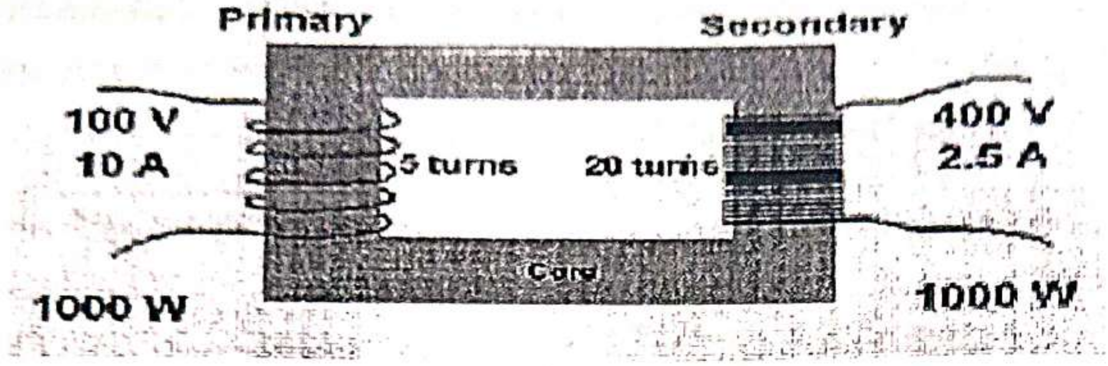


Fig :29: Step-Up Transformer

Applications:

Generally these **Step-Up Transformers** are used in industries applications only.

Turns Ratio and Voltage

The ratio of the number of turns on the primary and secondary coils determines the ratio of the voltages...

$$\frac{V_s}{V_p} = \frac{N_s}{N_p}$$

...where V_p is the primary (input) voltage, V_s is the secondary (output) voltage, N_p is the number of turns on the primary coil, and N_s is the number of turns on the secondary coil.

Diodes:

Diodes allow electricity to flow in only one direction. The arrow of the circuit symbol shows the direction in which the current can flow. Diodes are the electrical version of a valve and early diodes were actually called valves.

Fig :30: Diode Symbol

A **diode** is a device which only allows current to flow through it in one direction. In this direction, the diode is said to be 'forward-biased' and the only effect on the signal is that there will be a voltage loss of around 0.7V. In the opposite direction, the diode is said to be 'reverse-biased' and no current will flow through it.

Rectifier:

The purpose of a rectifier is to convert an AC waveform into a DC waveform (OR) Rectifier converts AC current or voltages into DC current or voltage. There are two different rectification circuits, known as **'half-wave'** and **'full-wave' rectifiers**. Both use components called **diodes** to convert **AC into DC.**

The Half-wave Rectifier:

The half-wave rectifier is the simplest type of rectifier since it only uses one diode, as shown in figure .

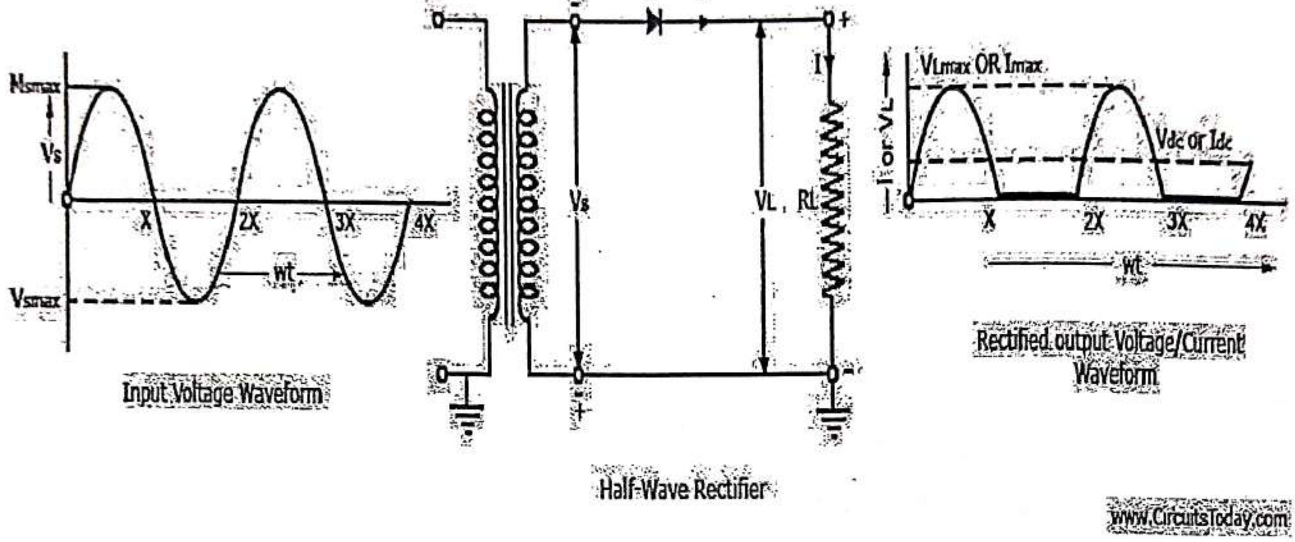


Fig :31: Circuit Diagram of a Half-Wave Rectifier

Figure 2 shows the AC input waveform to this circuit and the resulting output. As you can see, when the AC input is positive, the diode is forward-biased and lets the current through. When the AC input is negative, the diode is reverse-biased and the diode does not let any current through, meaning the output is 0V. Because there is a 0.7V voltage loss across the diode, the peak output voltage will be 0.7V less than Vs.

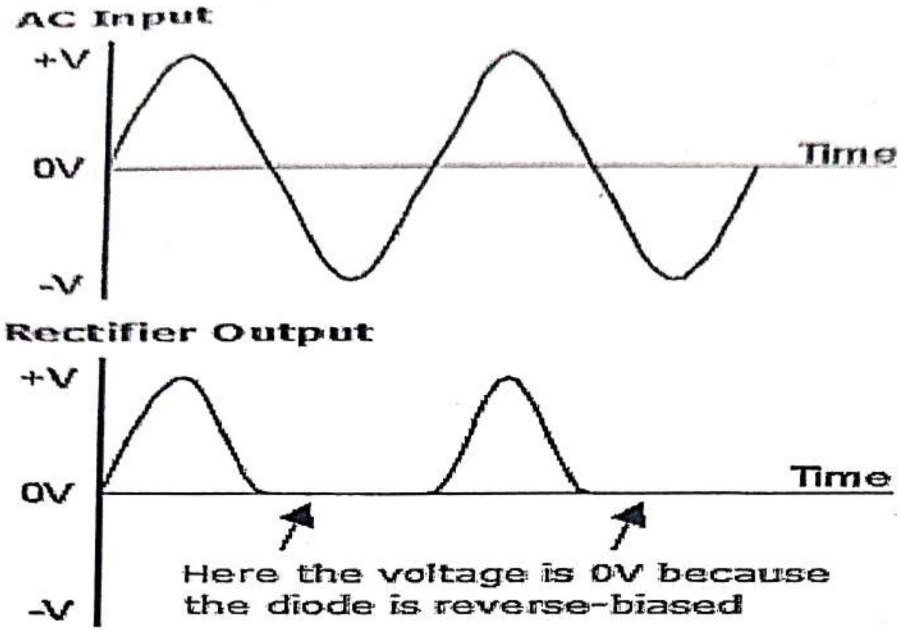


Fig :32: Input and Out put Waveforms of a Half-Wave Rectifier

While the output of the half-wave rectifier is DC (it is all positive), it would not be suitable as a power supply for a circuit. Firstly, the output voltage continually varies between 0V and Vs-0.7V, and secondly, for half the time there is no output at all.

The Full-wave Rectifier:

The circuit in figure 3 addresses the second of these problems since at no time is the output voltage 0V. This time four diodes are arranged so that both the positive and negative parts of the AC waveform are converted to DC. The resulting waveform is shown in figure 4.

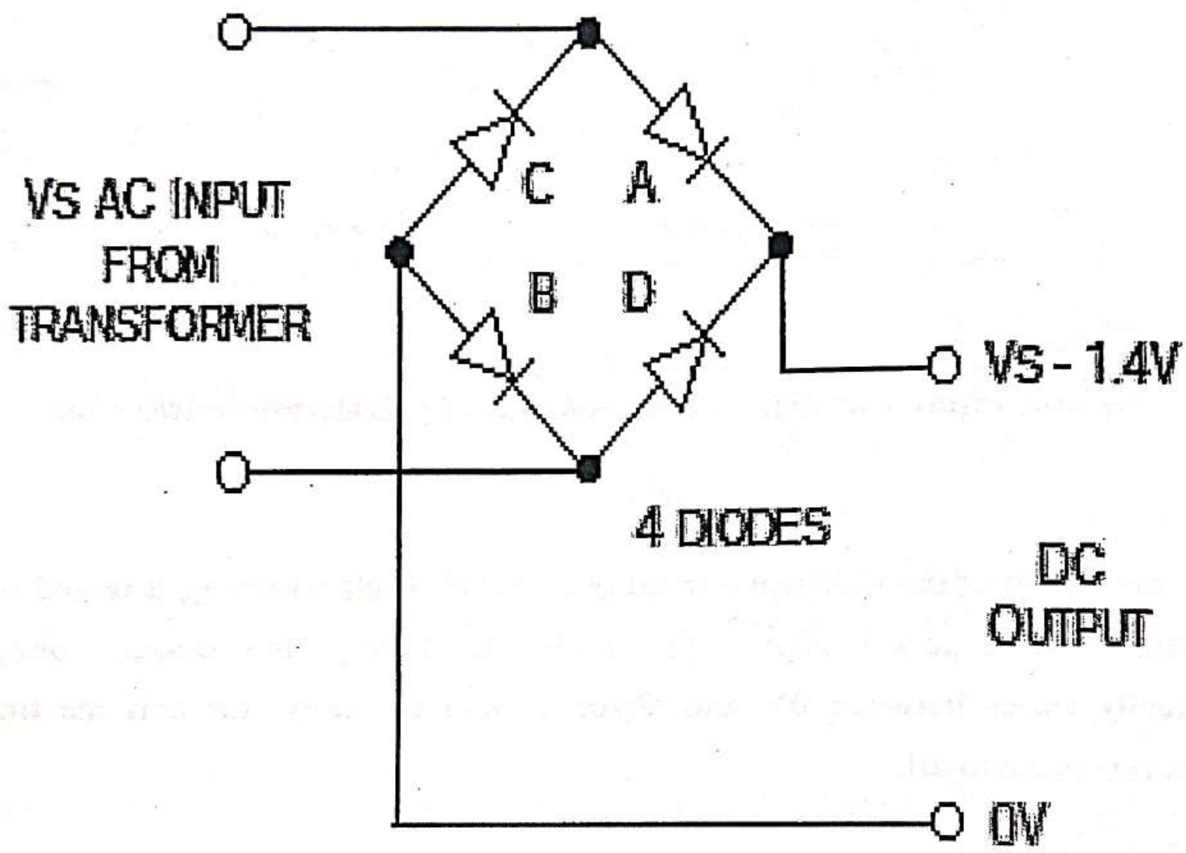


Fig :33: Circuit Diagram of a Bridge Rectifier Circuit

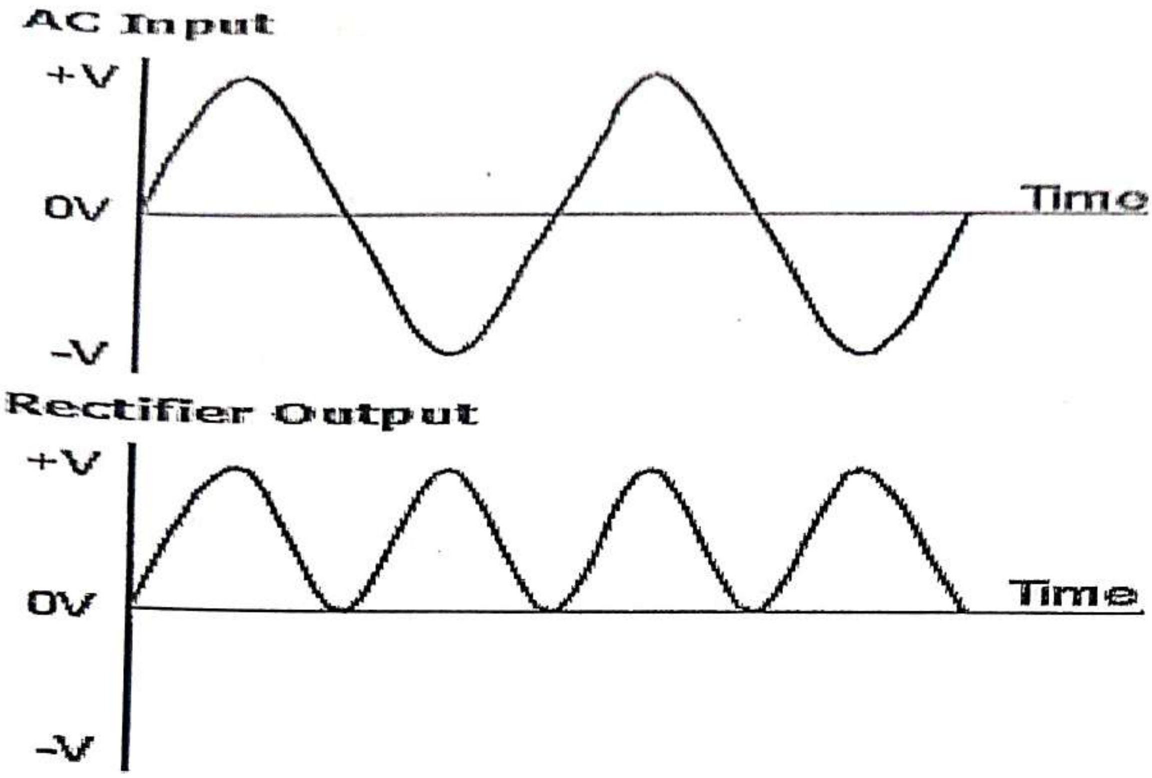


Fig :34: Input and Output of a Full-Wave Rectification

When the AC input is positive, diodes A and B are forward-biased, while diodes C and D are reverse-biased. When the AC input is negative, the opposite is true - diodes C and D are forward-biased, while diodes A and B are reverse-biased.

While the full-wave rectifier is an improvement on the half-wave rectifier, its output still isn't suitable as a power supply for most circuits since the output voltage still varies between 0V and Vs-1.4V. So, if you put 12V AC in, you will 10.6V DC out.

Capacitor Filter:

The **capacitor-input filter**, also called "Pi" filter due to its shape that looks like the Greek letter pi, is a type of electronic filter. Filter circuits are used to remove unwanted or undesired frequencies from a signal.

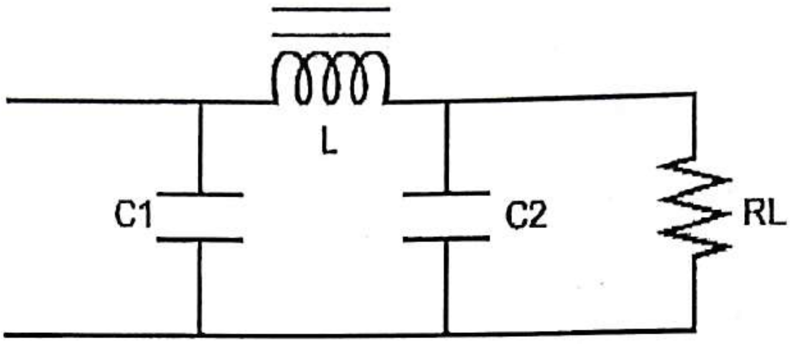


Fig :35: Circuit Diagram of a Capacitor Filter

A typical capacitor input filter consists of a filter capacitor C1, connected across the rectifier output, an inductor L, in series and another filter capacitor connected across the load.

1. The capacitor C1 offers low reactance to the AC component of the rectifier output while it offers infinite reactance to the DC component. As a result the capacitor shunts an appreciable amount of the AC component while the DC component continues its journey to the inductor L
2. The inductor L offers high reactance to the AC component but it offers almost zero reactance to the DC component. As a result the DC component flows through the inductor while the AC component is blocked

3. The capacitor C2 bypasses the AC component which the inductor had failed to block. As a result only the DC component appears across the load RL.

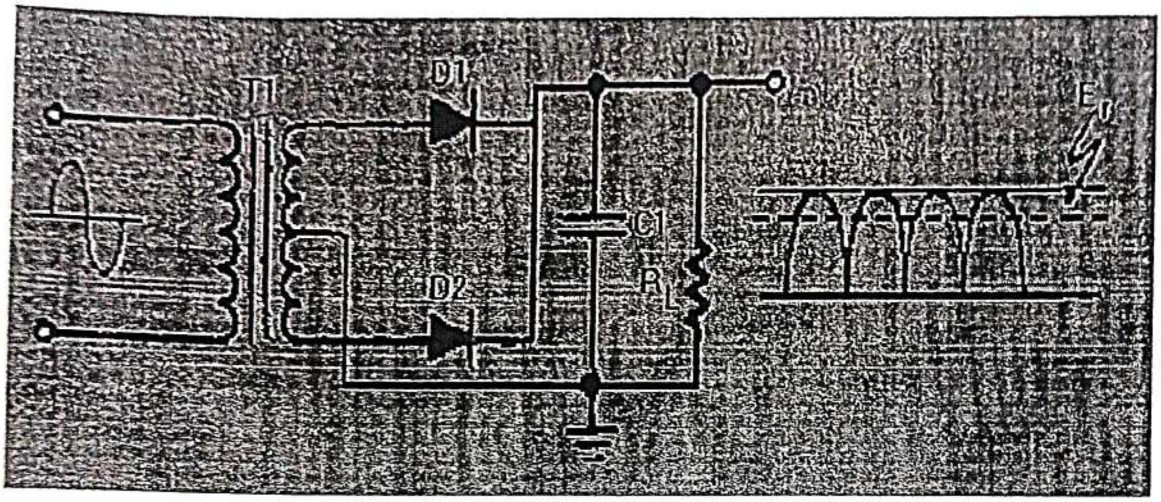


Fig :36: Center Tapped Full-Wave Rectifier With a Capacitor Filter

Voltage Regulator:

A **voltage regulator** is an electrical regulator designed to automatically maintain a constant voltage level. It may use an electromechanical mechanism, or passive or active electronic components. Depending on the design, it may be used to regulate one or more AC or DC voltages.

There are two types of regulators.

- Positive Voltage Series (78xx) and

➤ Negative Voltage Series (79xx)

78xx: '78' indicate the positive series and 'xx'indicates the voltage rating. Suppose 7805 produces the maximum 5V.'05'indicates the regulator output is 5V.

79xx:'78' indicate the negative series and 'xx'indicates the voltage rating. Suppose 7905 produces the maximum -5V.'05'indicates the regulator output is -5V.

These regulators consists the three pins there are

Pin1: It is used for input pin.

Pin2: This is ground pin for regulator

Pin3: It is used for output pin. Through this pin we get the output.

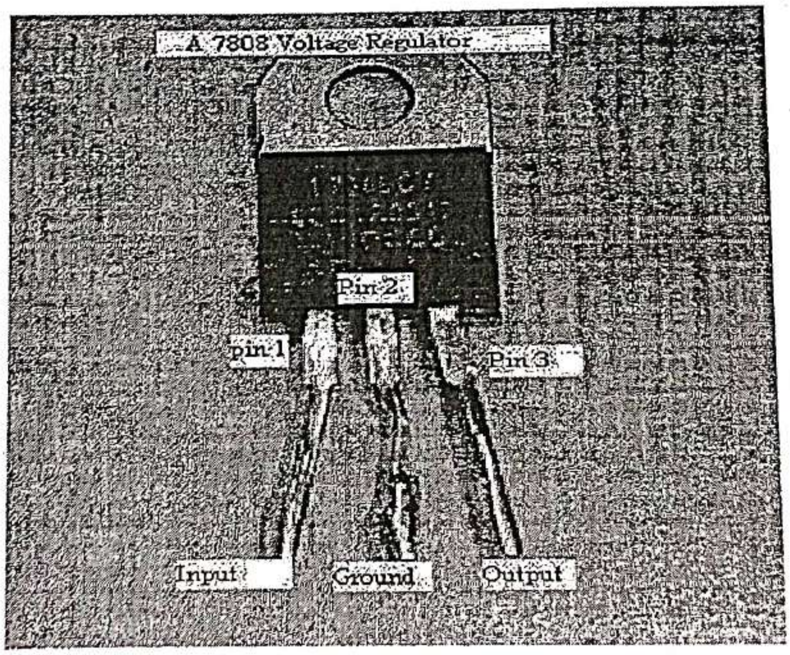


Fig :37: IC 7808 Voltage Regulator Pin Diagram

SOFTWARE EXPLANATION

ABOUT SOFTWARE:

Software's used are:

*Keil software for c programming

*Express PCB for lay out design

*Express SCH for schematic design

What's New in μVision3?

μVision3 adds many new features to the Editor like Text Templates, Quick Function Navigation, and Syntax Coloring with brace high lighting Configuration Wizard for dialog based startup and debugger setup. μVision3 is fully compatible to μVision2 and can be used in parallel with μVision2.

What is μVision3?

μVision3 is an IDE (Integrated Development Environment) that helps you write, compile, and debug embedded programs. It encapsulates the following components:

- A project manager.
- A make facility.
- Tool configuration.
- Editor.
- A powerful debugger.

To help you get started, several example programs (located in the **\C51\Examples, \C251\Examples, \C166\Examples,** and **\ARM\...\Examples)** are provided.

- **HELLO** is a simple program that prints the string "Hello World" using the Serial Interface.
- **MEASURE** is a data acquisition system for analog and digital systems.
- **TRAFFIC** is a traffic light controller with the RTX Tiny operating system.
- **SIEVE** is the SIEVE Benchmark.
- **DHRY** is the Dhrystone Benchmark.
- **WHETS** is the Single-Precision Whetstone Benchmark.

Additional example programs not listed here are provided for each device architecture.

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.

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.

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.

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 **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 database. Just select the micro controller 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

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.

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

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.

Start Debugging:

You start the debug mode of μVision2 with the Debug – Start/Stop Debug Session command. Depending on the Options for Target – Debug Configuration, μVision2 will load the application program and run the startup code μVision2 saves the editor screen layout and restores the screen layout of the last debug session.

If the program execution stops, μVision2 opens an editor window with the source text or shows CPU instructions in the disassembly window. The next executable statement is marked with a yellow arrow. During debugging, most editor features are still available.

For example, you can use the find command or correct program errors. Program source text of your application is shown in the same windows. The µVision2 debug mode differs from the edit mode in the following aspects:

_ The "Debug Menu and Debug Commands" described areAvailable. The additional debug windows are discussed in the following.

_ The project structure or tool parameters cannot be modified. All build

Commands are disabled.

Disassembly Window:

The Disassembly window shows your target program as mixed source and assembly program or just assembly code. A trace history of previously executed instructions may be displayed with Debug – View Trace Records. To enable the trace history, set Debug – Enable/Disable Trace Recording.

If you select the Disassembly Window as the active window all program step commands work on CPU instruction level rather than program source lines. You can select a text line and set or modify code breakpoints using toolbar buttons or the context menu commands.

You may use the dialog Debug – Inline Assembly... to modify the CPU instructions. That allows you to correct mistakes or to make temporary changes to the target program you are debugging.

B)Keil Software:

Installing the Keil software on a Windows PC

- Insert the CD-ROM in your computer's CD drive
- On most computers, the CD will "auto run", and you will see the Keil installation menu. If the menu does not appear, manually double click on the Setup icon, in the root directory: you will then see the Keil menu.
- On the Keil menu, please select "Install Evaluation Software". (You will not require a license number to install this software).
- Follow the installation instructions as they appear.

Loading the Projects:

The example projects for this book are NOT loaded automatically when you install the Keil compiler.
These files are stored on the CD in a directory "/Pont". The files are arranged by chapter: for example, the project discussed in Chapter 3 is in the directory "/Pont/Ch03_00-Hello".

Rather than using the projects on the CD (where changes cannot be saved), please copy the files from CD onto an appropriate directory on your hard disk.

Note: you will need to change the file properties after copying: file transferred from the CD will be 'read only'.

Configuring the Simulator

Open the Keil µVision2

go to Project – Open Project and browse for Hello in Ch03_00 in Pont and open it.

KEILSOFTWARETOOL(STEPS):

Click on the Keil uVision Icon on DeskTop

The following fig will appear

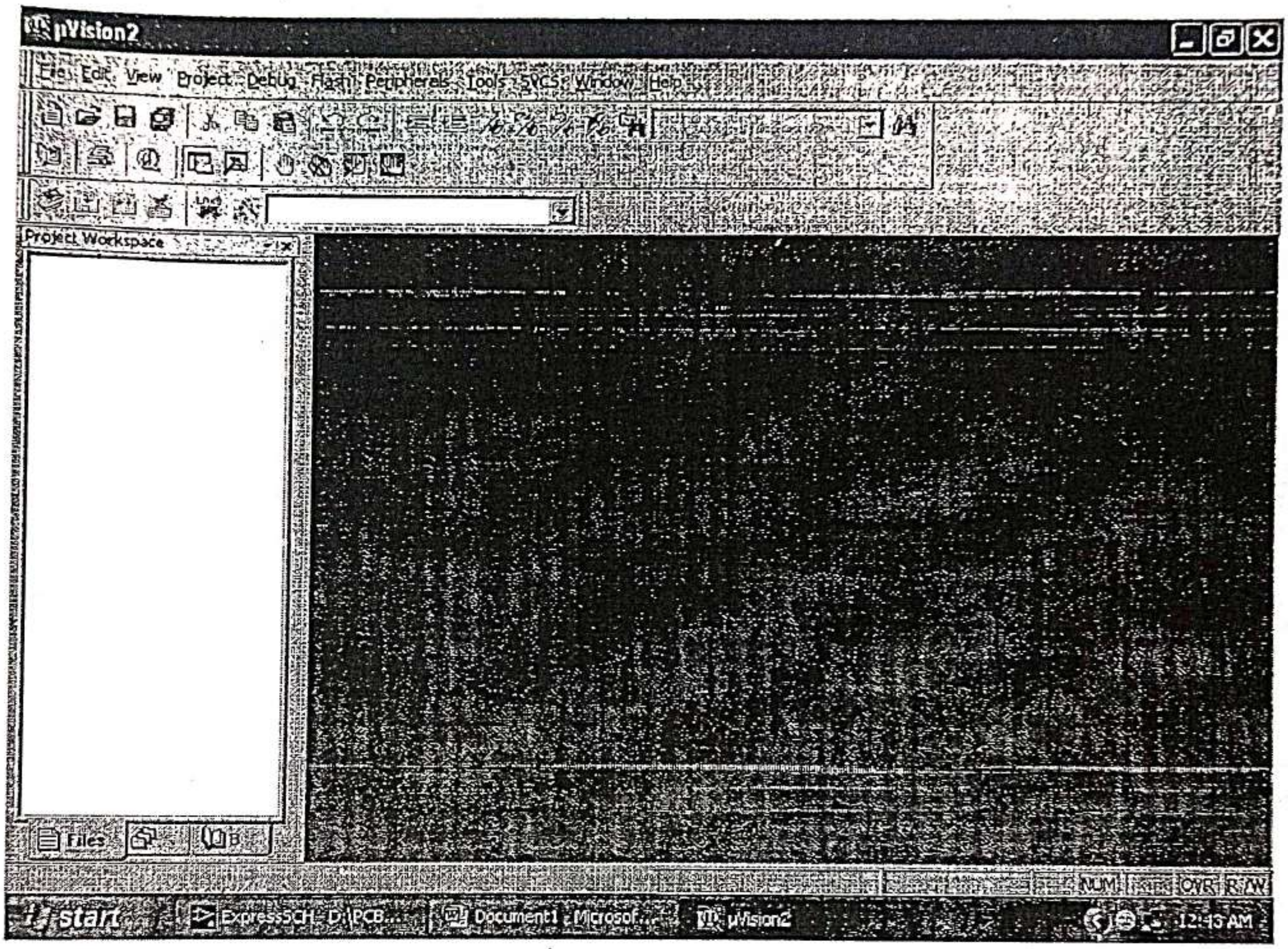


Fig :38: KEIL Software Starting Window

1. Click on the Project menu from the title bar
2. Then Click on New Project

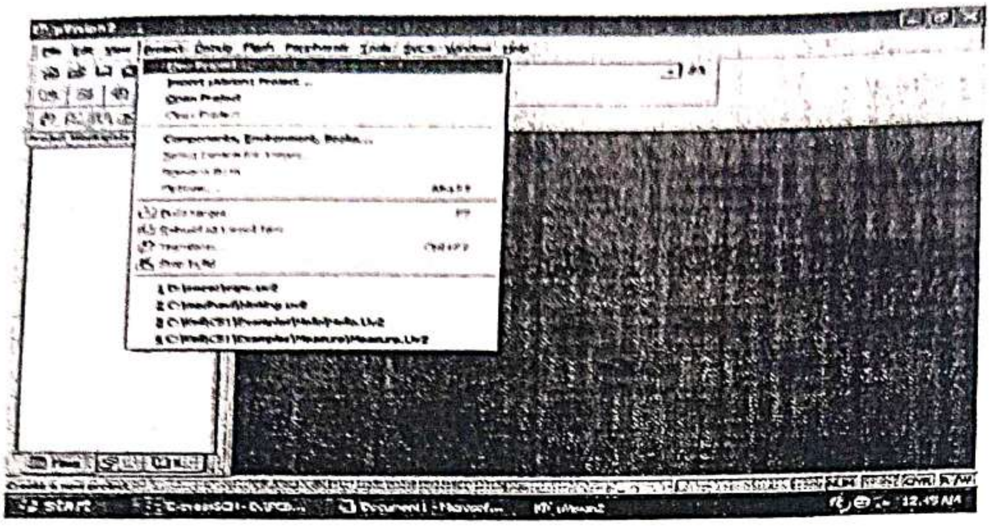


Fig :39: Create and Saving of File Using KEIL Software

3. Save the Project by typing suitable project name with no extension in u r own folder

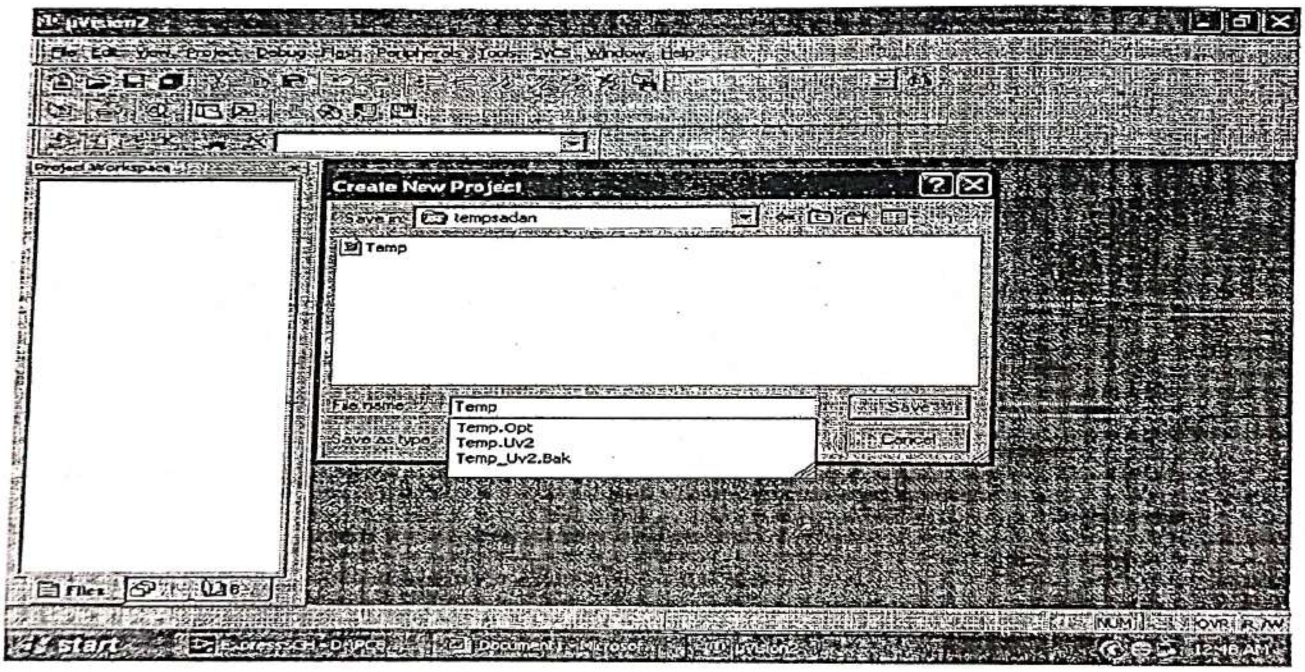


Fig :40: Creating of New Project File

4. Then Click on Save button above.

5. Select the component for u r project. i.e. Atmel……

6. Click on the + Symbol beside of Atmel

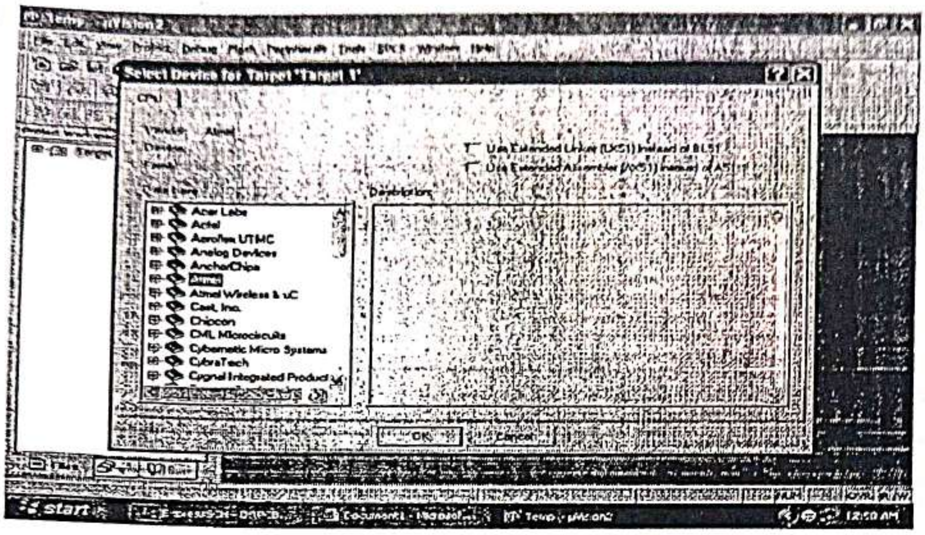


Fig :41: KEIL Software Window To Choose the Controller

7. Select AT89C52 as shown below

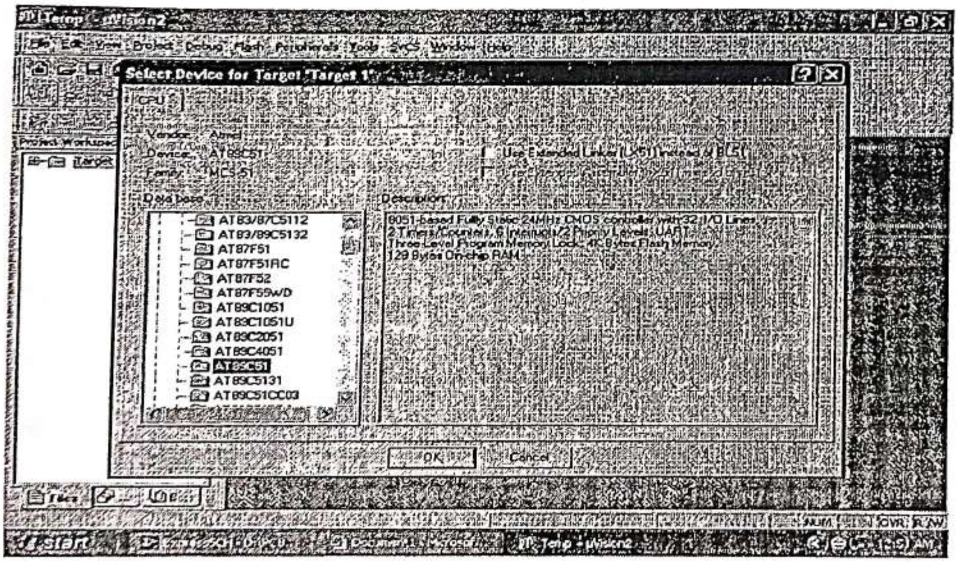


Fig :42: KEIL Software window

8. Then Click on "OK"

9. The Following fig will appear

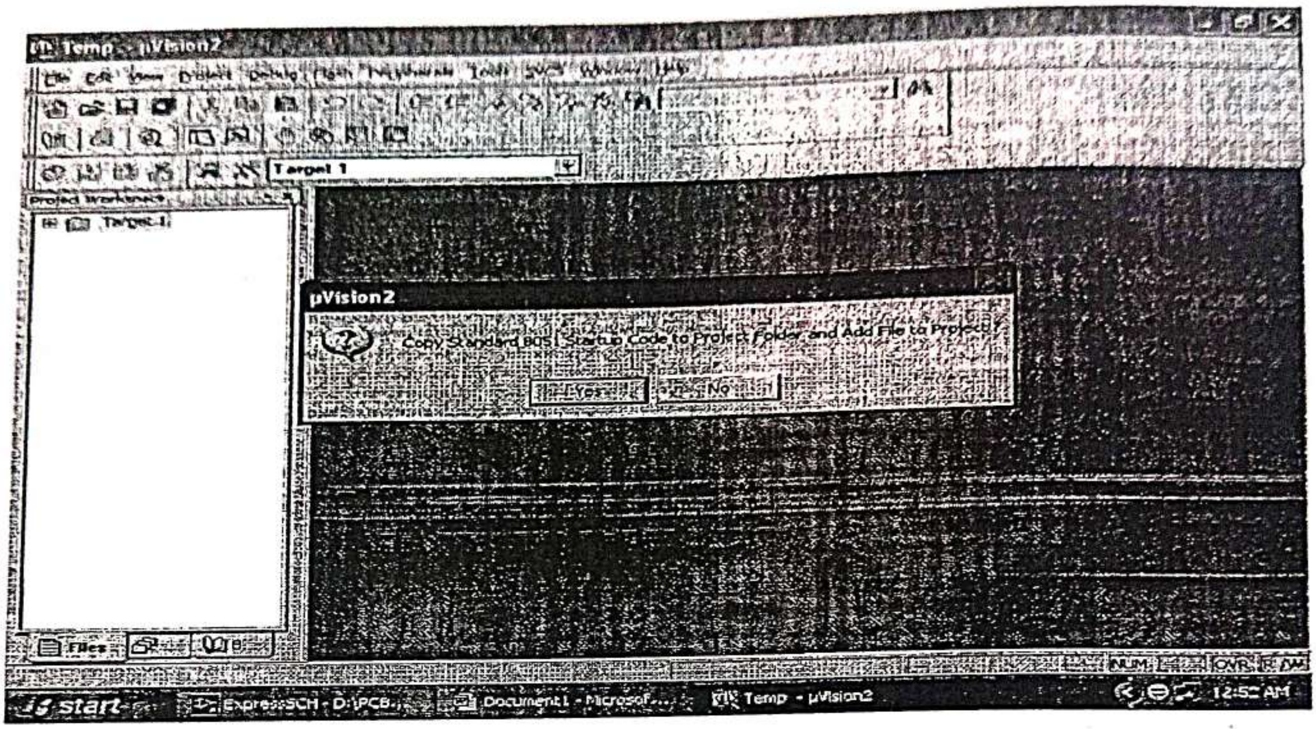


Fig :43: KEIL Software Window to Copy Standard 8051 Startup Code to Project Folder and add File to Project

10. Then Click either YES or NO.........mostly "NO"

11. Now your project is ready to USE

12. Now double click on the Target1, you would get another option "Source group 1" as shown in next page.

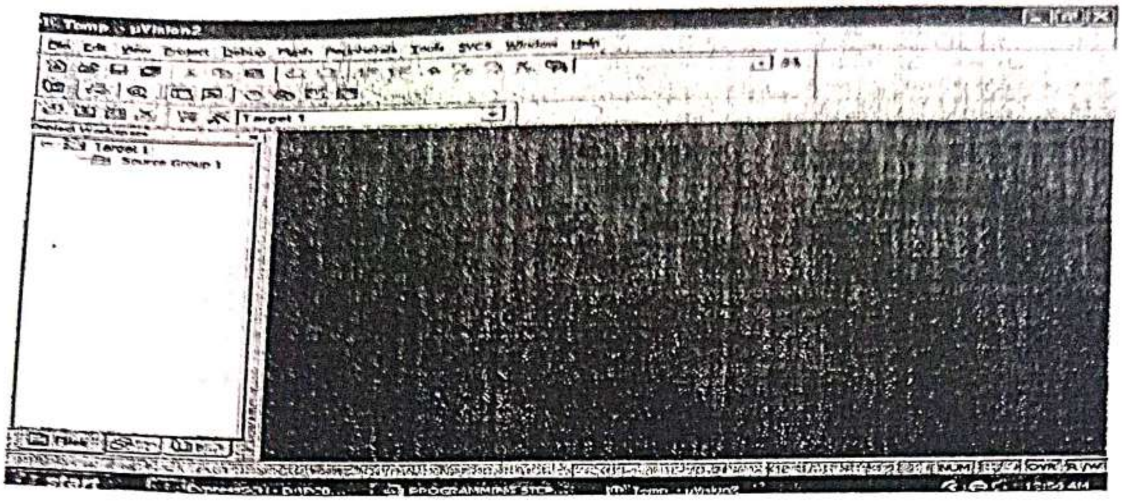


Fig :44: Steps to create Source Group 1 in KEIL Software

13. Click on the file option from menu bar and select "new"

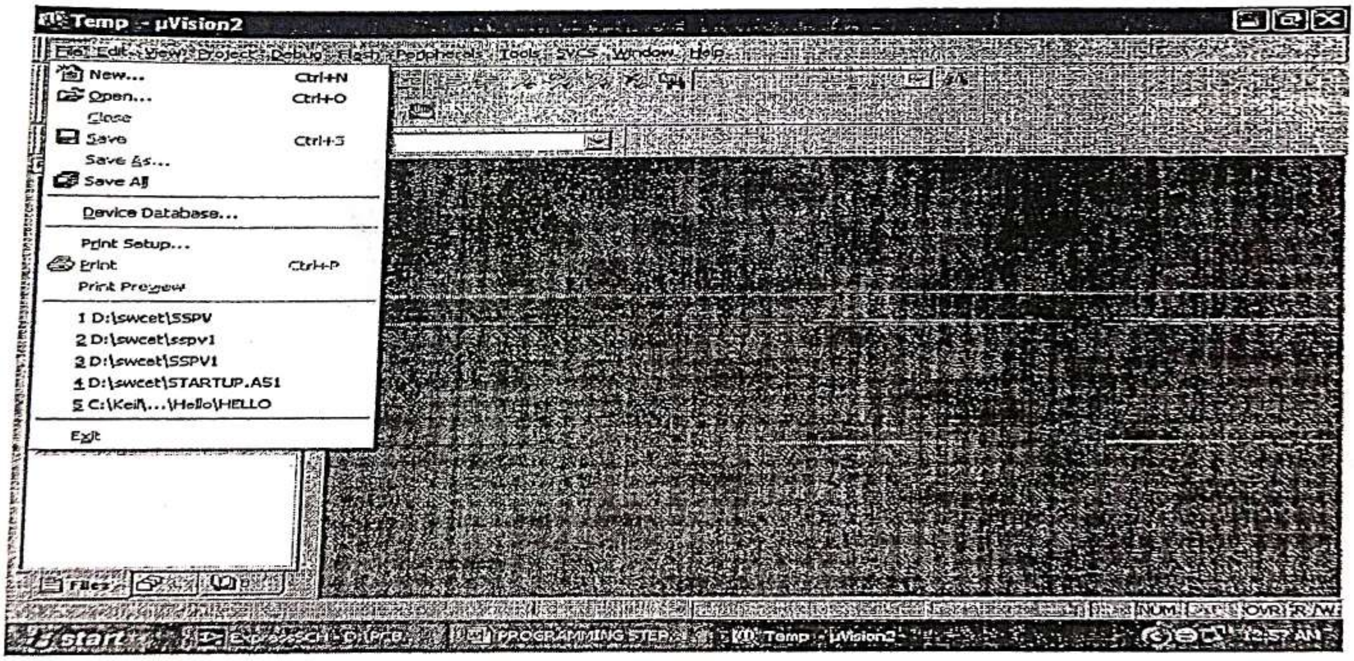


Fig :45: Steps to create New file in KEIL software

14. The next screen will be as shown in next page, and just maximize it by double clicking on its blue boarder.

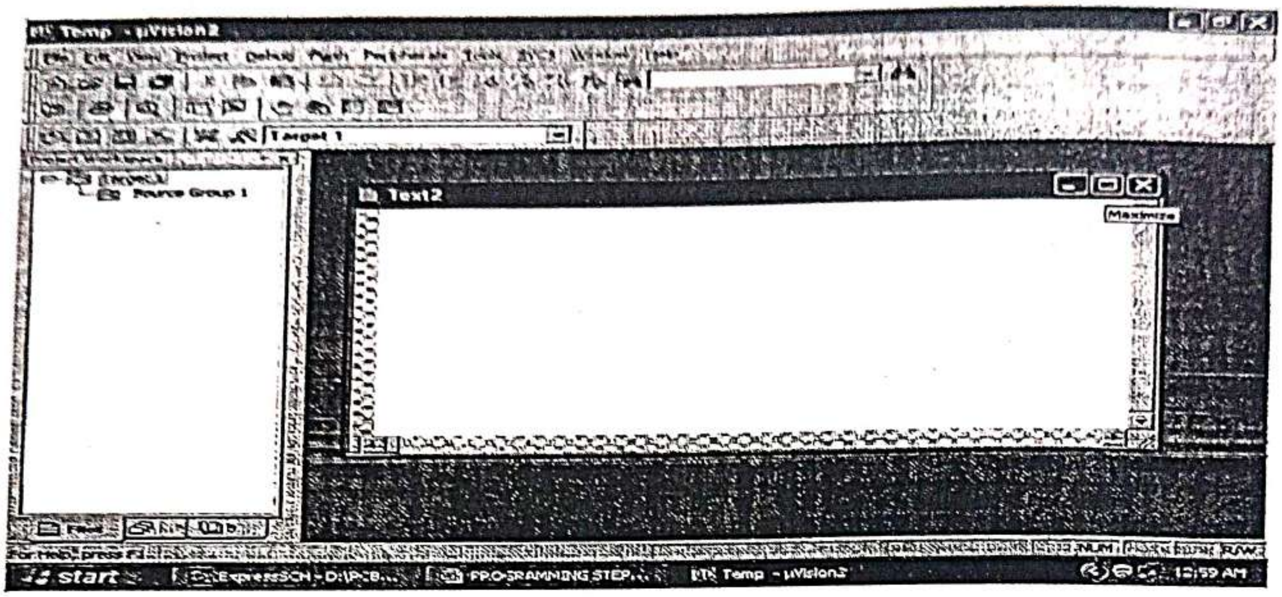


Fig :46: Steps to Open and write Program in KEIL software

15. Now start writing program in either in "C" or "ASM"

16. For a program written in Assembly, then save it with extension ". asm" and for "C" based program save it with extension " .C"

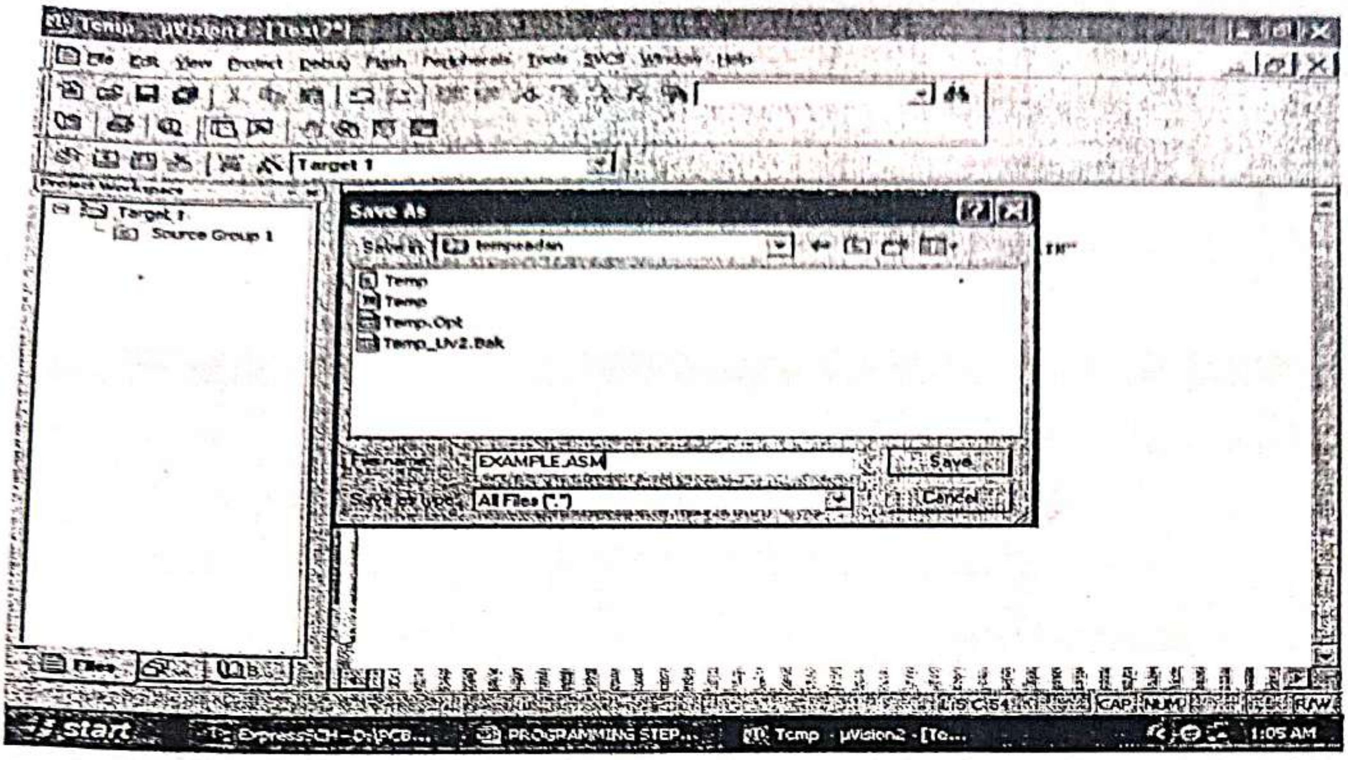


Fig :47: Steps to Save File in KEIL software

17. Now right click on Source group 1 and click on **"Add files to Group Source"**

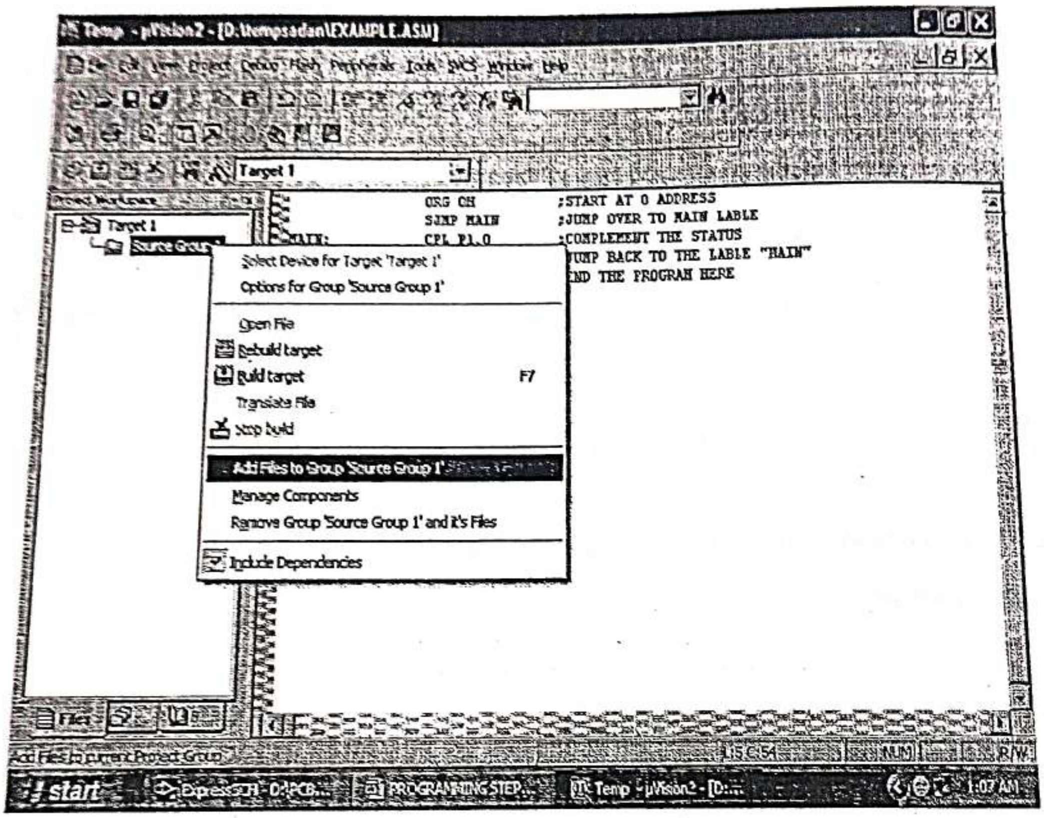


Fig :48: Steps to add files to group "source group 1" in KEIL Software

18. Now you will get another window, on which by default "C" files will appear.

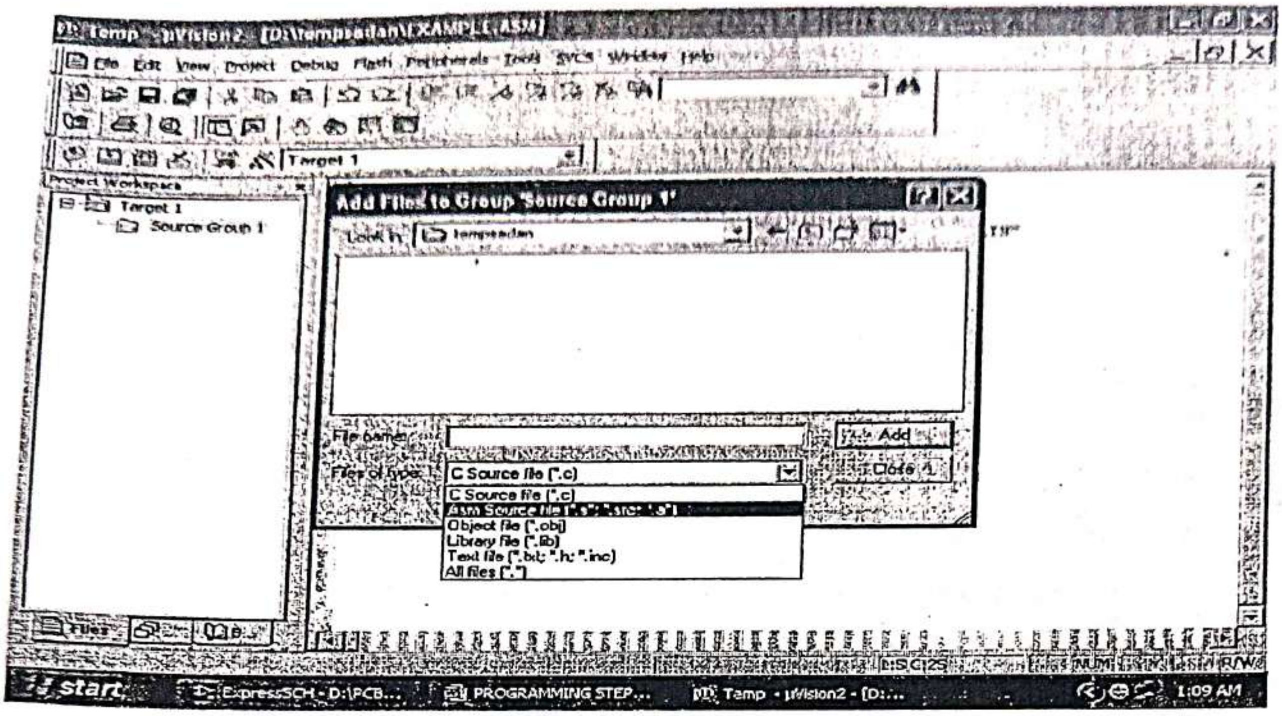


Fig :49: Steps to add files to group "source group 1" in KEIL Software

Now select as per your file extension given while saving the file

19. Click only one time on option "**ADD**"

20. Now Press function key F7 to compile. Any error will appear if so happen.

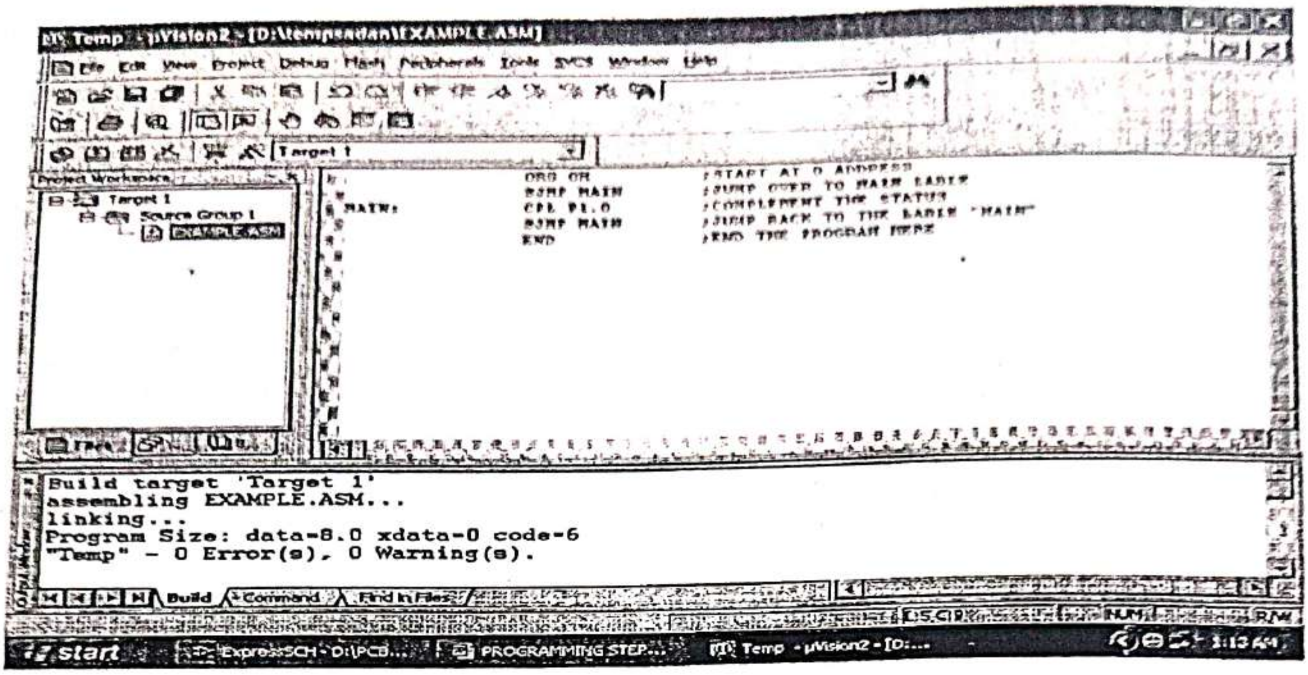


Fig :50: KEIL Software window to find errors

21. If the file contains no error, then press Control+F5 simultaneously.

22. The new window is as follows

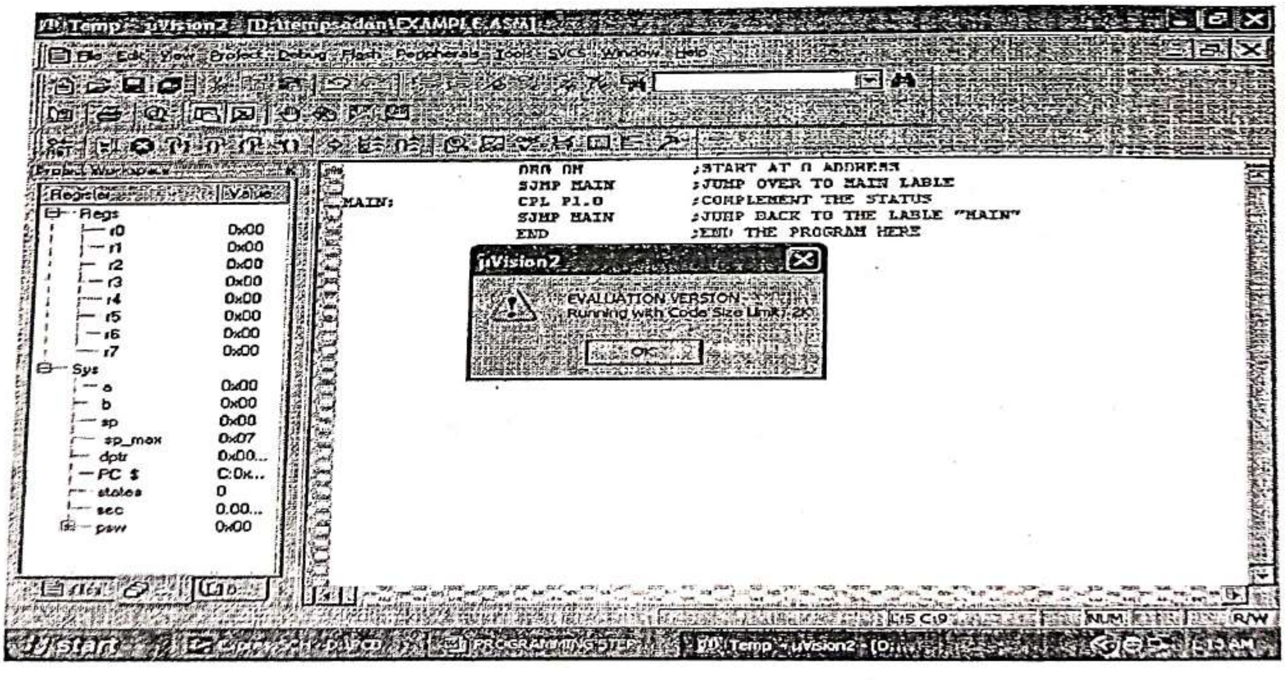


Fig :51: Steps to select Peripherals and Ports in KEIL Software

23. Then Click "OK"

24. Now Click on the Peripherals from menu bar, and check your required port as shown in fig below

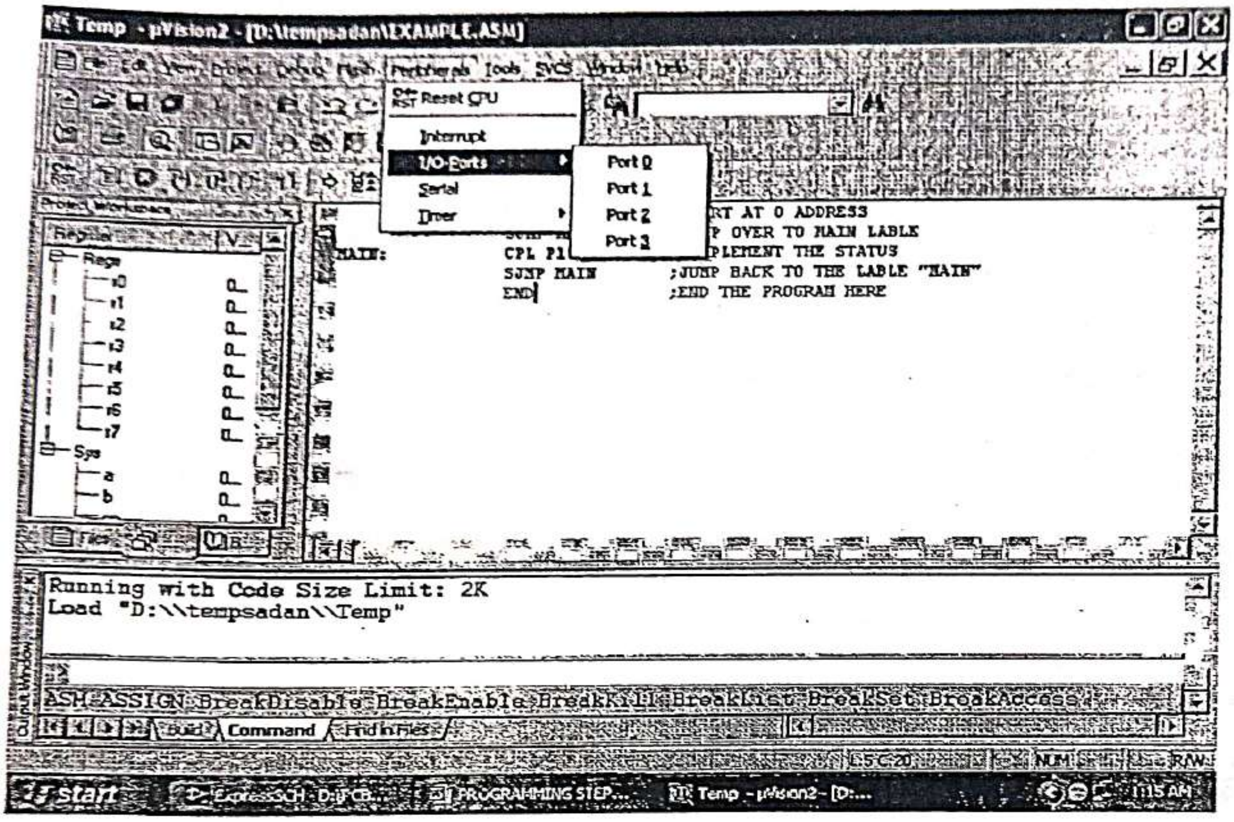


Fig :52: Steps to select Peripherals and Ports in KEIL Software

25. Drag the port a side and click in the program file.

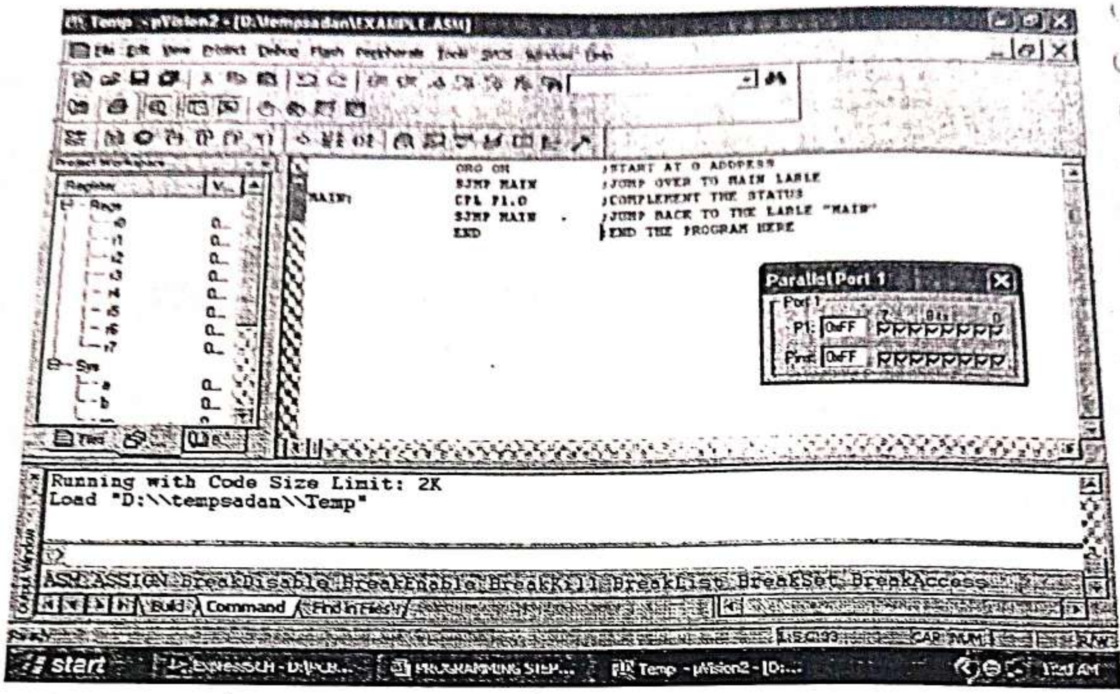


Fig :53: Steps to select Registers and Ports in KEIL Software

26. Now keep Pressing function key "F11" slowly and observe.
27. You are running your program successfully

PROGRAM CODE

FOR

AUTOMATIC FAULT DETECTION AND LOCATION IN POWER TRANSMISSION LINES

```
#include<reg52.h>
#include "lcd.h"
#include "uart.h"
#include "gsm.h"
sbit l1=P1^3;
sbit l2=P1^1;
sbit l3=P1^2;
int x=5;
int y=5;
int z=5;

void xx()
{
x=0;
}
void yy()
{
y=0;
```

```
}

void zz()
{
z=0;
}
void l1_sms()
{
tx("AT+CMGS=");
  tx1('"');
  tx(pastnumber1);
  tx1('"');
  tx("\r\n");
  do{
   rcv = receive();
   }while(rcv != '>');
  tx("POWER FAIL AT STREET NO:1\r\n");
   tx1(0x1A);
        okcheck();
        }
void l2_sms()
{
tx("AT+CMGS=");
```

```
   tx1("");
   tx(pastnumber1);
   tx1("");
   tx("\r\n");
   do{
    rcv = receive();
   }while(rcv != '>');
   tx("POWER FAIL AT STREET NO:2\r\n");
   tx1(0x1A);
        okcheck();
}
void l3_sms()
{
tx("AT+CMGS=");
  tx1("");
  tx(pastnumber1);
  tx1("");
  tx("\r\n");
  do{
   rcv = receive();
  }while(rcv != '>');
  tx("POWER FAIL AT STREET NO:3\r\n");
  tx1(0x1A);
```

```
        okcheck();
}
void main()
{
lcd_rw=0;

serinit();
lcdinit();
msgdisplay("GSM INITLIZING..");
delay(100);
gsm_init();
delay(100);
msgdisplay("GSM INIT COMPLT");
lcdcmd(1);
msgdisplay("POWR FAIL IDENTI");
lcdcmd(0xc0);
msgdisplay("SYS USING GSM");


while(1)
{
if(l1==0)
{
```

```
if(x==5)
{
l1_sms();
xx();

}
else
{
lcdcmd(1);
msgdisplay("POWER FAIL AT   STREET NO:1");
delay(200);
}
 }
else
{
x=5;
}

////////////////////////
 if(l2==0)
{
if(y==5)
{
```

```
l2_sms();
yy();
}
else
{
lcdcmd(1);
msgdisplay("POWER FAIL AT   STREET NO:2");
delay(200);
}
 }
else
{
y=5;
}
/////////////////////////////////////
 if(l3==0)
{
if(z==5)
{
l3_sms();
zz();
}
else
```

```
{
lcdcmd(1);
msgdisplay("POWER FAIL AT   STREET NO:3");
delay(200);
}
}
else
{
z=5;
}
if(l1==1 && l2==1 && l3==1)
{
lcdcmd(0x80);
msgdisplay("POWR FAIL IDENTI");
lcdcmd(0xc0);
msgdisplay("SYS USING GSM");
}
else
{
;;;
        }
}
}
```