P-3421

MICRO CONTROLLER BASED MILEAGE METER ENHANCED WITH DISTANCE CALCULATOR, TIME PREDICTOR AND HELMET SENSOR

A Project Report

Submitted by

J.ARCHANA	-	**0710105004**
K.ARVINDH RAMANNA	-	**0710105008**
P.PARTHIBAN	-	**0710105035**
O.P.T.PRADHEEP KUMAR	-	**0710105037**

in partial fulfillment for the award of the degree

of

Bachelor of Engineering

in

Electrical and Electronics Engineering

DEPARTMENT OF ELECTRICAL & ELECTRONICS ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY
COIMBATORE – 641 049

(An Autonomous Institution Affiliated to Anna University, Coimbatore.)

ANNA UNIVERSITY: COIMBATORE

APRIL 2011

ANNA UNIVERSITY: COIMBATORE

BONAFIDE CERTIFICATE

Certified that this project report entitled **"MICRO CONTROLLER BASED MILEAGE METER ENHANCED WITH DISTANCE CALCULATOR, TIME PREDICTOR AND HELMET SENSOR"** is the bonafide work of

J.Archana	Register No: 0710105004
K.Arvindh ramanna	Register No: 0710105008
P.Parthiban	Register No: 0710105035
O.P.T.Pradheep kumar	Register No: 0710105037

Who carried out the project work under my Supervision

Signature of the Head of the Department
DR. RANI THOTTUNGAL
HOD

Signature of the Supervisor
MR. J. RAM PRABU
ASST. PROF

Certified that the candidate with university Register No. 0710105004, 0710105008, 0710105035, 0710105037 was examined in project viva voce Examination held on 18.04.11

Internal Examiner

External Examiner

DEPARTMENT OF ELECTRICAL & ELECTRONICS ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY, COIMBATORE 641 006

(An Autonomous Institution Affiliated to Anna University, Coimbatore.)

DEDICATED

TO

THE ALMIGHTY

ABSTRACT

This project is done with a motive of sophisticating the travel by introducing gadgets such as mileage meter enhanced with the concepts of distance calculator, time predictor and helmet sensor. There are two microcontrollers employed to work on the real time data and give them suitably to computer. The software "Lab view" is used to compute and display the real time values from micro processor with a better visual and accuracy. Here we use motor as a model for vehicle and its variable speed using potentiometer is calculated by an opto-interrupter. The fuel tank setup is interpreted by means of a potentiometer, as the vehicles in market employ float– potentiometric sensors. An IR sensor is used to detect the heart beat and is fixed to a helmet. A wireless RF transmitter and receiver is used to transmit signals from heart beat sensors to the microprocessor in the vehicle setup. Based on the heart beat, processor ensures whether the person wears a helmet or not. When a person enters the distance that he has to travel, a new algorithm written in graphical coding using Labview gives details about distance yet to travel, approximate time of travel considering variation in speed and an indication about fuel getting low beyond pre-entered reserve value. Apart from this, from a pre-entered mileage, the distance that the vehicle can travel with the fuel in tank is also displayed. Thus, all these gadgets make the travel more certain.

ACKNOWLEDGEMENT

I humbly submit all the glory and thanks to the almighty for showering the blessings and giving the necessary wisdom for accomplishing this project.

I would like to express my deep sense of graduate and profound thanks to my guide **Mr.J.Ram Prabu,** Assistant Professor, Electrical and Electronics Engineering Department, for his valuable guidance, support, constant encouragement and co-operation rendered throughout the project.

I take immense pleasure in thanking **Dr.Rani Thottungal, HOD** Department of Electrical and Electronics Engineering, Kumaraguru College of Technology, for her constant encouragement and support.

I would like to express my heartful thanks to our beloved Principal **Dr.S.Ramachandran,** for his support.

I am also thankful to all my **teaching and technical supporting staffs** of Electrical and Electronics Engineering department, for their kind help and encouragement.

Finally, I thank my parents for giving me the moral support and abundant blessings in all of my activities and to my dear friends who made me endure my difficult times with their unfailing humor and warm wishes.

CONTENTS

3. ANALOG TO DIGITAL CONVERTER

4. MICROCONTROLLER

7. DECODER

8. HEART BEAT SENSOR

9. TRANSMITTER AND RECEIVER

10. DC MOTOR

11. HARDWARE DESCRIPTION

LIST OF TABLES

LIST OF FIGURES

ABREVIATIONS

IR	-	Infra Red Detector
Tx	-	Transmitter
Rx	-	Receiver
ADC	-	Analog to digital converter
DC	-	Direct current

CHAPTER 1

1. INTRODUCTION

1.1 Introduction to project

The automobile world is in thirst for new, reliable and accurate gadgets that make the travel certain. This project will help them to satisfy some of their needs in terms of sophistication. We here use the micro controller ATMEL 89S52 to do the basic calculations of real time and the complex calculations are taken care by LABVIEW software. Lab view supports the project by giving more accuracy, reliability and a better visual. Two micro controllers are used for two different kits. One micro controller takes care of the direct values obtained from the vehicle unit and the other for heartbeat circuit operation.

The Brushed DC shunt motor is used as a model for vehicle. The speed of the motor is varied using a potentiometer. The speed is detected using a OPTO-INTERRUPTER and the values are passed to MICROCONTROLLER I. For the fuel tank setup we use a potentiometer. The variable values from a potentiometer are given to ANALOG TO DIGITAL CONVERTER 0809. The digital values are fed to MICROCONTROLLER I.

The values transmitted from Heart beat sensor are received by a 433 MHz RECEIVER and decoded using HT12D DECODER. The decoded values are also given to MICROCONTROLLER I. The signal processed by micro controller gets weak, so we use a LEVEL CONVERTER to improve the quality of weak signal. The computer that requires real time values from Micro controller is interfaced using USB – SERIAL CONVERTER cable.

An IR TRANSMITTER – RECEIVER is used to track the variation in heartbeat. The varying pulses from IR circuit are fed into a heart beat sensor circuit. The result from heartbeat circuit is sent to MICRO CONTROLLER II. The micro controller processes the heartbeat signals and sends it to ENCODER - HT12E. the data is encoded and transmitted to vehicle unit by means of a 433MHz TRANSMITTER.

1.2 OBJECTIVES

- To make travelling more secured –

  HELMET SENSOR

- For more accurate data about the travel –

  MILEAGE METER AND

  DIGITAL SPEEDOMETER

- For sophistication -

  DISTANCE CALCULATOR & TIME PREDICTOR

1.3 ORGANIZATION OF THE REPORT

- Chapter 1 deals with the overview of the project and the objectives
- Chapter 2 deals with the power supply used
- Chapter 3 deals the Analog to Digital conversion with detailed explanation
- Chapter 4 deals with the Microcontroller and its overview
- Chapter 5 deals with the Opto-interrupter
- Chapter 6 deals with the Encoder and its operation
- Chapter 7 deals with the Decoder and its operation
- Chapter 8 deals with the Heartbeat sensor description and operation
- Chapter 9 deals with Transmitter and receiver
- Chapter 10 deals with the DC motor operation and selection
- Chapter 11 deals with the Hardware description, Block diagram and Lab View display
- Chapter 12 deals with the Algorithm and formulae used
- Chapter 13 Explains future scope and conclusion

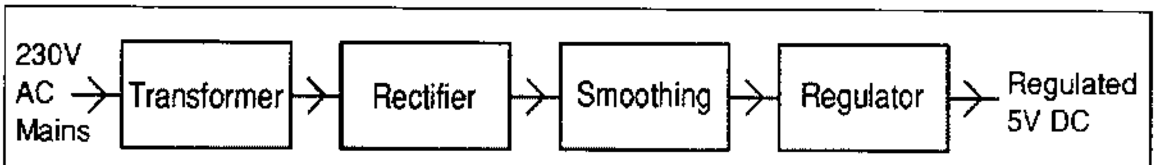
CHAPTER 2

2. POWER SUPPLY

2.1 INTRODUCTION

The present chapter introduces the operation of power supply circuits built using filters, rectifiers, and then voltage regulators. Starting with an ac voltage, a steady dc voltage is obtained by rectifying the ac voltage, then filtering to a dc level, and finally, regulating to obtain a desired fixed dc voltage. The regulation is usually obtained from an IC voltage regulator unit, which takes a dc voltage and provides a somewhat lower dc voltage, which remains the same even if the input dc voltage varies, or the output load connected to the dc voltage changes.

There are many types of power supply. Most are designed to convert high voltage AC mains electricity to a suitable low voltage supply for electronic circuits and other devices. A power supply can by broken down into a series of blocks, each of which performs a particular function.



Block Diagram of a Regulated Power Supply System

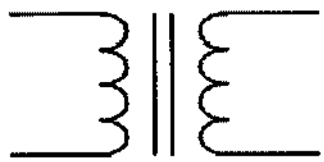
Each of the blocks is described in more detail below:

- Transformer - steps down high voltage AC mains to low voltage AC.
- Rectifier - converts AC to DC, but the DC output is varying.
- Smoothing - smooths the DC from varying greatly to a small ripple.
- Regulator - eliminates ripple by setting DC output to a fixed voltage.

Power supplies made from these blocks are described below with a circuit diagram and a graph of their output:

- Transformer only
- Transformer + Rectifier
- Transformer + Rectifier + Smoothing
- Transformer + Rectifier + Smoothing + Regulator

2.2 Transformer



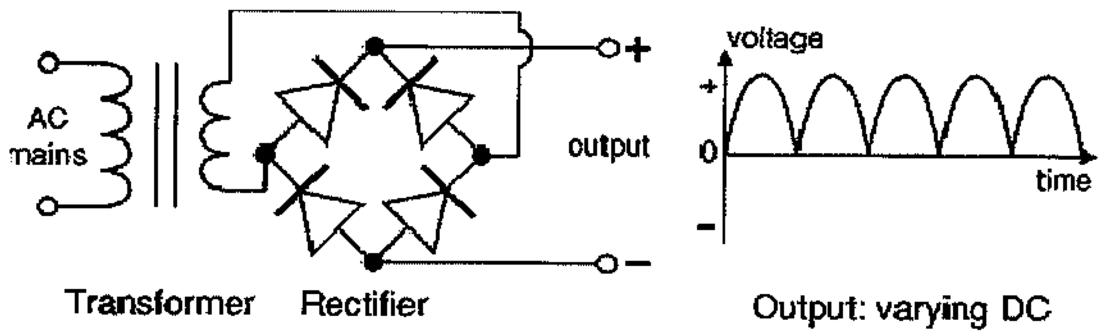
Transformers convert AC electricity from one voltage to another with little loss of power. Transformers work only with AC and this is one of the reasons why mains electricity is AC.

Step-up transformers increase voltage, step-down transformers reduce voltage. Most power supplies use a step-down transformer to reduce the dangerously high mains voltage (230V in UK) to a safer low voltage.

The input coil is called the primary and the output coil is called the secondary. There is no electrical connection between the two coils; instead they are linked by an alternating magnetic field created in the soft-iron core of the transformer. The two lines in the middle of the circuit symbol represent the core.

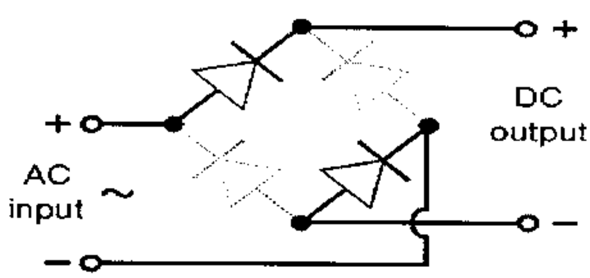
2.3 Rectifier

There are several ways of connecting diodes to make a rectifier to convert AC to DC. The bridge rectifier is the most important and it produces full-wave varying DC. A full-wave rectifier can also be made from just two diodes if a centre-tap transformer is used, but this method is rarely used now that diodes are cheaper. A single diode can be used as a rectifier but it only uses the positive (+) parts of the AC wave to produce half-wave varying DC.



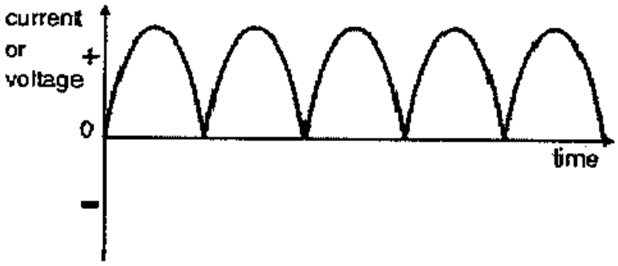
2.3.1 Bridge rectifier

A bridge rectifier can be made using four individual diodes, but it is also available in special packages containing the four diodes required. It is called a full-wave rectifier because it uses the entire AC wave (both positive and negative sections). 1.4V is used up in the bridge rectifier because each diode uses 0.7V when conducting and there are always two diodes conducting, as shown in the diagram below. Bridge rectifiers are rated by the maximum current they can pass and the maximum reverse voltage they can withstand (this must be at least three times the supply RMS voltage so the rectifier can withstand the peak voltages). Please see the Diodes page for more details, including pictures of ridge rectifiers.



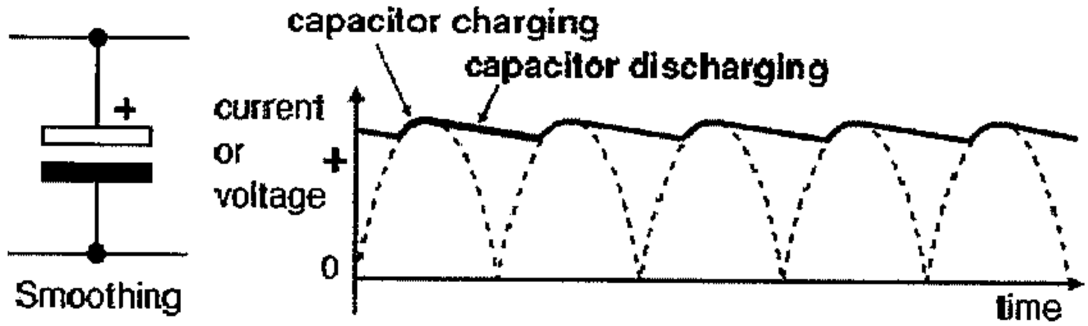
Alternate pairs of diodes conduct, changing over the connections so the alternating directions of AC are converted to the one direction of DC.

Output: full-wave varying DC: (using the entire AC wave):



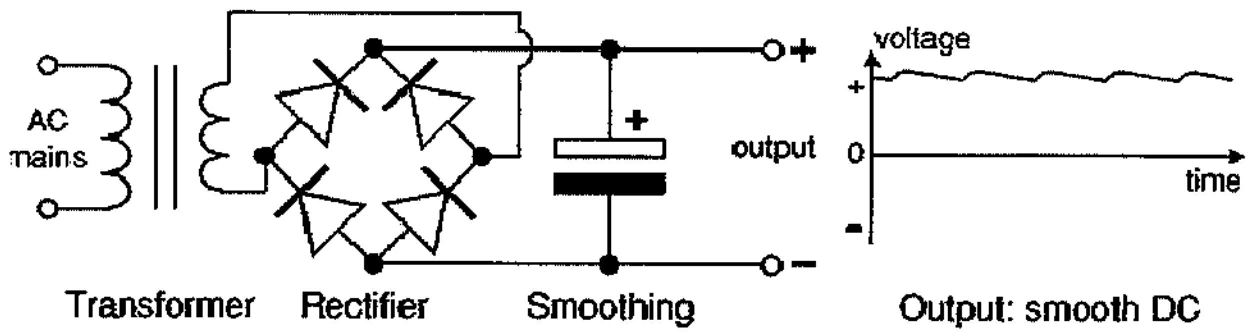
2.4 Smoothing

Smoothing is performed by a large value electrolytic capacitor connected across the DC supply to act as a reservoir, supplying current to the output when the varying DC voltage from the rectifier is falling. The diagram shows the unsmoothed varying DC (dotted line) and the smoothed DC (solid line). The capacitor charges quickly near the peak of the varying DC, and then discharges as it supplies current to the output.



Note that smoothing significantly increases the average DC voltage to almost the peak value (1.4 × RMS value). For example 6V RMS AC is rectified to full wave DC of about 4.6V RMS (1.4V is lost in the bridge rectifier), with smoothing this increases to almost the peak value giving 1.4 × 4.6 = 6.4V smooth DC.

Smoothing is not perfect due to the capacitor voltage falling a little as it discharges, giving a small ripple voltage. For many circuits a ripple which is 10% of the supply voltage is satisfactory. A larger capacitor will give fewer ripples. The capacitor value must be doubled when smoothing half-wave DC.



The smooth DC output has a small ripple. It is suitable for most electronic circuits.

2.5 Regulator

Voltage regulator ICs are available with fixed (typically 5, 12 and 15V) or variable output voltages. They are also rated by the maximum current they can pass. Negative voltage regulators are available, mainly for use in dual supplies. Most regulators include some automatic protection from excessive current ('overload protection') and overheating ('thermal protection').

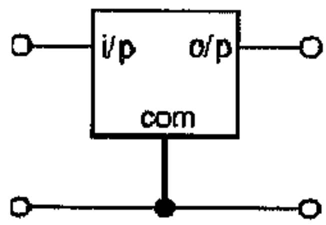
The LM78XX series of three terminal regulators is available with several fixed output voltages making them useful in a wide range of applications. One of these is local on card regulation, eliminating the distribution problems associated with single point regulation. The voltages available allow these regulators to be used in logic systems, instrumentation, HiFi, and other solid state electronic equipment. Although designed primarily as fixed voltage regulators these devices can be used with external components to obtain adjustable voltages and current.

Many of the fixed voltage regulator ICs has 3 leads and look like power transistors, such as the 7805 +5V 1A regulator shown on the right. They include a hole for attaching a heat sink if necessary.

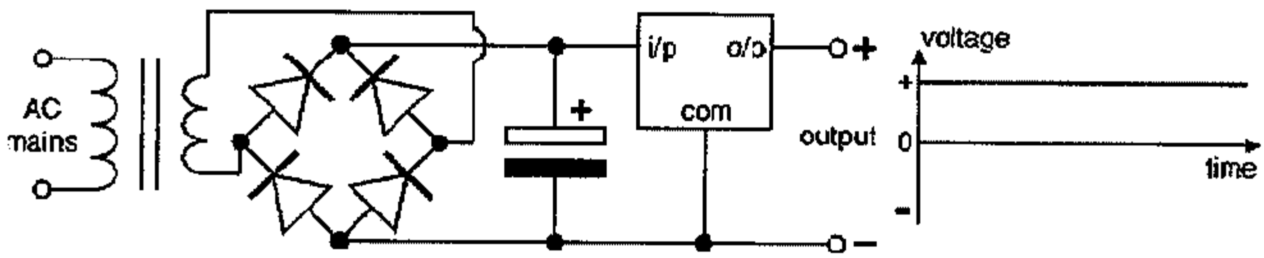
Positive regulator

1. input pin
2. ground pin
3. output pin

It regulates the positive voltage



Regulator



The regulated DC output is very smooth with no ripple. It is suitable for all electronic circuits.

CHAPTER 3

3. ANALOG TO DIGITAL CONVERTER

3.1 Introduction

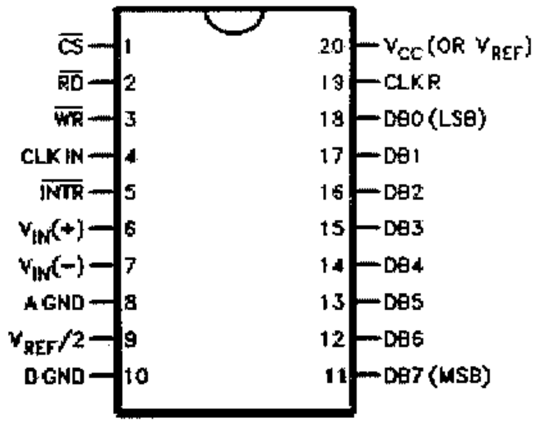
A/D Converter or Analog-to-digital converter is a device that converts data from analog to digital form. For example; an audio CD is made by converting analog sound signals into digital data. Typically, an ADC is an electronic device that converts an input analog voltage (or current) to a digital number. The digital output may be using different coding schemes, such as binary and two's complement binary. However, some non-electronic or only partially electronic devices, such as rotary encoders, can also be considered ADCs.

Analog-to-digital conversion is an electronic process in which a continuously variable (analog) signal is changed, without altering its essential content, into a multi-level (digital) signal. The input to an analog-to-digital converter (ADC) consists of a voltage that varies among a theoretically infinite number of values. Examples are sine waves, the waveforms representing human speech, and the signals from a conventional television camera.

The output of the ADC, in contrast, has defined levels or states. The number of states is almost always a power of two -- that is, 2, 4, 8, 16, etc. The simplest digital signals have only two states, and are called binary. All whole numbers can be represented in binary form as strings of ones and zeros.

Digital signals propagate more efficiently than analog signals, largely because digital impulses, which are well-defined and orderly, are easier for electronic circuits to distinguish from noise, which is chaotic. This is the chief advantage of digital modes in communications. Computers "talk" and "think" in terms of binary digital data; while a microprocessor can analyze analog data, it must be converted into digital form for the computer to make sense of it.

3.2 Pin Diagram of ADC 0804



3.3 Working

Steps to start the conversion

1. Make chip select (CS) signal low.
2. Make write (WR) signal low.
3. Make chip select (CS) high.
4. Wait for INTR pin to go low (means conversion ends).

Once the conversion in ADC is done, the data is available in the output latch of the ADC. Data of the new conversion is only available for reading after ADC0804 made INTR pin low or say when the conversion is over. Below are the steps to read output from the ADC0804.

1. Make chip select (CS) pin low.
2. Make read (RD) signal low.
3. Read the data from port where ADC is connected.
4. Make read (RD) signal high.
5. Make chip select (CS) high.

3.3.1 Operation and application of 0804

1. Compatible with 8080 µP derivatives-no interfacing logic needed - access time - 135 ns
2. Easy interface to all microprocessors, or operates "stand alone"
3. Differential analog voltage inputs
4. Logic inputs and outputs meet both MOS and TTL voltage level specifications
5. Works with 2.5V (LM336) voltage reference
6. On-chip clock generator
7. 0V to 5V analog input voltage range with single 5V supply
8. No zero adjust required
9. 20-pin molded chip carrier or small outline package
10. Operates ratiometrically or with 5 V_{DC}, 2.5 V_{DC}, or analog span adjusted voltage reference

3.4 Specification of 0804

1.	Supply Voltage	6.5V
2.	Voltage at any input	-0.3V to +0.3V
3.	Temperature Range	0C to 70C
4.	Conversion Time	<100us
5.	Resolution	$\pm$1LSB

CHAPTER 4

4. MICROCONTROLLER

4.1 Introduction

The microcontroller 89S52 is manufactured by Atmel. This is 8051 based Full Static CMOS controller with Three-Level Program Memory Lock, 32 I/O lines, 3 Timers/Counters, 8-Interrupts Sources, DPTRs, 8K Flash Memory, 256 Bytes On-chip RAM.

4.2 Features

1. Extensive Boolean processing (Single - bit Logic) Capabilities.
2. 8 Bit CPU optimized for control applications.
3. On - Chip Flash Program Memory.
4. On - Chip Data RAM.
5. Bi-directional and Individually Addressable I/O Lines.
6. Multiple 16-Bit Timer/Counters.

4.3 Pin configurations

4.3.1. Pin details

Pins 1-8: Port 1 Each of these pins can be configured as input or output.

Pin 9: RS Logical one on this pin stops microcontroller's operating and erases the contents of most registers. logical zero to this pin, the program starts execution from the beginning.

Pins 10-17: Port 3 Similar to port 1, each of these pins can serve as universal input or output. Besides, all of them have alternative functions:

Pin 10: RXD Serial asynchronous communication input or Serial synchronous communication output.

Pin 11: TXD Serial asynchronous communication output or Serial synchronous communication clock output.

Pin 12: INT0 Interrupt 0 input

Pin 13: INT1 Interrupt 1 input

Pin 14: T0 Counter 0 clock input

Pin 15: T1 Counter 1 clock input

Pin 16: WR Signal for writing to external (additional) RAM

Pin 17: RD Signal for reading from external RAM

Pin 18, 19: X2, X1 Internal oscillator input and output. A quartz crystal which determines operating frequency is usually connected to these pins. Instead of quartz crystal, the miniature ceramics resonators can be also used for frequency stabilization. Later versions of the microcontrollers operate at a frequency of 0 Hz up to over 50 Hz.

Pin 20: GND Ground

Pin 21-28: Port 2 If there is no intention to use external memory then these port pins are configured as universal inputs/outputs. In case external memory is used then the higher address byte, i.e. addresses A8-A15 will appear on this port. It is important to know that even memory with capacity of 64Kb is not used (i.e. note all bits on port are used for memory addressing) the rest of bits are not available as inputs or outputs.

Pin 29: PSEN If external ROM is used for storing program then it has a logic-0 value every time the microcontroller reads a byte from memory.

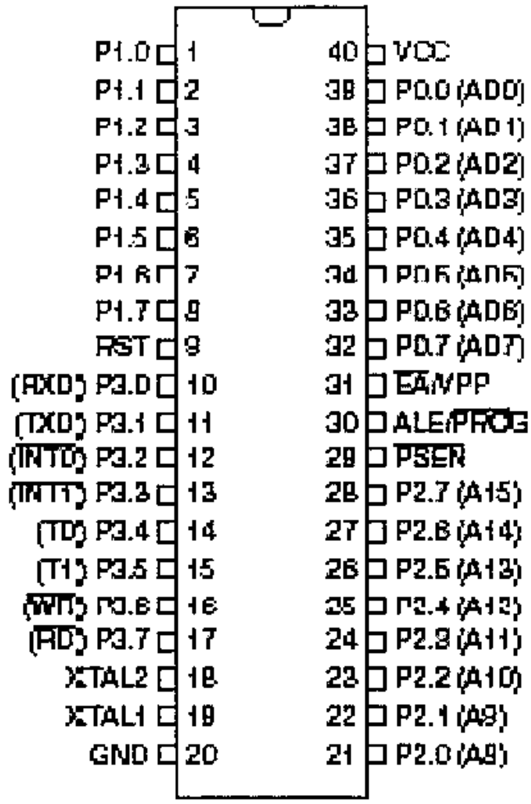
Pin 30: ALE Prior to each reading from external memory, the microcontroller will set the lower address byte (A0-A7) on P0 and immediately after that activates the output ALE. Upon receiving signal from the ALE pin, the external register (74HCT373 or 74HCT375 circuit is usually embedded) memorizes the state of P0 and uses it as an address for memory chip. In the second part of the microcontroller's machine cycle, a signal on this pin stops being emitted and P0 is used now for data transmission (Data Bus). In this way, by means of only one additional (and cheap) integrated circuit, data multiplexing from the port is performed. This port at the same time used for data and address transmission.

Pin 31: EA By applying logic zero to this pin, P2 and P3 are used for data and address transmission with no regard to whether there is internal memory or not. That means that even there is a program written to the microcontroller, it will not be executed, the program written to external ROM will be used instead. Otherwise, by applying logic one to the EA pin, the microcontroller will use both memories, first internal and afterwards external (if it exists), up to end of address space.

Pin 32-39: Port 0 Similar to port 2, if external memory is not used, these pins can be used as universal inputs or outputs. Otherwise, P0 is configured as address output (A0-A7) when the ALE pin is at high level (1) and as data output (Data Bus), when logic zero (0) is applied to the ALE pin.

Pin 40: VCC Power supply +5V.

4.3.2. Pin diagram



4.4. Functional operation

4.4.1. Input/ Output Ports

There are four I/O ports available in AT89S52. They are port 0, port 1, port 2, and port 3. All these ports are eight bit ports. All these ports can be controlled as eight-bit port or it can be controlled individually. One of the main feature of this micro controller is it can control the port pins individually. In 89S52 port 1 is available for user, Port 3 is combined with interrupts. This can be used as interrupts (or) I/O ports, ports 2 & port 0 is combined with address bus & data bus.

All these port lines are available with internal pull-ups except port 0. If we want to use port 0 as I/O port we have to use pull up resistors. This Micro controller is working at a maximum speed of 24MHz.

4.4.2. Memory Organization

All Atmel Flash micro controllers have separate address spaces for program and data memory. The logical separation of program and data memory allows the data memory to be accessed by 8 bit addresses. Program memory can only be read. There can be up to 64K bytes of directly addressable program memory. The read strobe for external program memory is the Program Store Enable Signal (PSEN). Data memory occupies a separate address space from program memory. Up to 64K bytes of external memory can be directly addressed in the external data memory space.The CPU generates read and write signals, RD and WR, during external data memory accesses. External program memory and external data memory can be combined by applying the RD and PSEN signal to the inputs of AND gate and using the output of the fate as the read strobe to the external program/data memory.

4.4.3. Program memory

After reset, the CPU begins execution from location 0000h. Each interrupt is assigned a fixed location in program memory. The interrupt causes the CPU to jump to that location, where it executes the service routine. If external Interrupt 0 is used, its service routine must begin at location 0003h. If the interrupt is not used its service location is available as general-purpose program memory. The interrupt service locations are spaced at 8 byte intervals 0003h for External interrupt 0, 000Bh for Timer 0, 0013h for External interrupt 1,001Bh for Timer1, and so on. If an Interrupt service routine is short enough it can reside entirely within that 8-byte interval. Longer service routines can use a jump instruction to skip over subsequent interrupt locations. The lowest addresses of program memory can be either in the on-chip Flash or in an external memory. To make this selection, strap the External Access (EA) pin to either Vcc or GND.

4.4.4. Data Memory

The Internal Data memory is divided into three blocks namely,

- The lower 128 Bytes of Internal RAM.
- The Upper 128 Bytes of Internal RAM.
- Special Function Register.

Internal Data memory Addresses are always 1 byte wide, which implies an address space of only 256 bytes. Direct addresses higher than 7Fh access one memory space and indirect addresses higher than 7Fh access a different Memory Space. The lowest 32 bytes are grouped into 4 banks of 8 registers. Program instructions call out these registers as R0 through R7. Two bits in the Program Status Word (PSW) select which register bank is in use. This architecture allows more efficient use of code space, since register instructions are shorter than instructions that use direct addressing.

The next 16-bytes above the register banks form a block of bit addressable memory space. The micro controller instruction set includes a wide selection of single - bit instructions and this instruction can directly address the 128 bytes in this area. These bit addresses are 00h through 7Fh The Special Function Register includes Port latches, timers, peripheral controls etc., direct addressing can only access these register. In general, all Atmel micro controllers have the same SFRs at the same addresses.Sixteen addresses in SFR space are both byte and bit Addressable. The bit Addressable SFRs are those whose address ends in 000B. The bit addresses in this area are 80h through FFh.

4.4.5. Oscillator And Clock Circuit

XTAL1 and XTAL2 are the input and output respectively of an inverting amplifier which is used as a crystal oscillator in the frequency range of 1.2 Mhz to 12 Mhz. XTAL2 is also used as the input to the internal clock generator. To drive the chip with an internal oscillator XTAL1 and XTAL2 are grounded. Since the input to the clock generator is divided by two flip flop there are no requirements on the duty cycle of the external oscillator signal. However, minimum high and low times must be observed.The clock generator divides the oscillator frequency by 2 and provides a two phase clock signal to the chip. The phase 1 signal is active during the first half of each clock period and the phase 2 signals are active during the second half of each clock period.

4.4.6. CPU Timing

A machine cycle consists of 6 states. Each state is divided into a phase /half, during which the phase 1 clock is active and phase 2 half. Arithmetic and Logical operations take place during phase1 and internal register to register transfer take place during phase 2.

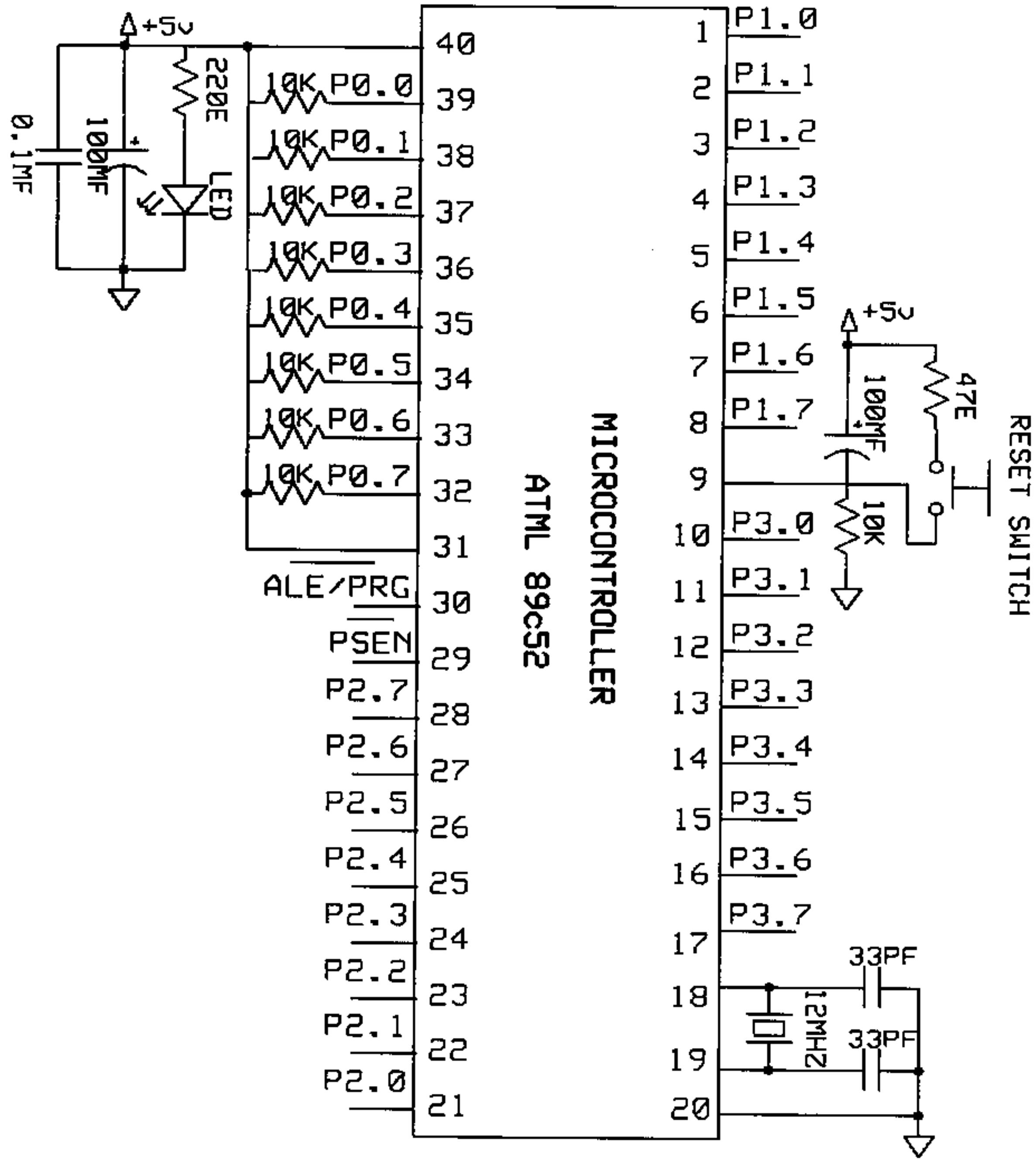
4.4.7. Flash Rom

8-kilo byte ROM is available in the Micro controller. It can be erased and reprogrammed. If the available memory is not enough for program, it can be interfaced with external ROM .It has 16 address lines, so maximum of (2^16) i.e.64 bytes of ROM can be interfaced with this Micro controller. Both internal and external ROM cannot be used simultaneously. For external accessing of ROM, a pin is provided in Micro controller itself i.e. pin no.31 EA. It should be high to use internal ROM, low to use external ROM

4.4.8. RAM

Internal 256 bytes of RAM are available for user. These 256 bytes of RAM can be used along with the external RAM. Externally 64-kilo bytes of RAM can be connected with micro controller. In internal RAM first 128 bytes of RAM is used for direct and indirect access and the remaining 128 bytes are used as only indirect access . The 128 bytes are used as special function registers (SFR) only direct access. These SFR's are used as control registers for timer, serial port etc.

4.4.9. Microcontroller Circuit



CHAPTER 5

5. OPTO INTERRUPTER

5.1 Introduction

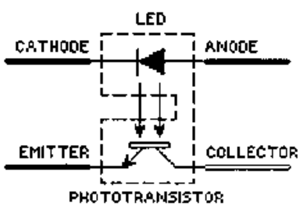
The opto-interrupter is an electronic device that consists of a light emitting diode (LED) and a phototransistor with a slot between them.

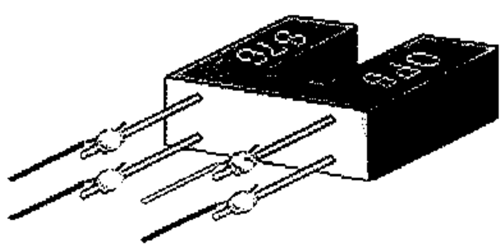
When voltage is applied to the LED it emits light like an electric bulb. However, the LED used in an opto-interrupter emits an infrared light beam which is invisible. Light emitting diodes are very reliable and consume a relatively small current. Big current may destroy them, therefore a resistor must be added to limit the current. Opto-interrupters supplied with the kit (or sold as a separate part) come with a 270 Ohm resistor which is suitable for voltages in the range from 3 to 6V.Phototransistors are specially designed transistors with the base region exposed.

These transistors are light sensitive, especially when infrared source of light is used. They have only two leads (collector and emitter). When there is no light the phototransistor is closed and does not allow a collector-emitter current to go through. The phototransistor opens only with the presence of sufficient light.

5.2 Functional operation

5.2.1. Functioning of opto-interrupter





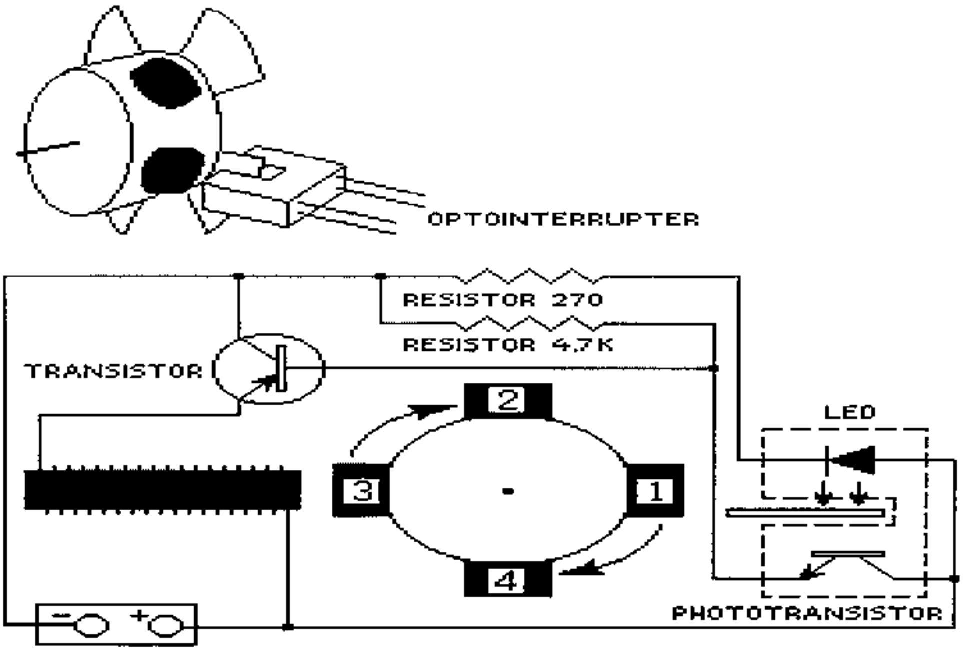
The motor with optical control uses a flat opaque disk with four blades attached to the rotor. When rotor spins the blades cross the slot and interfere with the beam from the LED to the phototransistor. The width of the blade greatly affects the efficiency of the motor as it determines how long the electromagnet stays on. The disk supplied with the kits is shown below. The blades are quite narrow so it consumes less current. With this width the motor may not start at 3V, but it works much longer in the range 4.5-6V.

If you decide to make a similar disk yourself keep in mind that it should be dense as paper or even cardboard might be semi-transparent for infrared light. The output from the phototransistor is very small and needs to be amplified. To make this motor as simple as possible and minimize the number of parts, a Darlington transistor was used. You may find information on this component at the idea of using opto-interrupters in brushless motors is not new.

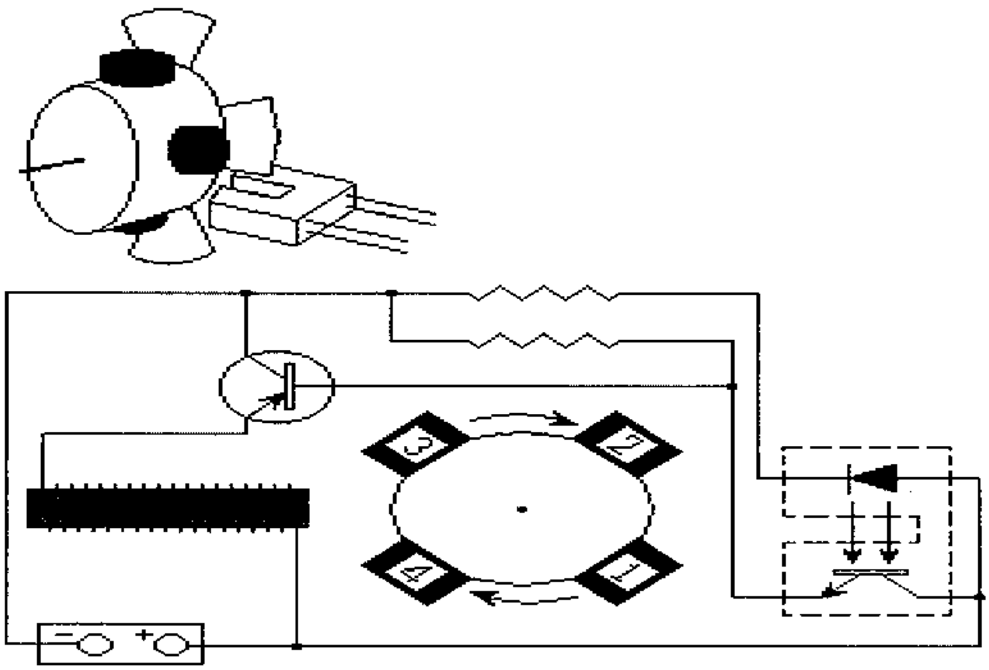
However, opto-interrupters are not widely used because the motors based on them are complicated and require a lot of parts. Dirt or a strong source of light pointed directly at the sensor may affect the performance of these motors.The motor with optical control is very reliable (more reliable than a reed switch motor) and should work indefinitely (actually, until mechanical parts wear down). This motor uses fewer parts than any other existing motors with optical sensors.

5.2.2. Working

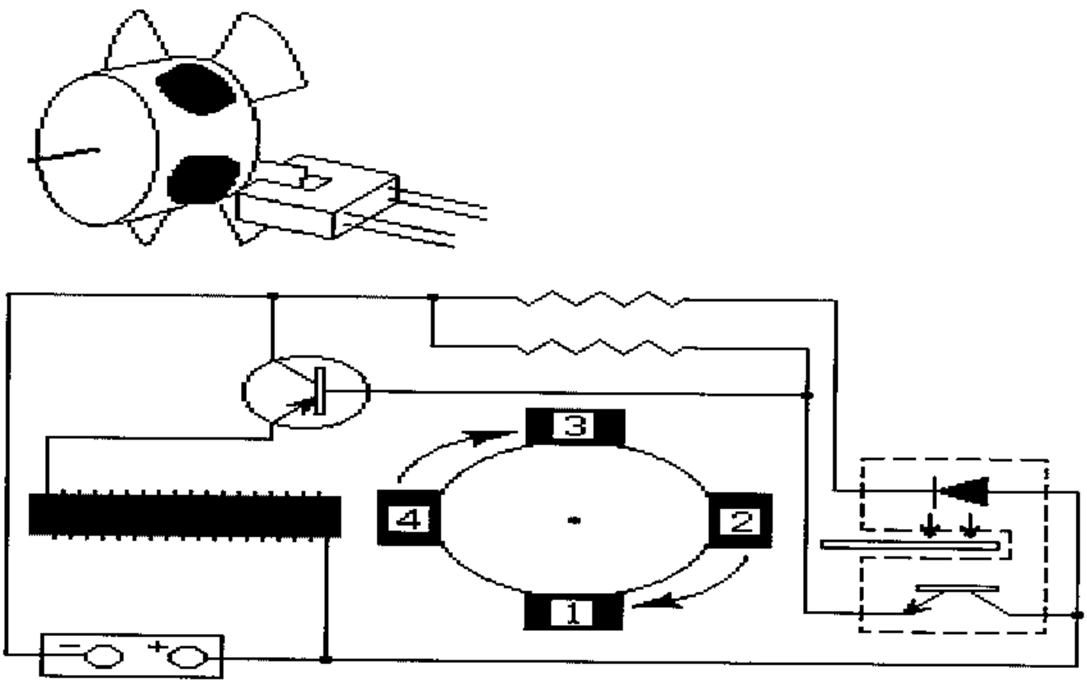
1. In the starting position, the disk blade interferes with the channel between the LED and the phototransistor. The phototransistor stays closed and does not allow the current to go through it. However, the PNP power Darlington transistor in this case is open due to the way it is connected. This allows a collector-emitter current to flow through the electromagnet. The electromagnet then pushes magnet #3 away.



2. When the magnets spin away, the disk moves out of the slot between the LED and phototransistor. The light signal to the phototransistor turns it on, and the phototransistor current turns the power transistor off. This disables the electromagnet.



3. The magnets continue to spin due to inertia until the next blade of the disk moves back into the slot. The phototransistor closes, and power transistor opens. Current through the electromagnet pushes magnet #4 away. This process continues until the power is disconnected.



CHAPTER 6

6. ENCODER

6.1. Introduction

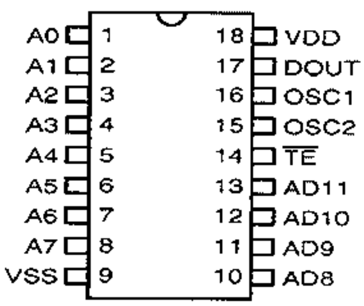
An encoder can be a device used to change a signal (such as a bit stream) or data into a code. The code serves any of a number of purposes such as compressing information for transmission or storage, encrypting or adding redundancies to the input code, or translating from one code to another. This is usually done by means of a programmed algorithm, especially if any part is digital, while most analog encoding is done with analog circuitry.

6.1.1. Encoder HT12E

The HT12E encoder is a CMOS IC built especially for remote control system applications. It is capable of encoding 8 bits of address (A0-A7) and 4 bits of data (AD8-AD11) information. Each address/data input can be set to one of the two logic states, 0 or 1. Grounding the pins is taken as a 0 while a high can be given by giving +5V or leaving the pins open (no connection). Upon reception of transmit enable (TE-active low), the programmed address/data are transmitted together with the header bits via an RF medium.

6.2. Pin details

6.2.1. Pin diagram



Pilot & Sync.	A0	A1	A2	A3	A4	A5	A6	A7	AD8	AD9	AD10	AD11
	1	1	1	[illegible]	1	[illegible]	1	1	0	0	0	0

6.2.2. Pin description

PIN NAME	I/O	INTERNAL CONNECTION	Description
A0-A7	I	NMOS TRANSMISSION GATE	Input pins for address A0-A7setting. They can be externally
			set to VDD or VSS.
AD8-AD11	I	NMOS TRANSMISSION	Input pins for data AD8-AD11 Input pins for data AD8-AD11
			setting. They can be externally set to VDD or VSS.
DOUT	O	CMOS OUT	Encoder data serial transmission output
TE	I	CMOS IN pull-high	Transmission Enable, active low
OSC 1	I	OSCILLATOR 1	Oscillator input pin
OSC2	O	OSCILLATOR 2	Oscillator output pin
VSS	I	-	Negative power supply(GND)
VDD	I	-	Positive power supply

6.3. Features

1. 2.4-12V Operation
2. Low power, high noise immunity CMOS technology
3. Low standby current of < 1µA at 5V supply
4. Built-in oscillator with only a 5% resister
5. Minimal external components

6.3.1. Electrical Characteristics Encoder

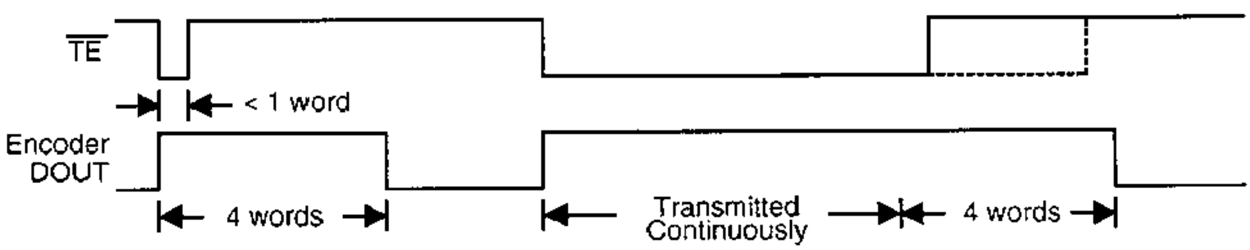
Symbol	Parameter	Test Condition		Min.	Typ.	Max.	Unit
		V_{DD}	Condition				
V_{DD}	Operating Voltage	—		2.4	5	12	V
I_{STB}	Stand-by Current	3V	Oscillator stops.	—	0.1	1	µA
		12V		—	2	4	µA
I_{DD}	Operating Current	3V	No load Fosc=3KHz	—	40	80	µA
		12V		—	150	300	µA
I_{DOUT}	Output Drive Current	5V	V_{OH}=0.9V_{DD} (Source)	-1	-1.6		mA
			V_{OL}=0.1V_{DD} (Sink)	1	1.6		mA
V_{IH}	"H" Input Voltage	—	—	0.8V_{DD}	—	V_{DD}	V
V_{IL}	"L" Input Voltage	—	—	0	—	0.2V_{DD}	V
Fosc	Oscillator Frequency	5V	Rosc=1.1MΩ	—	3	—	KHz
$R_{\overline{TE}}$	$\overline{TE}$ Pull-High Resistance	5V	$V_{\overline{TE}}$=0V	—	1.5	3	MΩ

(Ta=25°C)

6.4. Functional operation

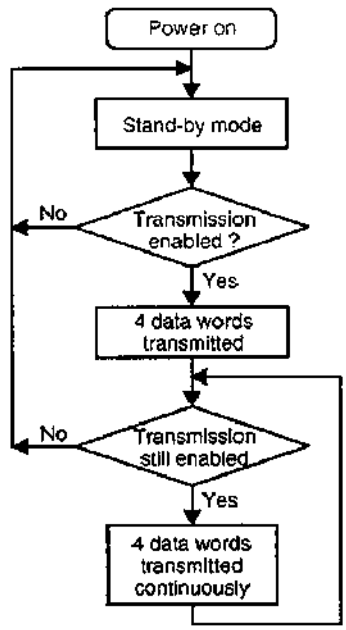
The encoder starts a 4 word transmission cycle upon reception of a transmit enable (TE active low). This cycle repeats itself as long as TE is held low. Once the TE goes high, the encoder completes its final cycle and stops as shown in Fig below.

6.4.1. Encoder Cycle Timing



As soon as a transmit enable occurs, the encoder scans and transmits the status of the 12 bits of address/data serially in the order A0 to AD11.

6.4.2. Encoder Operation Flowchart

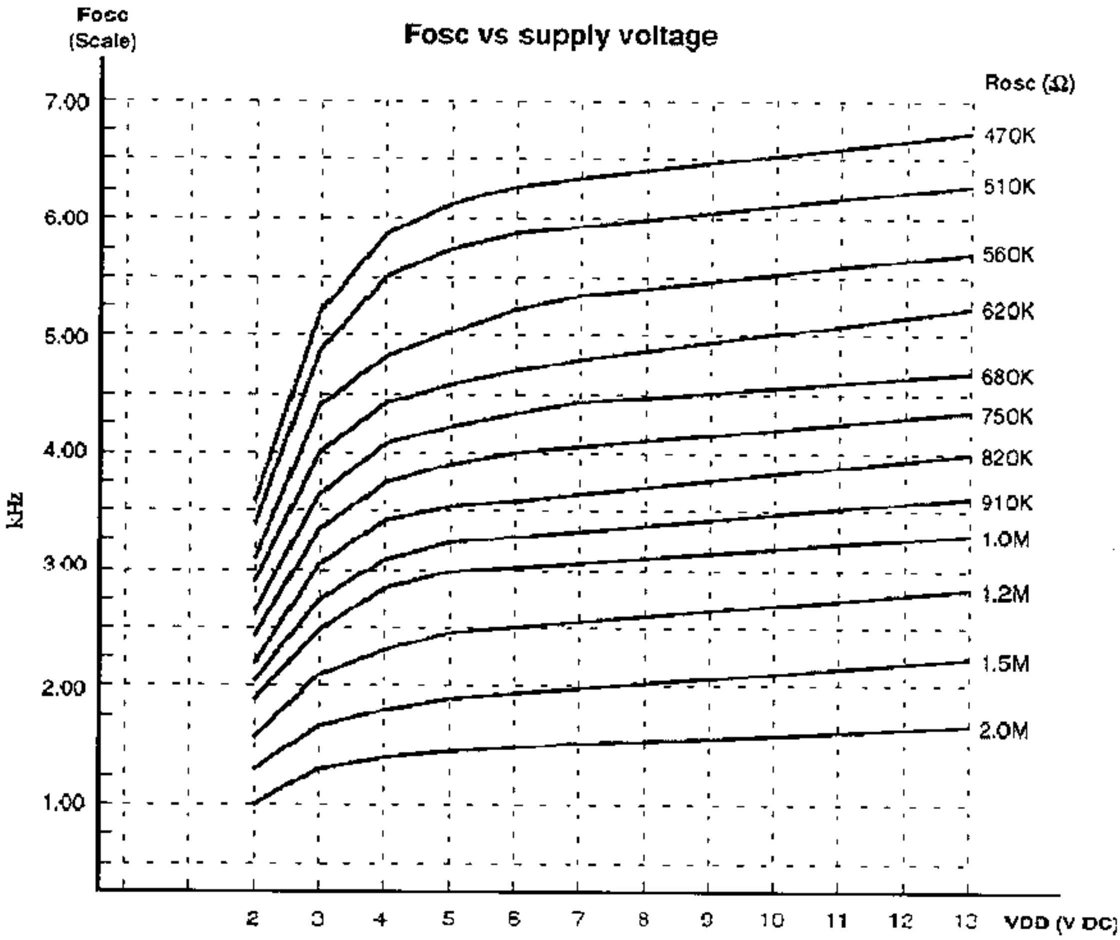


Encoder operation can be represented by a flowchart as shown in Fig .As an illustration of the way the data is sent serially, if all the 8 address lines were left open (no connection) and all 4 data lines were grounded, then the serial output would look like all open circuit address lines will be read as logic high and all 4 data bits will be read as 0 since they were grounded.

6.4.3. Encoder Oscillation Frequency

Since the encoder comes with a built in RC oscillator, its oscillation frequency can be set by connecting a resistor between OSC1 (pin 16) and OSC2 (pin15). The oscillation frequency depends on the resistor value as well as the supply voltage, as shown in Fig.

6.4.4. Encoder Oscillation Graph



CHAPTER 7

7. DECODER

7.1. Introduction

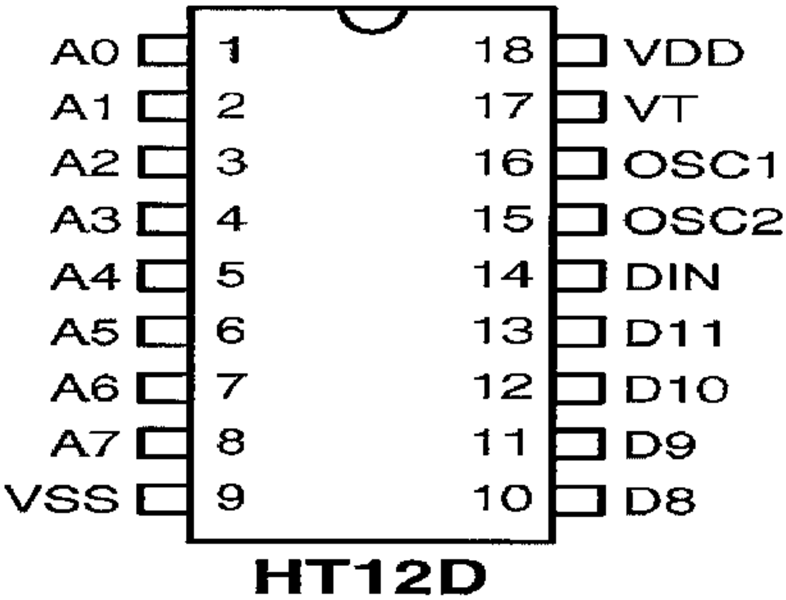
A **decoder** is a device which does the reverse of an encoder, undoing the encoding so that the original information can be retrieved. The same method used to encode is usually just reversed in order to decode. In digital electronics this would mean that a decoder is a multiple-input, multiple-output logic circuit that converts coded inputs into coded outputs. Enable inputs must be on for the decoder to function, otherwise its outputs assume a single "disabled" output code word. Decoding is necessary in applications such as data multiplexing, 7 segment display and memory address decoding.

7.1.1. Decoder HT12D

The HT12D is a decoder IC made especially to pair with the HT12E encoder. It is a CMOS IC made for remote control system application. The decoder is capable of decoding 8 bits of address (A0-A7) and 4 bits of data (AD8-AD11) information. For proper operation, a pair of encoder/decoder with the same number of addresses and data format should be chosen. The decoders receive serial addresses and data from programmed encoders that are transmitted by a carrier using an RF or an IR transmission medium. They compare the serial input data three times continuously with their local addresses. If no error or unmatched codes are found, the input data codes are decoded and then transferred to the output pins. The VT pin also goes high to indicate a valid transmission. The decoders are capable of decoding information that consists of N bits of address and 12_N bits of data. Of this series, the HT12D is arranged to provide 8 address bits and 4 data bits, and HT12F is used to decode 12 bits of address information.

7.2. Pin details

7.2.1. Pin diagram



7.2.2. Pin description

pin name	I/O	internal connection	description
A0-A7	I	NMOS TRANSMISSION GATE	Input pins for address A0-A7 setting. They can
			be externally set to VSS or left open.
D8-D11	o	CMOS OUT	Output data pins, power on-state is low.
DIN	I	CMOS IN	Serial data input pin.
VT	o	CMOS OUT	Valid Transmission, active high
OSC 1	I	OSCILLATOR 1	Oscillator input pin
OSC2	o	OSCILLATOR 2	Oscillator output pin
VSS	I	-	Negative power supply(GND)
VDD	I	-	Positive power supply

7.3. Features

1. Operating voltage: 2.4V~12V
2. Low power and high noise immunity CMOS Technology
3. Low standby current
4. Capable of decoding 12 bits of information
5. Binary address setting
6. Received codes are checked 3 times
7. Address/Data number combination
8. HT12D: 8 address bits and 4 data bits
9. HT12F: 12 address bits only
10. Built-in oscillator needs only 5% resistor
11. Valid transmission indicator
12. Easy interface with an RF or an infrared transmission medium

7.3.1. Electrical Characteristics of Decoder

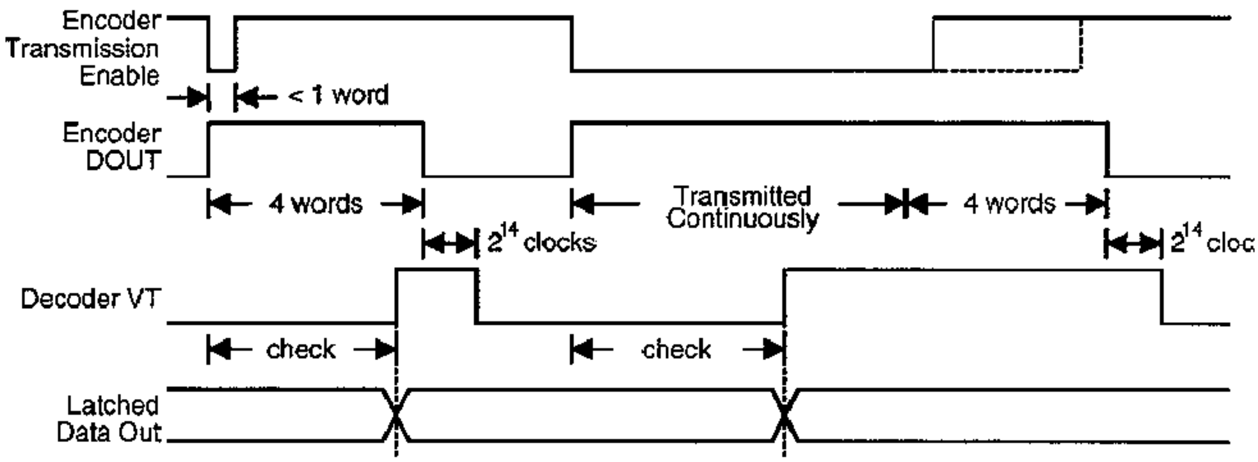
Ta=25°C

Symbol	Parameter	Test Conditions		Min.	Typ.	Max.	Unit
		V_{DD}	Conditions				
V_{DD}	Operating Voltage	—	—	2.4	5	12	V
I_{STB}	Standby Current	5V	Oscillator stops	—	0.1	1	μA
		12V		—	2	4	μA
I_{DD}	Operating Current	5V	No load, f_{OSC}=150kHz	—	200	400	μA
I_O	Data Output Source Current (D8~D11)	5V	V_{OH}=4.5V	1	-1.6		mA
	Data Output Sink Current (D8~D11)	5V	V_{OL}=0.5V	1	1.6		mA
I_{VT}	VT Output Source Current	5V	V_{OH}=4.5V	-1	-1.6		mA
	VT Output Sink Current		V_{OL}=0.5V	1	1.6		mA
V_{IH}	"H" Input Voltage	5V	—	3.5	—	5	V
V_{IL}	"L" Input Voltage	5V	—	0	—	1	V
f_{OSC}	Oscillator Frequency	5V	R_{OSC}=51kΩ		150		kHz

7.4. Functional operation

HT12D receives digital serial data from its DIN(pin14). A signal in the DIN activates the oscillator which then decodes the incoming address and data.

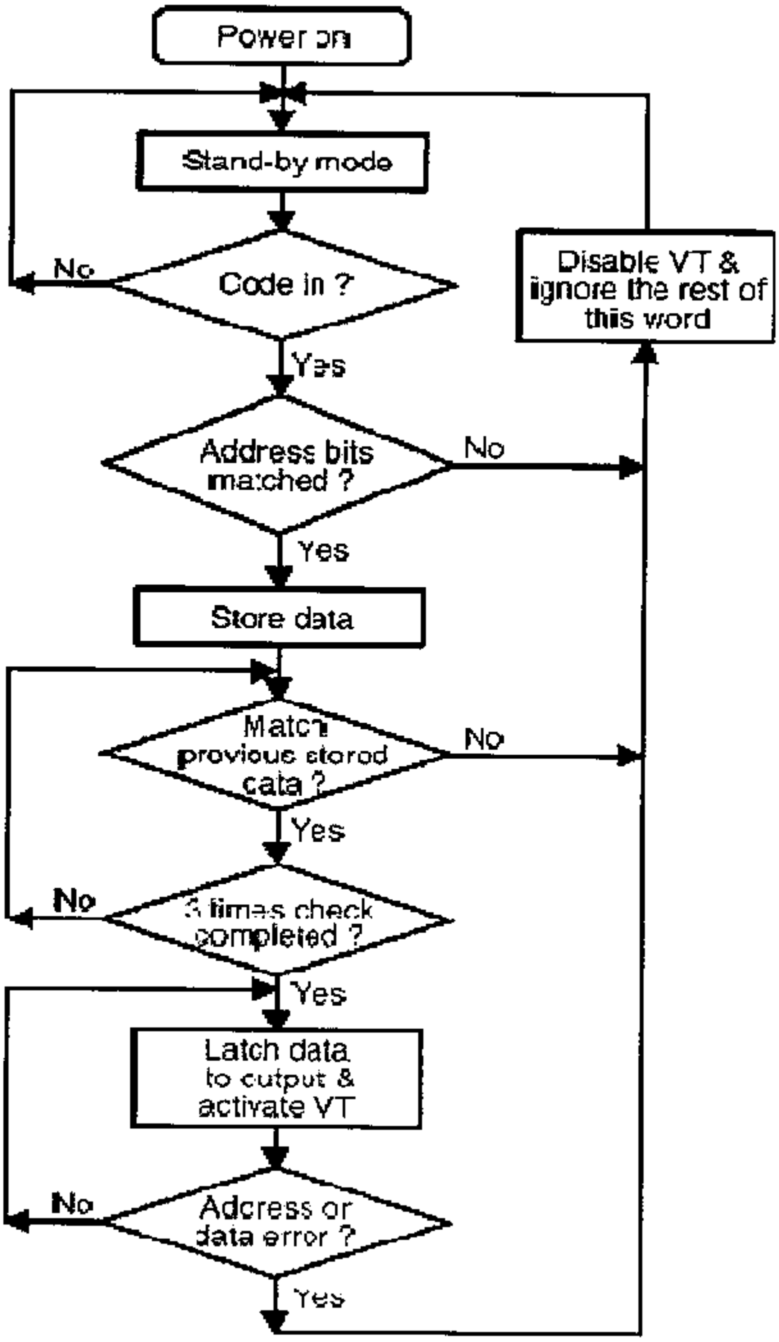
7.4.1. Decoder Timing



After decoding, it checks the serial input data three times continuously with its local addresses. If no error or unmatched codes are found, the input data codes are decoded and then transferred to the data output pins. This pin remains high for 214=16384 decoder clocks after the encoder's DOUT pin goes low. Since the decoder operates at 150 KHz, it takes 150000*16384=0.1 seconds for the VT pin to go low. This pin also goes low if the address code is incorrect or no signal is received. The 4 data pins are latched to their respective pins, meaning that the previous data remains on the pins unless a new data arrives to replace the existing one.

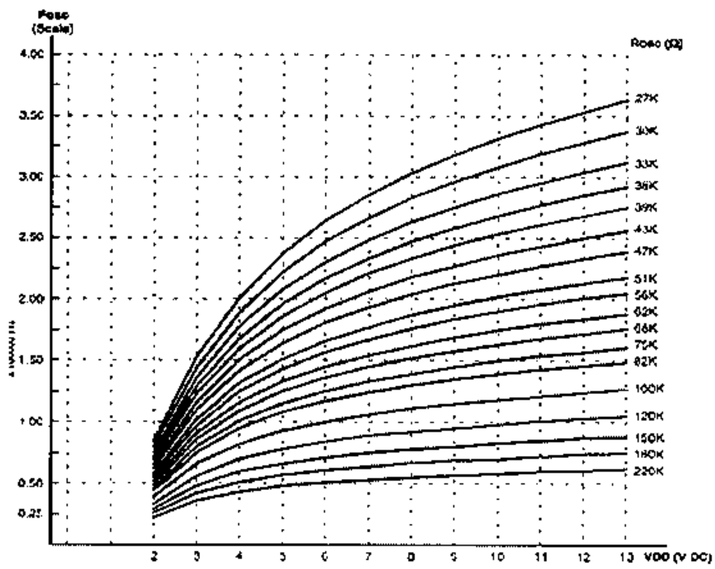
7.4.2. FLOWCHART

The oscillator is disabled in the standby state and activated when a logic high signal applies to the DIN pin. That is to say, the DIN should be kept low if there is no signal input.



7.4.3. DECODER OSCILLATION FREQUENCY

The decoder has a built in oscillator hence its clock can be set by connecting a resistor between OSC1 (pin 16) and OSC2 (pin 15). The oscillation frequency depends on the resistor value as well as the power supply as shown below. This project uses a 5V supply and it is recommended by the Holtek that oscillator frequency of decoder = 50*oscillator frequency of encoder. Since the HT12E encoder works at 3 KHz, the decoder frequency has to be 150 KHz. This requires a 51k resistor.



7.5 Applications

1. Burglar alarm system
2. Smoke and fire alarm system
3. Garage door controllers
4. Car door controllers
5. Car alarm system
6. Security system
7. Cordless telephones, etc

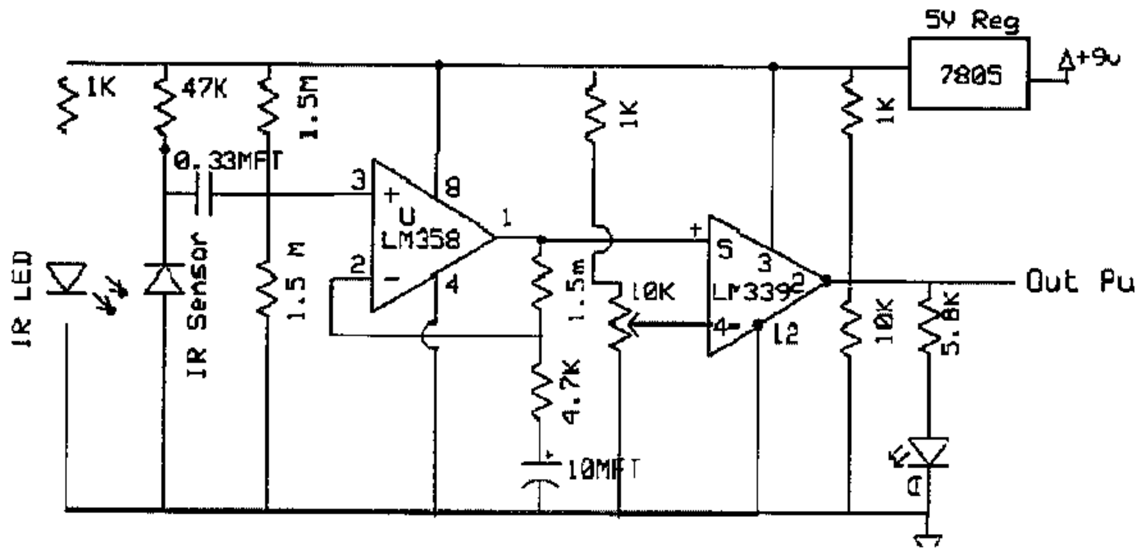
CHAPTER 8

8. HEART BEAT SENSOR

8.1 Introduction

Here we are using IR sensor for detecting the HEART BEAT. IR has less noise and ambient light than at normal optical wavelengths. The light is produced only when current passes through in the forward direction and block current in the reverse direction. Plethysmograph is an infrared photoelectric sensor used to record changes in pulsatile blood flow from the finger. The Plethysmograph operates by recording changes in blood volume as the arterial pulse expands and contracts the microvasculature.

8.2 circuit diagram



8.3 operation

This is a non-invasive measurement for changes in finger blood flow during wakefulness and sleep. Pulse wave amplitude (PWA) is the most frequently used parameter obtained by finger plethysmography. PWA is directly and positively correlated to finger blood flow. The hypothesis of this study was that finger plethysmography detects pharmacologically induced changes in finger blood flow, in particular changes induced by stimulation and blockade of vascular a-receptors. Due to the anatomic structure of the finger we expected that alterations of vascular tone following sympathetic activation or inhibition might be reflected by changes of PWA. A change in finger blood flow, reflected by PWA is derived from the finger plethysmography. PWA derived from finger plethysmography allows continuous, noninvasive measurement of changes in finger blood flow during wakefulness and sleep. Finally, to demonstrate the ability of finger plethysmography to continuously monitor vascular tone, PWA responses to obstructive breathing and concomitant arousal events in patients with obstructive sleep apnea were recorded and analysed.

8.4 Applications

1. Monitors pulse rate and rhythm.
2. Alphanumeric LCD Display.
3. Pulse monitoring by sound beeps and LED indication.
4. Shows Pulse Rate per minute after every 5 pulse count. 10 or 15 pulse count average also selectable.
5. Infra-Red Optical Finger /Ear Lobe Clip sensor.
6. Bar Graph to display signal sensitivity.
7. 15 Memories.
8. Lightweight, Easy to handle, Simple operation.
9. Works on power and battery both.

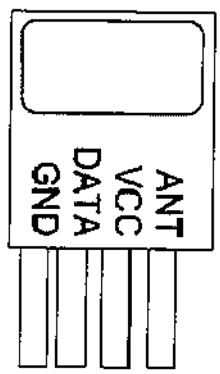
CHAPTER 9

9. TRANSMITTER AND RECEIVER

9.1 Introduction

It is an ideal for remote control applications where low cost and longer range is required. The transmitter operates from a 1.5-12V supply, making it ideal for battery-powered applications. The transmitter employs a SAW-stabilized oscillator, ensuring accurate frequency control for best range performance. Output power and harmonic emissions are easy to control, making FCC and ETSI compliance easy.

9.2 Operation of transmitter



9.2.1 PIN Description

ANT : 50 ohm antenna output. The antenna port impedance affects output power and harmonic emissions. An L-C low-pass filter may be needed to sufficiently filter harmonic emissions. Antenna can be single core wire of approximately 17cm length or PCB trace antenna.

VCC: Operating voltage for the transmitter. VCC should be bypassed with a .01uF ceramic capacitor and filtered with a 4.7uF tantalum capacitor. Noise on the power supply will degrade transmitter noise performance.

DATA :Digital data input. This input is CMOS compatible and should be driven with CMOS level inputs.

GND : Transmitter ground. Connect to ground plane.

9.2.2 Operation

OOK (On off Keying) modulation is a binary form of amplitude modulation. When a logical 0 (data line low) is being sent, the transmitter is off, fully suppressing the carrier. In this state, the transmitter current is very low, less than 1mA. When a logical 1 is being sent, the carrier is fully on. In this state, the module current consumption is at its highest, about 11mA with a 3V power supply.

On-off keying (OOK) is a type of modulation that represents digital data as the presence or absence of a carrier wave. In its simplest form, the presence of a carrier for a specific duration represents a binary one, while its absence for the same duration represents a binary zero. Some more sophisticated schemes vary these durations to convey additional information.

OOK is the modulation method of choice for remote control applications where power consumption and cost are the primary factors. Because OOK transmitters draw no power when they transmit a 0, they exhibit significantly better power consumption than FSK transmitters.

OOK data rate is limited by the start-up time of the oscillator. High-Q oscillators which have very stable center frequencies take longer to start-up than low-Q oscillators. The start-up time of the oscillator determines the maximum data rate that the transmitter can send.

9.2.3 Functions

Data Rate

The oscillator start-up time is on the order of 40uSec, which limits the maximum data rate to 4.8 Kbit/sec

SAW stabilized oscillator

The transmitter is basically a negative resistance LC oscillator whose center frequency is tightly controlled by a SAW resonator. SAW (Surface Acoustic Wave) resonators are fundamental frequency devices that resonate at frequencies much higher than crystals.

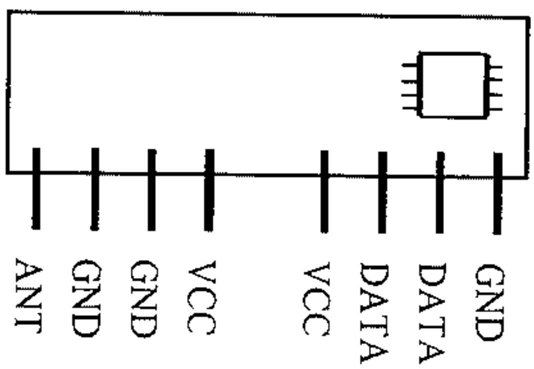
9.2.4 Features

1. 433.92 MHz Frequency
2. Low Cost
3. 1.5-12V operation
4. 11mA current consumption at 3V
5. Small size
6. 4 dBm output power at 3V

9.2.5 Application

1. Remote Keyless Entry (RKE)
2. Remote Lighting Controls
3. On-Site Paging
4. Asset Tracking
5. Wireless Alarm and Security Systems
6. Long Range RFID
7. Automated Resource Management

9.3 Operation of receiver



9.3.1 Pin description

ANT: Antenna input.

GND: Receiver Ground. Connect to ground plane.

VCC (5V): VCC pins are electrically connected and provide operating voltage for the Receiver.

VCC can be applied to either or both. VCC should be bypassed with a .1μF ceramic capacitor. Noise on the power supply will degrade receiver sensitivity.

DATA: Digital data output. This output is capable of driving one TTL or CMOS load. It is a CMOS compatible output.

9.3.2 Operation

The RF Receiver module uses a super-regenerative AM detector to demodulate the incoming AM carrier. A super regenerative detector is a gain stage with positive feedback greater than unity so that it oscillates. An RC-time constant is included in the gain stage so that when the gain stage oscillates, the gain will be lowered over time proportional to the RC time constant until the oscillation eventually dies. When the oscillation dies, the current draw of the gain stage decreases, charging the RC circuit, increasing the gain, and ultimately the oscillation starts again. In this way, the oscillation of the gain stage is turned on and off at a rate set by the RC time constant. This rate is chosen to be super-audible but much lower than the main oscillation rate. Detection is accomplished by measuring the emitter current of the gain stage. Any RF input signal at the frequency of the main oscillation will aid the main oscillation in restarting. If the amplitude of the RF input increases, the main oscillation will stay on for a longer period of time, and the emitter current will be higher. Therefore, we can detect the original base-band signal by simply low-pass filtering the emitter current. The average emitter current is not very linear as a function of the RF input level. It exhibits a 1/ln response because of the exponentially rising nature of oscillator start-up. The steep slope of a logarithm near zero results in high sensitivity to small input signals.

9.3.3 Functions

Data Slicer

The data slicer converts the base-band analog signal from the super-regenerative detector to a CMOS/TTL compatible output. Because the data slicer is AC coupled to the audio output, there is a minimum data rate. AC coupling also limits the minimum and maximum pulse width. Typically, data is encoded on the transmit side using pulse-width modulation (PWM) or non-return-to-zero (NRZ).

The most common source for NRZ data is from a UART embedded in a micro-controller. Applications that use NRZ data encoding typically involve microcontrollers. The most common source for PWM data is from a remote control IC such as the HC-12E.

Data is sent as a constant rate square-wave. The duty cycle of that square wave will generally be either 33% (a zero) or 66% (a one). The data slicer on the STR-433 is optimized for use with PWM encoded data, though it will work with NRZ data if certain encoding rules are followed.

Power Supply

The STR-433 is designed to operate from a 5V power supply. It is crucial that this power supply be very quiet. The power supply should be bypassed using a 0.1uF low-ESR ceramic capacitor and a 4.7uF tantalum capacitor. These capacitors should be placed as close to the power pins as possible. The STR- 433 is designed for continuous duty operation. From the time power is applied, it can take up to 750mSec for the data output to become valid.

Antenna Input

It will support most antenna types, including printed antennas integrated directly onto the PCB and simple single core wire of about 17cm. The performance of the different antennas varies. Any time a trace is longer than 1/8th the wavelength of the frequency it is carrying, it should be a 50 ohm microstrip.

9.3.4 Features

1. Low Cost
2. 5V operation
3. 3.5mA current drain
4. No External Parts are required
5. Receiver Frequency: 433.92 MHZ
6. Typical sensitivity: -105dBm
7. IF Frequency: 1MHz

9.3.5 Applications

1. Car security system
2. Sensor reporting
3. Automation system
4. Remote Keyless Entry (RKE)
5. Remote Lighting Controls
6. On-Site Paging
7. Wireless Alarm and Security Systems
8. Long Range RFID
9. Automated Resource Management

9.3.6 Advantages

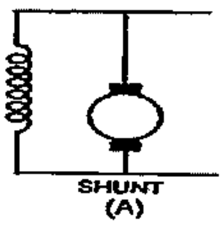
1. High level of integration.
2. Easy read of hardware design.
3. Increased reliability.
4. Micro controller is pretested by manufacturer.
5. Communicate easily with PC via serial port.
6. Operates with 5v only.

CHAPTER 10

10. DC MOTOR

10.1 Introduction

DC motors are part of the electric motors using DC power as energy source. These devices transform electrical energy into mechanical energy. The basic principle of DC motors is same as electric motors in general, the magnetic interaction between the rotor and the stator that will generate spin.



WE USE BRUSHED DC SHUNT MOTOR

10.2 working operations

Simple motor has six parts:

1. Armature or rotor
2. Commutator
3. Brushes
4. Axle
5. Field magnet
6. DC power supply of some sort

Graphically, the working principles of DC motors can be seen in Figure 1.

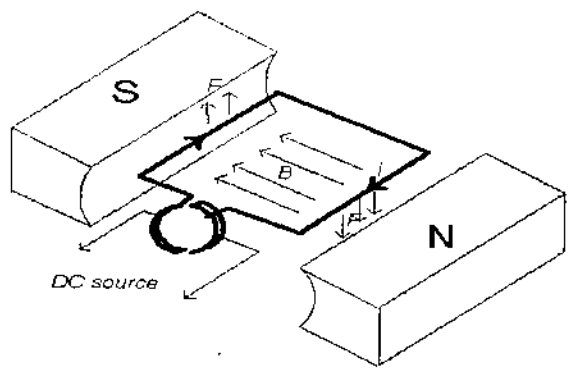


Figure 1. Principle of DC motor

When DC electric current flowing in the coil in accordance with the direction of the arrow, while the direction of the magnetic field B is from north to south pole, the coil will be driven by the force F in the direction as shown in Figure 1. This condition occurs continuously so will result in rotation on the axis of the coil. The direction of the electric current in the coil is fixed, because of the split ring on the end of the coil.

Brushed Motor

A Brushed Motor has a rotating set of wound wire coils called an armature which acts as an electromagnet with two poles. A mechanical rotary switch called a commutator reverses the direction of the electric current twice every cycle, to flow through the armature so that the poles of the electromagnet push and pull against the permanent magnets on the outside of the motor. As the poles of the armature electromagnet pass the poles of the permanent magnets, the commutator reverses the polarity of the armature electromagnet. During the instant of switching polarity, inertia keeps the classical motor going in the proper direction.

Brushed motor Advantages

1. Two wire control
2. Replaceable brushes for extended life
3. Low cost of construction
4. No controller is required for fixed speeds
5. Operates in extreme environments due to lack of electronics

CHAPTER 11

11. HARDWARE DESCRIPTION

11.1 Description

The micro controller ATMEL 89S52 is used here to take care of simple analytical operations. Two micro controllers are used for two different kits. One micro controller takes care of the direct values obtained from the vehicle unit and the other for heartbeat circuit operation.

Vehicle Unit:

The Brushed DC shunt motor is used as a model for vehicle. The speed of the motor is varied using a potentiometer. The speed is detected using a OPTO-INTERRUPTER and the values are passed to MICROCONTROLLER I. For the fuel tank setup we use a potentiometer. The variable values from a potentiometer are given to ANALOG TO DIGITAL CONVERTER 0809. The digital values are fed to MICROCONTROLLER I.

The values transmitted from Heart beat sensor are received by a 433 MHz RECEIVER and decoded using HT12D DECODER. The decoded values are also given to MICROCONTROLLER I. The signal processed by micro controller gets weak, so we use a LEVEL CONVERTER to improve the quality of weak signal. The computer that requires real time values from Micro controller is interfaced using USB – SERIAL CONVERTER cable.

PORT 0, PORT 1, PORT3 of MICRO CONTROLLER I is used here.

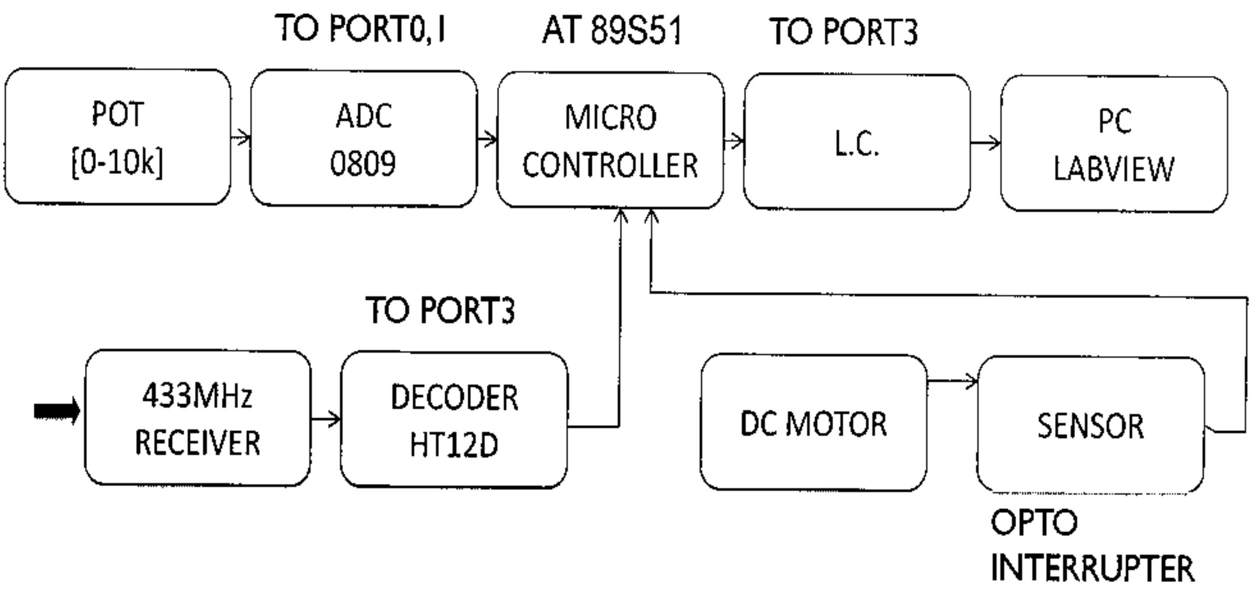
Heartbeat Unit:

An IR TRANSMITTER – RECEIVER is used to track the variation in heartbeat. The varying pulses from IR circuit are fed into a heart beat sensor circuit. The result from heartbeat circuit is sent to MICRO CONTROLLER II. The micro controller processes the heartbeat signals and sends it to ENCODER - HT12E. the data is encoded and transmitted to vehicle unit by means of a 433MHz TRANSMITTER.

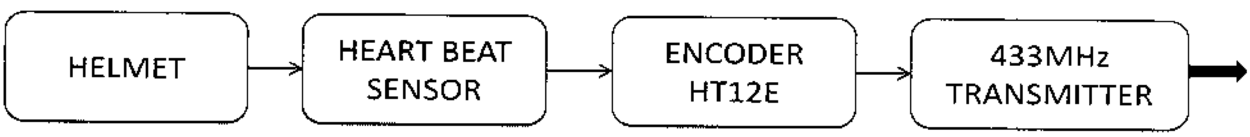
PORT 2 of MICRO CONTROLLER II is used here.

11.2 Block Diagram

Vehicle Unit:



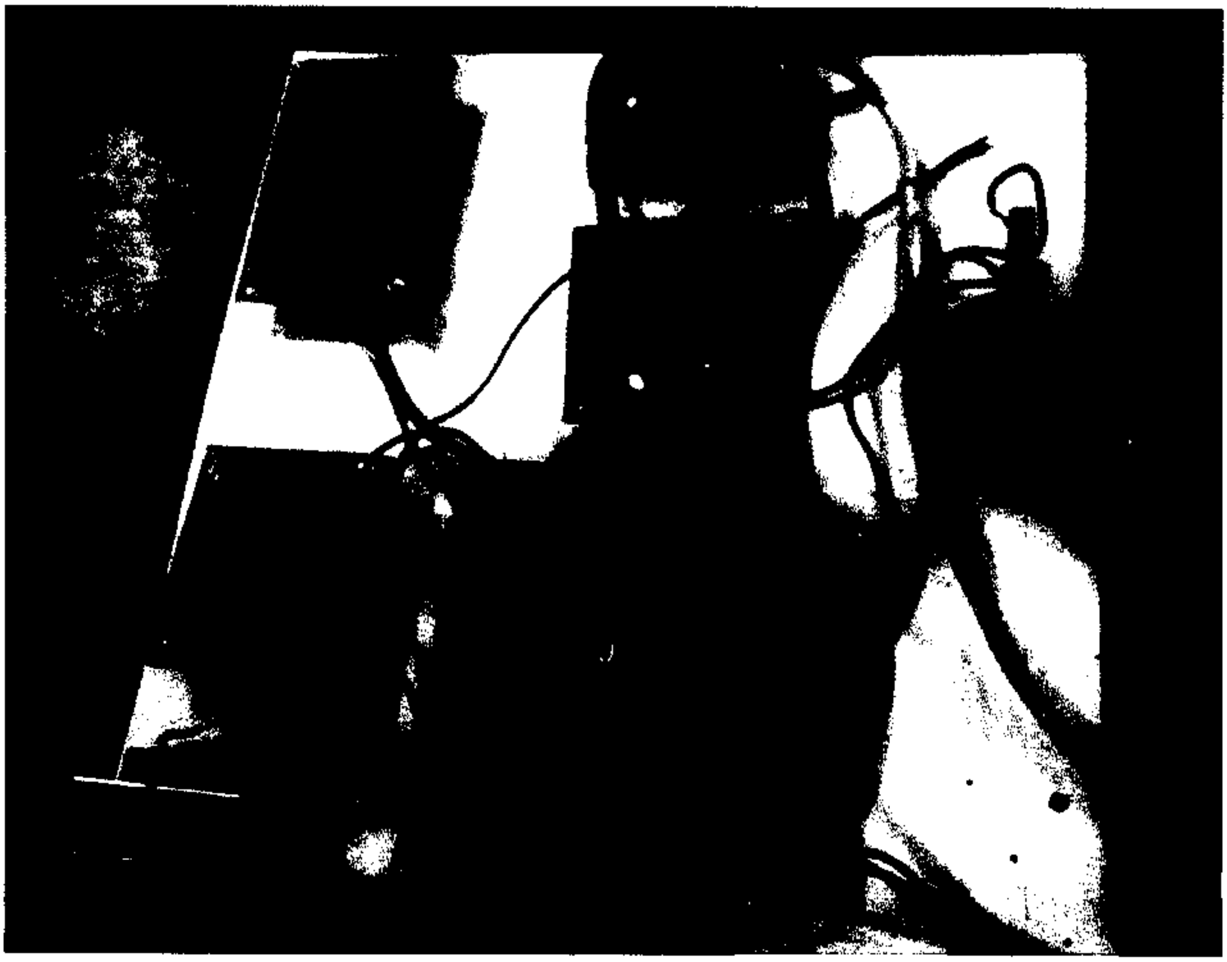
Heartbeat sensor Unit:



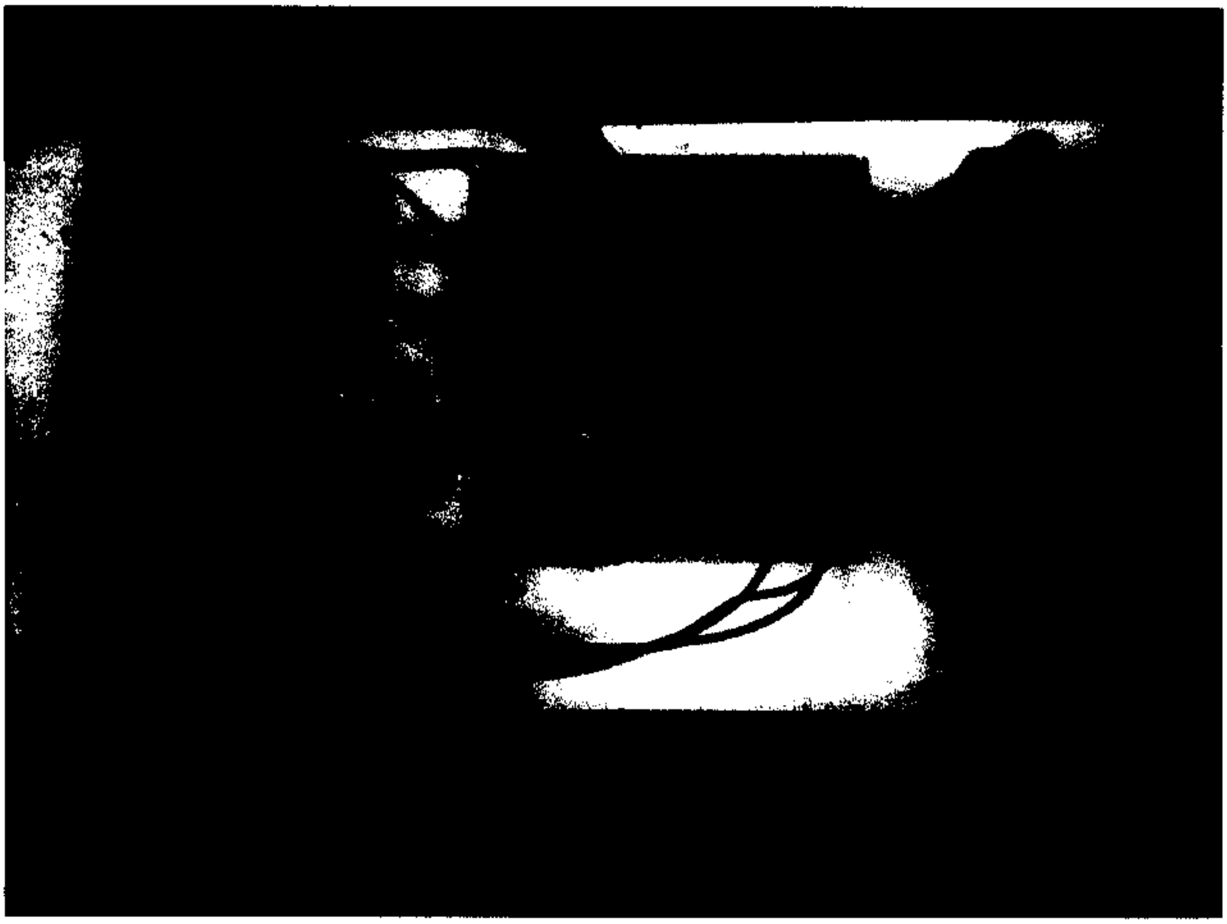
POT (0-10 K)	: Representation of a potentiometric-float
LC	: Level converter
Helmet	: IR sensors
DC motor	: Brushed shunt DC motor
RF transmitter & receiver	: 433 MHz Transmitter & Receiver

11.3 Hardware kit

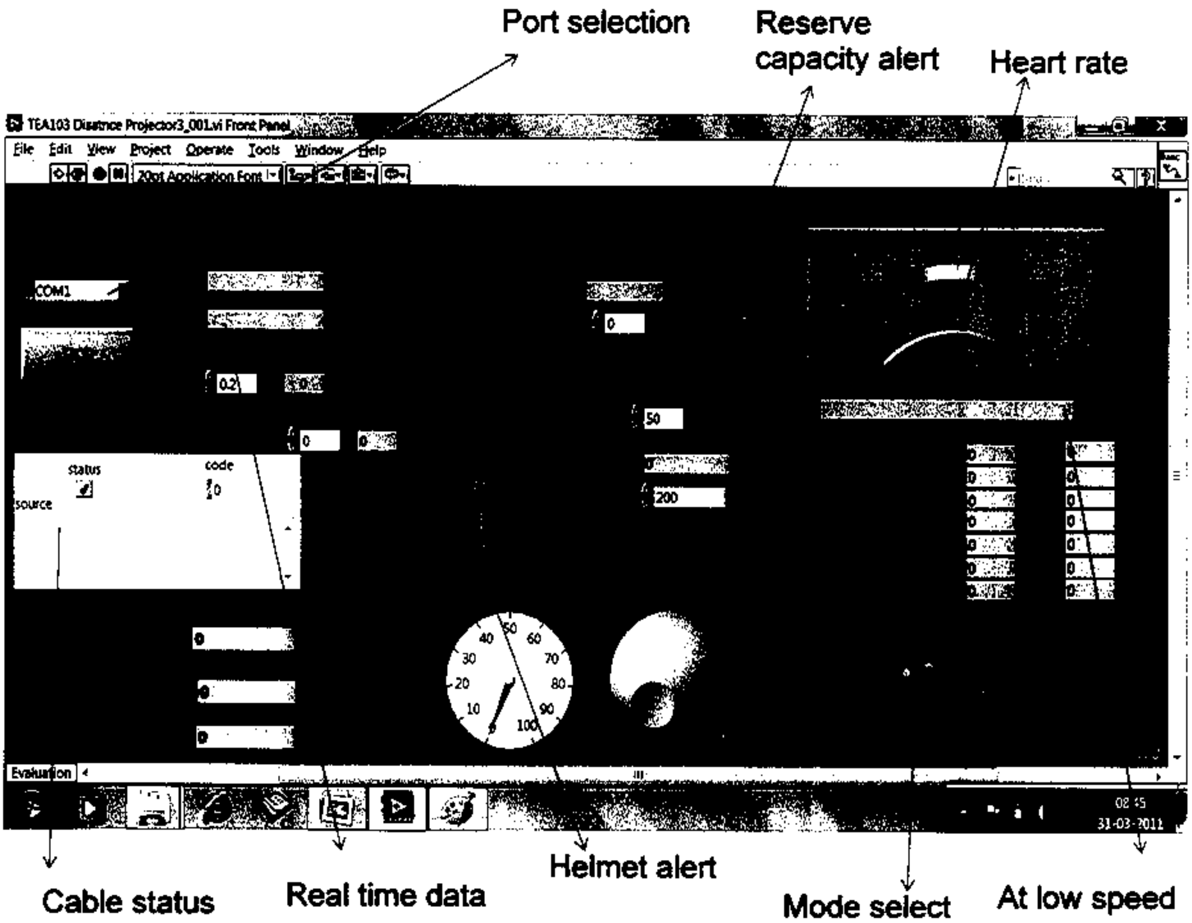
11.3.1 Vehicle Unit:



11.3.2 Heartbeat Unit:



11.4 LABVIEW DISPLAY



CHAPTER 12

12. ALGORITHM/FORMULAE

1. SPEED:

Data received: RPM

Therefore speed is RPM * 2 * pi * r

r is the radius of wheel.

2. ODOMETER:

Reducing the values of speed to milli-second level

Ex: expressing 3600 km / hr as, 0.001 km/ ms.

By continuous summation of distance, we obtain odometer readings.

3. DISTANCE YET TO TRAVEL:

Pre entered distance – odometer value updating for every 3 second

4. TIME OF TRAVEL:

Step 1: (Speed 1 + Speed 2 + Speed 3 + Speed 4 + Speed 5) /5 = Avg speed.

Step 2: (Avg speed + new speed)/2 = Avg. speed

Step 3: At calculation of every Avg. speed

Perform (Speed / Avg. speed) = Time.

Step 4: Avg. time calculation is also similar to speed average calculation

Step 5: Display Avg. speed and Avg. time

CHAPTER 13

13. CONCLUSION AND FUTURE SCOPE

13.1 CONCLUSION

This project purely works on sophisticating the people by improving the qualities of real time data and giving out new gadgets for travel. Most of the conventional system are analog type, the motive of this project is also to digitalize all the parameters and to give a new look to the existing technology. The question of building a new computer and its inquiring costs in the vehicle may arise, but by downloading the programs in advanced ARM processor and mass installation of this unit considerably reduces the cost of the product.

The concepts of displaying Time of travel, Distance yet to travel, Helmet sensing and fetching data from real time users are the advantages of this project. We chose this project hoping that these gadgets are new to the market and are less focused.

13.2 FUTURE SCOPE

The future scope is to improve our algorithm for better accuracy in time of travel considering many more factors. We would like to enhance the project by sensing the heartbeat of old aged people and sending a message in case of emergency. Also we are working in a transducer based technology replacing float type fuel measurement for better accuracy. All these gadgets will be non negotiable factors in near future.

REFERENCE

BIBLIOGRAPHY

BOOKS:

1. THE 8051 MICROCONTROLLER and EMBEDDED SYSTEMS - MUHAMMAD ALI MAZIDI.

2. SOLUTIONS OF 8051 MICROCONTROLLER BY KENNETH J. AYALA

REFERENCES

http://www.aplusinc.com.tw

www.datasheetcatalog.com

http://iec.org

http://www.neevia.com

http://www.leadtek.com

http://howstuffworks.com

http://gcn.com

http://www.simcom-sh.com

http://www.nskelectronics.com

APPENDIX

APPENDIX I:

1. VEHICLE UNIT CODING:

```
lrs         equ 0a0h
lrw         equ 0a1h
len         equ 0a2h

rpm_count0  equ 5ah
rpm_count1  equ 5bh
rpm_count2  equ 5ch
rpm_count3  equ 5dh
rpm_sense   equ 0b2h

heartrate0  equ 53h
heartrate1  equ 54h

ale    equ 80h
eoc         equ 81h
oe          equ   82h
clk    equ 83h
adc         equ p1

lvl3   equ 50h
lvl2     equ 51h
lvl1     equ 52h

;****************************************

        org 0000h
            mov p2,#00h
            mov p0,#0ffh
            mov p3,#0ffh
            mov p1,#0ffh
            acall lcddelay
        acall lcd_init
            acall lcddelay
        mov dptr,#line1
        acall message
        mov dptr,#line2
        acall message
        acall serial_init
            acall txdelay
            acall txdelay
        mov a,#"O"
            acall senddata
            mov a,#"K"
            acall senddata
            acall txdelay
            mov a,#0c0h
```

```
            acall command

skip:
            mov p3,#0ffh
waitq11:    jb p3.3,waitq11
waitq22:    jnb p3.3,waitq22
                  mov a,p3
                  swap a
                  anl a,#0fh
;                 add a,#30h
;                 acall display
;                 mov a,p3
;                 swap a
;                 anl a,#0fh
             cjne a,#0ah,waitq11
              mov a,#0c7h
              acall command


waitq33:    jb p3.3,waitq33
waitq44:    jnb p3.3,waitq44
                  mov a,p3
                  swap a
                  anl a,#0fh
                  add a,#30h
                  mov heartrate1,a

waitq55:    jb p3.3,waitq55
waitq66:    jnb p3.3,waitq66
                  mov a,p3
                  swap a
                  anl a,#0fh
                  add a,#30h
                  mov heartrate0,a


            acall lcddelay
        acall read_adc
        mov r0,a
        acall convert
        acall adcdelay

            acall lcddelay
        acall read_adc
        mov r0,a
        acall convert


        mov tmod,#21h
        mov r3,#20
        mov rpm_count0,#00h
        mov rpm_count1,#00h
            mov rpm_count2,#00h
            mov rpm_count3,#00h
```

```
again:
        mov tl0,#08h
        mov th0,#01h
        setb tr0
back:   jnb rpm_sense,chk1
            acall rpmcount
chk1: jnb tf0,back
            clr tr0
            clr tf0
            djnz r3,again
            mov a,#80h
            acall command
            mov a,rpm_count3
            add a,#30h
            mov rpm_count3,a
            acall display
            mov a,rpm_count2
            add a,#30h
            mov rpm_count2,a
            acall display
            mov a,rpm_count1
            add a,#30h
            mov rpm_count1,a
            acall display
            mov a,rpm_count0
            add a,#30h
            mov rpm_count0,a
            acall display
        acall senddata2pc
        acall txdelay
        ljmp skip



rpmcount:
        mov a,rpm_count0
        add a,#01h
        mov rpm_count0,a

        cjne a,#0ah,escw1
        mov rpm_count0,#00h
        mov a,rpm_count1
        inc a
        mov rpm_count1,a

        cjne a,#0ah,escw1
        mov rpm_count1,#00h
        mov a,rpm_count2
        inc a
        mov rpm_count2,a

        cjne a,#0ah,escw1
        mov rpm_count2,#00h
        mov a,rpm_count3
```

```
          inc a
          mov rpm_count3,a

escwl:        ;jb rpm_sense,escwl
              ;acall smsdelay
;             acall smsdelay
              ret

senddata2pc:
              mov a,#"<"
              acall senddata

              mov a,lvl3
              acall senddata
              mov a,lvl2
              acall senddata
              mov a,lvl1
              acall senddata

          mov a,#"N"
          acall senddata
          mov a,#":"
          acall senddata
          mov a,rpm_count3
          acall senddata
          mov a,rpm_count2
          acall senddata
          mov a,rpm_count1
          acall senddata
          mov a,rpm_count0
          acall senddata


          mov a,#"H"
          acall senddata
          mov a,#":"
          acall senddata
          mov a,heartrate1
          acall senddata
          mov a,heartrate0
          acall senddata
              ret


senddata:
          mov sbuf,a
l2:       jnb ti,l2
          clr ti
          ret

serial_init:
          mov tmod,#21h
          mov th1,#0fdh
          mov scon,#50h
```

```
            setb tr1
            clr ti
            ret


read_adc
                setb ale
                clr oe
                setb eoc

                clr clk
                nop
                setb clk
                nop
                clr ale

adc1
                clr clk
                nop
                setb clk
                nop
                jnb eoc,adc1

                setb oe
                clr clk
                nop
                setb clk
                nop

                clr clk
                nop
                setb clk
                nop

                clr clk
                nop
                setb clk
                nop



;               ret

                mov c,adc.7
                mov acc.0,c
                mov c,adc.6
                mov acc.1,c
                mov c,adc.5
                mov acc.2,c
                mov c,adc.4
                mov acc.3,c
                mov c,adc.3
                mov acc.4,c
```

```
                mov c,adc.2
                mov acc.5,c
                mov c,adc.1
                mov acc.6,c
                mov c,adc.0
                mov acc.7,c
                ret

convert:
        mov a,r0
        mov b,#02h
        mul ab
        subb a,#02h
        mov b,#10
        div ab
        mov 77h,b
        mov b,#10
        div ab
        mov 76h,b
        mov 75h,a
        mov a,#0c0h
        acall command
        mov a,75h
        add a,#30h
        mov lvl3,a
        acall display
            mov a,76h
        add a,#30h
        mov lvl2,a
        acall display
        mov a,77h
        add a,#30h
        mov lvl1,a
        acall display
        ret


lcd_init:
        setb len
        mov a,#2ch
        mov p2,a
        clr len
        acall command
        mov a,#0eh
        acall command
        mov a,#06h
        acall command
        mov a,#01h
        acall command
        ret
```

```
message:
        clr a
        movc a,@a+dptr
        inc dptr
        cjne a,#80h,comd
        acall command
        sjmp message
comd:
        cjne a,#0c0h,data
        acall command
        sjmp message
data:
        cjne a,#0ffh,disp
        ret
disp:
        acall display
        sjmp message


command:
                acall ready
                mov 7eh,a
                anl a,#0f0h
                mov p2,a
                clr lrs
                clr lrw
                setb len
                clr len

                acall ready
                mov a,7eh
                swap a
                anl a,#0f0h
                mov p2,a
                clr lrs
                clr lrw
                setb len
                clr len
                ret

display:
                        acall ready
                        mov 7eh,a
                        anl a,#0f0h
                        mov p2,a
                        setb lrs
                        clr lrw
                        setb len
                        clr len

                        acall ready
                        mov a,7eh
                        swap a
                        anl a,#0f0h
```

```
                    mov p2,a
                    setb lrs
                    clr lrw
                    setb len
                    clr len
                    ret
ready:
                    acall lcddelay
                    ret
line1:
        db 80h,"*** K.L.N C E***",0ffh
line2:
        db 0c0h,"T:   I:    H:    ",0ffh


smdelay:
        mov 72h,#01h
dl3:    mov 73h,#7fh
dl2:    mov 74h,#7fh
dl1:    djnz 74h,dl1
        djnz 73h,dl2
        djnz 72h,dl3
        ret


adcdelay:
                mov 72h,#02h
tdy3:           mov 73h,#00h
tdy2:           mov 74h,#00h
tdy1:           djnz 74h,tdy1
                djnz 73h,tdy2
                djnz 72h,tdy3
                ret


txdelay:
                mov 72h,#04h
tldy3:          mov 73h,#00h
tldy2:          mov 74h,#00h
tldy1:          djnz 74h,tldy1
                djnz 73h,tldy2
                djnz 72h,tldy3
                ret


lcddelay:
        mov 72h,#03h
dly3:   mov 73h,#0fh
dly2:   mov 74h,#10h
dly1:   djnz 74h,dly1
        djnz 73h,dly2
        djnz 72h,dly3
        ret
```

```
tldy1:          djnz r0,tldy1
                djnz r1,tldy2
                djnz r2,tldy3
                ret

longdelay:
                mov r2,#02h
tdy3:           mov r1,#00h
tdy2:           mov r0,#00h
tdy1:           djnz r0,tdy1
                djnz r1,tdy2
                djnz r2,tdy3
                ret

waitdelay:
                mov r2,#09h
ty3:            mov r1,#00h
ty2:            mov r0,#00h
ty1:            djnz r0,ty1
                djnz r1,ty2
                djnz r2,ty3
                ret

;*********************************

lcddelay:
        mov r2,#01h
dly3:   mov r1,#0fh
dly2:   mov r0,#10h
dly1:   djnz r0,dly1
        djnz r1,dly2
        djnz r2,dly3
        ret
```

2. HEART BEAT CODING:

```
tb0      equ 0a0h
tb1      equ 0a1h
tb2      equ 0a2h
tb3      equ 0a3h
te       equ 0a4h


hb_sense    equ 0b3h
hb_count0   equ 6bh
hb_count1   equ 6ch



         org 0000h
             mov p2,#0ffh
             mov p1,#0ffh
             mov p0,#0ffh
             mov p3,#0ffh
         mov hb_count0,#00h
         mov hb_count1,#00h

hold:
         mov tmod,#10h
         mov r3,#25
         mov hb_count0,#00h
         mov hb_count1,#00h

again:
         mov tl1,#08h
         mov th1,#01h
         setb tr1
back:    jnb hb_sense,chk1
             acall hbcount
chk1: jnb tf1,back
             clr tr1
             clr tf1
             djnz r3,again

             mov a,#0cdh
             acall command
             mov a,hb_count1
             add a,#30h
             mov hb_count1,a
             acall display
             mov a,hb_count0
             add a,#30h
             mov hb_count0,a
             acall display
```

```
            mov a,#0ah
            acall senddata
            mov a,hb_count1
            acall senddata
            mov a,hb_count0
            acall senddata
            sjmp hold


command:

            ret

display:
            ret


senddata:
            anl a,#0fh
esc:  rrc a
          mov tb0,c
          rrc a
          mov tb1,c
          rrc a
          mov tb2,c
          rrc a
          mov tb3,c
          clr te
          acall txdelay
          setb te
          acall waitdelay
        ret


hbcount:
        mov a,hb_count0
        add a,#01h
        mov hb_count0,a
        cjne a,#0ah,escw1
        mov hb_count0,#00h
        mov a,hb_count1
        inc a
        mov hb_count1,a
        cjne a,#0ah,escw1
escw1:      acall longdelay
;           jb hb_sense,escw1
            ret


txdelay:
                mov r2,#03h
tldy3:          mov r1,#00h
tldy2:          mov r0,#00h
```