COLLISION PREVENTION AND GATE SECURITY IN RAILWAY

A PROJECT REPORT

Submitted by

P.MALATHI -71201105022

V.RAJESWARI -71201105040

V.T.VADIVUKKARASI -71201105066

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

ELECTRICAL AND ELECTRONICS ENGINEERING

Under the guidance of

Mr.C.UdhayaShankar

KUMARAGURU COLLEGE OF TECHNOLOGY, COIMBATORE

ANNA UNIVERSITY: CHENNAI 600 025

APRIL 2005

ANNA UNIVERSITY : CHENNAI-600 025

BONAFIDE CERTIFICATE

Certified that this project report titled **"COLLISION PREVENTION AND GATE SECURITY IN RAILWAY"** is the bonafide work of

P.MALATHI	-	**Register No. 71201105022**
V.RAJESWARI	-	**Register No. 71201105040**
V.T.VADIVUKKARASI	-	**Register No. 71201105066**

who carried out the project work under my supervision.

8/4/05

Signature of the Head of the Department

8/4

Signature of the guide

Prof.K.Regupathy Subramanian, B.E., M.Sc.
DEAN/EEE,
Kumaraguru college
of technology.

Mr.C.UdhayaShankar, M.Tech.
Lecturer, EEE,
Kumaraguru college
of technology.

CERTIFICATE OF EVALUATION

College : KUMARAGURU COLLEGE OF TECHNOLOGY

Branch : Electrical And Electronics Engineering

Semester : Eighth semester

Sl.No.	Name of the students	Title of the project	Name of the supervisor with designation
01	P.MALATHI	**"Collision prevention and Gate security in Railway"**	Mr.C.UDHAYASHANKAR LECTURER.
02	V.RAJESWARI		
03	V.T.VADIVUKKARASI		

The report of the project work submitted by the above students in partial fulfillment for the award of Bachelor of Engineering degree in Electrical and Electronics Engineering of Anna University were evaluated and confirmed to be report of the work done by the above students and then evaluated.

(INTERNAL EXAMINER) **(EXTERNAL EXAMINER)**

DEDICATED

TO OUR

BELOVED PARENTS

AND

WELL WISHERS

ACKNOWLEDGEMENT

The successful completion of our project can be attributed to the combined efforts made by us and the contribution made in one form or the other, by the individuals we hereby acknowledge.

We are highly privileged to thank **Dr.K.K.Padmanabhan, B.Sc., (Engg)., M.Tech., Ph.D,, MISTE** and **F.I.E** principal, Kumaraguru College of Technology for providing the necessary facilities for successful completion of the project.

We express our heartfelt gratitude and thanks to the Head of Electrical and Electronics Department, **Dr.T.M.Kameswaran, B.E., M.Sc (Engg)., Ph.D., F.I.E.,** for encouraging us to choose this project and for being with us right from the beginning of the project and guiding us at every step.

We wish to express our deep sense of gratitude and indebtedness to the Dean, Electrical and Electronics Department, **Prof.K.RegupathySubramaniam, B.E (Hons)., M.Sc.,** for his enthusiasm and encouragement, which has been instrumental in the success of the project.

We wish to place on record our deep sense of gratitude and profound thanks to our guide **Mr.C.UdhayaShankar, M.Tech.,** Lecturer, Electrical and Electronics department, for his valuable guidance, constant encouragement, continuous support and co-operation rendered throughout the project.

We are also thankful to all teaching and non teaching staffs of Electrical and Electronics engineering department for their kind help and encouragement in making our project successful.

Last but not least we extend our sincere thanks to all our parents and friends who have contributed their ideas and encouraged us for completing the project successfully.

ABSTRACT

The main objective of our project is to avoid the number of railway accidents occurring in the unmanned level crossings by providing necessary and detailed information regarding the train arrival and its current position to the public at the gate and also to prevent the direct collision occurring between the two trains traveling in the same track.

The gate control part is done by glowing appropriate LEDs relevant to the train position and switching ON the buzzer at appropriate time along with automated gate closing. The information regarding the time that will be taken for the train to reach the gate and its velocity will be displayed at the gate using LCD display continuously. The opening and closing of the gate will be controlled using servomotor operated by relays.

The collision prevention part is done by transmitting and receiving the train and track number information within a particular area, so that if any train comes in the same track, further enhancements can be done either to stop the train or to change the track. The microcontroller used here is ATMEL 89C51.

TABLE OF CONTENTS

List of tables

List of figures

CHAPTER 1

INTRODUCTION:

1.1.A strategy of railway accidents in India:

Accident is an occurrence in the course of working of Railway which does or may affect the safety of the Railway passengers or servants or which affect the safety of others, or which does or may cause delay to a train or loss to the Railway. The term `**accident**' envelopes a wide spectrum of occurrences or consequences not necessarily leading to a mishap. `Failures of railway equipment' are also treated as technical and potential `accidents' for the purposes of managing the assets safely. Asset failures are continuously monitored and efforts made to oversee that they do not cause actual accidents. Consequential train accidents include train accidents having **serious** repercussion in terms of loss of human life, injury, damage to Railway property or interruption to rail traffic of laid down threshold levels and values. These consequential train accidents include ***Collision, Derailments, Fire in Trains, Collisions of trains at Level Crossings and few miscellaneous incidents.*** All other train accidents, which are below the threshold values, are treated as "other train accidents". Such categorization is, broadly, in consonance with practices adopted in many of world railways, though varying in details and degree. **Indicative Accidents,** distinct from Consequential Train Accidents include all cases of `train passing signal at danger', `averted collision', breach of block rules etc.

Collisions are the most dreaded accidents for any railway-man. These can be `side collisions', `Rear-end' and `head-on collisions'.

Trains ramming into another from behind are called rear-end collisions, while trains colliding on the same track from opposite ends, are called head-on collisions and are the most fatal of all accidents. Side collisions can occur either in station area, while converging or diverging or by fouling the adjacent track in multiple lines territory. Rear-end collisions and head-on collisions can occur at stations or between the stations. Of the total consequential train accidents, that occurred during the last decade, the percentage of collisions involving passenger carrying trains was 4% only, but they are highly volatile mishaps and call for necessary steps to prevent them at all costs.

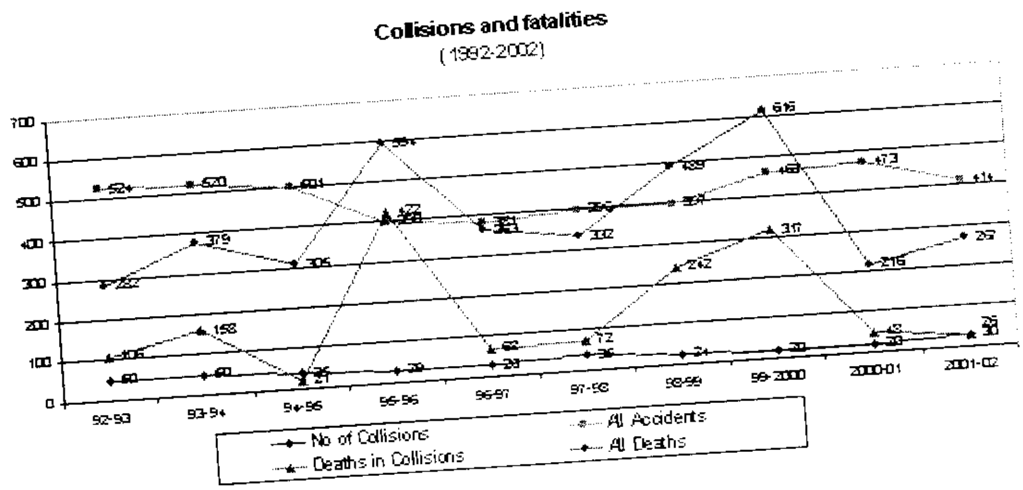


Fig. 1.1

Of the total consequential train accidents, that occurred during the last decade, the accidents at ***level crossings*** were about 16%. Accidents at manned level crossings were at the level of 4% of the total consequential train accidents, whereas unmanned level crossings accidents shared 12% of mishaps.

Accidents at unmanned level crossings occur primarily due to dashing of road vehicles with the oncoming trains and cause fatality of the road user. A road vehicle driver, though having the advantage of maneuverability and lesser braking distances and shorter reaction time as compared to train drivers, fails to maintain the level of alertness, normally expected, while crossing such intersections, where he is supposed to take necessary precautions as stipulated in the Motor Vehicles Act 1988. It has been observed that over 85% of all accidents occurring at unmanned level crossings, involved passenger carrying trains, reflecting that the road vehicle drivers normally misjudge the speed of the oncoming trains and take chances while crossing the rail track.

On an average, every year, 141 persons died and 158 injured in the accidents occurring at unmanned level crossings only during the last decade, contributing a share of 37% of the total fatalities in all accidents. 9% of the total fatalities occurred in manned level crossing accidents, thus indicating that **46% of train accident fatalities take place at rail-road intersections** only.

1.2.NEED FOR OUR PROJECT:

Whether it is systems failure or human error that causes accidents, experts believe a majority of them could have been avoided if proper signaling equipment was installed. Trains are recklessly added, but the corresponding investment in technology and equipment, which will automatically ensure safety, is disregarded. A growing network, populist demands and poor funds have given the Indian Railways a bloody track record in the past decade. Since 1986, railway ministry reveal more than 5,000 people have died in train accidents across the country.

India has the second-largest rail network in the world and it transports the largest number of passengers – four billion people – annually. Rail experts argue that accidents will inevitably occur in such a mammoth system.

"While sophisticated communication and automation systems have considerably reduced the number of accidents in the West, there has been a steep increase in the number of accidents in India," says Rajmani Singh who retired from the Railway Board last year.

Thus from the data as seen above tells us the importance of safety measures to be adopted in railways. If that is automated, then it becomes easier to control the disasters. The size and complexity of the previous operations, growing traffic and changing technologies, placed inevitably a heavy burden on manual information system. So need for modernization in such systems is important now a days.

In **our project**, the *time* taken for the train to cover the *distance* that is already known between two keys is sensed using the timers. From the distance and the time known, the *velocity of the train* can be calculated. Thus the time that will be taken for the train to reach the gate is known approximately and the status of the train can be displayed at the gate location. This is a part of the project. And the other is that the *transmitter* kept at one train transmits the train information to the *receiver* kept at the other train if traveling in the same track at a reasonable distance. This will be useful to stop the trains by some mechanism or the other. Microcontroller employed for this purpose is 89C51.

CHAPTER 2

PROJECT DESCRIPTION:

2.1.GENERAL BLOCK DIAGRAM

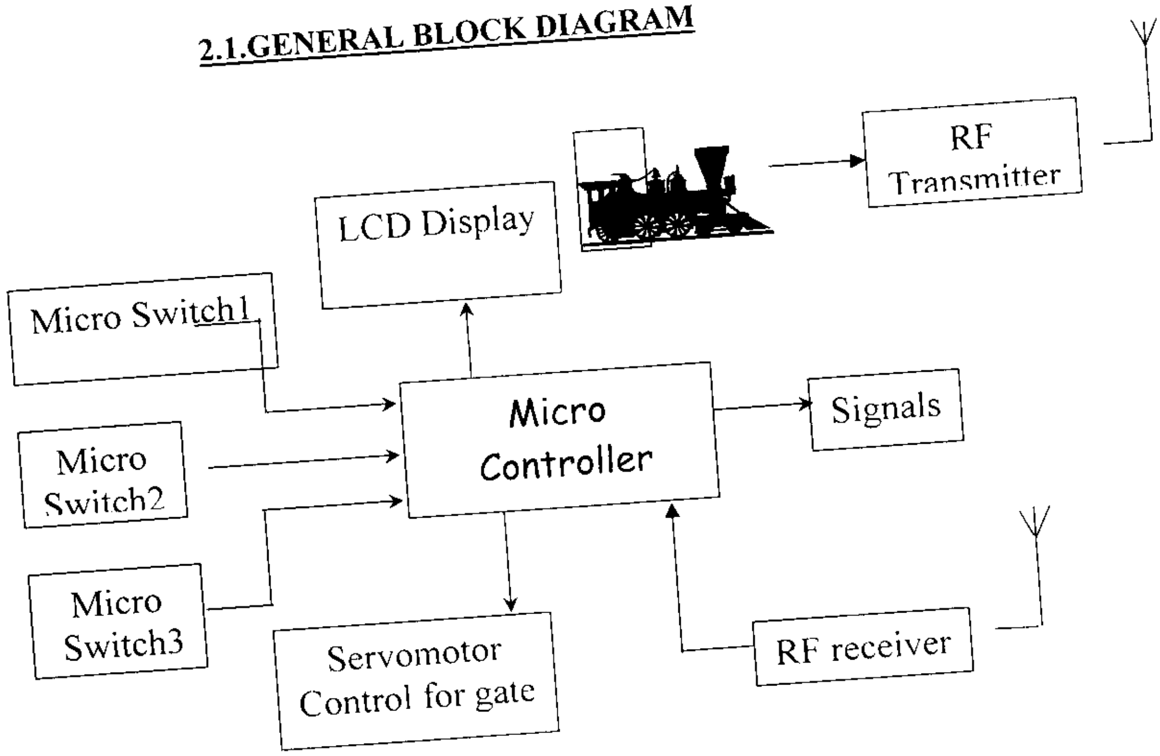


Fig 2.1

2.2.INTRODUCTION TO MICROCONTROLLER

INTRODUCTION:

A Micro controller consists of a powerful CPU tightly coupled with memory RAM, ROM or EPROM, various I / O features such as Serial ports, Parallel Ports, Timer/Counters, Interrupt Controller, Data Acquisition interfaces-Analog to Digital Converter (ADC), Digital to Analog Converter (DAC), everything integrated onto a single Silicon Chip.

Any microcomputer system requires memory to store a sequence of instructions making up a program, parallel port or serial port for communicating with an external system, timer / counter for control purposes like generating time delays, Baud rate for the serial port, apart from the controlling unit called the **Central Processing Unit**

ADVANTAGES OF MICROCONTROLLERS:

1. If a system is developed with a microprocessor, the designer has to go for external memory such as RAM, ROM or EPROM and peripherals and hence the size of the PCB will be large enough to hold all the required peripherals. But, the micro controller has got all these peripheral facilities on a single chip so development of a similar system with a micro controller reduces PCB size and cost of the design.

   One of the major differences between a micro controller and a microprocessor is that a controller often deals with bits, not bytes as in the real world application, for example switch contacts can only be open or close, indicators should be lit or dark and motors can be either turned on or off and so forth.

INTRODUCTION TO ATMEL MICROCONTROLLER

SERIES: 89C51 Family

TECHNOLOGY: CMOS

The major Features of 8-bit Micro controller **ATMEL 89C51**:

- 8 Bit CPU optimized for control applications
- Extensive Boolean processing (Single - bit Logic) Capabilities.
- On - Chip Flash Program Memory
- On - Chip Data RAM
- Bi-directional and Individually Addressable I/O Lines
- Multiple 16-Bit Timer/Counters
- Full Duplex UART
- Multiple Source / Vector / Priority Interrupt Structure
- On - Chip Oscillator and Clock circuitry.
- On - Chip EEPROM
- SPI Serial Bus Interface
- Watch Dog Timer

MEMORY ORGANIZATION:

* Logical Separation of Program and Data Memory *

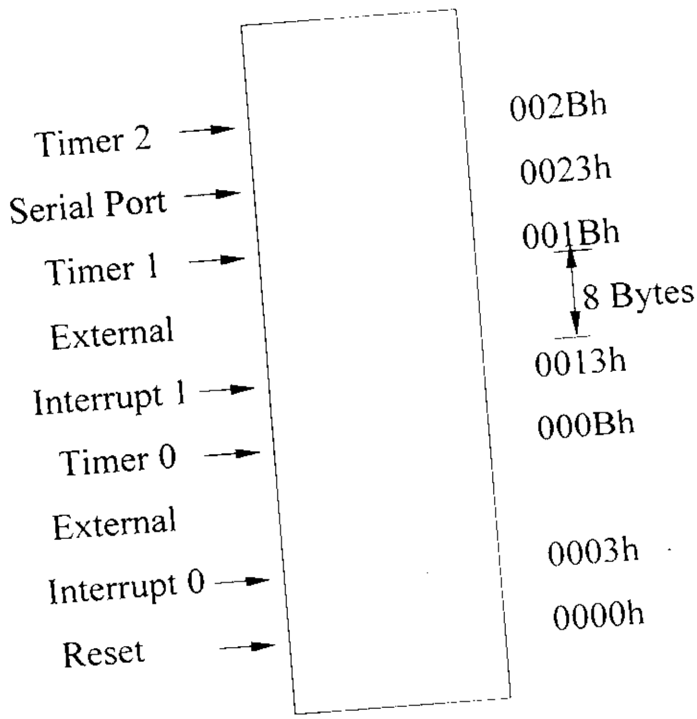
All Atmel Flash micro controllers have separate address spaces for program and data memory.The logical separation of program and data memory allows the data memory to be accessed by 8 bit addresses. Which can be more quickly stored and manipulated by an 8 bit CPU Nevertheless

16 Bit data memory addresses can also be generated through the DPTR register.

PROGRAM MEMORY:

Fig shows the map of the lower part of the program memory, after reset, the CPU begins execution from location 0000h. As shown in Fig 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. External Interrupt 0 for example, is assigned to location 0003h. If external Interrupt 0 is used, its service routine must begin at location 0003h. If the interrupt in 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.



Longer service routines can use a jump instruction to skip over subsequent interrupt locations. If other interrupts are in use. 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. For example, in the AT89C51 with 4K bytes of on-chip Flash, if the EA pin is strapped to Vcc, program fetches to addresses 0000h through 0FFFh are directed to internal Flash. Program fetches to addresses 1000h through FFFFh are directed to external memory.

DATA MEMORY:

The Internal Data memory is dived into three blocks namely,

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

89C51 MEMORY STRUCTURE

ADDRESSING MODES:

DIRECT ADDRESSING:

In direct addressing, the operand specified by an 8-bit address field in the instruction. Only internal data RAM and SFR's can be directly addressed.

INDIRECT ADDRESSING:

In Indirect addressing, the instruction specifies a register that contains the address of the operand. Both internal and external RAM can indirectly address. The address register for 8-bit addresses can be either the Stack

Pointer or R0 or R1 of the selected register Bank. The address register for 16-bit addresses can be only the 16-bit data pointer register, DPTR.

INDEXED ADDRESSING:

Program memory can only be accessed via indexed addressing this addressing mode is intended for reading look-up tables in program memory. A 16 bit base register (Either DPTR or the Program Counter) points to the base of the table, and the accumulator is set up with the table entry number. Adding the Accumulator data to the base pointer forms the address of the table entry in program memory.

REGISTER INSTRUCTION:

The register banks, which contains registers R0 through R7, can be accessed by instructions whose opcodes carry a 3-bit register specification. Instructions that access the registers this way make efficient use of code, since this mode eliminates an address byte. When the instruction is executed, one of four banks is selected at execution time by the row bank select bits in PSW.

PROGRAM STATUS WORD:

Program Status Word Register in Atmel Flash Micro controller:

The Program Status Word contains Status bits that reflect the current status of the CPU. The PSW shown in Fig resides in SFR space. The PSW contains the Carry Bit, The auxiliary Carry (For BCD Operations) the two - register bank select bits, the Overflow flag, a Parity bit and two user Definable status Flags.

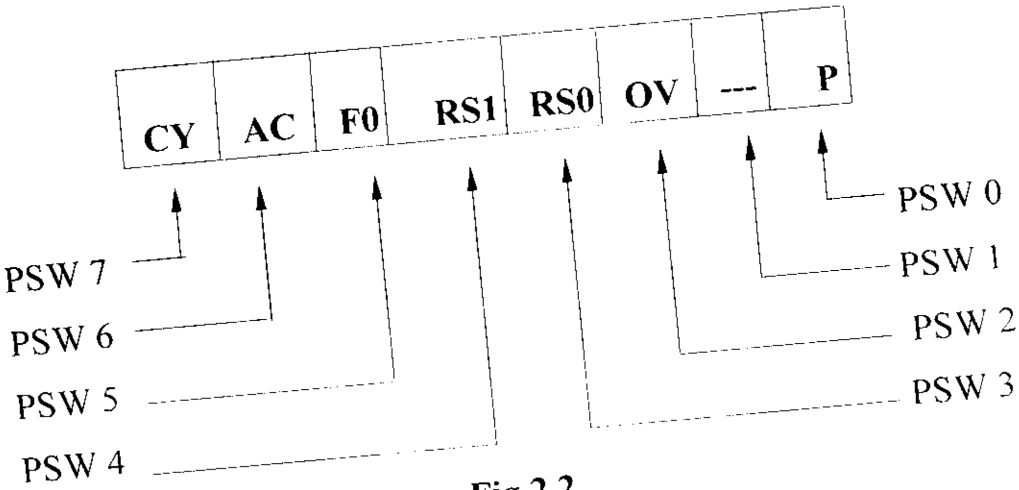


Fig 2.2

PSW 0: Parity of Accumulator Set By Hardware To 1 if it contains an Odd number of 1s, Otherwise it is reset to 0.

PSW1: User Definable Flag

PSW2: Overflow Flag Set By Arithmetic Operations

PSW3: Register Bank Select

PSW4: Register Bank Select

PSW5: General Purpose Flag.

PSW6: Auxiliary Carry Flag Receives Carry Out from Bit 1 of Addition Operands

PSW7: Carry Flag Receives Carry Out From Bit 1 of ALU Operands.

The Carry Bit, in addition to serving as a Carry bit in arithmetic operations also serves the as the "Accumulator" for a number of Boolean Operations .The bits RS0 and RS1 select one of the four register banks. A number of instructions register to these RAM locations as R0 through R7.The status of the RS0 and RS1 bits at execution time determines which of the four banks is selected.

The Parity bit reflect the Number of 1s in the Accumulator .P=1 if the Accumulator contains an even number of 1s, and P=0 if the Accumulator contains an even number of 1s. Thus, the number of 1s in the Accumulator plus P is always even. Two bits in the PSW are uncommitted and can be used as general-purpose status flags.

INTERRUPTS

The AT89C51 provides 5 interrupt sources: Two External interrupts, two-timer interrupts and a serial port interrupts. The External Interrupts INT0 and INT1 can each 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 to hardware clears the flag that generated an external interrupt *only* if the interrupt WA transition - activated. If the interrupt was level - activated, then the external requesting source (rather than the on-chip hardware) controls the requested flag. Tf0 and Tf1 generate the Timer 0 and Timer 1 Interrupts, which are set by a rollover in their respective Timer/Counter Register (except for Timer 0 in Mode 3). When a timer interrupt is generated, the on-chip hardware clears the flag that generated it when the service routine is vectored to. The logical OR of RI and TI generate the Serial Port Interrupt. Neither of these flag is cleared by hardware when the service routine is vectored to. In fact, the service routine normally must determine whether RI or TI generated the interrupt and the bit must be cleared in software.

IE: Interrupt Enable Register

EA	-	**ET2**	**ES**	**ET1**	**EX1**	**ET0**	**EX0**

Enable bit = 1 enabled the interrupt

Enable bit = 0 disables it.

Symbol	Position	Function
EA	IE.	Global enable / disable all interrupts. If EA = 0, no interrupt will be Acknowledge. If EA = 1, each interrupt source is individually enabled to deabled by setting or clearing its enable bit.
-	IE.6	Undefined / reserved
ET2	IE.5	Timer 2 Interrupt enable Bit
ES	IE.4	Serial Port Interrupt enabled bit.
ET1	IE.3	Timer 1 Interrupt enable bit.
EX1	IE.2	External Interrupt 1 enable bit.
ET0	IE.1	Timer 0 Interrupt enable bit.
EX0	IE.0	External Interrupt 0 enable bit.

Table 2.1

2.3.LCD DISPLAY

INTRODUCTION:

Short for *liquid crystal display*, a type of display used digital watches and many portable computers. LCD displays utilize two sheets of polarizing material with a liquid crystal solution between them. An electric current passed through the liquid causes the crystals to align so that light cannot pass through them. Each crystal, therefore, is like a shutter, either allowing light to pass through or blocking the light.

Monochrome LCD images usually appear as blue or dark gray images on top of a grayish-white background. Color LCD displays use two basic techniques for producing color: *Passive matrix* is the less expensive of the two technologies. The other technology, called *thin film transistor* (TFT) or *active-matrix*, produces color images that are as sharp as traditional CRT displays, but the technology is expensive.

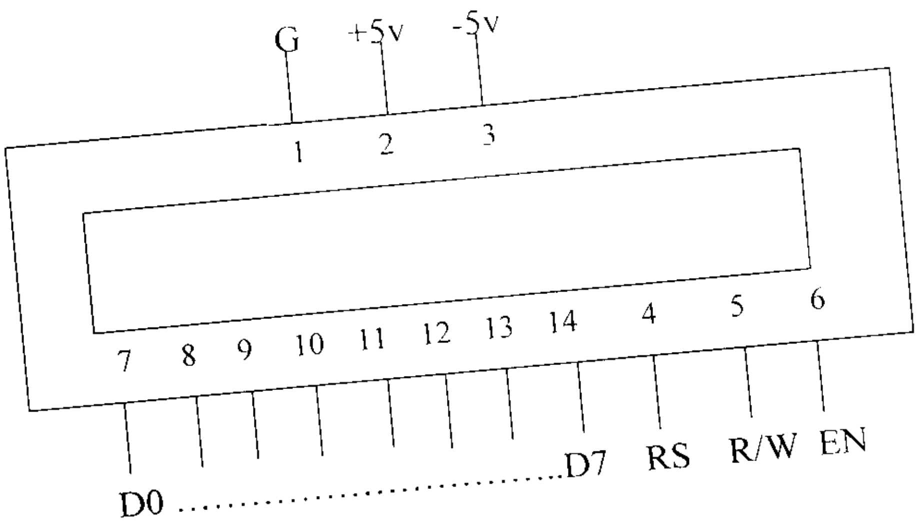
When the LCD is in the off state, light rays are rotated by the two polarisers and the liquid crystal, such that the light rays come out of the LCD without any orientation, and hence the LCD appears transparent. When sufficient voltage is applied to the electrodes, the liquid crystal molecules would be aligned in a specific direction. The light rays passing through the LCD would be rotated by the polarisers, which would result in activating / highlighting the desired characters.

The LCD's are lightweight with only a few millimeters thickness. Since the LCD's consume less power, they are compatible with low power electronic circuits, and can be powered for long durations.

The power supply should be of +5V, with maximum allowable transients of 10mv. To achieve a better / suitable contrast for the display, the voltage (VL) at pin 3 should be adjusted properly.

The ground terminal of the power supply must be isolated properly so that no voltage is induced in it. The module should be isolated from the other circuits, so that stray voltages are not induced, which could cause a flickering display.

The LCD variables en, rw and rs are interfaced to the pins 5,6 and 7 pins of the port 2. LCD initialization is done by sending corresponding hexadecimal values .For the display to appear in the first line, 0x80 is to be sent and for second line, 0xc0 is sent. The location at which the display has to appear is to be sent appropriately.



RS – reset

R/W – read/ write

EN - enable

2.4.SERVO MOTOR

The motors used for the automatic control systems are called Servomotors. They are used to convert an electrical signal applied to them into an angular displacement of the shaft. They can either operate in a continuous duty or step duty depending on the construction.

In general, a servomotor should have the following features,

- Linear relationship between the speed and the electric control signal.
- Steady state stability
- Wide range of speed control
- Linearity of mechanical characteristics throughout the entire speed range.
- Low mechanical and electrical inertia and fast response.

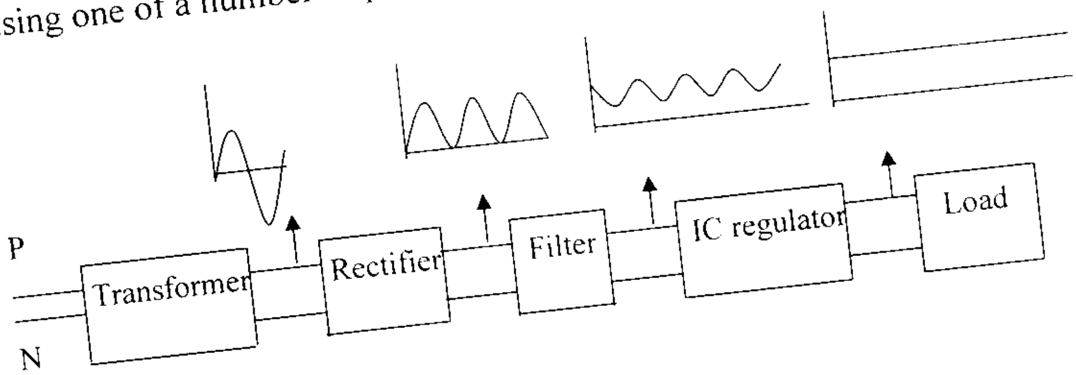
There are varieties of servomotors available for control system applications. The suitability of a motor for a particular application depends on the characteristics of the system, the purpose of the system and its operating conditions.

Here, the servomotor is used for the gate control mechanism, which is to open and close the gate. The ON/OFF and the directions of the motor are controlled using two relays respectively. After the train crosses the second switch, the red LED and the buzzer is ON to indicate the train arrival and the motor is made to run in forward direction to close the gate. After the train crosses the third switch and green LED is ON, the motor is made to run in the reverse direction to close the gate.

2.5.CIRCUIT EXPLANATION

2.5.1.POWER SUPPLY CIRCUIT:

A block diagram containing the parts of a typical power supply and the voltage at various points in the unit is shown. The ac voltage, typically 120 V rms, is connected to a transformer, which steps that ac voltage down to the level for the desired dc output. A diode rectifier then provides a full-wave rectified voltage that is initially filtered by a simple capacitor filter to produce a dc voltage. This resulting dc voltage usually has some ripple or ac voltage variation. A regulator circuit can use this dc input to provide a dc voltage that not only has much less ripple voltage but also remains the same dc value even if the input dc voltage varies somewhat, or the load connected to the output dc voltage changes. This voltage regulation is usually obtained using one of a number of popular voltage regulator IC units.



IC VOLTAGE REGULATORS:

Voltage regulators comprise a class of widely used ICs. Regulator IC units contain the circuitry for reference source, comparator amplifier, control device, and overload protection all in a single IC. IC units provide regulation of either a fixed positive voltage, a fixed negative voltage.

A power supply can be built using a transformer connected to the ac supply line to step the ac voltage to a desired amplitude, then rectifying that ac voltage, filtering with a capacitor and RC filter, if desired, and finally regulating the dc voltage using an IC regulator. The regulators can be selected for operation with load currents from hundreds of milliamperes to tens of amperes, corresponding to power ratings from mw to tens of watts.

5V POWER SUPPLY

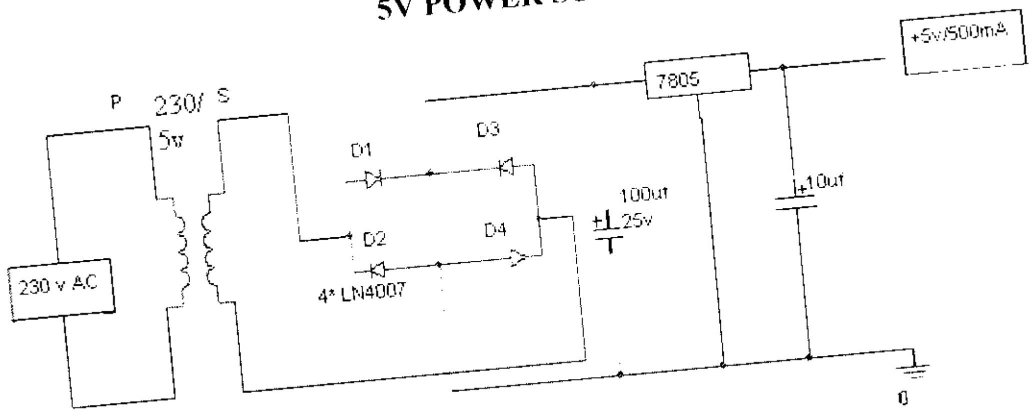


Fig 2.3

THREE-TERMINAL VOLTAGE REGULATORS:

The following Fig shows the basic connection of a three-terminal voltage regulator IC to a load. The fixed voltage regulator has an unregulated dc input voltage, Vi, applied to one input terminal, a regulated output dc voltage, Vo, from a second terminal, with the third terminal connected to ground. For a selected regulator, IC device specifications list a voltage range over which the input voltage can vary to maintain a regulated output voltage over a range of load current.

Fixed Positive Voltage Regulators:

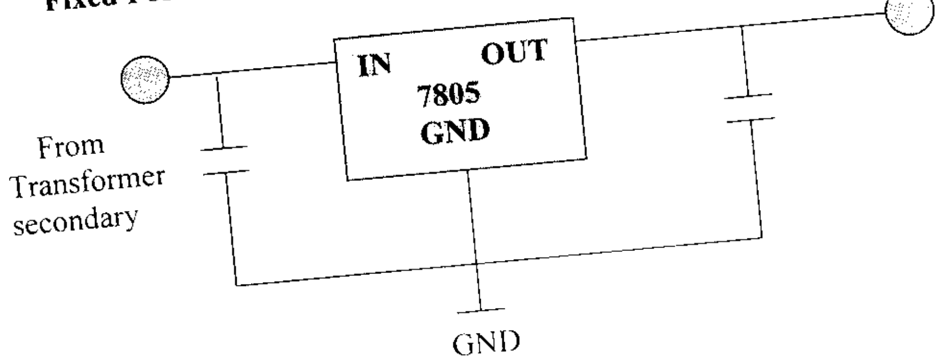


Fig 2.4

Positive Voltage Regulators in 7800 series

IC Part	Output Voltage (V)	Minimum Vi (V)
7805	+5	7.3
7806	+6	8.3
7808	+8	10.5
7810	+10	12.5
7812	+12	14.6
7815	+15	17.7
7818	+18	21.0
7824	+24	27.1

Table 2.2

The series 78 regulators provide fixed regulated voltages from 5 to 24 V. Figure shows how one such IC, a 7812, is connected to provide voltage regulation with output from this unit of +12V dc. An unregulated input voltage Vi is filtered by capacitor C1 and connected to the IC's IN terminal. The IC's OUT terminal provides a regulated + 12V which is filtered by capacitor C2 (mostly for any high-frequency noise). The third IC terminal is connected to ground (GND).

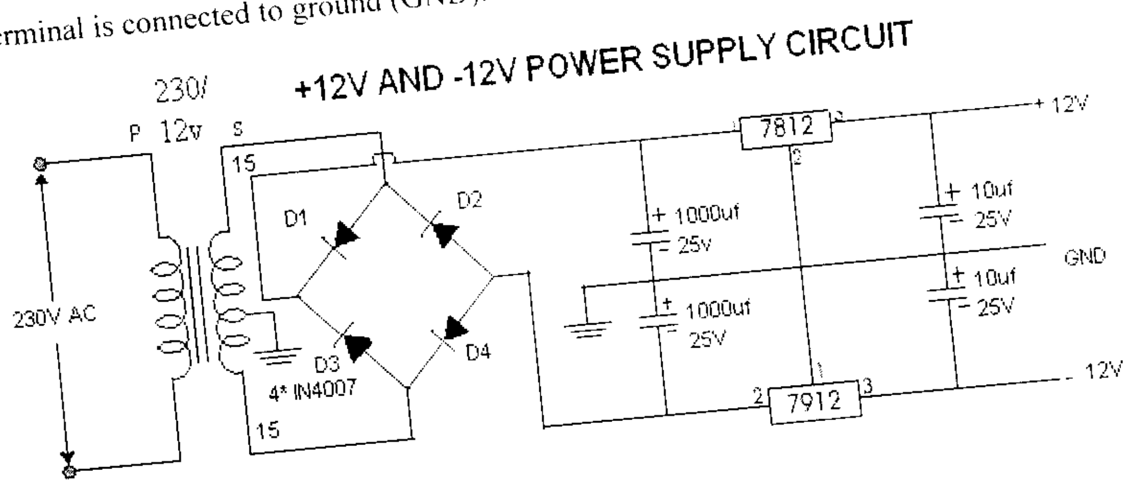


Fig 2.5

2.5.2.RELAY CIRCUIT:

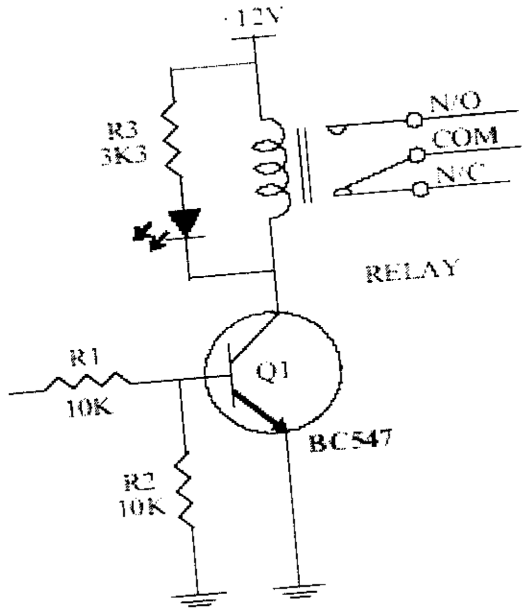


Fig 2.6

In this circuit transistor BC547 is used as a switch. The control signal is given to the base terminal of the transistor. The collector is attached to the relay coil. Relays are electromechanical devices. There are two types of relays.

1. Normally closed
2. Normally opened

We are using normally opened type relay. When the controller output is high the transistor will be in the ON state, so relay is energized. When the controller output from is low the transistor will be in the OFF state, so relay is de-energized. So according to the controller output the relay can be switched ON or OFF, thus giving the required output.

In case of relays used for servo motor control, relay1 is used for ON or OFF of the motor and relay2 is used for forward or reverse rotation of the motor.

2.5.3.KEY PAD

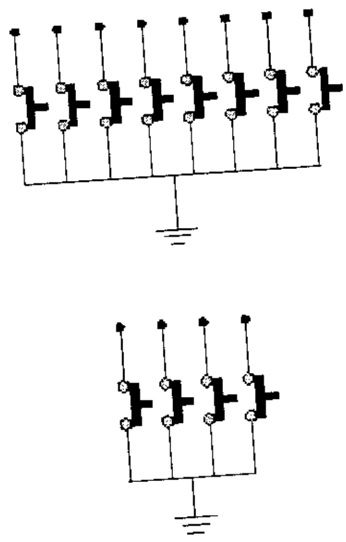


Fig 2.7

The keypad is used in the collision prevention part to enter the train and the track number accordingly and is interfaced with the micro controller. Four keys are placed among which two are used for entering the train number and the other two for entering the track number. Keys 1 and 2 are programmed to enter the train number and 3 and 4 are used for entering the track number. Thus according to the requirement the number of keys can be varied and we can obtain several combinations out of them.

In case of gate control, in the place of sensors, keys are used here at positions 1,2 and 3.The first two keys are used for calculating and displaying the time taken by the train to reach the gate and the train velocity and to close the gate. The third key is used to open the gate after the train leaves the gate.

ASK modulation

There are three ways in which the bandwidth of the channel carrier may be altered simply. These techniques give rise to **amplitude-shift-keying (ASK), frequency-shift-keying (FSK)** and **phase-shift-keying (PSK)**, respectively and **ASK is used here.**

ASK describes the technique the carrier wave is multiplied by the digital signal f(t). Mathematically, the modulated carrier signal δ(t) is:

$$\delta(t) = f(t) \sin(2\Pi\ fc\ t + \varphi) \quad , \quad fc = 433.93MHz.$$

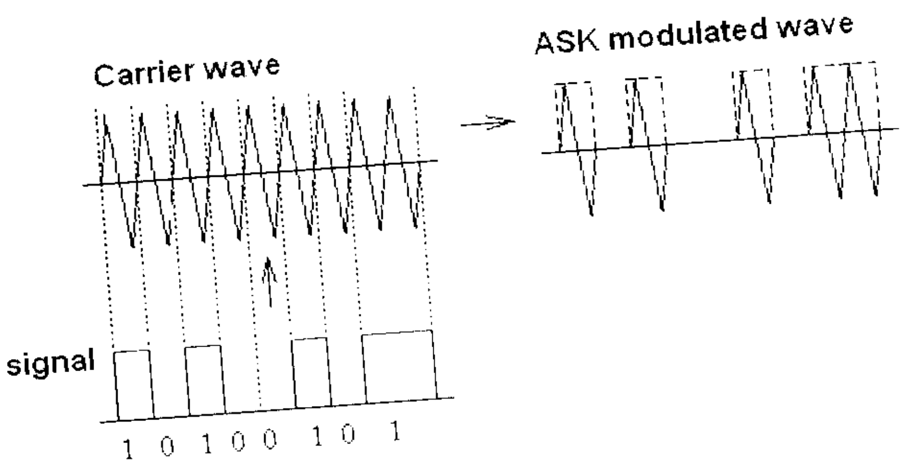


Fig 2.8

Amplitude modulation has the property of translating the spectrum of the modulation f(t) to the carrier frequency. The bandwidth of the signal remains unchanged. The fact that AM simply shifts the signal spectrum is often used to convert the carrier frequency to a more suitable value without altering the modulation. This process is known variously as **mixing, up-conversion** or **down-conversion**. Some form of conversion will always be present when the channel carrier occupies a frequency range outside the modulation frequency range.

2.5.4.RF TRANSMITTER CIRCUIT

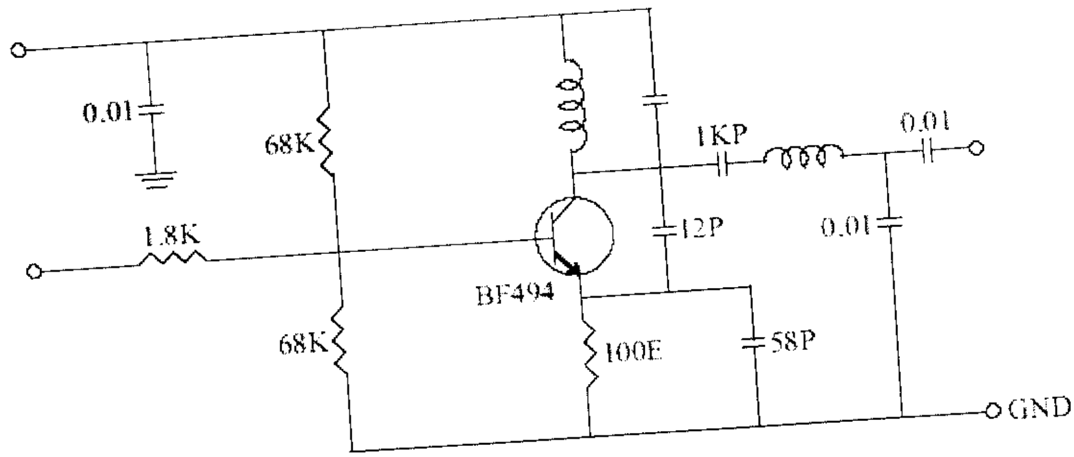


Fig 2.9

The RF Transmitter works on the concept of ASK modulation.

- It is driven by a single 9V supply from a battery
- The carrier frequency is 433.9 MHz
- It uses a common emitter amplifier biased with a voltage divider circuit
- The parallel circuit components of 0.1mH inductor and 0.01uF capacitor make up the tank circuit
- The 0.01uF capacitor at the signal entry point is an bypass

2.5.5.RF RECEIVER CIRCUIT

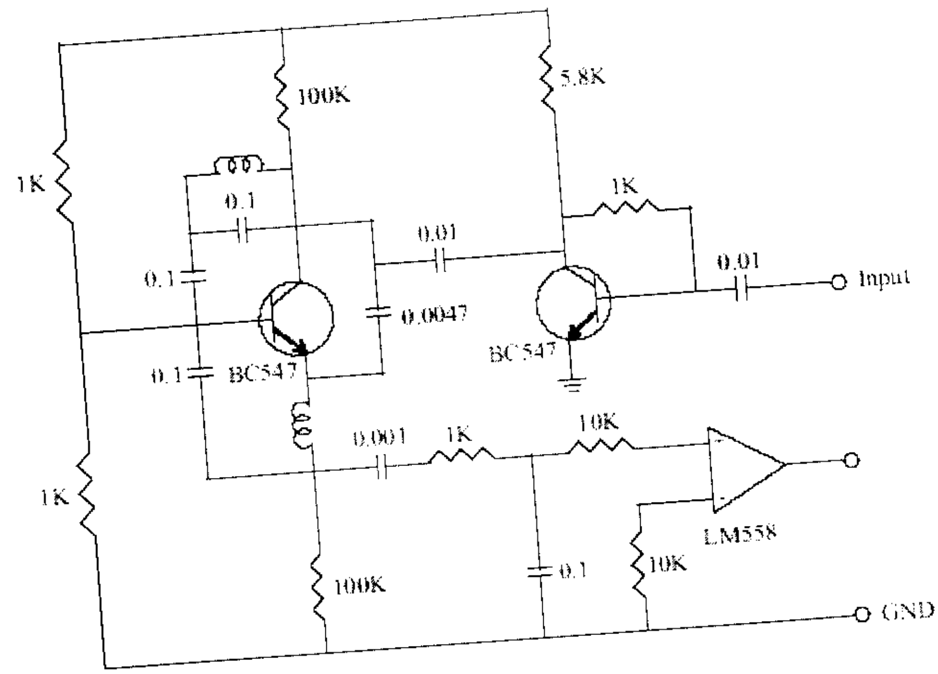


Fig 2.10

- The circuit consists of a preamplifier that serves two purposes, to give reasonably well-defined input impedance and provide isolation between the oscillations of the super regenerator and the antenna, reducing unwanted radiation.
- The super-regenerator input is applied to the tank circuit via a small value capacitor to minimize the loading of the circuit.
- The non-linearity of the device when operated in this mode also detects the AM signals and the output is taken at the biasing resistor and is low pass filtered using an RC network.

2.6. MAIN CIRCUITS

2.6.1.TRANSMITTER CIRCUIT

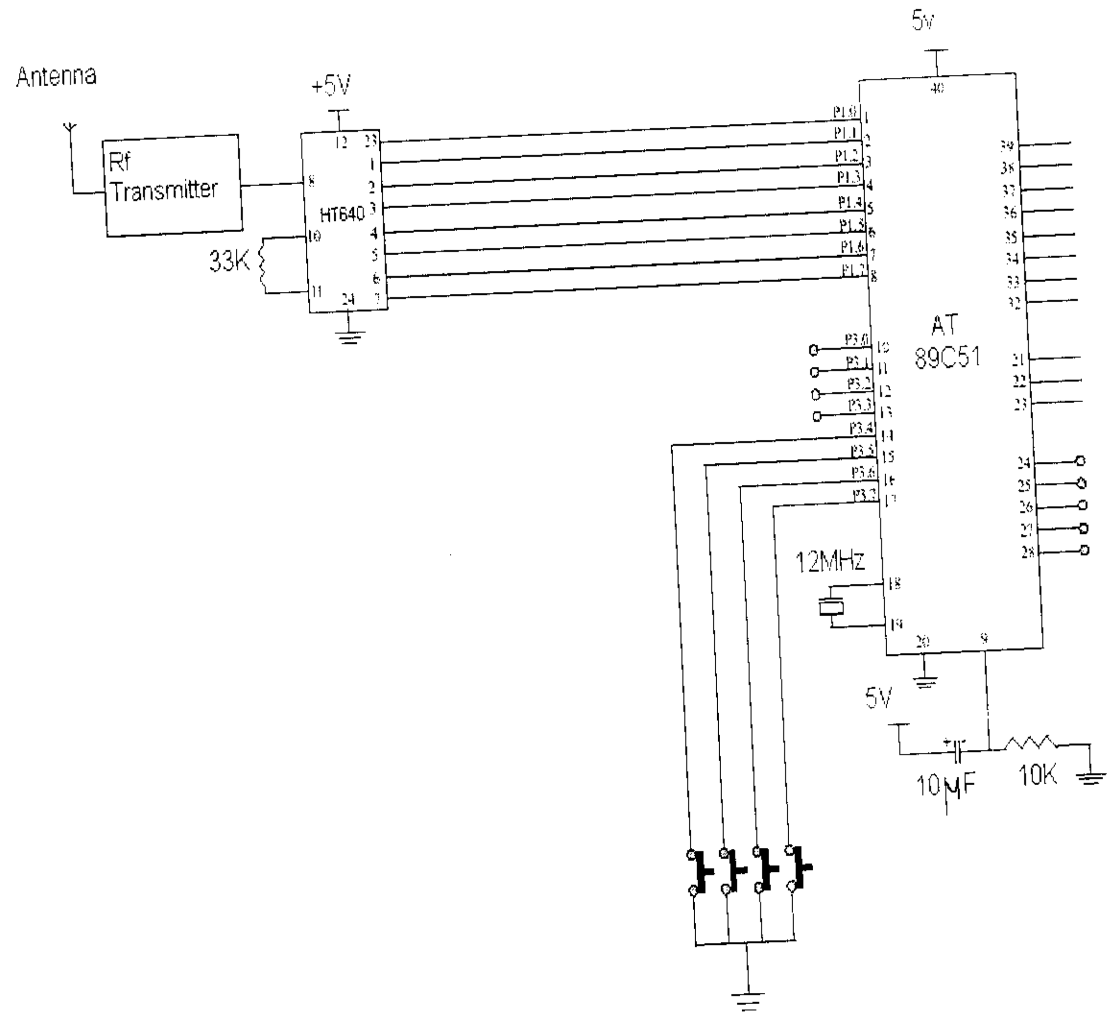


Fig 2.11

2.6.2 RECEIVER CIRCUIT

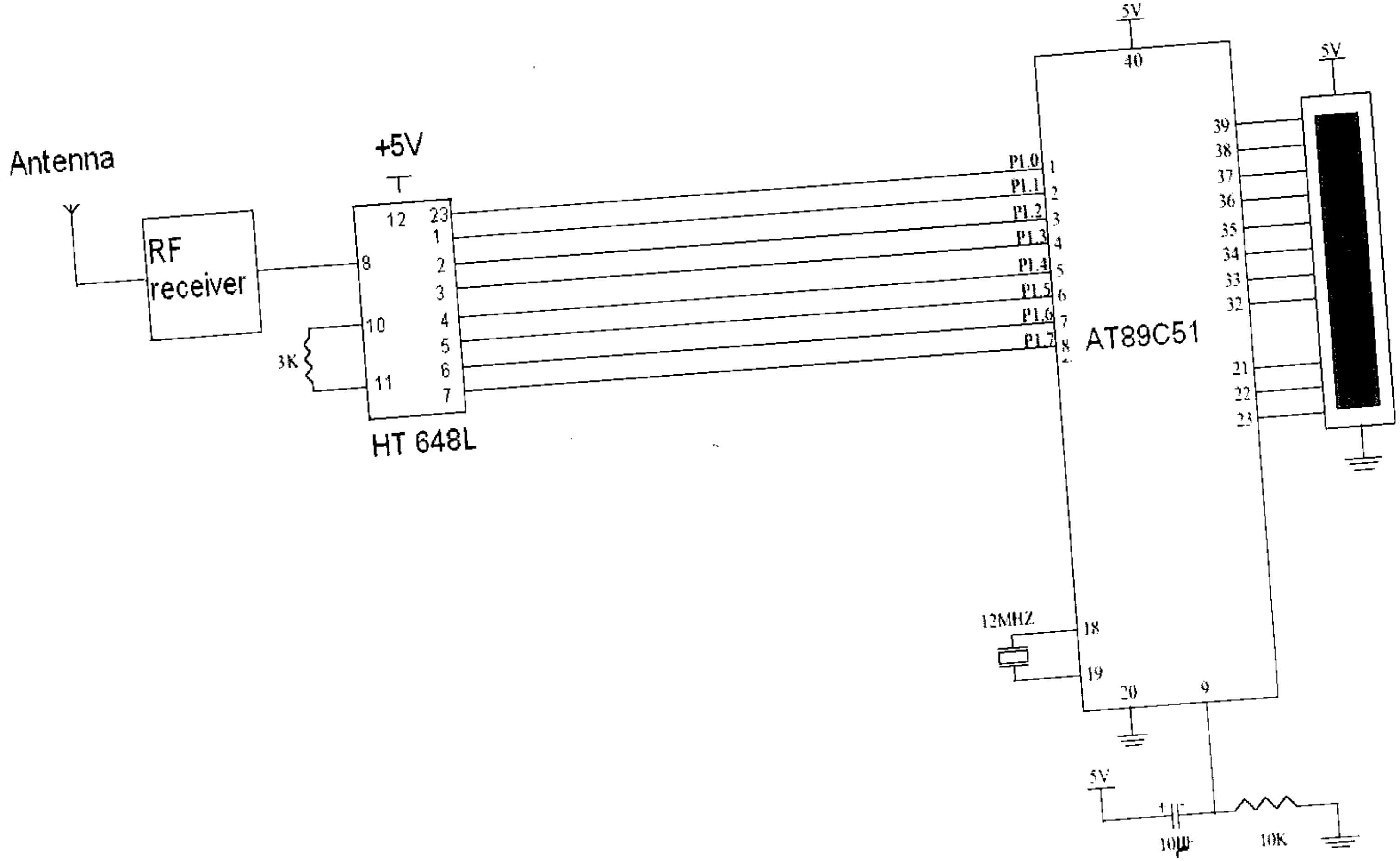


Fig 2.12

EXPLANATION:

TRANSMITTER:

- There are four keys, two for entering the train number and the other two for entering the track number as explained earlier.
- When the train number and the track number are entered, the micro controller sends the entered number to the encoder.i.e., HT640, so that the parallel data is converted to serial for transmission.
- Then the serial data is transmitted through the rf transmitter which is explained earlier.
- The keys 1,2,3 and 4 are connected to the pins 14,15,16 and 17 of the micro controller respectively.
- Two keys are provided with a combination of numbers, which represents the train number.
- The other two keys are programmed with a single digit number, which is the track number.
- When keys 1 and 3 are pressed, then a train number and a track number are transmitted and when keys 1 and 4 are pressed, then the same train number with a different track number is sent.
- similarly for the combinations of key 2 with 3 and 4.

RECEIVER:

- The transmitted train number and the track number are received through the rf receiver and the data is sent to the decoder i.e., HT648L.
- The serial data is converted to parallel again and then given to the micro controller,with the help of the decoder HT648L.
- The data given to the micro controller is then sent to the LCD display where the entered train number and the track number are displayed.

ENCODER AND DECODER

HT640 encoder:

The HT640 encoders are a series of CMOS LSIs for remote control system applications. They are capable of encoding 18 bits of information, which consists of N address bits, and 18 N data bits. Each address/data input is externally programmable if bonded out. It is otherwise set floating internally. The programmable address/ data is transmitted together with the header bits via an RF or an infrared transmission medium upon receipt of a trigger signal. The capability to select a TE trigger type or a DATA trigger type further enhances the application flexibility HT640 Encoder series of encoders.

FEATURES:

- Operating voltage: 2.4V~12V
- Low power and high noise immunity CMOS technology
- Low standby current
- Three words transmission
- Built-in oscillator needs only 5% resistor
- Easy interface with an RF or infrared transmission media
- Minimal external components

HT648L decoder:

The HT648L decoders are a series of CMOS LSIs for remote control system applications. They are paired with the 318 series of encoders. The HT648L decoder receives serial address and data from that series of encoders that are transmitted by a carrier using an RF or an IR transmission medium. It then compares the serial input data twice continuously with its local address. If no errors or unmatched codes are encountered, 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 HT648L decoders are capable of decoding 18 bits of information that consists of N bits of address and 18 N bits of data.

FEATURES:

- Operating voltage: 2.4V~12V
- Low power and high noise immunity CMOS technology
- Low standby current
- Capable of decoding 18 bits of information
- 8~18 address pins
- 0~8 data pins
- Trinary address setting
- Two times of receiving check
- Built-in oscillator needs only a 5% resistor
- Valid transmission indictor
- Easily interface with an RF or an infrared transmission medium
- Minimal external components

2.6.3.GATE CONTROL CIRCUIT

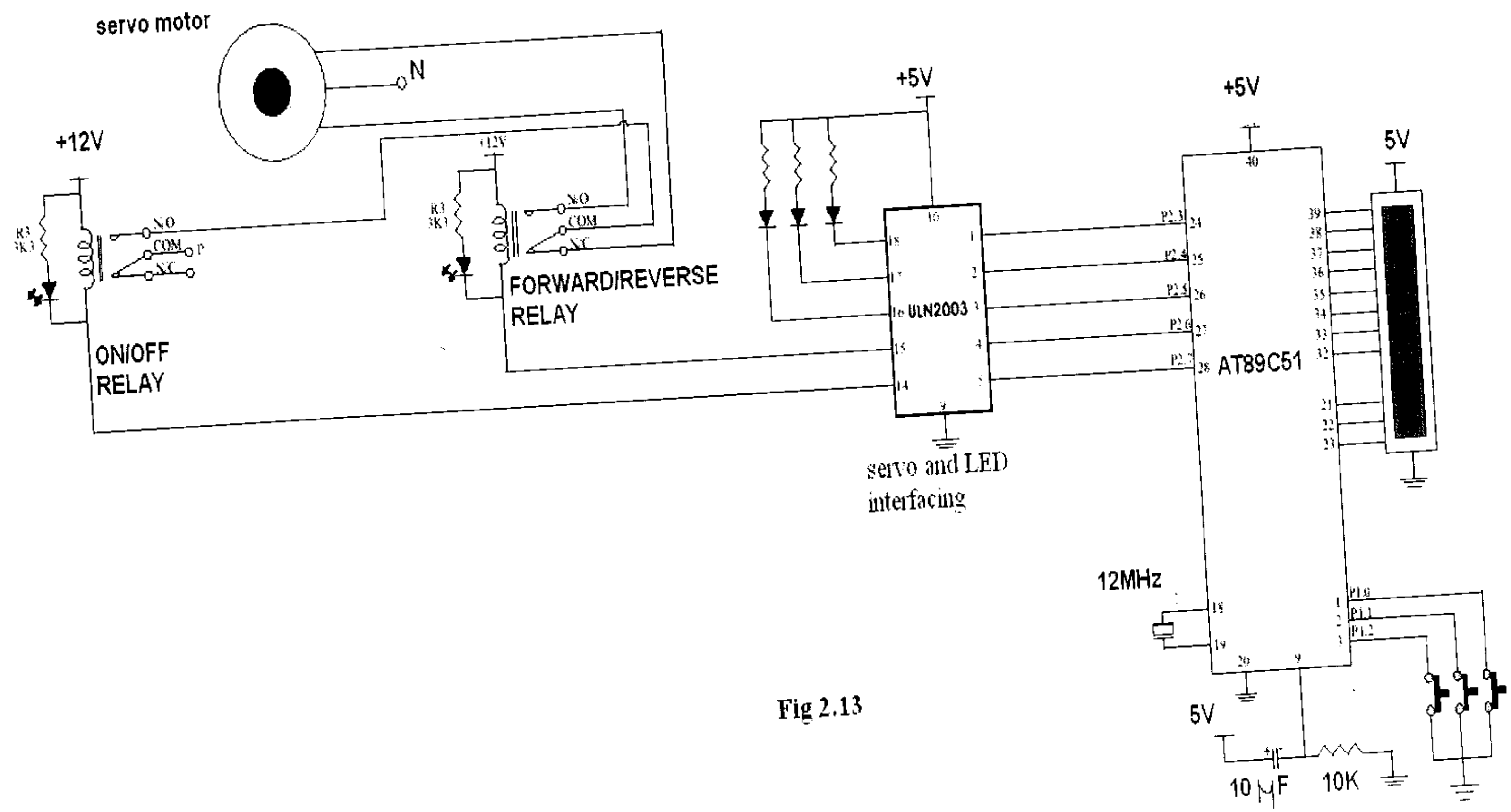


Fig 2.13

GATE CONTROL CIRCUIT:

The basic component used for gate control the keys which are provided at the places of the sensors i.e., sensors are provided when used practically. Three keys are employed here each connected to the micro controller. In this case, when the train reaches sensor1,the micro controller starts calculating the time till the train reaches the sensor2.As the distance between the two switches and also the time difference between the two are known, using these information, the micro controller is made to calculate the velocity of the train and the time taken by the train to reach the gate. These two are displayed using LCD display.

After the train leaves the sensor2, the calculated time is decremented such that, when the train reaches the gate, the display shows zero.

To alert the people that the gate is going to close, an alarm system is provided which produces a sound when there is a few seconds to gate closing. Opening and closing of the gate is done by using a servo motor which can be made to rotate in both forward and reverse direction easily through relays. Two relays are used here, one for ON and OFF of the motor and the other one for forward and reverse rotation of the motor.

Three LEDs are provided here, yellow, red and green to glow at corresponding time intervals i.e., when the train reaches the sensor1, sensor2 and when it leaves the sensor3 respectively.

The LEDs are connected to the micro controller through an interface, ULN2003.

2.7.SOFTWARE DESCRIPTION:

2.7.1.GATE CONTROL:

```
#include<reg51.h>
void scan_full();
void scan();
void ser_out_st(unsigned char *,unsigned char);
void ser_out(unsigned char);
void ser_init();
void dela(unsigned int);
void chk_ndata_sub();
void display_all();
void chk_start();
void chk_ndata();
void htd(unsigned int);
void display();
void read(unsigned char);
void write(unsigned char);
void lcd_init(void);
void lcd_dis(unsigned char *,unsigned char);
void delay(unsigned int);
void del(unsigned char);
sbit en=P2^7;       //initializing port 2 to lcd variables
sbit rw=P2^6;
sbit rs=P2^5;
sbit p7=P1^7;//matrix keys connected to  pins of port1
sbit p6=P1^6;
```

```
sbit p4=P1^4;
sbit p3=P1^3;
sbit p0=P1^0;
sbit ndata  = P3^4;
sbit red   = P2^4;//2.2        //  initializing  pins of port 2 to LEDs
sbit green   = P2^3;//2.1
sbit yellow  = P2^2;//2.0
sbit forward  = P2^0;//2.4   // forward and reverse bits for motor
sbit reverse  = P3^3;//2.3
sbit buzzer   = P2^1;        // initializing port pin to buzzer
bit z,y,start_bit,ndata_bit,flag,flag1,forward_bit,key1,key2,key3,one_bit;
unsigned char thous,hund,ten,one,hund_r;
unsigned int thous_r;
unsigned int time,sec,train;
unsigned char datum[5],val,add=0xc5,dtmf,e;
float velocity,time_sec;
main()
{
 buzzer=0;     // buzzer bit initialised
 forward=reverse=0;     // reverse and forward bit initialised
 lcd_init();     // initializing LCD varaibles
 read(0x01);
 while(1)
 {
 first:
    red=yellow=0;     // initialize red and yellow LEDs to zero
   read(0x80);    // to display at first line of LCD
   lcd_dis("EMBEDDED RAILWAY",16);
```

```
read(0xc0);  // to display at second line of LCD
lcd_dis("   CONTROLLER   ",16);
val=0;
EA=ET0=1;
TMOD=0x21;      // to operate  the timer at mode 2
TH0=TL0=0;
scn:
  scan();
  if(!key1) goto scn;  // scanning key 1
  key1=0;
  read(0x01);
  yellow=1;  // yellow LED ON to indicate train is approaching
  green=0;
  flag=1;   // timer in incrementing mode
  time=0;
  TR0=1;  // timer ON
 scn1:
   scan();
   read(0x80);  //display at first line
   lcd_dis("    ALERT...    ",16);
   read(0xc1);    // display position at second line of LCD
   lcd_dis("TIME :",6);
   read(0xcB);    // display position at second line of LCD
   lcd_dis("sec",3);
   htd(sec);  // fn to convert the hex data to decimal
   read(0Xc7);
   display();
   if(!key2) goto scn1;   // if key2 is pressed
```

```
key2=0;
TR0=0;
flag=0;
display_all();
velocity=0.5*3600/sec;   // calculating velocity
htd(velocity);
read(0X86);
display();
time_sec=2*3600/velocity; // calculating time to reach the gate
flag1=1;  // timer in decrementing mode
time=0;
TR0=1;
scn2:
  scan();
  if(time_sec==0)
  {
   TR0=0;
  }
  htd(time_sec);        // hexa to decimal conversion
  read(0XC7);         // display location
  display();
  if(time_sec<=35 && time_sec>30)
  {
   buzzer=1;   // buzzer ON when 35s remain to reach the gate
  }
  if(time_sec<=30 && time_sec>29)
  {
   yellow=0;     //yellow Led is OFF
```

```
red=1;   // Red LED ON when 30s remain to reach the gate
read(0x80);
lcd_dis("  GATE CLOSED   ",16);
read(0xc1);    //display location at second line
lcd_dis("TIME :",6);
read(0xcB);
lcd_dis("sec",3);
if(one_bit==0)  forward_bit=1;
}
if(forward_bit==1)
{
one_bit=1; // servo motor ON/OFF control
forward_bit=0;
forward=1;      // servo motor forward/ reverse control
dela(125);  // calling delay fn.
forward=0;
reverse=0;
buzzer=0; // buzzer OFF
}
if(!key3)  goto scn2;  //scanning key3
key3=0;
time=flag=flag1=0; //reset the flags
TR0=1;
read(0x80);   //display at first line
lcd_dis("  GATE OPEN    ",16);
read(0xC0);
lcd_dis(" HAPPY JOURNEY  ",16);
time=0;
```

```
reverse=1;  // motor run in reverse direction
delay(50000);  //delay fns. called
delay(50000);
delay(50000);
forward=1;      // in forward direction
delay(50000);  // delay fns. called
delay(50000);
delay(25000);
forward=0;  // forward and reverse bits reset
reverse=0;
time=flag=flag1=sec=time_sec=one_bit=0;
red=0;  // red Led is OFF
green=1; // green LED ON to indicate train has crossed the gate
TR1=0; // timers reset
TR0=0;
EA=ET0=0;
TMOD=0; // timer mode reest
TH0=TL0=0;
goto first;      // continuing scanning process
}
}
void t0_isr(void) interrupt 1  // interrupt service routine
{
TR0=0;
T0 = 0;
time++;
if(flag==1 && time>=0x10)  //timer in incrementing mode
{
```

```
time=0;
sec++;
if(sec>=999)  sec=999;
}
if(flag1==1 && time>=0x10)  // timer in decrementing mode
{
time=0;
time_sec--;
if(time_sec==0xffff)   time_sec=0;
}
TR0=1;
}
void display()        //display fn.
{
write(hund+0x30);
write(ten+0x30);
write(one+0x30);
}
void htd(unsigned int hex_val)      // hexa to decimal conversion
{
thous=hex_val/1000;
thous_r=hex_val%1000;
hund=thous_r/100;
hund_r=thous_r%100;
ten=hund_r/10;
one=hund_r%10;
}
```

```
void delay(unsigned int b)// the only delay routine, which is user variable &
is decided by the values given in the brackets
{
   do b-=1;
   while (b!=0);
}
void del(unsigned char del)     // delay fn.
{
 while(del--)
 {
  delay(65000);
 }
}
void lcd_init()  //initialisng LCD display
{
 read(0x38);    // reading at appropriate locations
 read(0x06);
 read(0x0c);
}
void read(unsigned char i)  // read fn.
{
 P0=i;
en=1;     // enable bit high
rs=rw=0;     // disable reset and read write bits
delay(125); // delay fn. called
en=0;     // enable bit low
delay(125);
}
```

```
void write(unsigned char i)  //write fn.
{
 P0=i;
 en=rs=1;  // enable and reset bit high
 rw=0;  // read write bit low
 delay(125);  // delay fn. called
 en=0; //enable bit low
 delay(125);
}
void lcd_dis(unsigned char *mess,unsigned char n)
{
 unsigned char i;    // i is the character to be displayed
 for(i=0;i<n;i++)   // n is the no. of characters to be displayed
 {
  write(mess[i]);
  delay(255); // delay fn. called
 }
}
void display_all()
{
 read(0x80);  // display at first line
 lcd_dis("SPEED:    Km/h ",16);
 read(0xc0); // display at second line
 lcd_dis(" TIME :000 sec  ",16);
}
void dela(unsigned int del)
{
 while(del--)
```

```
{
  htd(time_sec); // hexa to decimal conversion
  read(0XC7);
  display();
 }
}
void ser_init()
{
 TH1=0xe6;    //e6=1200 b/s,72=110 b/s
 TMOD=0x21;   //auto-reload mode for serial comm
 TR1=1;
 delay(255);
 SCON=0x58;     // SCON reg for serial commn.
 }
 void ser_out(unsigned char ser_data)
 {
  SBUF=ser_data;
  delay(255);
  SCON=0x58;
  delay(2000);
 }
 void ser_out_st(unsigned char *mess,unsigned char n)
 {
  unsigned char i;
  for(i=0;i<n;i++)
  {
   ser_out(mess[i]);
  }
```

```
}
void scan()
 {
    p0=0;
     p3=1;
     if(p6==0)
      {
      delay(20000);  // delay fn. called
       key1=1;
       key2=key3=0;
                        }

       p3=0;
        p0=1;
        if(p7==0)
         {
         delay(20000);
          key3=1;
          key1=key2=0;
          }
          else if(p4==0)
           {
           delay(20000);
            key2=1;
            key1=key3=0;
            }
          }
```

2.7.2.TRANSMITTER:

```
#include<reg51.h>
void transm(unsigned char);
sbit key1  = P3^4;    //initializing pins of port3 to keys
sbit key2  = P3^5;
sbit key3  = P3^6;
sbit key4  = P3^7;
main()
{while(1)
 {
  if(key1==0)  // key1 for train no.
  {
 again:
   if(key3==0)  //key3 for track no.
   {
    transm(0x03); // transm fn. called
    transm(0x04);
    transm(0x05);
    transm(0x01);
    goto end;
   }
   else if(key4==0)  // key4 for track no.
   {
    transm(0x03);  // transm fn. called
    transm(0x04);
    transm(0x05);
    transm(0x02);
    goto end;
```

```
 }  else goto again;
}
else if(key2==0)    // key2 for train no.
{
again1:
 if(key3==0)
 {
  transm(0x06);
  transm(0x07);
  transm(0x08);
  transm(0x01);
  goto end;
 }
 else if(key4==0)
 {
  transm(0x06);
  transm(0x07);
  transm(0x08);
  transm(0x02);
  goto end;
 }
 else goto again1;
}
else if(key3==0)
{
 transm(0x09);
}
end:;
```

```
}
}
void delay(unsigned int b)  //delay fn.
{
 do b-=1;
 while (b!=0);
}
void transm(unsigned char da)
{
    P1=da;
    delay(20000);
    P1=0xff;
    delay(20000);
}
```

2.7.3.RECEIVER:

```
#include<reg51.h>
void disp_all();
void htd(unsigned int);
void display();
void read(unsigned char);
void write(unsigned char);
void lcd_init(void);
void lcd_dis(unsigned char *,unsigned char);
void delay(unsigned int);
sbit ack    = P1^4;   // ack bit assigned to port 1
sbit rs     = P2^0;   // rs bit assigned to pin 0 of port2
sbit rw     = P2^1;   //rw bit assigned to pin1 of port2
```

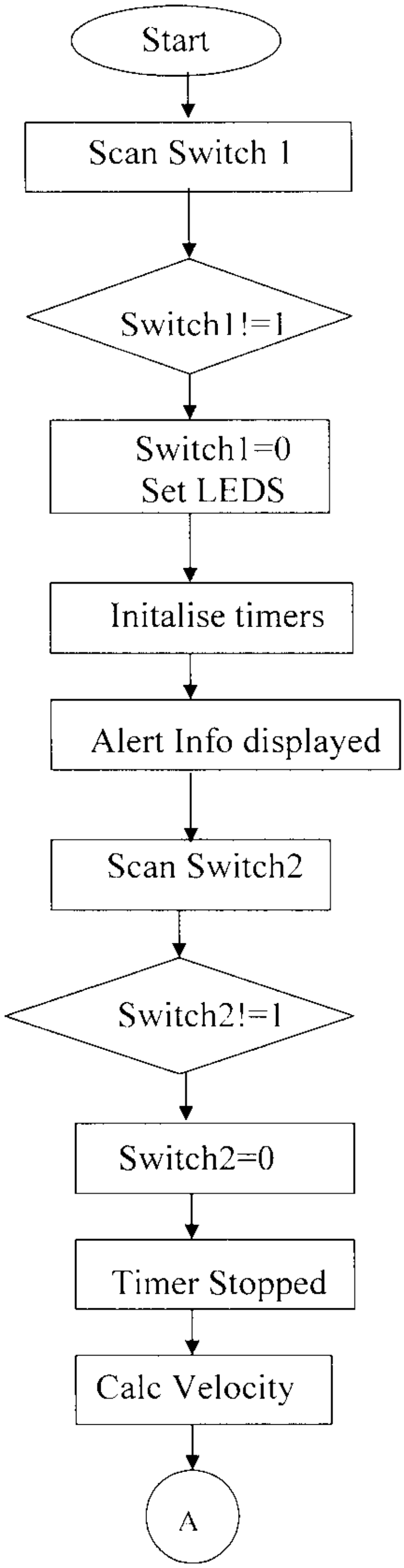
```
sbit en      = P2^2;    // en bit assigned to pin 2 of port2
unsigned char thous,hund,ten,one,hund_r;
unsigned int thous_r;
unsigned char u,address=0x89,dat;
main()
{
 lcd_init();  // LCD variables initialised
 disp_all();
 delay(65000); // delay fn. called
 delay(65000);
 read(0x01);     // display at appropriate location
 read(0x80);     // display at first line
 lcd_dis("TRAIN NO:      ",16);
 read(0xc0);    //display at second line
 lcd_dis("TRACK NO:      ",16);
 while(1)
 {
    if((ack==0)&&(u==0))  u=1;
    if((ack==1)&&(u==1))
    {     u=0;
     dat=P1&0x0f;
     if(dat!=0x0f && dat!=0x09)
     {
      read(address);
      write(dat+0x30);
      address++;
      if(address==0x8c)  address=0xc9;
      if(address==0xca)  address=0x89;
```

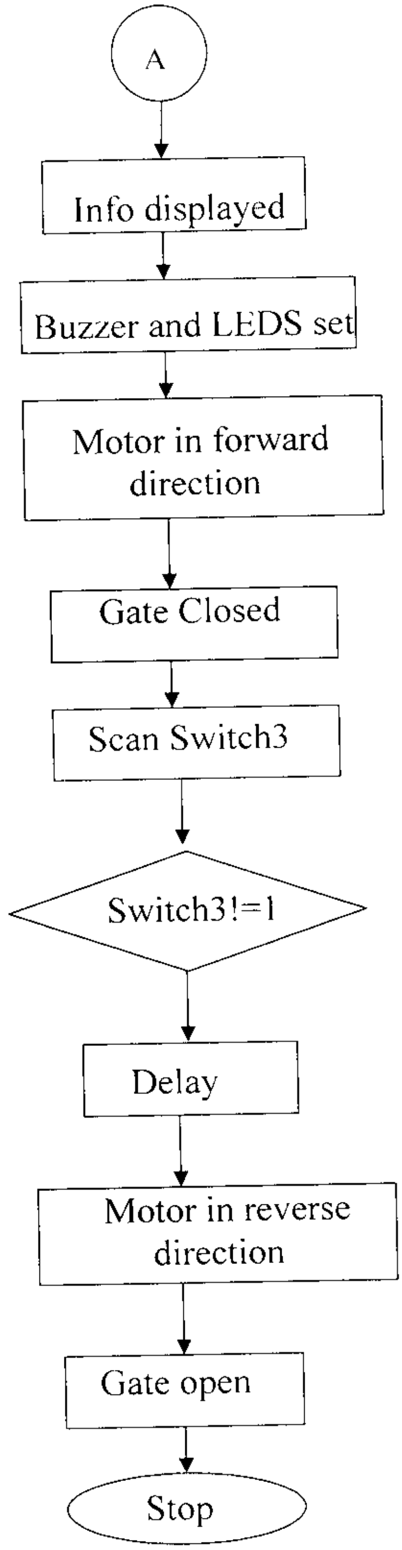
```
}
else if(dat==0x09)
 {
  read(0x01);
  read(0x80);    // display at first line
  lcd_dis("TRAIN NO:      ",16);
  read(0xc0);    // display at second line
  lcd_dis("TRACK NO:      ",16);
  address=0x89;
  }
  } }
}
void delay(unsigned int b)// the only delay routine, which is user variable &
is decided by the values given in the brackets
{  do b-=1;
  while (b!=0);
 }
 void lcd_init()
 {
  read(0x38);
  read(0x01);
  read(0x06);
  read(0x0c);
  }
  void read(unsigned char add)
  {
   P0=add;
   rs=rw=0;  // low the rs and rw bits
```

```
en=1;  // en bit high
delay(125);    // call delay
en=0;   // low the en bit
delay(125);
}
void write(unsigned char dat)
{ P0=dat;
 rs=en=1; // rs and en bit made high
 rw=0;  // rw bit made low
 delay(125);
 en=0;    // en bit made low
 delay(125);    // call delay
 }
 void lcd_dis(unsigned char *mess,unsigned char n)
 { unsigned char i;
  for(i=0;i<n;i++)     // n  -  no. of characters to be displayed
  {
   write(mess[i]);    // i – character to be diaplayed
   delay(125);
  }}
  void disp_all()
  {
   read(0x01);
   read(0x80); // display at first line
   lcd_dis(" Railway gate   ",16);
   read(0xc0);  //display at second line
   lcd_dis("  Controller    ",16)
  }
```

2.7.4.FLOWCHART

Fig 2.14




A
Info displayed
Buzzer and LEDS set
Motor in forward direction
Gate Closed
Scan Switch3
Switch3!=1
Delay
Motor in reverse direction
Gate open
Stop

CHAPTER 3

CONCLUSION:

In our project we implemented an automatic gate control system that includes appropriate warning signals at the gate, thus avoiding danger to life and property of the public by eliminating manual operation. In collision prevention system we took measures to prevent direct collision between the trains which is becoming a serious problem to the economy of the country.

FUTURE ENHANCEMENTS:

- Provision of sensors at the gate to sense if anyone is crossing the gate at the time of its closing so that appropriate measures could be taken to prevent their damage.
- To avoid the human involvement in entering the data such as train no. and its track no. for transmission.
- Real time simulation can be done.

APPENDICES

APPENDIX 1

Serial number	Components	Price In Rs.
1.	IC ULN2003	40
2.	IC 7805	15
3.	IC 7812	15
4.	IC 7912	15
5.	AT 89C51	90
6.	Crystal Oscillator-12 MHz	18
7.	10μF Capacitor	2
8.	1000μF Capacitor	8
9.	Key	12
10.	LCD Display	650
11.	ON/OFF Relay	60
12.	Resistor 3K	0.15
13.	Servo motor	950
14.	Gate Model	1600
15.	Resistor 33k	0.50
16.	Resistor 10k	0.25
17.	RF Transmitter	250
18.	RF Receiver	250
19.	HT648L and HT640	400
20.	Diodes	0.50
21.	Transformer	190
22.	PCB per Square cm.	2

Instruction Set

Microcontroller Instruction Set

For interrupt response time information, refer to the hardware description chapter.

Instructions that Affect Flag Settings[1]

Instruction	Flag			Instruction	Flag		
	C	OV	AC		C	OV	AC
ADD	X	X	X	CLR C	O		
ADDC	X	X	X	CPL C	X		
SUBB	X	X	X	ANL C,bit	X		
MUL	O	X		ANL C,/bit	X		
DIV	O	X		ORL C,bit	X		
DA	X			ORL C,/bit	X		
RRC	X			MOV C,bit	X		
RLC	X			CJNE	X		
SETB C	1						

Note: 1. Operations on SFR byte address 208 or bit addresses 209-215 (that is, the PSW or bits in the PSW) also affect flag settings.

The Instruction Set and Addressing Modes

R_n	Register R7-R0 of the currently selected Register Bank.
direct	8-bit internal data location's address. This could be an Internal Data RAM location (0-127) or a SFR [i.e., I/O port, control register, status register, etc. (128-255)].
@R_i	8-bit internal data RAM location (0-255) addressed indirectly through register R1or R0.
#data	8-bit constant included in instruction.
#data 16	16-bit constant included in instruction.
addr 16	16-bit destination address. Used by LCALL and LJMP. A branch can be anywhere within the 64K byte Program Memory address space.
addr 11	11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2K byte page of program memory as the first byte of the following instruction.
rel	Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is -128 to +127 bytes relative to first byte of the following instruction.
bit	Direct Addressed bit in Internal Data RAM or Special Function Register.

ATMEL

Instruction Set Summary

	0	1	2	3	4	5	6	7
0	NOP	JBC bit, rel [3B, 2C]	JB bit, rel [3B, 2C]	JNB bit, rel [3B, 2C]	JC rel [2B, 2C]	JNC rel [2B, 2C]	JZ rel [2B, 2C]	JNZ rel [2B, 2C]
1	AJMP (P0) [2B, 2C]	ACALL (P0) [2B, 2C]	AJMP (P1) [2B, 2C]	ACALL (P1) [2B, 2C]	AJMP (P2) [2B, 2C]	ACALL (P2) [2B, 2C]	AJMP (P3) [2B, 2C]	ACALL (P3) [2B, 2C]
2	LJMP addr16 [3B, 2C]	LCALL addr16 [3B, 2C]	RET [2C]	RETI [2C]	ORL dir, A [2B]	ANL dir, A [2B]	XRL dir, a [2B]	ORL C, bit [2B, 2C]
3	RR A	RRC A	RL A	RLC A	ORL dir, #data [3B, 2C]	ANL dir, #data [3B, 2C]	XRL dir, #data [3B, 2C]	JMP @A + DPTR [2C]
4	INC A	DEC A	ADD A, #data [2B]	ADDC A, #data [2B]	ORL A, #data [2B]	ANL A, #data [2B]	XRL A, #data [2B]	MOV A, #data [2B]
5	INC dir [2B]	DEC dir [2B]	ADD A, dir [2B]	ADDC A, dir [2B]	ORL A, dir [2B]	ANL A, dir [2B]	XRL A, dir [2B]	MOV dir, #data [3B, 2C]
6	INC @R0	DEC @R0	ADD A, @R0	ADDC A, @R0	ORL A, @R0	ANL A, @R0	XRL A, @R0	MOV @R0, @data [2B]
7	INC @R1	DEC @R1	ADD A, @R1	ADDC A, @R1	ORL A, @R1	ANL A, @R1	XRL A, @R1	MOV @R1, #data [2B]
8	INC R0	DEC R0	ADD A, R0	ADDC A, R0	ORL A, R0	ANL A, R0	XRL A, R0	MOV R0, #data [2B]
9	INC R1	DEC R1	ADD A, R1	ADDC A, R1	ORL A, R1	ANL A, R1	XRL A, R1	MOV R1, #data [2B]
A	INC R2	DEC R2	ADD A, R2	ADDC A, R2	ORL A, R2	ANL A, R2	XRL A, R2	MOV R2, #data [2B]
B	INC R3	DEC R3	ADD A, R3	ADDC A, R3	ORL A, R3	ANL A, R3	XRL A, R3	MOV R3, #data [2B]
C	INC R4	DEC R4	ADD A, R4	ADDC A, R4	ORL A, R4	ANL A, R4	XRL A, R4	MOV R4, #data [2B]
D	INC R5	DEC R5	ADD A, R5	ADDC A, R5	ORL A, R5	ANL A, R5	XRL A, R5	MOV R5, #data [2B]
E	INC R6	DEC R6	ADD A, R6	ADDC A, R6	ORL A, R6	ANL A, R6	XRL A, R6	MOV R6, #data [2B]
F	INC R7	DEC R7	ADD A, R7	ADDC A, R7	ORL A, R7	ANL A, R7	XRL A, R7	MOV R7, #data [2B]

Note: Key: [2B] = 2 Byte, [3B] = 3 Byte, [2C] = 2 Cycle, [4C] = 4 Cycle, Blank = 1 byte/1 cycle

Instruction Set Summary (Continued)

	8	9	A	B	C	D	E	F
0	SJMP REL [2B, 2C]	MOV DPTR,# data 16 [3B, 2C]	ORL C, /bit [2B, 2C]	ANL C, /bit [2B, 2C]	PUSH dir [2B, 2C]	POP dir [2B, 2C]	MOVX A, @DPTR [2C]	MOVX @DPTR, A [2C]
1	AJMP (P4) [2B, 2C]	ACALL (P4) [2B, 2C]	AJMP (P5) [2B, 2C]	ACALL (P5) [2B, 2C]	AJMP (P6) [2B, 2C]	ACALL (P6) [2B, 2C]	AJMP (P7) [2B, 2C]	ACALL (P7) [2B, 2C]
2	ANL C, bit [2B, 2C]	MOV bit, C [2B, 2C]	MOV C, bit [2B]	CPL bit [2B]	CLR bit [2B]	SETB bit [2B]	MOVX A, @R0 [2C]	MOVX wR0, A [2C]
3	MOVC A, @A + PC [2C]	MOVC A, @A + DPTR [2C]	INC DPTR [2C]	CPL C	CLR C	SETB C	MOVX A, @RI [2C]	MOVX @RI, A [2C]
4	DIV AB [2B, 4C]	SUBB A, #data [2B]	MUL AB [4C]	CJNE A, #data, rel [3B, 2C]	SWAP A	DA A	CLR A	CPL A
5	MOV dir, dir [3B, 2C]	SUBB A, dir [2B]		CJNE A, dir, rel [3B, 2C]	XCH A, dir [2B]	DJNZ dir, rel [3B, 2C]	MOV A, dir [2B]	MOV dir, A [2B]
6	MOV dir, @R0 [2B, 2C]	SUBB A, @R0	MOV @R0, dir [2B, 2C]	CJNE @R0, #data, rel [3B, 2C]	XCH A, @R0	XCHD A, @R0	MOV A, @R0	MOV @R0, A
7	MOV dir, @R1 [2B, 2C]	SUBB A, @R1	MOV @R1, dir [2B, 2C]	CJNE @R1, #data, rel [3B, 2C]	XCH A, @R1	XCHD A, @R1	MOV A, @R1	MOV @R1, A
8	MOV dir, R0 [2B, 2C]	SUBB A, R0	MOV R0, dir [2B, 2C]	CJNE R0, #data, rel [3B, 2C]	XCH A, R0	DJNZ R0, rel [2B, 2C]	MOV A, R0	MOV R0, A
9	MOV dir, R1 [2B, 2C]	SUBB A, R1	MOV R1, dir [2B, 2C]	CJNE R1, #data, rel [3B, 2C]	XCH A, R1	DJNZ R1, rel [2B, 2C]	MOV A, R1	MOV R1, A
A	MOV dir, R2 [2B, 2C]	SUBB A, R2	MOV R2, dir [2B, 2C]	CJNE R2, #data, rel [3B, 2C]	XCH A, R2	DJNZ R2, rel [2B, 2C]	MOV A, R2	MOV R2, A
B	MOV dir, R3 [2B, 2C]	SUBB A, R3	MOV R3, dir [2B, 2C]	CJNE R3, #data, rel [3B, 2C]	XCH A, R3	DJNZ R3, rel [2B, 2C]	MOV A, R3	MOV R3, A
C	MOV dir, R4 [2B, 2C]	SUBB A, R4	MOV R4, dir [2B, 2C]	CJNE R4, #data, rel [3B, 2C]	XCH A, R4	DJNZ R4, rel [2B, 2C]	MOV A, R4	MOV R4, A
D	MOV dir, R5 [2B, 2C]	SUBB A, R5	MOV R5, dir [2B, 2C]	CJNE R5, #data, rel [3B, 2C]	XCH A, R5	DJNZ R5, rel [2B, 2C]	MOV A, R5	MOV R5, A
E	MOV dir, R6 [2B, 2C]	SUBB A, R6	MOV R6, dir [2B, 2C]	CJNE R6, #data, rel [3B, 2C]	XCH A, R6	DJNZ R6, rel [2B, 2C]	MOV A, R6	MOV R6, A
F	MOV dir, R7 [2B, 2C]	SUBB A, R7	MOV R7, dir [2B, 2C]	CJNE R7, #data, rel [3B, 2C]	XCH A, R7	DJNZ R7, rel [2B, 2C]	MOV A, R7	MOV R7, A

Note: Key: [2B] = 2 Byte, [3B] = 3 Byte, [2C] = 2 Cycle, [4C] = 4 Cycle, Blank = 1 byte/1 cycle

Mnemonic		Description	Byte	Oscillator Period
ACALL	addr11	Absolute Subroutine Call	2	24
LCALL	addr16	Long Subroutine Call	3	24
RET		Return from Subroutine	1	24
RETI		Return from interrupt	1	24
AJMP	addr11	Absolute Jump	2	24
LJMP	addr16	Long Jump	3	24
SJMP	rel	Short Jump (relative addr)	2	24
JMP	@A+DPTR	Jump indirect relative to the DPTR	1	24
JZ	rel	Jump if Accumulator is Zero	2	24
JNZ	rel	Jump if Accumulator is Not Zero	2	24
CJNE	A,direct,rel	Compare direct byte to Acc and Jump if Not Equal	3	24
CJNE	A,#data,rel	Compare immediate to Acc and Jump if Not Equal	3	24
CJNE	R_n,#data,rel	Compare immediate to register and Jump if Not Equal	3	24
CJNE	@R_i,#data,rel	Compare immediate to indirect and Jump if Not Equal	3	24
DJNZ	R_n,rel	Decrement register and Jump if Not Zero	2	24
DJNZ	direct,rel	Decrement direct byte and Jump if Not Zero	3	24
NOP		No Operation	1	12

Table 1. AT89 Instruction Set Summary[1]

Mnemonic		Description	Byte	Oscillator Period
ARITHMETIC OPERATIONS				
ADD	A,R_n	Add register to Accumulator	1	12
ADD	A,direct	Add direct byte to Accumulator	2	12
ADD	A,@R_i	Add indirect RAM to Accumulator	1	12
ADD	A,#data	Add immediate data to Accumulator	2	12
ADDC	A,R_n	Add register to Accumulator with Carry	1	12
ADDC	A,direct	Add direct byte to Accumulator with Carry	2	12
ADDC	A,@R_i	Add indirect RAM to Accumulator with Carry	1	12
ADDC	A,#data	Add immediate data to Acc with Carry	2	12
SUBB	A,R_n	Subtract Register from Acc with borrow	1	12
SUBB	A,direct	Subtract direct byte from Acc with borrow	2	12
SUBB	A,@R_i	Subtract indirect RAM from ACC with borrow	1	12
SUBB	A,#data	Subtract immediate data from Acc with borrow	2	12
INC	A	Increment Accumulator	1	12
INC	R_n	Increment register	1	12
INC	direct	Increment direct byte	2	12
INC	@R_i	Increment direct RAM	1	12
DEC	A	Decrement Accumulator	1	12
DEC	R_n	Decrement Register	1	12
DEC	direct	Decrement direct byte	2	12
DEC	@R_i	Decrement indirect RAM	1	12
INC	DPTR	Increment Data Pointer	1	24
MUL	AB	Multiply A & B	1	48
DIV	AB	Divide A by B	1	48
DA	A	Decimal Adjust Accumulator	1	12

Note: 1. All mnemonics copyrighted © Intel Corp., 1980.

Mnemonic		Description	Byte	Oscillator Period
LOGICAL OPERATIONS				
ANL	A,R_n	AND Register to Accumulator	1	12
ANL	A,direct	AND direct byte to Accumulator	2	12
ANL	A,@R_i	AND indirect RAM to Accumulator	1	12
ANL	A,#data	AND immediate data to Accumulator	2	12
ANL	direct,A	AND Accumulator to direct byte	2	12
ANL	direct,#data	AND immediate data to direct byte	3	24
ORL	A,R_n	OR register to Accumulator	1	12
ORL	A,direct	OR direct byte to Accumulator	2	12
ORL	A,@R_i	OR indirect RAM to Accumulator	1	12
ORL	A,#data	OR immediate data to Accumulator	2	12
ORL	direct,A	OR Accumulator to direct byte	2	12
ORL	direct,#data	OR immediate data to direct byte	3	24
XRL	A,R_n	Exclusive-OR register to Accumulator	1	12
XRL	A,direct	Exclusive-OR direct byte to Accumulator	2	12
XRL	A,@R_i	Exclusive-OR indirect RAM to Accumulator	1	12
XRL	A,#data	Exclusive-OR immediate data to Accumulator	2	12
XRL	direct,A	Exclusive-OR Accumulator to direct byte	2	12
XRL	direct,#data	Exclusive-OR immediate data to direct byte	3	24
CLR	A	Clear Accumulator	1	12
CPL	A	Complement Accumulator	1	12
RL	A	Rotate Accumulator Left	1	12
RLC	A	Rotate Accumulator Left through the Carry	1	12
LOGICAL OPERATIONS (continued)				

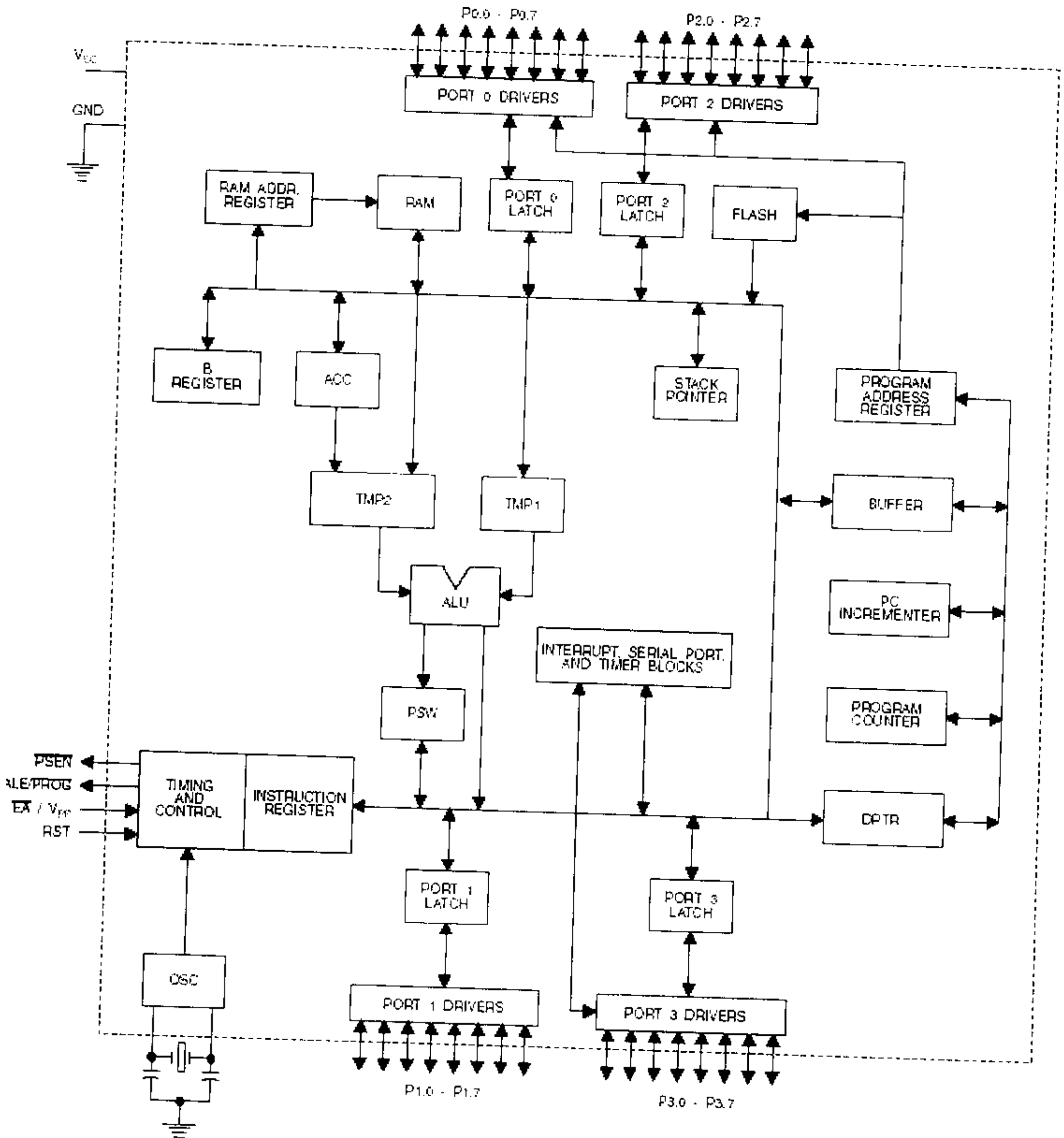
Instruction Set

Mnemonic		Description	Byte	Oscillator Period
RR	A	Rotate Accumulator Right	1	12
RRC	A	Rotate Accumulator Right through the Carry	1	12
SWAP	A	Swap nibbles within the Accumulator	1	12
DATA TRANSFER				
MOV	A,R_n	Move register to Accumulator	1	12
MOV	A,direct	Move direct byte to Accumulator	2	12
MOV	A,@R_i	Move indirect RAM to Accumulator	1	12
MOV	A,#data	Move immediate data to Accumulator	2	12
MOV	R_n,A	Move Accumulator to register	1	12
MOV	R_n,direct	Move direct byte to register	2	24
MOV	R_n,#data	Move immediate data to register	2	12
MOV	direct,A	Move Accumulator to direct byte	2	12
MOV	direct,R_n	Move register to direct byte	2	24
MOV	direct,direct	Move direct byte to direct	3	24
MOV	direct,@R_i	Move indirect RAM to direct byte	2	24
MOV	direct,#data	Move immediate data to direct byte	3	24
MOV	@R_i,A	Move Accumulator to indirect RAM	1	12
MOV	@R_i,direct	Move direct byte to indirect RAM	2	24
MOV	@R_i,#data	Move immediate data to indirect RAM	2	12
MOV	DPTR,#data16	Load Data Pointer with a 16-bit constant	3	24
MOVC	A,@A+DPTR	Move Code byte relative to DPTR to Acc	1	24
MOVC	A,@A+PC	Move Code byte relative to PC to Acc	1	24
MOVX	A,@R_i	Move External RAM (8-bit addr) to Acc	1	24
DATA TRANSFER (continued)				

Mnemonic		Description	Byte	Oscillator Period
MOVX	A,@DPTR	Move Exernal RAM (16-bit addr) to Acc	1	24
MOVX	@R_i,A	Move Acc to External RAM (8-bit addr)	1	24
MOVX	@DPTR,A	Move Acc to External RAM (16-bit addr)	1	24
PUSH	direct	Push direct byte onto stack	2	24
POP	direct	Pop direct byte from stack	2	24
XCH	A,R_n	Exchange register with Accumulator	1	12
XCH	A,direct	Exchange direct byte with Accumulator	2	12
XCH	A,@R_i	Exchange indirect RAM with Accumulator	1	12
XCHD	A,@R_i	Exchange low-order Digit indirect RAM with Acc	1	12
BOOLEAN VARIABLE MANIPULATION				
CLR	C	Clear Carry	1	12
CLR	bit	Clear direct bit	2	12
SETB	C	Set Carry	1	12
SETB	bit	Set direct bit	2	12
CPL	C	Complement Carry	1	12
CPL	bit	Complement direct bit	2	12
ANL	C,bit	AND direct bit to CARRY	2	24
ANL	C,/bit	AND complement of direct bit to Carry	2	24
ORL	C,bit	OR direct bit to Carry	2	24
ORL	C,/bit	OR complement of direct bit to Carry	2	24
MOV	C,bit	Move direct bit to Carry	2	12
MOV	bit,C	Move Carry to direct bit	2	24
JC	rel	Jump if Carry is set	2	24
JNC	rel	Jump if Carry not set	2	24
JB	bit,rel	Jump if direct Bit is set	3	24
JNB	bit,rel	Jump if direct Bit is Not set	3	24
JBC	bit,rel	Jump if direct Bit is set & clear bit	3	24
PROGRAM BRANCHING				

APPENDIX 2

Block Diagram



The AT89C51 provides the following standard features: 4K bytes of Flash, 128 bytes of RAM, 32 I/O lines, two 16-bit timer/counters, a five vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator and clock circuitry. In addition, the AT89C51 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port and interrupt system to continue functioning. The Power-down Mode saves the RAM contents but freezes the oscillator disabling all other chip functions until the next hardware reset.

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 may also be configured to be the multiplexed low-order address/data bus during accesses to external program and data memory. In this mode P0 has internal pullups.

Port 0 also receives the code bytes during Flash programming, and outputs the code bytes during program verification. External pullups are required during program verification.

Port 1

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

Port 1 also receives the low-order address bytes during Flash programming and verification.

Port 2

Port 2 is an 8-bit bi-directional I/O port with internal pullups. 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 pullups and can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current (I_{IL}) because of the internal pullups.

Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @ DPTR). In this application, it uses strong internal pullups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ RI), 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 pullups. 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 pullups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (I_{IL}) because of the pullups.

Port 3 also serves the functions of various special features of the AT89C51 as listed below:

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

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

RST

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

ALE/$\overline{PROG}$

Address Latch Enable output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input ($\overline{PROG}$) during Flash programming.

In normal operation ALE is emitted at a constant rate of 1/6 the oscillator frequency, and may be used for external timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external Data Memory.

If desired, ALE operation can be disabled by setting bit 0 of SFR location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction. Otherwise, the pin is weakly pulled high. Setting the ALE-disable bit has no effect if the microcontroller is in external execution mode.

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 V_{CC} for internal program executions.

This pin also receives the 12-volt programming enable voltage (V_{PP}) during Flash programming, for parts that require 12-volt V_{PP}.

XTAL1

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

XTAL2

Output from the inverting oscillator amplifier.

Oscillator Characteristics

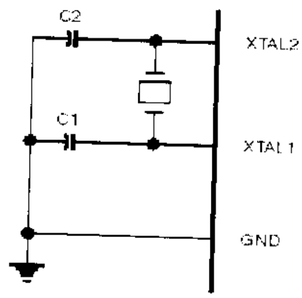
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 1. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven as shown in Figure 2. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.

Idle Mode

In idle mode, the CPU puts itself to sleep while all the on-chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the special functions registers remain unchanged during this mode. The idle mode can be terminated by any enabled interrupt or by a hardware reset.

It should be noted that when idle is terminated by a hard ware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write to a port pin when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

Figure 1. Oscillator Connections



Note: C1, C2 = 30 pF ± 10 pF for Crystals
= 40 pF ± 10 pF for Ceramic Resonators

Status of External Pins During Idle and Power-down Modes

Mode	Program Memory	ALE	PSEN	PORT0	PORT1	PORT2	PORT3
Idle	Internal	1	1	Data	Data	Data	Data
Idle	External	1	1	Float	Data	Address	Data
Power-down	Internal	0	0	Data	Data	Data	Data
Power-down	External	0	0	Float	Data	Data	Data

APPENDIX 3

National Semiconductor

May 2000

LM78XX Series Voltage Regulators

General Description

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 currents.

The LM78XX series is available in an aluminum TO-3 package which will allow over 1.0A load current if adequate heat sinking is provided. Current limiting is included to limit the peak output current to a safe value. Safe area protection for the output transistor is provided to limit internal power dissipation. If internal power dissipation becomes too high for the heat sinking provided, the thermal shutdown circuit takes over preventing the IC from overheating.

Considerable effort was expanded to make the LM78XX series of regulators easy to use and minimize the number of external components. It is not necessary to bypass the output, although this does improve transient response. Input bypassing is needed only if the regulator is located far from the filter capacitor of the power supply.

For output voltage other than 5V, 12V and 15V the LM117 series provides an output voltage range from 1.2V to 57V.

Features

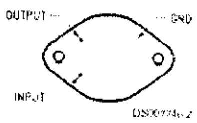
- Output current in excess of 1A
- Internal thermal overload protection
- No external components required
- Output transistor safe area protection
- Internal short circuit current limit
- Available in the aluminum TO-3 package

Voltage Range

LM7805C	5V
LM7812C	12V
LM7815C	15V

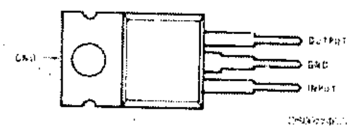
Connection Diagrams

Metal Can Package
TO-3 (K)
Aluminum



Bottom View
Order Number LM7805CK, LM7812CK or LM7815CK
See NS Package Number KC02A

Plastic Package
TO-220 (T)



Top View
Order Number LM7805CT, LM7812CT or LM7815CT
See NS Package Number T03B

Absolute Maximum Ratings (Note 3)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/ Distributors for availability and specifications.

Input Voltage	
(V_O = 5V, 12V and 15V)	35V
Internal Power Dissipation (Note 1)	Internally Limited
Operating Temperature Range (T_A)	0°C to +70°C
Maximum Junction Temperature	
(K Package)	150°C
(T Package)	150°C
Storage Temperature Range	−65°C to +150°C
Lead Temperature (Soldering, 10 sec.)	
TO-3 Package K	300°C
TO-220 Package T	230°C

Electrical Characteristics LM78XXC (Note 2)

0°C ≤ T_J ≤ 125°C unless otherwise noted.

Output Voltage				5V			12V			15V			Units
Input Voltage (unless otherwise noted)				10V			19V			23V			
Symbol	Parameter	Conditions		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
V_O	Output Voltage	Tj = 25°C, 5 mA ≤ I_O ≤ 1A		4.8	5	5.2	11.5	12	12.5	14.4	15	15.6	V
		P_D ≤ 15W, 5 mA ≤ I_O ≤ 1A		4.75		5.25	11.4		12.6	14.25		15.75	V
		V_{MIN} ≤ V_{IN} ≤ V_{MAX}		(7.5 ≤ V_{IN} ≤ 20)			(14.5 ≤ V_{IN} ≤ 27)			(17.5 ≤ V_{IN} ≤ 30)			V
ΔV_O	Line Regulation	I_O = 500 mA	Tj = 25°C		3	50		4	120		4	150	mV
			ΔV_{IN}	(7 ≤ V_{IN} ≤ 25)			14.5 ≤ V_{IN} ≤ 30)			(17.5 ≤ V_{IN} ≤ 30)			V
			0°C ≤ Tj ≤ +125°C			50			120			150	mV
			ΔV_{IN}	(8 ≤ V_{IN} ≤ 20)			(15 ≤ V_{IN} ≤ 27)			(18.5 ≤ V_{IN} ≤ 30)			V
		I_O ≤ 1A	Tj = 25°C			50			120			150	mV
			ΔV_{IN}	(7.5 ≤ V_{IN} ≤ 20)			(14.6 ≤ V_{IN} ≤ 27)			(17.7 ≤ V_{IN} ≤ 30)			V
			0°C ≤ Tj ≤ +125°C			25			60			75	mV
			ΔV_{IN}	(8 ≤ V_{IN} ≤ 12)			(16 ≤ V_{IN} ≤ 22)			(20 ≤ V_{IN} ≤ 26)			V
ΔV_O	Load Regulation	Tj = 25°C	5 mA ≤ I_O ≤ 1.5A		10	50		12	120		12	150	mV
			250 mA ≤ I_O ≤ 750 mA			25			60			75	mV
		5 mA ≤ I_O ≤ 1A, 0°C ≤ Tj ≤ +125°C				50			120			150	mV
I_Q	Quiescent Current	I_O ≤ 1A	Tj = 25°C			8			8			8	mA
			0°C ≤ Tj ≤ +125°C			8.5			8.5			8.5	mA
ΔI_Q	Quiescent Current Change	5 mA ≤ I_O ≤ 1A				0.5			0.5			0.5	mA
		Tj = 25°C, I_O ≤ 1A				1.0			1.0			1.0	mA
		V_{MIN} ≤ V_{IN} ≤ V_{MAX}		(7.5 ≤ V_{IN} ≤ 20)			(14.8 ≤ V_{IN} ≤ 27)			(17.9 ≤ V_{IN} ≤ 30)			V
		I_O ≤ 500 mA, 0°C ≤ Tj ≤ +125°C				1.0			1.0			1.0	mA
		V_{MIN} ≤ V_{IN} ≤ V_{MAX}		(7 ≤ V_{IN} ≤ 25)			(14.5 ≤ V_{IN} ≤ 30)			(17.5 ≤ V_{IN} ≤ 30)			V
V_N	Output Noise Voltage	T_A = 25°C, 10 Hz ≤ f ≤ 100 kHz			40			75			90		µV
$\frac{\Delta V_{IN}}{\Delta V_{OUT}}$	Ripple Rejection	f = 120 Hz	I_O ≤ 1A, Tj = 25°C or	62	80		55	72		54	70		dB
			I_O ≤ 500 mA, 0°C ≤ Tj ≤ +125°C	62			55			54			dB
		V_{MIN} ≤ V_{IN} ≤ V_{MAX}		(8 ≤ V_{IN} ≤ 18)			(15 ≤ V_{IN} ≤ 25)			(18.5 ≤ V_{IN} ≤ 28.5)			V
R_O	Dropout Voltage	Tj = 25°C, I_{OUT} = 1A			2.0			2.0			2.0		V
	Output Resistance	f = 1 kHz			8			18			19		mΩ

Electrical Characteristics LM78XXC (Note 2) (Continued)

$0°C \le T_J \le 125°C$ unless otherwise noted.

	Output Voltage		5V			12V			15V			Units
	Input Voltage (unless otherwise noted)		10V			19V			23V			
Symbol	Parameter	Conditions	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
	Short-Circuit Current	Tj = 25°C		2.1			1.5			1.2		A
	Peak Output Current	Tj = 25°C		2.4			2.4			2.4		A
	Average TC of V_{OUT}	$0°C \le Tj \le +125°C$, I_O = 5 mA		0.6			1.5			1.8		mV/°C
V_{IN}	Input Voltage Required to Maintain Line Regulation	Tj = 25°C, $I_O \le 1A$	7.5			14.6			17.7			V

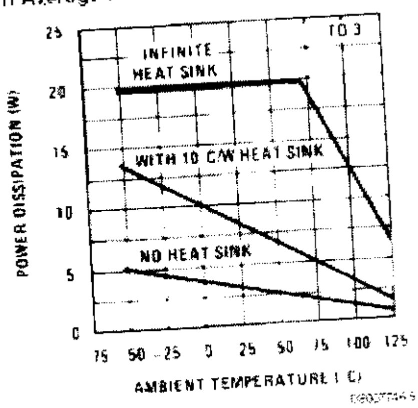
Note 1: Thermal resistance of the TO-3 package (K, KC) is typically 4°C/W junction to case and 35°C/W case to ambient. Thermal resistance of the TO-220 package (T) is typically 4°C/W junction to case and 50°C/W case to ambient.

Note 2: All characteristics are measured with capacitor across the input of 0.22 µF, and a capacitor across the output of 0.1µF. All characteristics except noise voltage and ripple rejection ratio are measured using pulse techniques ($t_W \le 10$ ms, duty cycle $\le 5\%$). Output voltage changes due to changes in internal temperature must be taken into account separately.

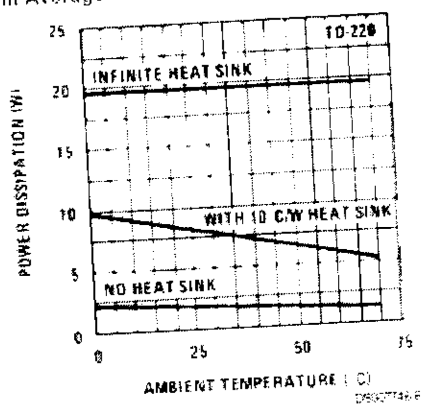
Note 3: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. For guaranteed specifications and the test conditions, see Electrical Characteristics.

Typical Performance Characteristics

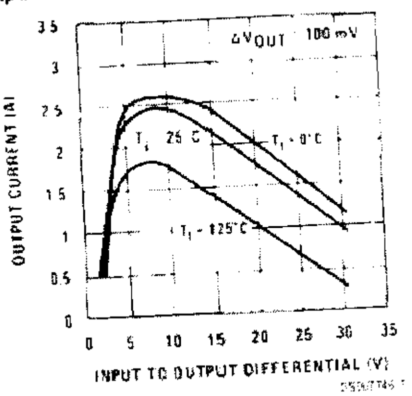
Maximum Average Power Dissipation



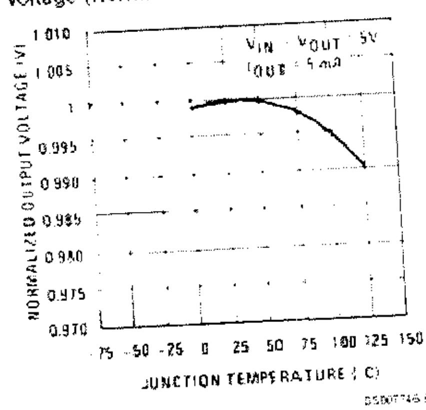
Maximum Average Power Dissipation



Peak Output Current



Output Voltage (Normalized to 1V at T_J = 25°C)



APPENDIX 4

FAIRCHILD SEMICONDUCTOR®

www.fairchildsemi.com

KA78XX/KA78XXA

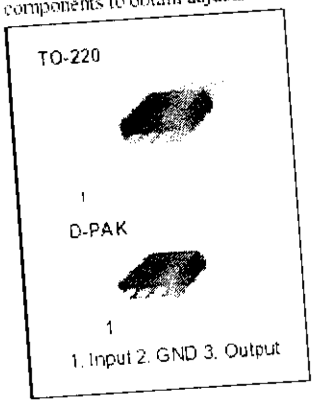
3-Terminal 1A Positive Voltage Regulator

Features

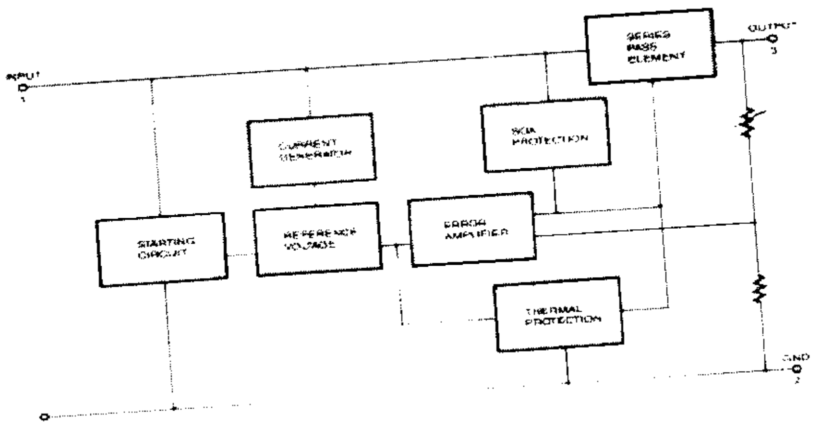
- Output Current up to 1A
- Output Voltages of 5, 6, 8, 9, 10, 12, 15, 18, 24V
- Thermal Overload Protection
- Short Circuit Protection
- Output Transistor Safe Operating Area Protection

Description

The KA78XX/KA78XXA series of three-terminal positive regulator are available in the TO-220/D-PAK package and with several fixed output voltages, making them useful in a wide range of applications. Each type employs internal current limiting, thermal shut down and safe operating area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltages and currents.



Internal Block Digram



Rev. 1.0.0

Electrical Characteristics (KA7806/KA7806R)

(Refer to test circuit ,0°C < T_J < 125°C, I_O = 500mA, V_I =11V, C_I= 0.33µF, C_O=0.1uF, unless otherwise specified)

Parameter	Symbol	Conditions		KA7806 Min.	KA7806 Typ.	KA7806 Max.	Unit
Output Voltage	VO	TJ =+25 °C		5.75	6.0	6.25	V
		5.0mA ≤ IO ≤ 1.0A, PO ≤ 15W VI = 8.0V to 21V		5.7	6.0	6.3	
Line Regulation (Note1)	Regline	TJ =+25 °C	VI = 8V to 25V	-	5	120	mV
			VI = 9V to 13V	-	1.5	60	
Load Regulation (Note1)	Regload	TJ =+25 °C	IO =5mA to 1.5A	-	9	120	mV
			IO =250mA to750mA	-	3	60	
Quiescent Current	IQ	TJ =+25 °C		-	5.0	8.0	mA
Quiescent Current Change	ΔIQ	IO = 5mA to 1A		-	-	0.5	mA
		VI = 8V to 25V		-	-	1.3	
Output Voltage Drift	ΔVO/ΔT	IO = 5mA		-	-0.8	-	mV/°C
Output Noise Voltage	VN	f = 10Hz to 100KHz, TA =+25 °C		-	45	-	µV/Vo
Ripple Rejection	RR	f = 120Hz VI = 9V to 19V		59	75	-	dB
Dropout Voltage	VDrop	IO = 1A, TJ =+25 °C		-	2	-	V
Output Resistance	rO	f = 1KHz		-	19	-	mΩ
Short Circuit Current	ISC	VI= 35V, TA=+25 °C		-	250	-	mA
Peak Current	IPK	TJ =+25 °C		-	2.2	-	A

Note:

1. Load and line regulation are specified at constant junction temperature. Changes in VO due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Typical Perfomance Characteristics

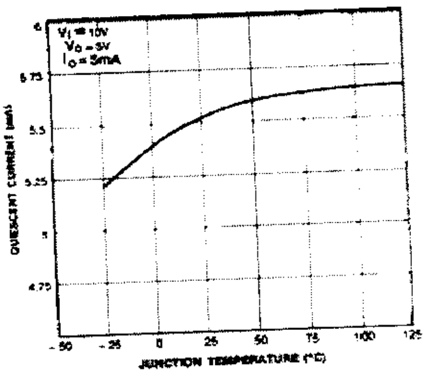


Figure 1. Quiescent Current

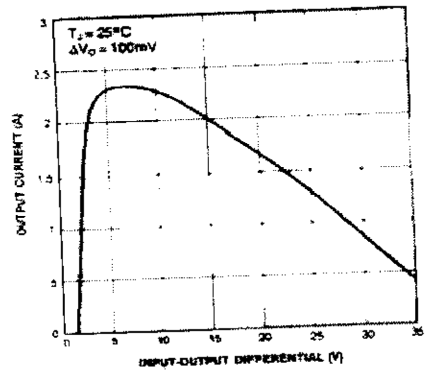


Figure 2. Peak Output Current

APPENDIX 5

Order this document by MC7900/D

MOTOROLA

MC7900 Series

Three-Terminal Negative Voltage Regulators

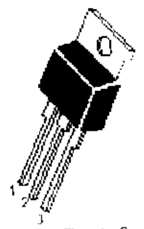
THREE–TERMINAL NEGATIVE FIXED VOLTAGE REGULATORS

The MC7900 series of fixed output negative voltage regulators are intended as complements to the popular MC7800 series devices. These negative regulators are available in the same seven–voltage options as the MC7800 devices. In addition, one extra voltage option commonly employed in MECL systems is also available in the negative MC7900 series.

Available in fixed output voltage options from −5.0 V to −24 V, these regulators employ current limiting, thermal shutdown, and safe–area compensation – making them remarkably rugged under most operating conditions. With adequate heatsinking they can deliver output currents in excess of 1.0 A.

- No External Components Required
- Internal Thermal Overload Protection
- Internal Short Circuit Current Limiting
- Output Transistor Safe–Area Compensation
- Available in 2% Voltage Tolerance (See Ordering Information)

T SUFFIX
PLASTIC PACKAGE
CASE 221A



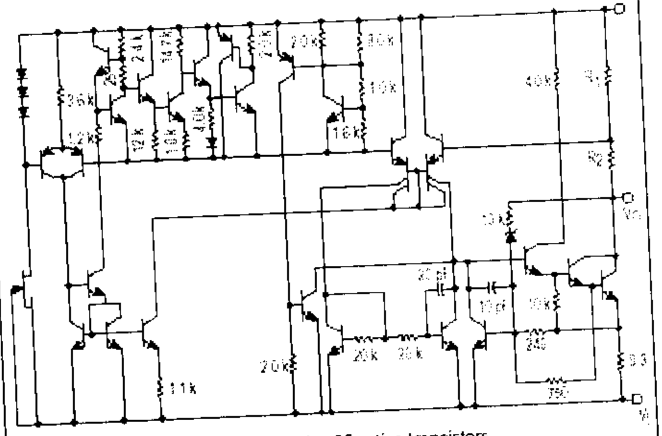
Heatsink surface connected to Pin 2.

Pin 1. Ground
2. Input
3. Output

D2T SUFFIX
PLASTIC PACKAGE
CASE 936
(D^2PAK)

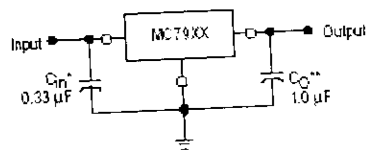
Heatsink surface (shown as terminal 4 in case outline drawing) is connected to Pin 2.

Representative Schematic Diagram



This device contains 26 active transistors.

STANDARD APPLICATION



A common ground is required between the input and the output voltages. The input voltage must remain typically 2.0 V above more negative even during the high point of the input ripple voltage.

XX, These two digits of the type number indicate nominal voltage.
* C_{in} is required if regulator is located an appreciable distance from power supply filter.
** C_O improve stability and transient response.

ORDERING INFORMATION

Device	Output Voltage Tolerance	Operating Temperature Range	Package
MC79XXACD2T	2%	T_J = 0° to +125°C	Surface Mount
MC79XXCD2T	4%		
MC79XXACT	2%		Insertion Mount
MC79XXCT	4%		
MC79XXBD2T	4%	T_J = −40° to +125°C	Surface Mount
MC79XXBT			Insertion Mount

XX indicates nominal voltage.

DEVICE TYPE/NOMINAL OUTPUT VOLTAGE

MC7905	5.0 V	MC7912	12 V
MC7905.2	5.2 V	MC7915	15 V
MC7906	6.0 V	MC7918	28 V
MC7908	8.0 V	MC7924	24 V

Rev 6

MAXIMUM RATINGS (T_A = +25°C, unless otherwise noted.)

Rating	Symbol	Value	Unit
Input Voltage (−5.0 V ≥ V_O ≥ −18 V) (24 V)	V_I	−35 −40	Vdc
Power Dissipation			
Case 221A			
T_A = +25°C	P_D	Internally Limited	W
Thermal Resistance, Junction–to–Ambient	θ_{JA}	65	°C/W
Thermal Resistance, Junction–to–Case	θ_{JC}	5.0	°C/W
Case 936 (D^2PAK)			
T_A = +25°C	P_D	Internally Limited	W
Thermal Resistance, Junction–to–Ambient	θ_{JA}	70	°C/W
Thermal Resistance, Junction–to–Case	θ_{JC}	5.0	°C/W
Storage Junction Temperature Range	T_{stg}	−65 to +150	°C
Junction Temperature	T_J	+150	°C

THERMAL CHARACTERISTICS

Characteristics	Symbol	Max	Unit
Thermal Resistance, Junction–to–Ambient	$R_{\theta JA}$	65	°C/W
Thermal Resistance, Junction–to–Case	$R_{\theta JC}$	5.0	°C/W

MC7905C
ELECTRICAL CHARACTERISTICS (V_I = −10 V, I_O = 500 mA, 0°C < T_J < +125°C, unless otherwise noted.)

Characteristics	Symbol	Min	Typ	Max	Unit
Output Voltage (T_J = +25°C)	V_O	−4.8	−5.0	−5.2	Vdc
Line Regulation (Note 1)	Reg_{line}				mV
(T_J = +25°C, I_O = 100 mA)					
−7.0 Vdc ≥ V_I ≥ −25 Vdc		–	7.0	50	
−8.0 Vdc ≥ V_I ≥ −12 Vdc		–	2.0	25	
(T_J = +25°C, I_O = 500 mA)					
−7.0 Vdc ≥ V_I ≥ −25 Vdc		–	35	100	
−8.0 Vdc ≥ V_I ≥ −12 Vdc		–	8.0	50	
Load Regulation, T_J = +25°C (Note 1)	Reg_{load}				mV
5.0 mA ≤ I_O ≤ 1.5 A		–	11	100	
250 mA ≤ I_O ≤ 750 mA		–	4.0	50	
Output Voltage −7.0 Vdc ≥ V_I ≥ −20 Vdc, 5.0 mA ≤ I_O ≤ 1.0 A, P ≤ 15 W	V_O	−4.75	–	−5.25	Vdc
Input Bias Current (T_J = +25°C)	I_{IB}	–	4.3	8.0	mA
Input Bias Current Change	ΔI_{IB}				mA
−7.0 Vdc ≥ V_I ≥ −25 Vdc		–	–	1.3	
5.0 mA ≤ I_O ≤ 1.5 A		–	–	0.5	
Output Noise Voltage (T_A = +25°C, 10 Hz ≤ f ≤ 100 kHz)	V_n	–	40	–	µV
Ripple Rejection (I_O = 20 mA, f = 120 Hz)	RR	–	70	–	dB
Dropout Voltage I_O = 1.0 A, T_J = +25°C	$V_I – V_O$	–	2.0	–	Vdc
Average Temperature Coefficient of Output Voltage I_O = 5.0 mA, 0°C ≤ T_J ≤ +125°C	$\Delta V_O/\Delta T$	–	−1.0	–	mV/°C

NOTE: 1. Load and line regulation are specified at constant junction temperature. Changes in V_O due to heating effects must be taken into account separately. Pulse testing with low duty cycle is used.

Figure 1. Worst Case Power Dissipation as a Function of Ambient Temperature

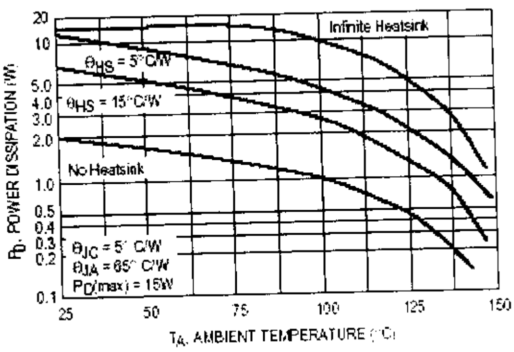


Figure 2. Peak Output Current as a Function of Input–Output Differential Voltage

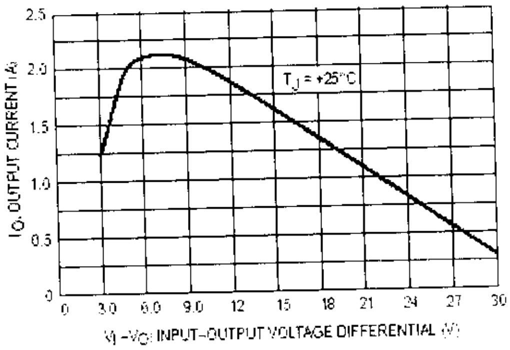


Figure 3. Ripple Rejection as a Function of Frequency

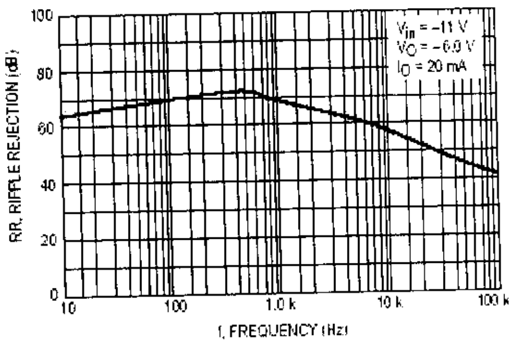


Figure 4. Ripple Rejection as a Function of Output Voltage

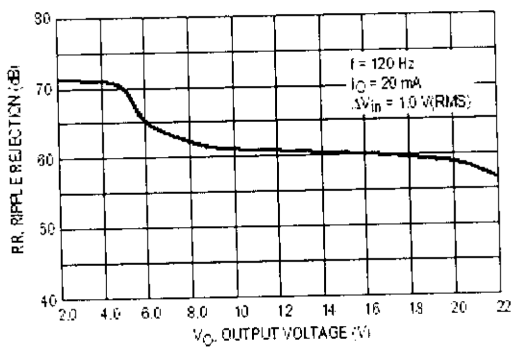


Figure 5. Output Voltage as a Function of Junction Temperature

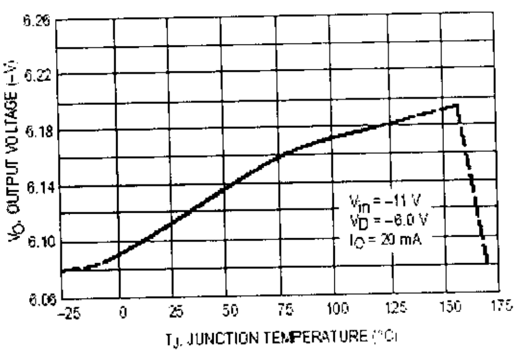
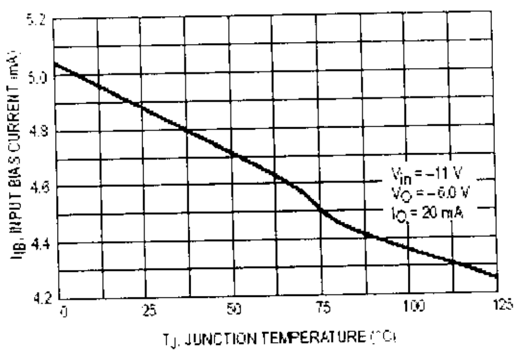


Figure 6. Quiescent Current as a Function of Temperature



APPENDIX 6

LCD General Information

Part Numbering ~ LCD ~ Glass Only (without drivers)

				A	B							C	D		E	F
L	C	D	-	S	4	0	1	C	4	0		T	F		-	1
LIQUID	CRYSTAL	DISPLAY		OPERATING TEMPERATURE	MODEL NUMBER							FLUID TYPE	POLARIZER		DISPLAY MODE	VIEWING DIR

A = OPERATING TEMPERATURE

S = Standard (0° TO 50° C)
H = High Temp (-30° to 75° C)
UH = Ultra High Temp (-40° to 85° C)

B = MODEL NUMBER

FOR: ALPHANUMERIC AND NUMERIC
A = Alphanumeric <1.0"
B = Alphanumeric ≥1.0"
C = Numeric <1.0"
D = Numeric ≥1.0"
M = Custom Display <1.0"
N = Custom Display ≥1.0"
401C40 = 4 Character x 1 line, 0.40"
4x1C45 = 4.5 Character x 1 line, 0.45"
x - represents 1/2 column
101D23 = 1 Character x 1 Line, 2.3"
101D40 = 1 Character x 1 Line, 4.0"

FOR: MATRIX AND GRAPHIC MODEL
01602D = 16 Characters x 2 Lines, Dot Matrix
12864G = 128 Columns x 64 Rows, Graphics

C = FLUID TYPE

T = Twisted Nematic (TN)
S = Super Twisted Nematic (STN)

D = POLARIZER MODE

R = Reflective
F = Transflective
M = Transmissive

*** ADDITIONAL PART NUMBERING SEQUENCE FOR NON DEFAULT SETTING***

E = DISPLAY MODE

Default = Positive image (no letter), or if more Description follows, use "-"
N = Negative image

F = VIEWING DIRECTION

DEFAULT = 6 O'CLOCK (NO NUMBER)
1 = 12 O'CLOCK

LCD General Information

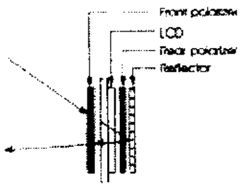
LCD Panels ~ Custom Design Guidelines | **Lighting and Mounting Methods**

Types of Display

Positive Type 1.8.8.8 Negative Type 1.8.8.8

Lighting Methods

(1) Reflective Mode

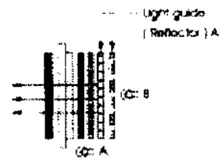


* It is necessary to use type under ambient light condition.

(2) Transflective Mode

* Ambient light is taken from the outside during day or in the dark and a back light is used in the dark.

(3) Transmissive Mode



Back-lighting only. In case of B, no reflector is used.

*A back light is always used.

Connector and LCD Mounting Method

To connect LCD to the drive circuit, following connectors are available.

Rubber Connector

LCD Mounting Method (example)

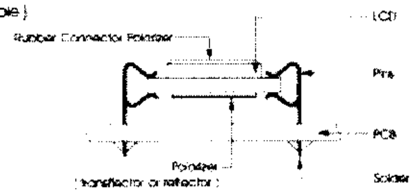
Structure: Alternate Lamination of conductive rubber and insulting rubber.
Connecting Method: Mechanical compression
Pitch (mm): Min 0.4

- Easy to assemble
- Adopted for many year
- Applicable even to narrow pads.
- Printed circuit boards need god plating or graphite coating

Pin Connector

LCD Mounting Method (example)

Structure: Metal pins fit onto the panel terminal pads.
Connecting Method: Soldering
Pitch (mm): 1.5, 1.8, 2.0, 2.54

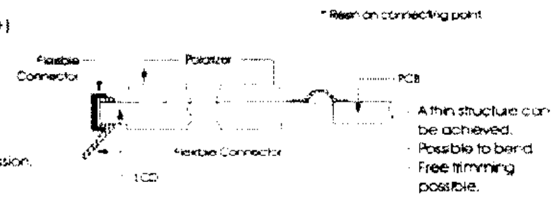


- Suitable for small production runs.

Flexible Connector

LCD Mounting Method (example)

Structure: Film with electroconductive thin film or printed graphite.
Connecting Method: Heat and pressure fitting. Soldering or mechanical compression.
Pitch (mm): Anisotropic Type : Min 1.25



- A thin structure can be achieved.
- Possible to bend
- Free trimming possible.

APPENDIX 7

HOLTEK

3^{18} Series of Encoders

Features

- Operating voltage: 2.4V~12V
- Low power and high noise immunity CMOS technology
- Low standby current
- Three words transmission
- Built-in oscillator needs only 5% resistor
- Easy interface with an RF or infrared transmission media
- Minimal external components

Applications

- Burglar alarm system
- Smoke and fire alarm system
- Garage door controllers
- Car door controllers
- Car alarm system
- Security system
- Cordless telephones
- Other remote control systems

General Description

The 3^{18} encoders are a series of CMOS LSIs for remote control system applications. They are capable of encoding 18 bits of information which consists of N address bits and 18–N data bits. Each address/data input is externally trinary programmable if bonded out. It is otherwise set floating internally. Various packages of the 3^{18} encoders offer flexible combinations of programmable address/data to meet various application needs. The programmable address/data is transmitted together with the header bits via an RF or an infrared transmission medium upon receipt of a trigger signal. The capability to select a TE trigger type or a DATA trigger type further enhances the application flexibility of the 3^{18} series of encoders.

Selection Table

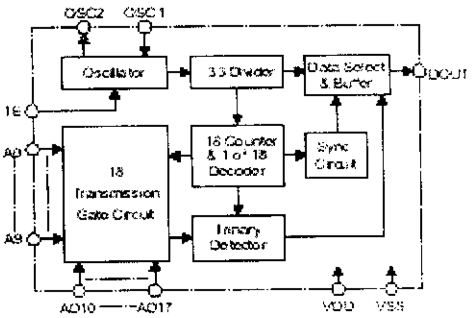
Function / Part No.	Address No.	Address/ Data No.	Data No.	Dummy Code No.	Oscillator	Trigger	Package
HT600	9	5	0	4	RC oscillator	TE	20 DIP/20 SOP
HT640	10	8	0	0	RC oscillator	TE	24 SOP/24 SDIP
HT680	8	4	0	6	RC oscillator	TE	18 DIP
HT6187	9	0	3	6	RC oscillator	D12,D14,D15	18 DIP/20 SOP
HT6207	10	0	4	4	RC oscillator	D12~D15	20 DIP/20 SOP
HT6247	12	0	6	0	RC oscillator	D12~D17	24 SOP/24 SDIP

Note: Address/Data represents addressable pins or data according to the decoder requirements.

Block Diagram

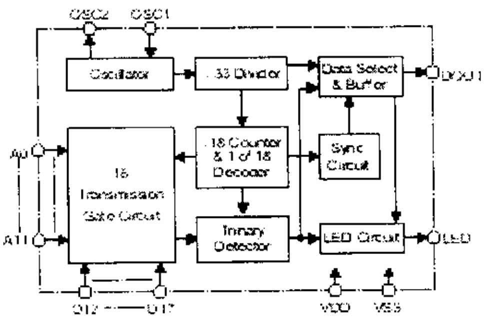
TE trigger

HT600/HT640/HT680



DATA trigger

HT6187/HT6207/HT6247

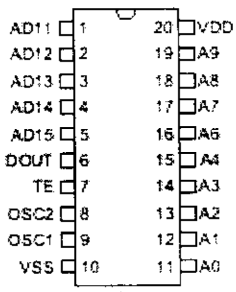


Note: The address/data pins are available in various combinations.

Pin Assignment

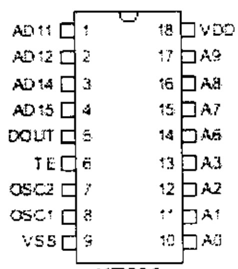
TE trigger type

9-Address
5-Address/Data



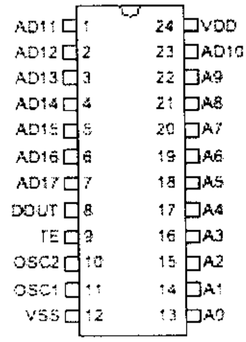
HT600
– 20 DIP/SOP

8-Address
4-Address/Data



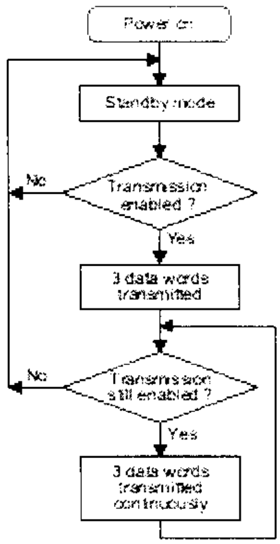
HT680
– 18 DIP/SOP

10-Address
8-Address/Data



HT640
– 24 SOP/SDIP

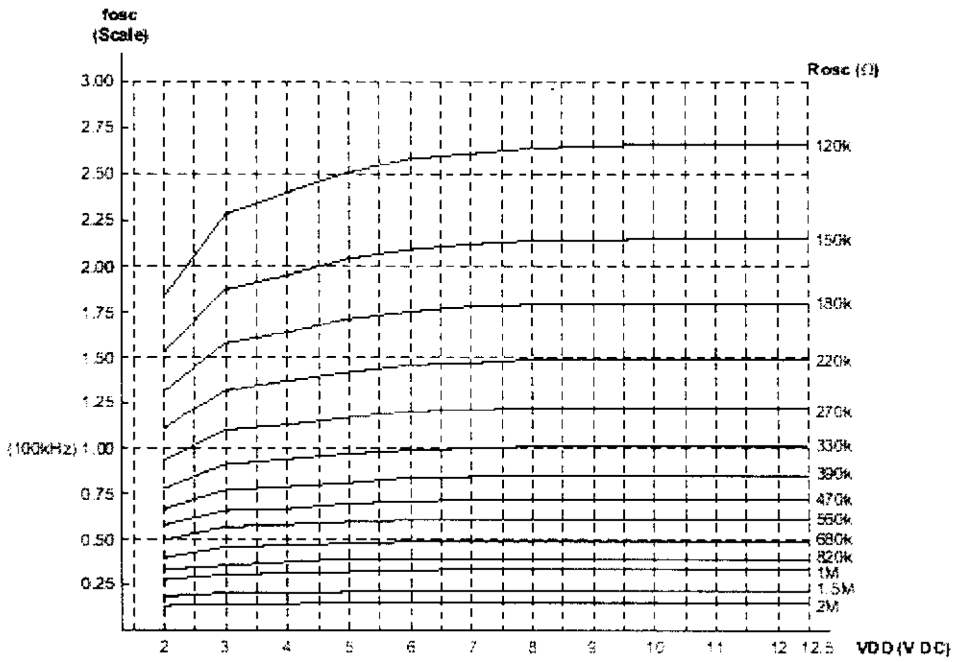
Flowchart



Notes: D12~D17 are transmission enables of the HT6187/HT6207/HT6247.

TE is the transmission enable of the HT600/HT640/HT680.

Oscillator frequency vs supply voltage



The recommended oscillator frequency is f_{OSCD} (decoder) ≅ f_{OSCE} (encoder).

APPENDIX 8

HOLTEK

3^{18} Series of Decoders

Features

- Operating voltage: 2.4V~12V
- Low power and high noise immunity CMOS technology
- Low standby current
- Capable of decoding 18 bits of information
- Pairs with HOLTEK's 3^{18} series of encoders
- 8~18 address pins
- 0~8 data pins
- Trinary address setting
- Two times of receiving check
- Built-in oscillator needs only a 5% resistor
- Valid transmission indictor
- Easily interface with an RF or an infrared transmission medium
- Minimal external components

Applications

- Burglar alarm system
- Smoke and fire alarm system
- Garage door controllers
- Car door controllers
- Car alarm system
- Security system
- Cordless telephones
- Other remote control systems

General Description

The 3^{18} decoders are a series of CMOS LSIs for remote control system applications. They are paired with the 3^{18} series of encoders. For proper operation a pair of encoder/decoder pair with the same number of address and data format should be selected (refer to the encoder/decoder cross reference tables).

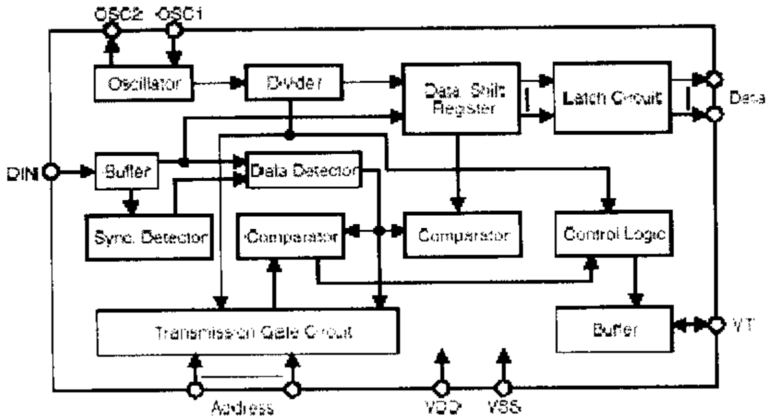
The 3^{18} series of decoders receives serial address and data from that series of encoders that are transmitted by a carrier using an RF or an IR transmission medium. It then compares the serial input data twice continuously with its local address. If no errors or unmatched codes are encountered, the input data codes are decoded and then transferred to the output pins. The VT pin also goes high to indicate a valid transmission.

The 3^{18} decoders are capable of decoding 18 bits of information that consists of N bits of address and 18-N bits of data. To meet various applications they are arranged to provide a number of data pins whose range is from 0 to 8 and an address pin whose range is from 8 to 18. In addition, the 3^{18} decoders provide various combinations of address/data number in different packages.

Selection Table

Function / Item	Address No.	Data No.	Data Type	VT	Oscillator	Trigger	Package
HT602L	12	2	L	√	RC oscillator	DIN active "Hi"	20 DIP/20 SOP
HT604L	10	4	L	√	RC oscillator	DIN active "Hi"	20 DIP/20 SOP
HT605L	9	5	L	√	RC oscillator	DIN active "Hi"	20 DIP/20 SOP
HT611	14	0	—	√	RC oscillator	DIN active "Hi"	20 DIP/20 SOP

Block Diagram

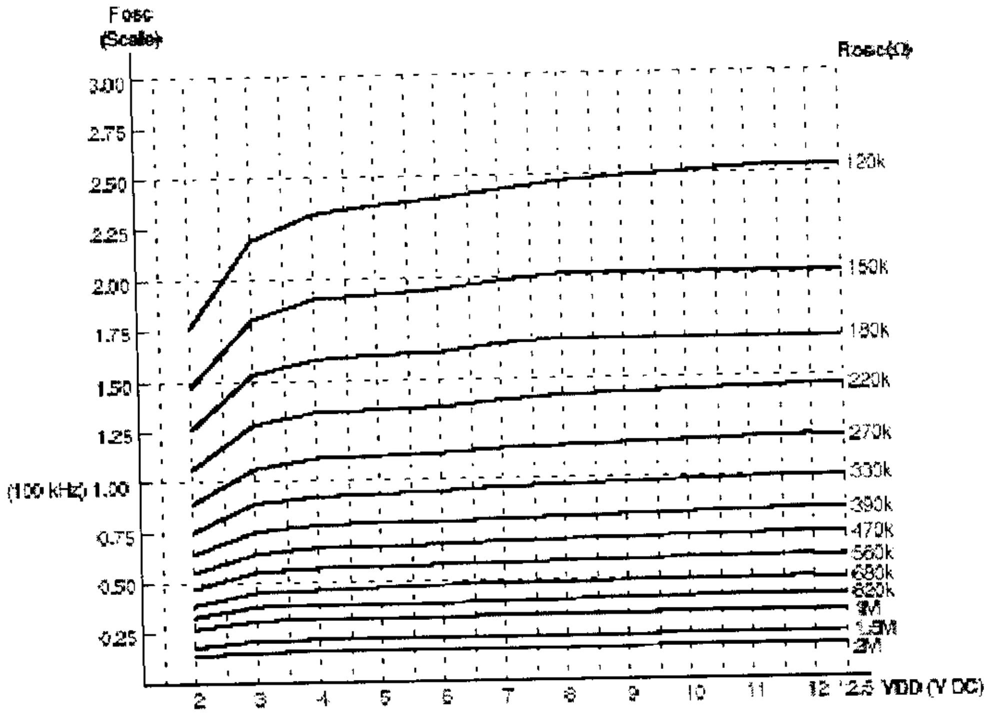


Note: The address/data pins are available in various combinations (refer to the address/data table).

Pin Description

Pin Name	I/O	Internal Connection	Description
A0~A17	I	TRANSMISSION GATE	Input pins for address A0~A17 setting They can be externally set to VDD, VSS, or left open.
D10~D17	O	CMOS OUT	Output data pins
DIN	I	CMOS IN	Serial data input pin
VT	O	CMOS OUT	Valid transmission, active high
OSC1	I	OSCILLATOR	Oscillator input pin
OSC2	O	OSCILLATOR	Oscillator output pin
VSS	I	—	Negative power supply (GND)
VDD	I	—	Positive power supply

Oscillator frequency vs supply voltage



The recommended oscillator frequency is F_{OSCD} (decoder) ≅ F_{OSCE} (encoder).