# Road Traffic Rules Violators' Monitoring System

by

MUHD HAFIZ BIN MOHD KHALILI

Dissertation Submitted in partial fulfilment of
the requirements for the
Bachelor of Engineering (Hons)
(Electrical and Electronics Engineering)

JUNE 2004

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

**CERTIFICATION OF ORIGINALITY**

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

MUHD HAFIZ BIN MOHD KHALILI

# ABSTRACT

This purpose of this system is to catch vehicle, which breaks the red traffic signal by sending the unique characters (e.g. vehicle's registration number) to the database and keep the record for future action. The main objective of this project is to transmit alphanumeric characters through RF medium and display the characters on the LCD matrix at the receiver.

The system is using digital RF transmitter and receiver integrated with 16F84A PIC micro controller for the ASCII code conversion, data encoding and decoding task. Thus, the next objective is to develop the programming code for the 16F84A to perform all these tasks. In the project, the assembly language is used to develop the programming code.

The VMS operates only when the traffic signal is red. In this project, a switch is used to trigger the transmitter in the vehicle at that particular time. The switch is only working under certain state condition (RED light is ON). In the future and practical used, the switch can be replaced with a Hall sensor. The hall sensor detects the magnetic flux introduced by the electromagnet, which is installed under the road surface.

The first section of the report will tell about the introduction of background of the project, the concept of RTRVMS, problem statement, objectives and scopes of study. The second section of the report is literature review of the components used in the system. The third section will be the methodology and project work including the project development. The forth section is the project findings and discussion throughout this semester and also the problem facing.

This simple system can be one of the solutions to control the road traffic violent in the future.

# ACKNOWLEDGEMENT

Thank God finally I've completed my final year design project (FYP). I would like to gratefully acknowledge the contributions of several people who have supported me to finish the two semester's course and finally completing this project. First, a million thanks to Universiti Teknologi PETRONAS (UTP) for providing me the place and resources to complete my project.

Thousand thanks to Dr. Nazir Ahmed Arain as my supervisor, for his great teaching and support. Great thanks for his supervision throughout my FYP and also for helping me to achieve the project objectives.

Special thanks to Mr. Zuki, a lecturer of giving me a support and knowledge in using PIC microcontroller. Thanks to Miss Hawa for their permission using the lab and providing the electronic components for my FYP. Thank you to Universiti Teknologi Petronas (UTP) lecturers and staffs, whom directly or indirectly involved in the FYP course.

Not forget, thank to my colleague, whom helping and be pleased to share their project experiences and knowledge with me. The knowledge and hand-on experiences in this project give me lots of benefit and very useful for my future working career.

## TABLE OF CONTENT

**List of Figures**



**List of Tables**

**List of Abbreviations**

| | |
|---|---|
| AM | Amplitude Modulation |
| ASCII | American Standard Code for Information Interchange |
| CPU | Central Processing Unit |
| DLE | Data Link Escape |
| EEPROM | Electrically Erasable Programmable Read Only Memory |
| EPROM | Erasable Programmable Read Only Memory |
| ETX | End of Text |
| FET | Field-effect Transistor |
| FM | Frequency Modulation |
| IC | Integrated Circuitry |
| I/O | Input/Output |
| LCD | Liquid Crystal Display |
| LED | Light Emitting Diode |
| PIC | Programmable Integrated Circuit |
| PLL | Phase-locked Loop |
| PM | Phase Modulation |
| RAM | Random Access Memory |
| RF | Radio Frequency |
| ROM | Read Only Memory |
| RTRVMS | Traffic Rules Violators' Monitoring System |
| SSB | Single Side Band |
| STX | Start of Text |
| TMS | Traffic Monitoring System |
| UARTS | Universal Asynchronous Receive/ Transmit |
| VCO | Voltage Controlled Oscillator |
| VMS | Vehicle Monitoring System |

# CHAPTER 1
# INTRODUCTION

## 1.1 Background

There are people driving on the road but do not care about the traffic light rules such as stop their vehicle when the traffic signal is red. Therefore, an electronic intelligent system can be developed to monitor traffic-rule violators and keep their record. The system can be made smart enough to identify the habitual rule-violators.

Currently, the camera systems are widely being used to catch the driver. The existing system required human to monitor the system and the installation cost is higher. Therefore, the Road Traffic Rules Violators' Monitoring System (RTRVMS) is designed with the lower cost and perhaps capable to replace the camera later.

## 1.2 RTRVMS Concept

This system is based on the communication between the vehicle's monitoring system (VMS) and the traffic monitoring system (TMS) using radio waves. When the traffic light is 'GREEN', the TMS will not send signal to activate the VMS located in the vehicles.

If the traffic signal is 'RED', the TMS will send a signal to activate the VMS in vehicles near to that location. If there is a vehicle still try to pass through, the VMS will automatically send the signal and the TMS will receive the unique ID number (e.g. vehicle's registration number) of that particular car and send it to central database located somewhere else.

The vehicle will only transmit its number when it's within the traffic-signal range, and traffic-signal will record the vehicle's number when the traffic signal is 'RED' and the vehicle is crossing the traffic-signal.

### 1.3 RTRVMS Block Diagram

Basically, the block diagram of the transmitter and receiver is integrated with the electronic and microcontroller circuit to transmit the data as shown in figure 1.1.

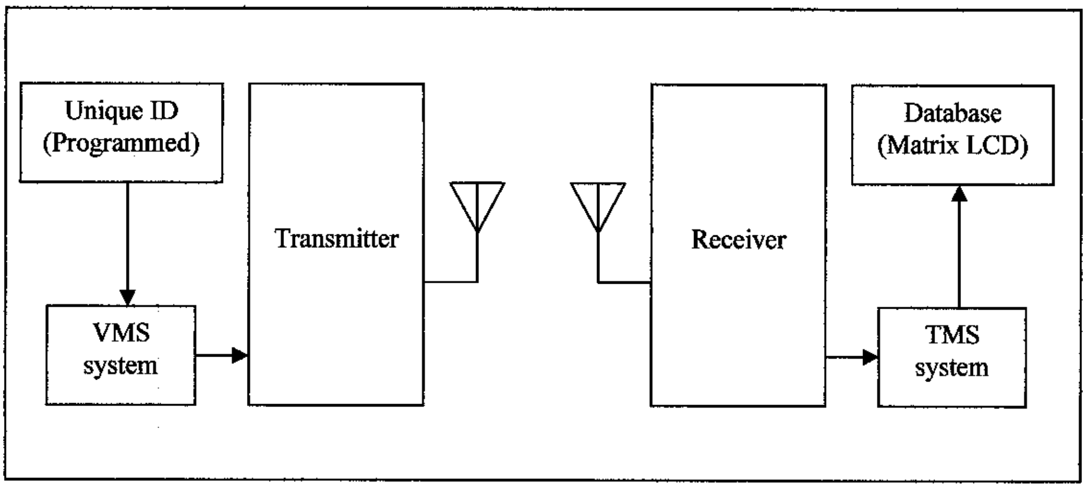



Figure 1.1: System Block Diagram

**Description**

The data is stored in the programme. The data is converted into binary digits, which in ASCII code standard as shown in the figure 1.1. The microcontroller will process data into digital signal for serial transmission. The digital RF transmitter will transmit the data signal to the digital RF receiver bit by bit. The receiver will receive the data bit by bit and store in the temporary register. Then, the bits are converted into the specific number according to the ASCII code for the record (display on the matrix LCD).

Note that the keypad and matrix LCD is used for experimental purpose. It is used to test the accuracy of transferred data. In the real operation, the unique ID is stored permanently in the chip. This is for security reason and the unique ID can only being changed by an authorised person, such as policeman and Road and Transport Department.

### 1.4 Problem Statement

In order to communicate between two systems, digital RF transmitters and receivers is the devices to transfer the data. The VMS should capable to transfer alphanumeric characters and receive by the TMS at one time. The signal should be digitally transferred the data in binary digit using ASCII code. The study needs to be done on RF communication area, hardware requirements and availability in the market.

Basically, the VMS and TMS systems are using transmitter and receiver and will be integrated with other hardware such as micro controller for the ASCII code conversion, data encoding and decoding and data processing. Some R&D works need to be done on microcontroller hardware and programming.

The VMS system is only activated when the traffic signal is 'RED'. A system can be designed to catch cars which cross the 'RED' light. Meanwhile, other cars VMS should not transfer the data simultaneously.

The project is focused on the system design and development using components available in the market. The components such as transmitter and receiver need not to be designed but some programming for microcontroller is involved in this project.

There is an experiment to test the ability of transferring the data using digital signal. It will be conducted using keypad connected to the transmitter as an input of the alphanumeric characters and the receiver will display the characters matrix LCD respectively.

## 1.5 Objectives and Scopes of Study

### 1.5.1 Objectives

a. To monitor and catch vehicle that breaks the red traffic signal
b. To transfer the alphanumeric characters using radio frequency (RF) medium
c. To develop a system that can transmits and receives the characters using ASCII code

### 1.5.2 Scopes of Study

Through out this project, there are certain studies need to be covered. The study scopes will cover as follows:

a. Theory of Digital Transmitter and Receiver or other relevant communication devices
b. Alphanumeric Characters to Binary Digit in ASCII code conversion using PIC 16F84A microcontroller
c. Microcontroller (encoding, decoding, data transfer)
d. Other supporting electronic components and devices to the system
e. Alternatives system design

# CHAPTER 2
# LITERATURE REVIEW

## 2.1 Transmitter Basic building block

The basic building blocks to be discussed are as follows:

- Oscillators
- Amplifier
- Multipliers
- Modulator
- Frequency synthesizer (PLL) circuits

### 2.1.1 Oscillators

Oscillators are necessary for generating the necessary RF signal. A good oscillator must have the following characteristics:

a. Good frequency stability
b. Adequate power output
c. Spectral purity
d. Simplicity

Frequency-determining elements are those elements in an oscillator circuit that affect or are intended to determine the oscillator frequency, such as LC tank circuit or a crystal. These elements must have as high Q factor as possible [1]. The stability of an oscillator is proportional to the rate of phase change with frequency in the oscillator loop, which consists of an amplifier and a feedback network [1]. The better oscillators in this regard have high Q frequency-determining elements that dominate the loop characteristics. Crystal-controlled oscillators are the best and simple oscillators [1].

### 2.1.2 Amplifier

Amplifiers used for low-power transmitters may be either of IC or discrete transistor construction.

### 2.1.3 Multipliers

Frequency multipliers are basically nonlinear circuit that produce harmonics of the operating frequency. A filter, which usually comprises tuned circuits, picks out the desired harmonic and rejects the input frequency and all other harmonics. Multipliers are normally used to double or triple the input frequency, but higher orders of multiplication are often performed. Multipliers may consist of discrete transistor amplifier stages, Schottky (hot carrier) diodes, varactors and snap diodes [1].

### 2.1.4 Modulator

A modulator is a circuit used to superimpose information on a carrier waveform. The amplitude of the carrier, a sine wave may be modulated with this modulating waveform (AM or SSB), or the instantaneous frequency (FM) or phase (phase modulation, PM) of the carrier wave can be modulated with it [1]. Another possibility is that the carrier wave may be simply turned on and off (pulse or CW).

### 2.1.5 Frequency synthesizer (PLL) circuits

Frequency synthesizer circuits are feedback servo loops in which a voltage controlled oscillator (VCO) generates a signal whose frequency is locked to a fixed reference signal. A signal sample from the VCO feeds a frequency that is suitable for the frequency divider circuit, which can be made to have a fixed or variable divide ratio (called N) and is compared to a crystal oscillator for both frequency and phase [1].

## 2.2 American Standard Code for Information Interchange (ASCII)

Data is transmitted in codes. To transmit characters for standard **text** and associated **control** characters (e.g."return", "backspace", "delete", "start-of-text", etc.), ASCII defines a 7-bit code word [3].

A subset of these are the standard printable characters (upper/lower case alphabet, numerals, punctuation and special symbols), the rest are control characters for formatting or used to control transmission (We may add an 8th redundant bit to provide a degree of **error control.** The 8th bit may be defined to give the resulting 8-bit **byte** "even" **parity,** say, so that the number of binary 1's in the word is an even number. "Odd" parity may also be used, or a dummy 8th bit may be inserted.) [3].

Nevertheless there is no such evident way to represent letters with **0**s and **1**s [3]. Then, to be able to do that, computers use the ASCII code, that is a table or list that contains all the letters in the alphabet plus other additional characters. In this code each character is always represented by the same order number. For example, in ASCII code the capital letter **A** is always represented by the order number **65**, which is easily representable using **0**s and **1**s in binary (**1000001**).

The standard ASCII code defines 128 character codes (from 0 to 127), of which, the first 32 are control codes (non-printable), and the other 96 are representable characters:

| * | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | N<br>UL | SO<br>H | ST<br>X | ET<br>X | EO<br>T | EN<br>Q | A<br>C<br>K | BE<br>L | BS | TA<br>B | LF | VT | FF | CR | SO | SI |
| 1 | DL<br>E | D<br>C1 | D<br>C2 | D<br>C3 | D<br>C4 | N<br>A<br>K | SY<br>N | ET<br>B | C<br>A<br>N | E<br>M | SU<br>B | ES<br>C | FS | GS | RS | US |
| 2 | | ! | " | # | $ | % | & | ' | ( | ) | * | + | , | - | . | / |
| 3 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | : | ; | < | = | > | ? |
| 4 | @ | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O |
| 5 | P | Q | R | S | T | U | V | W | X | Y | Z | [ | \ | ] | ^ | _ |
| 6 | ` | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o |
| 7 | p | q | r | s | t | u | v | w | x | y | z | { | \| | } | ~ | |

Table 2.1: ASCII code [3]

This panel is organized to be easily read in hexadecimal: row numbers represent the first digit and the column numbers represent the second one. For example, the **A** character is located at the **4th**row and the **1st** column, for that it would be represented in hexadecimal as **0x41** (**$65_{10}$**).

## 2.3 Digital Data Transmission

### 2.3.1 SYNCHRONIZATION

A sequence of binary digits transmitted at some regular rate over a link in the form of an electrical signal (pulse sequence) and has to be recovered at the receiver by manipulating the electrical signal [1]. The important consideration here is that of **clocking or timing** of the instants at which the receiver will be looking for individual bits [3]. If the receiver clock is running at a rate different from that of the transmitter, or if there is an offset between the two clocks even when they are running at the same rate, we will not be able to recover the bits properly.

There are two approaches to obtain the necessary timing information [3].

- In **asynchronous** transmission the receiver has a **free-running clock** that is running nominally at a fixed multiple of the transmitter clock rate (e.g. transmitter running at 9600 clock cycles/sec, receiver clock running at 8 times this rate, nominally 76,800 cycles/sec). Short bit sequences are emitted with framing bits that allow the receiver to know when to start looking for bits. For each short sequence, provided the receiver starts at approximately the correct time, it can maintain approximately correct timing for the duration.

- In **synchronous** transmission, transmit **clock information** has to be **embedded** within the bit stream, so that the receiver can either **extract** a clock signal for its use or can use the embedded clock information to **fine-tune its own local clock** to keep it in synchronism with the transmitter clock.

### 2.3.2 ASYNCHRONOUS TRANSMISSION

Bits are represented by one of two electrical states on the line or different voltage levels [3]. Suppose the line is in "idle" condition (in logic "high" or "1" state, say. This means that the electrical state of the line is that corresponding to bit "1" on the line. An idle line has to be in some electrical state.) Then one 7 or 8 bit data group is to be sent. These bits are **preceded** by a "start bit' (logic "low" or 0) and **followed** by one to two "stop" bits (logic "high" or 1). The receiver operates by looking for the

**first transition from "high" to "low"** after its previous idle or "stop" bits; it continuously samples at rate nominally N times the transmit clock rate. Upon finding the first transition from high to low, it sets the next sampling instant to be at N/2 receive clock samples (nominally half of transmit clock period) after the first transition; this should be near the center of the "start" bit. It then samples at multiples of N receiver clock pulses after this for the required number of data bits. The ending stop bits return the line to logic high (idle state value), so that the next character or byte will again produce a high to low transition with its start bit [3].

The receiver works better (samples more nearly at the center of the bit intervals) if N is larger; receiver clocks with N=8 or 16 are typical. This is because the faster the receiver clock is, the sooner after the actual transition from "high" to "low" it will discover it [3]. The circuits that provide this type of transmitter and receiver function are called UARTS (Universal Asynchronous Receive/ Transmit) [1].

The main point here is that the receiver clock does not actually extract a clock signal from the incoming signal, nor does it try to synchronize its own clock with the transmitter clock [3]. It assumes that its clock is reasonably accurate to maintain approximately correct timing over short spans of a few bits.

### 2.3.3 Frame Synchronization in Asynchronous Transmission

If a block of characters is to be sent and received as one frame of data, we need a means to signal to the receiver the beginning and end of a frame. This can be accomplished by sending special characters "STX" (start of text) and "ETX" (end of text) when transmitting printable ASCII characters (because these special characters do not occur in the middle of such data) [3].

To transmit arbitrary binary digits, this needs to be modified, because any of the special control characters may occur within the data itself. For this we can use **character stuffing** [3]. The special DLE control character (data link escape) is used to precede the STX and ETX characters marking frame start and stop. The receiver looks for these two-character combinations to determine frame boundaries.

During the transmission, the transmitter inspects the outgoing bytes for the DLE character. Whenever this is found, a second DLE is stuffed in before the next byte. At the receiver, any pair of DLE's after frame start is decoded as a single DLE. The DLE-ETX pair of course denotes end of frame.

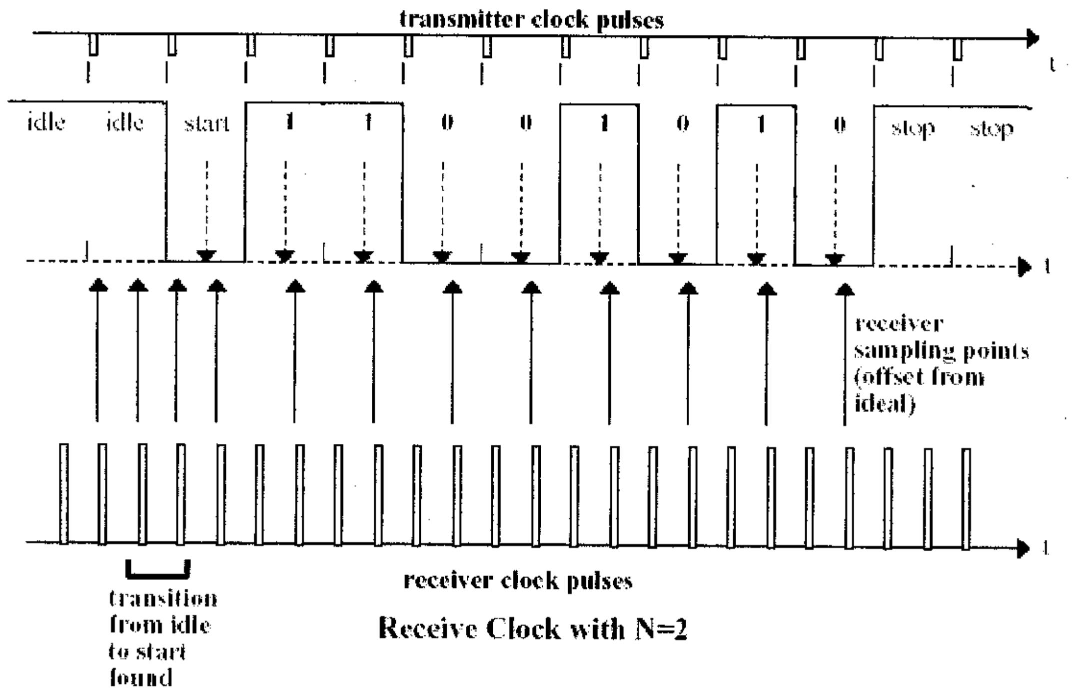



Figure 2.1: Asynchronous Transmission (Start/Stop) [3]

"Direction of transmission" labels in figures 2.1 always point in the opposite direction of "time". Earlier events are seen at the receiver before later events; thus the data bits in the figure arrive at the receiver as 11001010. We could have shown a "direction of transmission" as an arrow pointing to the left. The transmitter and receiver clock waveforms generated by the timing circuits are usually a periodic square-wave at the clock frequency. The actual sampling instants are derived from the clock waveform by looking only at the rising edges or only at the falling edges (transitions) of the clock waveform, which occur once each clock cycle.

## 2.4 PIC 16F8X microcontroller

PIC16F8X devices have enhanced core features, eight-level deep stack and multiple internal and external interrupt sources. The separate instruction and data buses of the Harvard architecture allow a 14-bit wide instruction word with a separate 8-bit wide data bus [7]. The two stage instruction allows all instructions to execute in a single cycle, except for program branches (which require two cycles). A total of 35 instructions (reduce instruction set) are available. Additionally, a large register set is used to achieve a very high performance level [7].

The PIC16F8X has up to 68-bytes of RAM, 64 bytes of Data EEPROM memory and 13 I/O pins. A timer/counter is also available [7]. The PIC16CXX family has special features to reduce external components, thus reducing cost, enhancing system reliability and reducing power consumption. There are four oscillator options of which the single pin RC oscillator provides a low-cost solution, the LP oscillator minimize power consumption, XT is a standard crystal and the HS is for High Speed crystal [7]. The SLEEP (power-down) mode offers power saving. A highly reliable Watchdog Timer with its own on-chip RC oscillator provides protection against software lock-up.

The devices with Flash program memory allow the same device package to be used for prototyping and production. The PIC16F8X fits perfectly in applications ranging from high speed automotive and appliance motor control to low-power remote sensors, electronic locks, security devices and smart cards [7]. The Flash/EEPROM technology makes customization of application programs (transmitter codes, motor speeds, receiver frequencies, security codes, etc.) extremely fast and convenient [7]. The small footprint packages make this microcontroller series perfect for all applications with space limitation. The PIC16CXX family is supported by a full-featured macro assembler, a software simulator, an in-circuit emulator, a low-cost development and a full-featured programmer [7].

**The details on PIC16F84 is in Appendix C*

## 2.5 Hall Effect Sensor

### 2.5.1 Introduction

The function of a Hall sensor is based on the physical principle of the Hall Effect named after its discoverer E. H. Hall. Refer to figure 2.2, a voltage is generated transversely to the current flow direction in an electric conductor (the Hall voltage), if a magnetic field is applied perpendicularly to the conductor [5]. As the Hall Effect is most pronounced in semiconductors, the most suitable Hall element is made of semi conductive material [5].

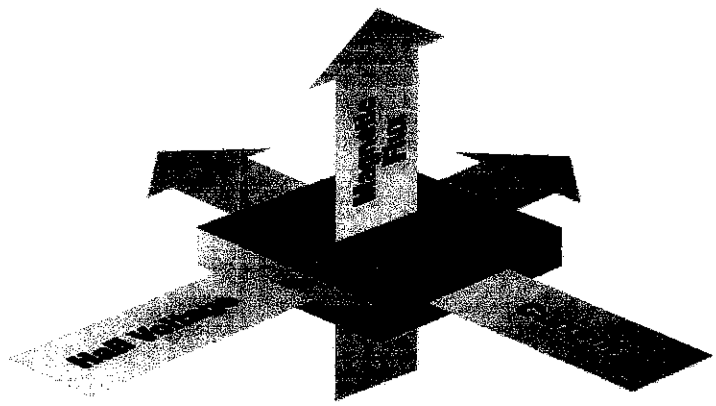


Figure 2.2: The Hall Effect [5]

In a semi conductive platelet, the Hall voltage is generated by the effect of an external magnetic field, which is acting perpendicularly to the direction of the current [5].

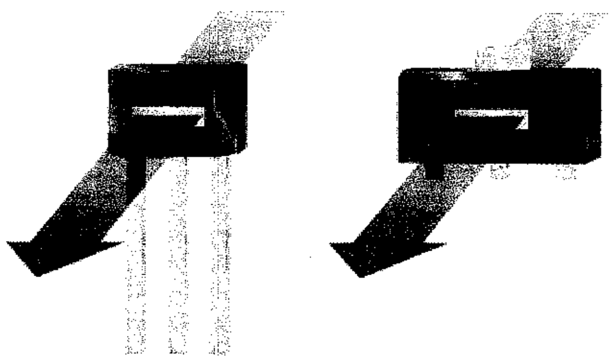


Figure 2.3: The magnetic flux component perpendicular to the chip surface is measured (arrows) [5]

The Hall plate with the current terminals and the taps for the Hall voltage are arranged on the surface of the crystal. This sensor element detects the components of the magnetic flux perpendicular to the surface of the chip and emits a proportional electrical signal which is processed in the evaluation circuits integrated on the sensor chip as shown in figure 2.3 [5].

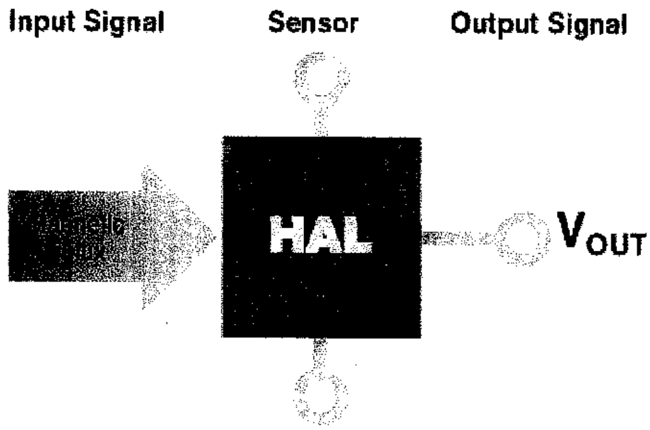



Figure 2.4: Functional principle of a Hall sensor [5]

In figure 2.4, it shows that the output voltage of the sensor and the switching state, respectively, depend on the magnetic flux density through the Hall plate [5].

The different types of Hall sensors are distinguished depending on the mode of signal processing and signal output. Fundamentally, the sensors can be divided into Hall Switches and linear Hall sensors.

### 2.5.2 The Hall Sensor Working Principal

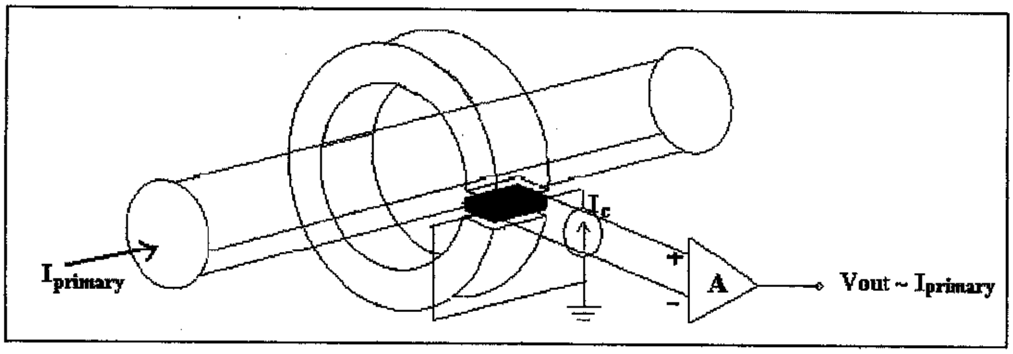



Figure 2.5: Open Loop Hall Effect Current Sensor

The current generates magnetic flux surrounding the electromagnet [5]. Once the car passes through the electromagnet, the sensor will sense the flux and turn the $V_{out}$ as a high output as in figure 2.5. The signal is received by the microcontroller and interprets it as logic 1 ('High' state). Hence, the data is transmitted.

### 2.5.3 Hall switches

Hall switches (switching sensors) have an integrated comparator with predefined switching points and a digital output which can be adapted to different logic systems. All Hall switches include an open-drain output transistor and require an external pull-up resistor to the supply voltage. A standard Hall switch has a single Hall plate and responds to the absolute value of the magnetic field perpendicular to the plate. The Hall switch is characterized by the magnetic switching points $B_{ON}$ (or $B_{OP}$) and $B_{OFF}$ (or $B_{RPN}$) [5].

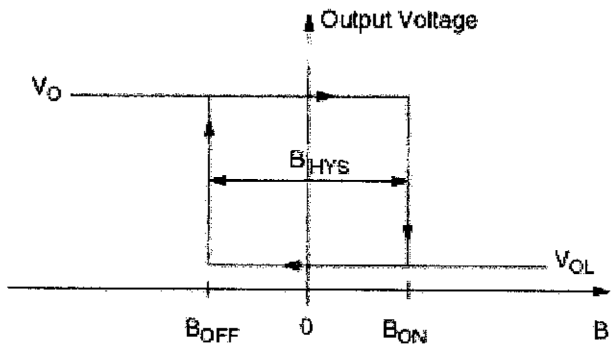



Figure 2.6: Definition of switching points

Refer to figure 2.5, if the magnetic flux exceeds $B_{ON}$, the output transistor is switched on; on dropping below $B_{OFF}$, the transistor is switched off. The magnetic hysteresis $B_{HYS}$ is the difference between the switching points $B_{ON}$ and $B_{OFF}$ [5].

### 2.5.4 Electrical Interface for Digital Hall Devices

The output stage of a digital Hall switch is simply an open-collector NPN transistor. The rules for use are the same as those for any similar switching transistor. When the transistor is OFF, there is a small output leakage current (typically a few nanoamperes) that usually can be ignored, and a maximum (breakdown) output voltage (usually 24 V), which must not be exceeded [9].

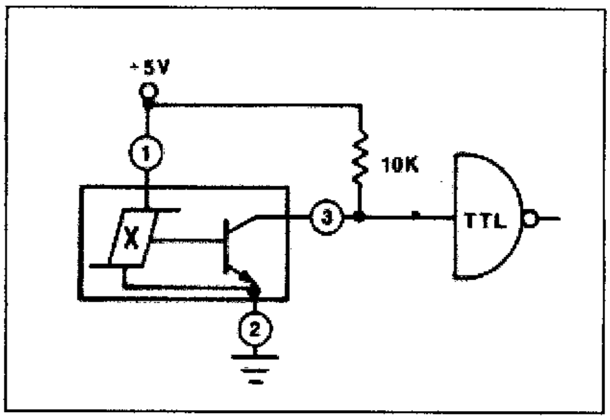



Figure 2.7: Common interface circuit [9]

When the transistor is ON, the output is grounded. The current flowing through the switch must be externally limited to less than a maximum value (usually 20 mA) to prevent damage. The voltage drop across the switch ($V_{CE}$(sat)) will increase for higher values of output current [9]. The voltage should be compatible with the OFF, or "logic zero," voltage of the control circuit.

With current-sinking logic families, such as DTL or the popular 7400 TTL series (figure 2.7), the Hall switch has only to sink one unit-load of current to the circuit common when it turns ON (1.6 mA maximum for TTL) [9]. Loads that require sinking currents up to 20 mA can be driven directly by the Hall switch [9].

## 2.6 PIC Output Interfacing

Many output devices will require a transistor switching circuit. In most cases a Darlington pair, formed from two transistors is ideal (figure 2.8) [11].

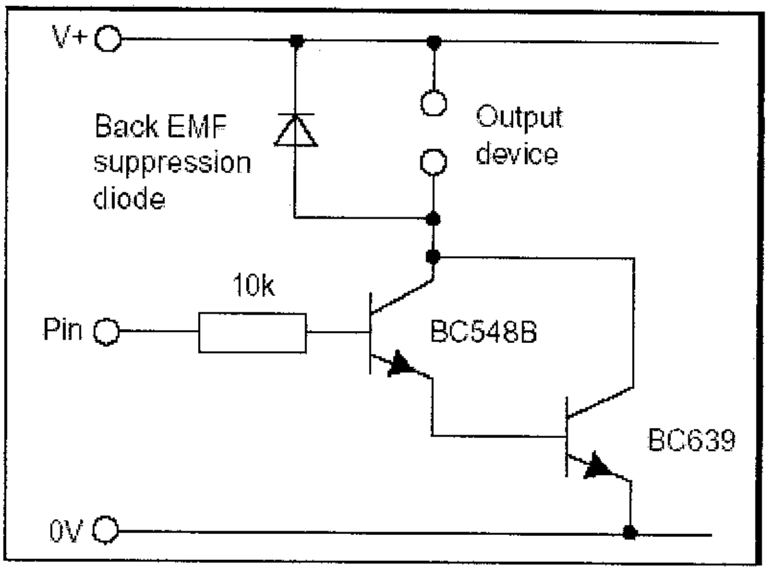



Figure 2.8: Transistor Switching Circuit [11]

If a number of output devices are being controlled it may be necessary to use a number of output transistors. In this case it will often be more convenient to use a (ULN2003) Darlington driver IC. This is simply a 16 pin 'chip' that contains 7 Darlington transistors similar in value to the BCX38B. The 'chip' also contains internal back e.m.f. suppression diodes and so no external (1N4001) diodes are required [11].

# CHAPTER 3
# METHODOLOGY AND PROJECT WORK

## 3.1 Procedure Identification

This is a practical based project and has the time frame. It must complete within the given time. The project is managed in stages including:

a. Literature Review

- The research is conducted by collecting the information from engineering books, catalogues, internet and other relevant sources.
- Theoretically there are seven steps in process design, which are:
  - Evaluate problem statement
  - Define the problems
  - Conduct a research
  - Analyze existing solution
  - Generate possible solution
  - Select the best solution
  - Design the selected solution

b. Conceptual Design

- The first sketch of the system is developed. The system is designed in schematic drawing and block diagram.
- The design shouldn't be in detailed but need to comply with the requirement given.
- The project understanding is very important at this stage.

c. Project Experiment

- The design is brought to the lab for experimental task.
- The result analysis is very important before proceed to the next stage.
- This stage should be completed at the end of the semester one and come out working model.

d. Detailed Design
   - All main components of the system are already firmed. The overall system is designed with detailed specification.

e. System Simulation and Implementation
   - The design is transferred on board and being simulated. At this stage, the real time error is recognised. The simulation is done until the system running correctly.
   - The practical approach is followed when the simulation has succeeded.

## 3.2 Hardware/ Software Tools

In order to complete this project, there are several tools (hardware/software) might be used for certain tasks such as:

### 3.2.1 Electronic Components

Hardware simulation will need power supply and electronic components. The components can be obtained from the lab. Some component such as digital RF transmitter and receiver, which is not available in the lab, need to be bought from the outside.

### 3.2.2 Microcontroller

There will be Programmable Integrated Circuit (PIC) microcontroller implementation in this project. The advantages of using the PIC are easy to be built, smaller size and work successful for digital processing.

### 3.2.3 Programming Software

The programming software is used to program the PIC chip. The chip needs special program to load the assembly language into the chip to make it function. The programming can be in C language or Assembly language. The program will determine the functionality of the PIC chip.

### 3.2.4 Hall Effect Sensor

The sensor is used to detect the electromagnetic force (emf) wave. It is used to trigger the transmitter in the vehicle in order to transmit the data.

### 3.2.5 Other relevant devices

In future study, there might some component will be added in the system such as sensors, navigation system and intelligent chip.

## 3.3 Project Milestones

This project is two semesters project. The project should be completed within the time given. The phase is divided into two semesters as below.

### 3.3.1 Semester One

The project research should be completed in this semester. All hardware and software has already been determined.

### 3.3.2 Semester Two

The project will focus on system design and microcontroller programming. The board simulation and hardware implementation should be delivered at the end of semester.

**Please refer appendix A for the Gantt chart*

# CHAPTER 4
# RESULTS AND DISCUSSION

## 4.1 Overall RTRVMS System

The RMTRVMS system is built to catch the drivers, who like to break the traffic rules. The focus will be on the scenario at the traffic light. As stated earlier, the system is only operating when the traffic signal is 'RED'. The system will identify the vehicles, which try to pass through the traffic light during the particular condition. The unique ID (vehicle identity e.g: vehicle registration number) is stored in binary code and the transmission method is using digital signal. The unique ID will be send directly to the central vehicle database, thus a compound or summon can be produce automatically.

The system can be divided into two sub-systems which are Traffic Monitoring System (TMS) and Vehicle Monitoring System (VMS) as in figure 4.1. The VMS system is installed in the vehicles. Meanwhile, the TMS system is installed at the traffic light and directly connected to Central vehicle database. The systems are communicating through simplex Radio frequency (RF) medium. In other word, they are using wireless connection.

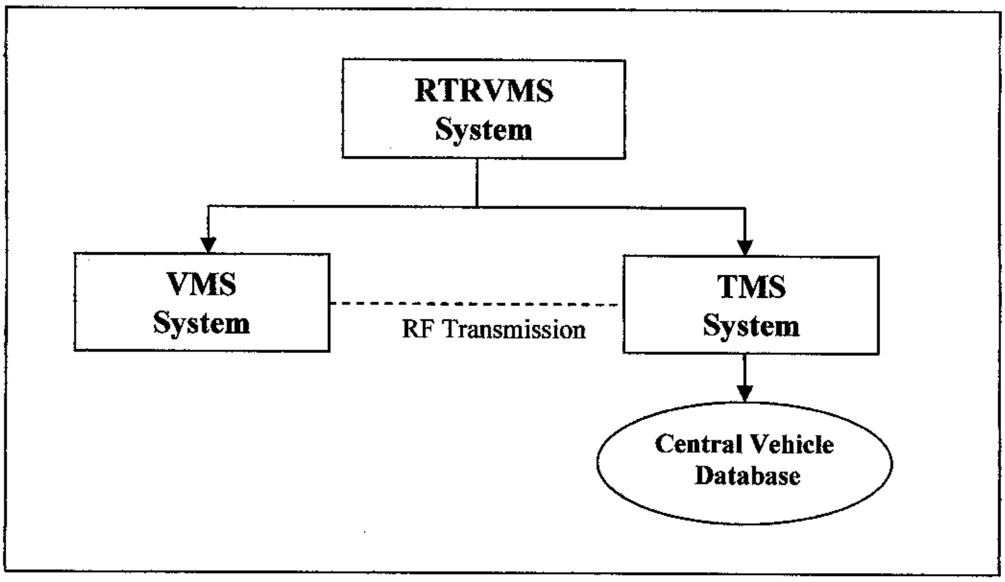



Figure 4.1: RTRVMS System

## 4.2 Existing Monitoring System

From the research, there is existing system, which have similar objective but different technology. The camera system is widely used but it has some disadvantages.

a. The camera captures the vehicle picture to get vehicle registration number. The driver may take advantage by putting the incorrect registration number.
b. A person is needed to monitor the vehicle registration number manually. The possibility of the human error is there.
c. The picture quality may vary due to whether and traffic condition.
d. The installation cost is high.

## 4.3 The Advantages of RTRVMS system

The RTRVMS is installed in the vehicles and also at the traffic light. The system is relatively smaller than the camera system. The advantages are:

a. The size is small and can be hidden from the driver view. The driver won't know the system location.
b. The unique ID is different for each vehicle. The risk of catching the wrong vehicles can be removed.
c. The system will automatically send the number to database. The database can automatically keep the record of that particular vehicle.
d. The installation cost is low since it uses the component that available in the market.

## 4.4 Basic Criteria

The basic criteria needs for the system:

a. The system should be able to transmit and receive 10 alphanumeric characters at one time using RF medium.
b. The reception range should be at least 50 ft radius of the traffic light location.
c. The system should use digital RF transmitter and receiver to transfer the data signal.
d. The system should be an independent (stand-alone) system.

### 4.5 System Requirements

#### 4.5.1 PIC Microcontroller

A PIC microcontroller is a single integrated circuit small enough to fit in the palm of a hand. 'Traditional' microprocessor circuits contain four or five separate integrated circuits – the microprocessor (CPU) itself, an EPROM program memory chip, some RAM memory and an input/output interface. With PIC microcontrollers all these functions are included within one single package, making them cost effective and easy to use.

There are few advantages of using microcontroller, such as:

a. Reduce the complexity and size (small footprint) of the circuit
b. Many functions in one single PIC chip
c. Easy to reprogram the PIC chip
d. The PIC chip is easy to obtain in the market
e. Work comfortably for digital system

PIC microcontrollers can be used as the 'brain' to control a large variety of products. In order to control devices, it is necessary to inter face (or 'connect') them to the PIC microcontroller. Flash microcontrollers are very popular for use in education since they can easily be reprogrammed many times. The most popular ICs are the 18 pin 16F84A and the newer 16F627.

It is found that the microcontroller is the best solution to built the transmitter and receiver circuit. The PIC model used is PIC16F84A. The PIC chip is available in the lab as well as its programmer device.

#### 4.5.2 Digital RF Transmitter and Receiver

The availability of digital RF transmitter and receiver is the problem that encountered last semester. This is product of Laipac Technology, Inc. The RF transmitter model is (TLP434A) and receiver (RLP34A). The transmitter and receiver are integrated with the microcontroller devices.

**The specification of the RF transmitter and receiver is available in the appendix C.*

### 4.5.3 Switch (Hall Effect Sensor)

In this project, the switch is used to trigger the transmitter. It can be replaced with Hall Effect sensor in the future. The sensor senses the presence of electromagnet and converts the signal into specific voltage. The sensor is robust compare to infra red sensor. The open loop Hall Effect current sensor since the power consumption is very little.

### 4.5.4 The Hall Sensor Installation

The size is very small and can be installed underneath the vehicle. The best location is at back of the car, which is far from the engine bay. The purpose is to avoid the sensor sense the electromagnetic generated by the engine components.

The electromagnet should be installed under the road surface in the traffic light area as in figure 4.2 and 4.3. The electromagnet should only energise when the traffic signal is RED.

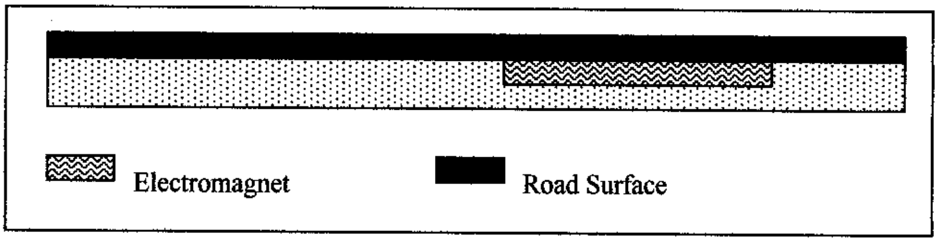



Figure 4.2: The electromagnet installation

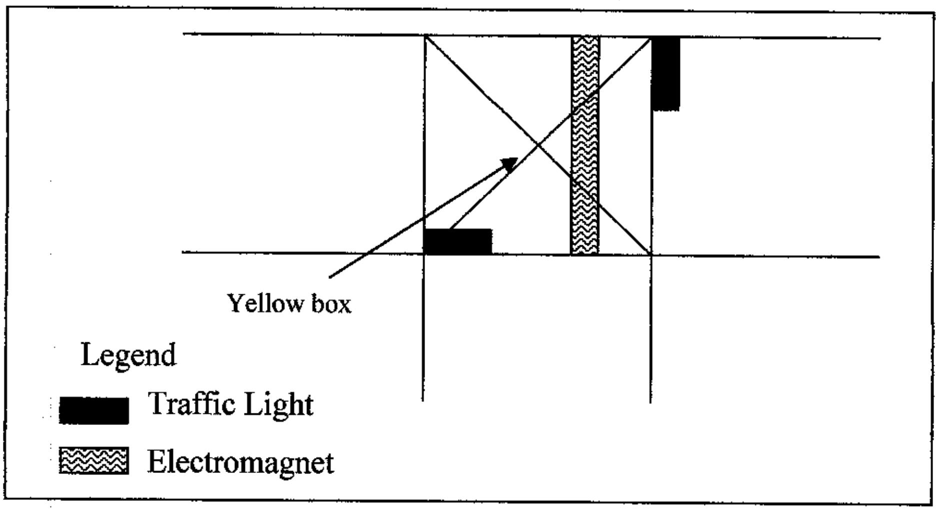



Figure 4.3: The installation of the system at the traffic light area

### 4.5.5 Alphanumeric display

The LCD is 16 characters by 2 line display, which incorporates a Hitachi HD44780 liquid crystal display controller driver chip. The HD44780 is an industry standard. It can be configured to drive a dot-matrix liquid crystal display under the control of a 4- or 8-bit microprocessor. It has an onboard character generator ROM which can display 192 character patterns.

Since all the functions such as display RAM, character generator, and liquid crystal driver, required for driving a dot-matrix liquid crystal display are internally provided on one chip, a minimal system can be interfaced with this controller/driver.

The LCD is integrated with PIC16F84A in order to produce the characters. The programme is simulated using PIC Simulator IDE.

## 4.6 Process Flow Chart

### 4.6.1 Transmitter (in the vehicle)

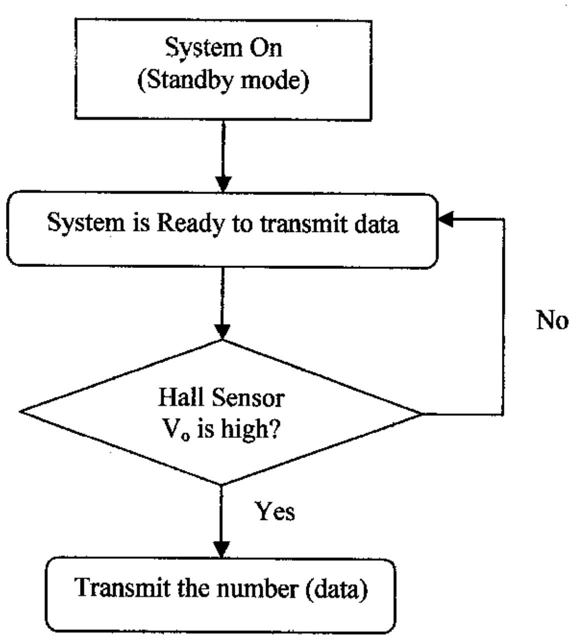



Figure 4.4: Flow chart of Transmitter in the vehicle

### 4.6.2 Receiver (at the traffic light)

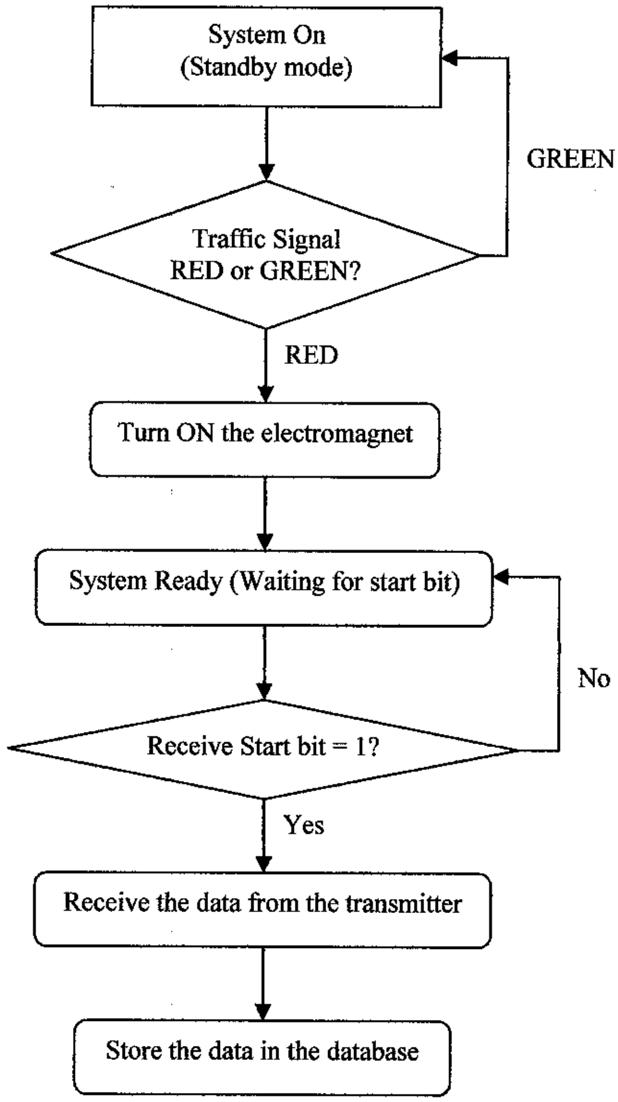



Figure 4.5: Flow chart of Receiver at the traffic light

## 4.7 Experiment Conducted

The experiment is conducted by using PIC Simulator IDE. This is trial software but contains some features such as oscillator, function generator, compiler, LCD display, microcontroller overview and step by step execution (monitoring). Figure 4.6 shows the main layout of the software. It shows the last instruction executed and the next instruction that will be executed.

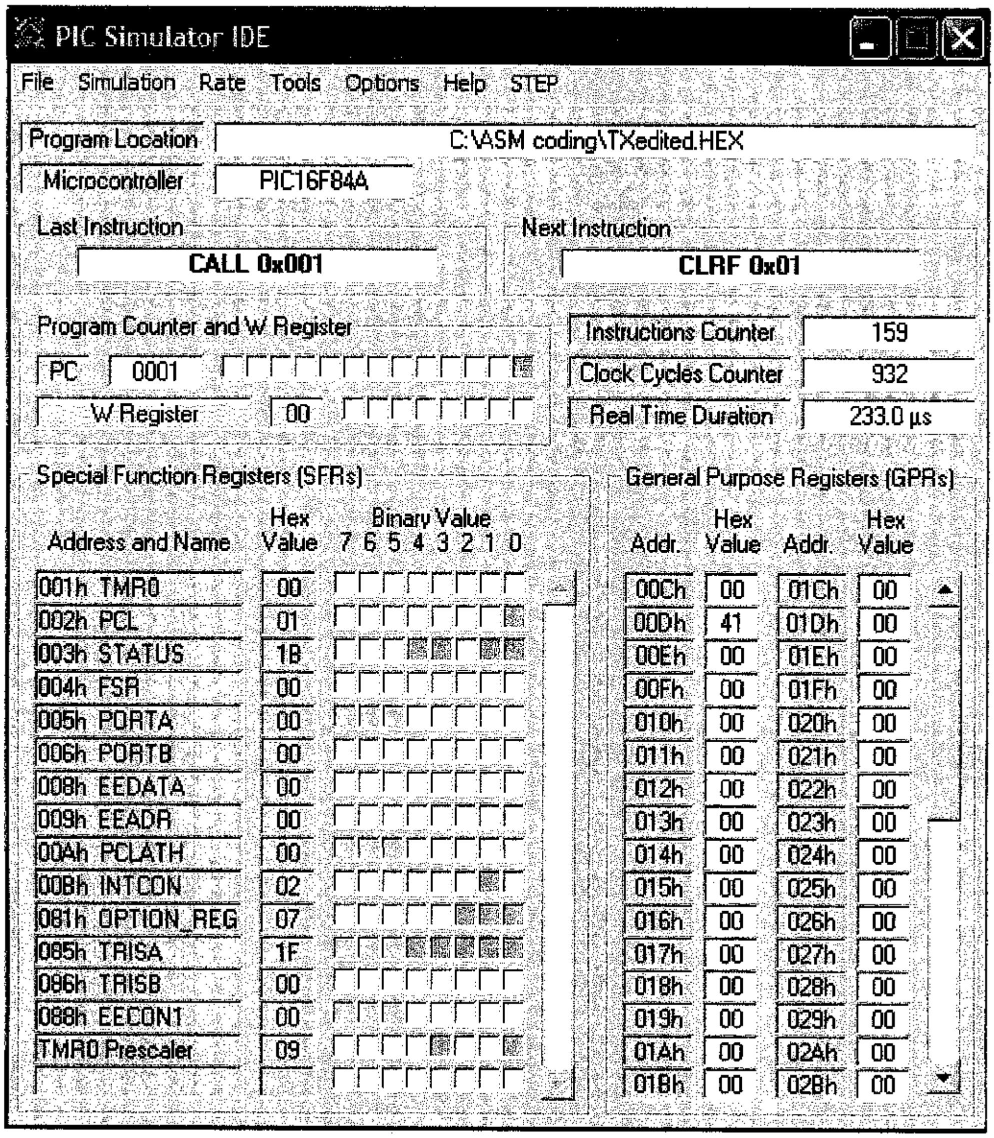



Figure 4.6: PIC Simulator IDE main programme

### 4.7.1 Initial Testing Experiment

To quickly test whether the PIC chip has been configured correctly the following assembler program and circuit (Figure 4.7) can be used.

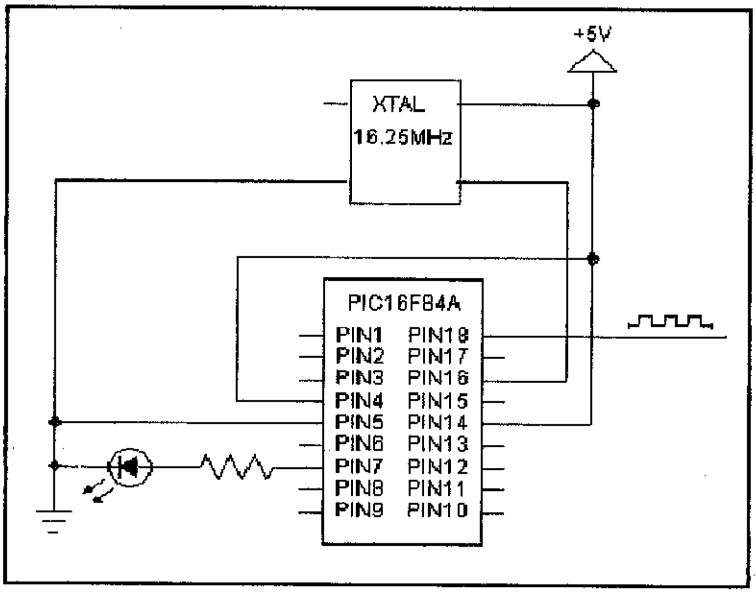



Figure 4.7: The schematic drawing

**Observation**

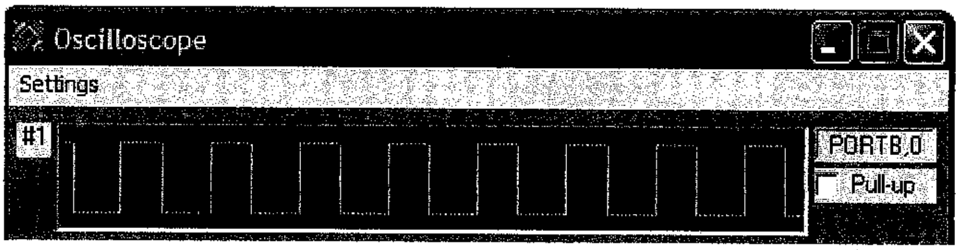



Figure 4.8: The graph shows blinking LED

Referring to figure 4.8, the LED is blinking ON for 1 second and OFF for 1 second with the clock supplied from the XT oscillator.

**Discussion**

The function is just set the bit at PORTB0 and PORTB1 with 1 and 0 and vice versa. The blinking function is called using delay subroutine. The programming code is in the appendix B.

### 4.7.2 Traffic light

For experiment purpose, the condition of the traffic light is set as in table 4.1.

| Time | RED | YELLOW | GREEN |
|---|---|---|---|
| 15 sec | OFF | OFF | ON |
| 2 sec | OFF | ON | OFF |
| 20 sec | ON | OFF | OFF |

Table 4.1: The traffic light condition

**Observation**

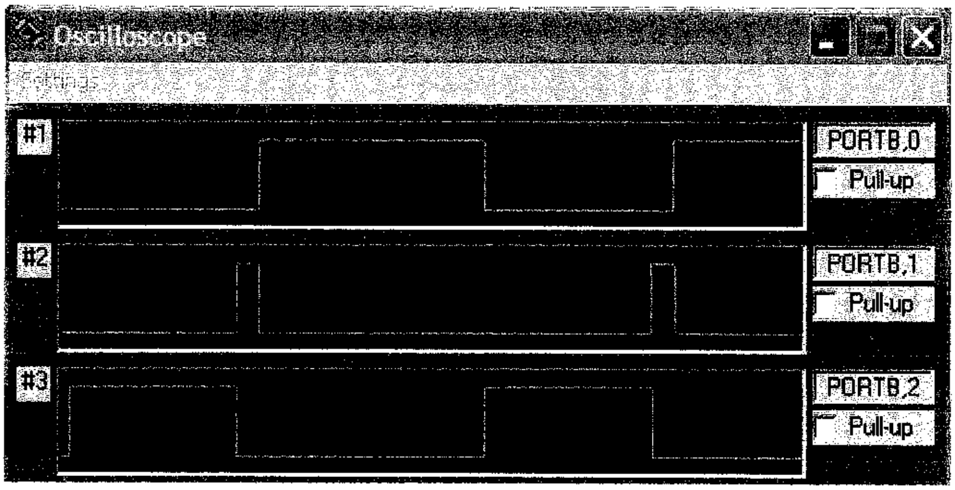



Figure 4.9: The graph shows the traffic light's working

Referring to figure 4.9; the PORTB2 which is green light turns ON for 15 seconds. After 15 seconds, the PORTB1 (yellow light) is set to high for 2 seconds. Then, the red light, which is PORTB0, turns ON for 20 seconds. The process is looping continuously.

**Discussion**

The programme is more likely the first experiment but with enhancement. There are 3 output from PORTB used to determine the light state. There are also 3 subroutines for delay, which are 2, 15 and 20 seconds. The output pins are connected to transistor switching circuit in order to switch the light ON and OFF. The programming code is in the appendix B.

### 4.7.3 Sending Experimental Data

Typically when sending data a protocol is developed so that the receiving end knows when to start and stop accepting data. A start bit is used for this purpose. The timing used is shown as in figure 4.10.

**Observation**

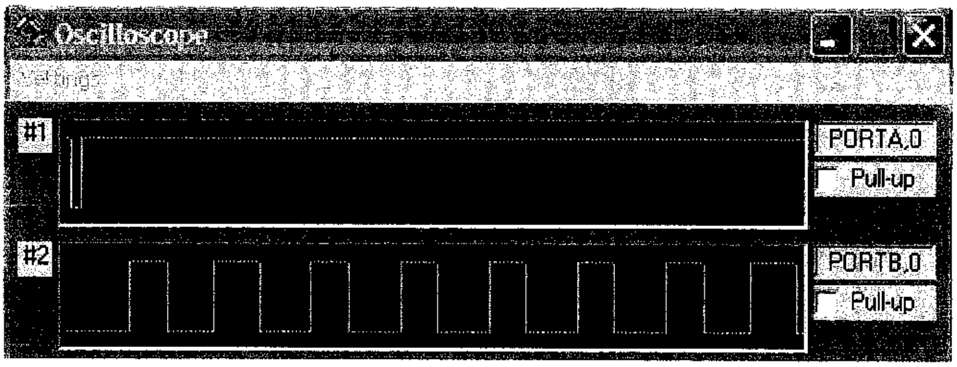



Figure 4.11: The graph shows bits are transmitted

When the PORTA 0 is triggered, the first bit, which is the start bit, is transmitted. It follows by the data 8-bit. In figure 4.11, the data value is 41H which means 01000001 in binary digit.

**Discussion**

For this project, the protocols consisted of a start bit and long wait in-between data packets. This is an extremely simplified method of sending and receiving data, the follow up project will create a more robust protocol that includes a decent checking scheme. Basically it takes the input from the sensors (connected to PORTA), calls the number in the register, adds a start bit, sends it serially through PORTB, waits, and then sends again until all the numbers have transmitted. The encoding and decoding are done by HT12E encoder and HT12D decoder from Holtek Semiconductor. The data is transmitted serially by the encoder. The full programming code is in the appendix B.

**Holtek HT12E and HT12D datasheet are attached in appendix C*

### 4.7.4 Receiving Experimental Data

On the other end of the process is the matching receiver PIC 16F84A chip which checks for the incoming data, stores in the temporary register, processes it, and produces the desired outputs.

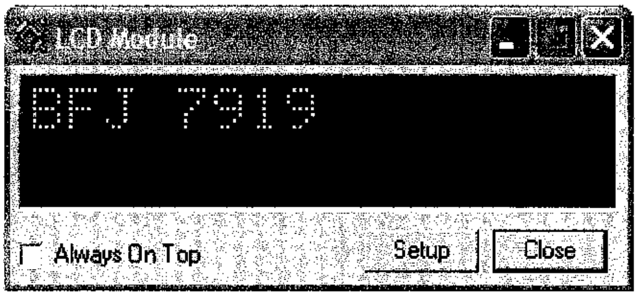



Figure 4.12: The LCD Module

**Observation**

When the traffic is RED, the PORTA1 is set to high. It will turn the receiver into waiting mode. Once the receiver receives the start bit, it will keep the following bits in the temporary register. Once the transmission is finished, the data is processed and convert into number display on the matrix LCD as in figure 4.12.

**Discussion**

Basically it waits for PORTA to go high and then begins reading in data after the initial bit. Typically the start bit would be different than the other bits sent and the receiver would check for this but because there is such a long wait between transmissions it is unnecessary. The system will receive the data when the traffic signal is RED. Otherwise, it will ignore the receive data. The programming code is in the appendix B.

### 4.7.5 Complete Circuit

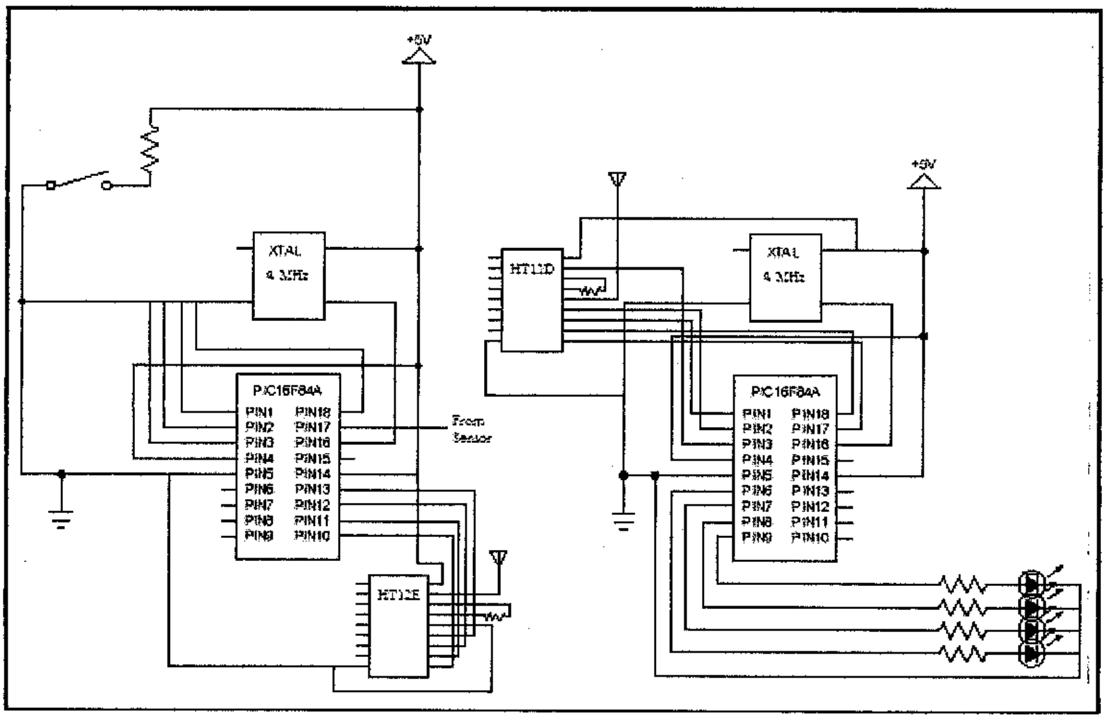



Figure 4.13: The Transmitter and Receiver Circuit

Finally, the circuit (as in figure 4.13) is designed. The data is in 4-bit x 2 (8-bit), which is capable to transfer the alphabet in the ASCII code format. To generate more than one alphabet, the coding is set to loop for few times depend on the number of alphabet required. There is one starting bit used to distinguish the previous data with the next one. The Holtek HT12E encoder is used with the RF transmitter, where Holtek HT12D decoder is used with the RF receiver. When the experiment is completed, the LEDs can be replaced with the matrix LCD display.

### 4.8 Problem Encountered

From the experiment and observation done, the transmitter/receiver kit available in the lab is also not suitable for this project. Thus, the experiment was done using simulation software.

The project progress is quite slow in the first semester since the stage was the literature review and lack of time.

The knowledge and information for PIC microcontroller is not enough due to time constraint.

# CHAPTER 5

# CONCLUSION AND RECOMMENDATION

This intelligent system is one of the solutions to control the road traffic violent. This may be applied in Malaysian road traffic to replace the existing camera system. The major challenge in this project is to achieve the main target.

The project development can be continued in focusing on practical implementation, PIC programming and integration of the database system.

In overall, the project may complete within the time given with hard R&D work. A good engineering practice will also contribute to design a good system for this project.

There are few enhancements and recommendations can be made to improve the system:

a. The switch can be replaced with the Hall sensor or other devices, which are more accurate and effective.

b. The matrix LCD can be replaced with the real vehicle database. This work needs knowledge in software programming such as Microsoft Visual Basics.

c. Intelligent Road Tax Sticker. There is special chip installed inside the sticker which is hidden from the user view. The sensor located at the traffic light will scan the chip in order to verify the number.

d. The security code should be installed for commercial purpose. This is to ensure that the system can only be accessed by the authorised person.

## REFERENCES


1. Rudolf F. Graf and William Sheets, 2001, *Build Your Own Low-Power Transmitters: Projects for the Electronics Experiment,* Newnes, USA

2. D.W Smith, 2000, *PIC in Practice,* Newnes, USA

3. Notes 99-7, *Digital Data Transmission*, TCOM 370

4. Myers Engineering International, Inc Websites

5. Micronas Hall Effect Sensor Websites

6. Internet Website: www.rapideducation.co.uk

7. Microchip Technology Institute Data Sheet
   Corporate Office,
   2355 West Chandler Blvd.
   Chandler, AZ 85224-6199
   Tel: 480-792-7200 Fax: 480-792-7277
   Technical Support: 480-792-7627
   Web Address: http://www.microchip.com

8. Laipac Technology, Inc.
   105 West Beaver Creek Rd.
   Unit 207 Richmond Hill Ontario L4B 1C6
   Canada
   Tel: (905)762-1228 Fax: (905)763-1737
   E-mail: info@laipac.com
   Web Address: www.laipac.com

9. Hall Effect IC Application Guide, Copyright © 1987, 1997
Allegro MicroSystems, Inc.
115 Northeast Cutoff, Box 15036,
Worcester, Massachusetts
01615-0036 (508) 853-5000

10. Thomas L. Floyd, 2002, *Electronic Devices Sixth Edition,* Prentice Hall International Inc, USA

11. Robert L. Boylestad and Louis Nashelsky, 2002, *Electronic Devices and Circuit Theory Eight Edition,* Prentice Hall International Inc, USA

# APPENDIX A

## SEMESTER ONE

| No. | Detail/ Week | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | | | | |
| 1 | Selection of Project Topic | ■ | | | | | | | | | | | | | |
| | - Propose Topic | | | | | | | | | | | | | | |
| | - Topic assigned to students | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | |
| 2 | Preliminary Research Work | | ■ | ■ | | | | | | | | | | | |
| | - Introduction (Problem Define) | | | | | | | | | | | | | | |
| | - Objective | | | | | | | | | | | | | | |
| | - List of references/literature | | | | | | | | | | | | | | |
| | - Project planning (methodology) | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | |
| 3 | Submission of Preliminary Report | | | ● | | | | | | | | | | | |
| | | | | | | | | | | | | | | | |
| 4 | Project Work | | | | ■ | ■ | ■ | ■ | | | | | | | |
| | - Reference/Literature Research | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | |
| 5 | Submission of Progress Report | | | | | | | | ● | | | | | | |
| | | | | | | | | | | | | | | | |
| 6 | Project work continue | | | | | | | | ■ | ■ | ■ | ■ | | | |
| | - Continue Literature Research | | | | | | | | | | | | | | |
| | - Practical/Laboratory Work | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | |
| 7 | Submission of Interim Report Final Draft | | | | | | | | | | | | | ● | |
| | | | | | | | | | | | | | | | |
| 8 | Oral Presentation | | | | | | | | | | | | | | ● W17 |
| | | | | | | | | | | | | | | | |
| 9 | Submission of Interim Report | | | | | | | | | | | | | | ● W18 |

● Suggested milestone

■ Work Done

## SEMESTER TWO

| No. | Detail/ Week | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Project Work Continue | | ■ | ■ | | | | | | | | | | | |
| | - Practical/Laboratory Work | | | | | | | | | | | | | | |
| | - PIC Programming Study | | | | | | | | | | | | | | |
| 2 | Submission of Progress Report 1 | | | ● | | | | | | | | | | | |
| 3 | Project Work Continue | | | | ■ | ■ | ■ | ■ | | | | | | | |
| | - Practical/Laboratory Work | | | | | | | | | | | | | | |
| | - Continue Programming Study | | | | | | | | | | | | | | |
| | - Sub-system prototyping | | | | | | | | | | | | | | |
| 4 | Submission of Progress Report 2 | | | | | | | | ● | | | | | | |
| 5 | Project work continue | | | | | | | | ■ | ■ | ■ | ■ | | | |
| | -Practical/Laboratory Work | | | | | | | | | | | | | | |
| | - System Integration and Testing | | | | | | | | | | | | | | |
| | - System Prototyping | | | | | | | | | | | | | | |
| 6 | Submission of Dissertation Final Draft | | | | | | | | | | | | ● | | |
| 7 | Oral Presentation | | | | | | | | | | | | | ● TBA | |
| 8 | Submission of Project Dissertation | | | | | | | | | | | | | | ● |

● Suggested milestone

■ Work Done

# APPENDIX B

```
;FLASHED LED PROGRAM CODE

;HEADER84.ASM    This sets PORTA as an INPUT (NB 1 means input)
;                and PORTB as an OUTPUT (NB 0 means output).

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

;EQUATES SECTION

STATUS          EQU     3                ;STATUS is file 3
PORTA           EQU     5                ;PORTA is file 5
PORTB           EQU     6                ;PORTB is file 6
COUNTER1        EQU     0CH              ;COUNT is file 0C
COUNTER2        EQU     0DH              ;COUNT is file 0D
COUNTER3        EQU     0EH              ;COUNT is file 0E
                                         ;a register to count
events

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

                LIST    P=16F84A         ;using the 16F84A
                #INCLUDE        <P16F84A.inc>
                ORG     0                ;start address memory
is 0
                GOTO    START            ;go to start

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

;DELAY subroutine

; The DELAY subroutine is used to delay the execution of the
next command by a
; specific period as determined by the values of COUNTER1,
COUNTER2 and COUNTER3

                ORG     0xD0

DELAY           MOVLW   0xC8             ; Set COUNTER1
                MOVWF   COUNTER1         ; to decimal 200

DELAY1          MOVLW   0xC8             ; Set COUNTER2
                MOVWF   COUNTER2         ; to decimal 200

DELAY2          MOVLW   0x07             ; Set COUNTER3
                MOVWF   COUNTER3         ; to decimal 7

DELAY3          DECFSZ  COUNTER3
                GOTO    DELAY3

                DECFSZ  COUNTER2
```

```
            GOTO     DELAY2

            DECFSZ   COUNTER1
            GOTO     DELAY1

            RETURN

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

;TRANSMIT SUBROUTINE

CHAR        MOVWF    PORTB
            CALL     DELAYP1          ;wait 1 seconds

            RETURN

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

;CONFIGURATION SECTION

START       BCF      STATUS,5         ;return to Bank0
            CLRF     PORTA            ;clears PORTA
            CLRF     PORTB            ;clears PORTB

            BSF      STATUS,5         ;turns to Bank1
            MOVLW    0x1F             ;5bits of PORTA are
input
            MOVWF    TRISA
            CLRF     TRISB            ;PORTB is output

            BCF      STATUS,5         ;Select Bank 0 for
the rest of the program

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

;Program starts now

BEGIN       BSF      PORTB,0                  ;turn ON B0
            BCF      PORTB,1                  ;turn OFF B1
            CALL     DELAY                    ;wait 1 second
            BCF      PORTB,0                  ;turn OFF B0
            BSF      PORTB,1                  ;turn ON B1
            CALL     DELAY                    ;wait 1 second
            GOTO     BEGIN                    ;repeat
            END
```

```
;TRAFFIC LIGHT PROGRAM CODE

;HEADER84.ASM    This sets PORTA as an INPUT (NB 1 means input)
;                and PORTB as an OUTPUT (NB 0 means output).
;                The OPTION register is set to /256 to give
;                timing pulses of 1/32 of a second.
;
;***********************************************************
;

;EQUATES SECTION

TMR0            EQU     1               ;TMR0 is file 1
STATUS          EQU     3               ;STATUS is file 3
PORTA           EQU     5               ;PORTA is file 5
PORTB           EQU     6               ;PORTB is file 6
ZEROBIT         EQU     2               ;ZEROBIT is bit 2
COUNTER1        EQU     0CH             ;COUNT is file 0C
COUNTER2        EQU     0DH             ;COUNT is file 0D
COUNTER3        EQU     0EH             ;COUNT is file 0E
COUNT           EQU     0FH             ;COUNT is file 0F
                                ;a register to count events

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

                LIST    P=16F84A        ;using the 16F84A
                #INCLUDE        <P16F84A.inc>
                ORG     0               ;start address memory
is 0
                GOTO    START           ;go to start

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

;DELAY subroutine

; The DELAY subroutine is used to delay the execution of the
next command by a
; specific period as determined by the values of COUNTER1,
COUNTER2 and COUNTER3

                ORG     0xD0

DELAY           MOVLW   0xC8            ; Set COUNTER1
                MOVWF   COUNTER1        ; to decimal 200

DELAY1          MOVLW   0xC8            ; Set COUNTER2
                MOVWF   COUNTER2        ; to decimal 200

DELAY2          MOVLW   0x07            ; Set COUNTER3
                MOVWF   COUNTER3        ; to decimal 5
```

```
)ELAY3          DECFSZ  COUNTER3
                GOTO    DELAY3

                DECFSZ  COUNTER2
                GOTO    DELAY2

                DECFSZ  COUNTER1
                GOTO    DELAY1

                RETLW   0

************************************************************

'CONFIGURATION SECTION

;TART           BCF     STATUS,5        ;return to Bank0
                CLRF    PORTA           ;clears PORTA
                CLRF    PORTB           ;clears PORTB

                BSF     STATUS,5        ;turns to Bank1
                MOVLW   0x1F            ;5bits of PORTA are
.nput
                MOVWF   TRISA
                CLRF    TRISB           ;PORTB is output

                BCF     STATUS,5        ;Select Bank 0 for
:he rest of the program

************************************************************

 Program starts now

;EGIN   MOVLW   .5
        MOVWF   COUNT

;EQ1    BCF     PORTB,0         ;turn OFF RED light
        BCF     PORTB,1         ;turn OFF YELLOW light
        BSF     PORTB,2         ;turn ON GREEN light
        CALL    DELAY

        DECFSZ  COUNT           ;wait 15 seconds
        GOTO    SEQ1

        MOVLW   .2
        MOVWF   COUNT

.EQ2    BCF     PORTB,0         ;turn OFF RED light
        BSF     PORTB,1         ;turn ON YELLOW light
        BCF     PORTB,2         ;turn OFF GREEN light
        CALL    DELAY
```

```
        DECFSZ  COUNT           ;wait 2 seconds
        GOTO    SEQ2

        MOVLW   .10
        MOVWF   COUNT

SEQ3    BSF     PORTB,0         ;turn ON RED light
        BCF     PORTB,1         ;turn OFF YELLOW light
        BCF     PORTB,2         ;turn OFF GREEN light
        CALL    DELAY1

        DECFSZ  COUNT           ;wait 20 seconds
        GOTO    SEQ3

        GOTO    BEGIN           ;repeat
        END
```

TRANSMITTER PROGRAM CODE

```
HEADER84.ASM    This sets PORTA as an INPUT (NB 1 means input)
                and PORTB as an OUTPUT (NB 0 means output).

***********************************************************


EQUATES SECTION

:TATUS          EQU     3               ;STATUS is file 3
'ORTA           EQU     5               ;PORTA is file 5
'ORTB           EQU     6               ;PORTB is file 6
:OUNTER1        EQU     0CH             ;COUNT is file 0C
:OUNTER2        EQU     0DH             ;COUNT is file 0D
:OUNTER3        EQU     0EH             ;COUNT is file 0E
:OUNTER4        EQU     0FH             ;COUNT is file 0E
                                        ;a register to count events

***********************************************************

                LIST    P=16F84A        ;using the 16F84A
                #INCLUDE        <P16F84A.inc>
                ORG     0               ;start address memory is 0
                GOTO    START           ;go to start

***********************************************************

DELAY subroutine

 The DELAY subroutine is used to delay the execution of the next command
y a
 specific period as determined by the values of COUNTER1, COUNTER2 and
OUNTER3

                ORG     0xD0

ELAY            MOVLW   0xC8            ; Set COUNTER1
                MOVWF   COUNTER1        ; to decimal 200

ELAY1           MOVLW   0xC8            ; Set COUNTER2
                MOVWF   COUNTER2        ; to decimal 200

ELAY2           MOVLW   0x07            ; Set COUNTER3
                MOVWF   COUNTER3        ; to decimal 7

ELAY3           DECFSZ  COUNTER3
                GOTO    DELAY3

                DECFSZ  COUNTER2
                GOTO    DELAY2

                DECFSZ  COUNTER1
                GOTO    DELAY1

                RETURN

0.1 second delay
ELAYP1          MOVLW   0x28            ; Set COUNTER4
                MOVWF   COUNTER4        ; to decimal 40

ELAY4           DECFSZ  COUNTER4
                GOTO    DELAY4
```

```
            RETURN

*********************************************************

TRANSMIT SUBROUTINE

:HAR        MOVWF   PORTB
            CALL    DELAYP1         ;wait 1 seconds

            RETURN

*********************************************************

CONFIGURATION SECTION

;TART       BCF     STATUS,5        ;return to Bank0
            CLRF    PORTA           ;clears PORTA
            CLRF    PORTB           ;clears PORTB

            BSF     STATUS,5        ;turns to Bank1
            MOVLW   0x1F            ;5bits of PORTA are input
            MOVWF   TRISA
            CLRF    TRISB           ;PORTB is output

            BCF     STATUS,5        ;Select Bank 0 for the rest of the
rogram

*********************************************************

Program starts now (characters transmit are "WLB 6165")

EGIN        BTFSS   PORTA,0         ;wait for switch press/sensor ON
            GOTO    BEGIN

:HAR        MOVLW   0xF0            ;transmit bit (4-bit HIGH)
            CALL    DELAYP1         ;wait 1 seconds
            MOVLW   0xF0            ;transmit bit (4-bit LOW)
            CALL    DELAYP1         ;wait 1 seconds

            MOVLW   0x50            ;transmit bit (4-bit HIGH)
            CALL    DELAYP1         ;wait 1 seconds
            MOVLW   0x70            ;transmit bit (4-bit LOW)
            CALL    DELAYP1         ;wait 1 seconds

            MOVLW   0x40            ;transmit bit (4-bit HIGH)
            CALL    DELAYP1         ;wait 0.5 seconds
            MOVLW   0xC0            ;transmit bit (4-bit LOW)
            CALL    DELAYP1         ;wait 1 seconds

            MOVLW   0x40            ;transmit bit (4-bit HIGH)
            CALL    DELAYP1         ;wait 0.5 seconds
            MOVLW   0x20            ;transmit bit (4-bit LOW)
            CALL    DELAYP1         ;wait 1 seconds

            MOVLW   0x20            ;transmit bit (4-bit HIGH)
            CALL    DELAYP1         ;wait 0.5 seconds
            MOVLW   0x00            ;transmit bit (4-bit LOW)
            CALL    DELAYP1         ;wait 1 seconds

            MOVLW   0x30            ;transmit bit (4-bit HIGH)
            CALL    DELAYP1         ;wait 0.5 seconds
            MOVLW   0x60            ;transmit bit (4-bit LOW)
```

```
CALL    DELAYP1         ;wait 1 seconds

MOVLW   0x30            ;transmit bit (4-bit HIGH)
CALL    DELAYP1         ;wait 0.5 seconds
MOVLW   0x10            ;transmit bit (4-bit LOW)
CALL    DELAYP1         ;wait 1 seconds

MOVLW   0x30            ;transmit bit (4-bit HIGH)
CALL    DELAYP1         ;wait 0.5 seconds
MOVLW   0x60            ;transmit bit (4-bit LOW)
CALL    DELAYP1         ;wait 1 seconds

MOVLW   0x30            ;transmit bit (4-bit HIGH)
CALL    DELAYP1         ;wait 0.5 seconds
MOVLW   0x50            ;transmit bit (4-bit LOW)
CALL    DELAYP1         ;wait 1 seconds

MOVLW   0x00            ;transmit bit (4-bit HIGH)
CALL    DELAYP1         ;wait 0.5 seconds
MOVLW   0x00            ;transmit bit (4-bit LOW)
CALL    DELAYP1         ;wait 1 seconds

GOTO    BEGIN
END
```

RECEIVER AND LCD PROGRAM CODE

HEADER84.ASM    This sets PORTA as an INPUT (NB 1 means input) and PORTB as an OUTPUT (NB 0 means output).

```
**********************************************************

EQUATES SECTION

'MR0            EQU     1               ;TMR0 is file 1
:TATUS          EQU     3               ;STATUS is file 3
'ORTA           EQU     5               ;PORTA is file 5
'ORTB           EQU     6               ;PORTB is file 6
:OUNTER1        EQU     0CH             ;COUNT is file 0C
:OUNTER2        EQU     0DH             ;COUNT is file 0D
:OUNTER3        EQU     0EH             ;COUNT is file 0E
:OUNTER4        EQU     0FH             ;COUNT is file 0F

**********************************************************

                LIST    P=16f84A        ;using the 16F84A
                #INCLUDE        <P16F84A.inc>
                ORG     0               ;start address memory is 0
                GOTO    START           ;go to start

**********************************************************
```

DELAY subroutine

The DELAY subroutine is used to delay the execution of the next command
y a
specific period as determined by the values of COUNTER1, COUNTER2 and
OUNTER3

```
                ORG     0xD0

'ELAY           MOVLW   0xC8            ; Set COUNTER1
                MOVWF   COUNTER1        ; to decimal 200

'ELAY1          MOVLW   0xC8            ; Set COUNTER2
                MOVWF   COUNTER2        ; to decimal 200

ELAY2           MOVLW   0x07            ; Set COUNTER3
                MOVWF   COUNTER3        ; to decimal 5

ELAY3           DECFSZ  COUNTER3
                GOTO    DELAY3

                DECFSZ  COUNTER2
                GOTO    DELAY2

                DECFSZ  COUNTER1
                GOTO    DELAY1

                RETURN

0.1 second delay
ELAYP1          MOVLW   0x28            ; Set COUNTER4
                MOVWF   COUNTER4        ; to decimal 40

ELAY4           DECFSZ  COUNTER4
```

```
                GOTO    DELAY4

                RETURN

:LOCK           BSF     PORTA,2
                NOP
                CALL    DELAYP1
                BCF     PORTA,2
                NOP
                CALL    DELAYP1

                RETURN

*************************************************************

\               MOVLW   2               ;enable the display
                MOVWF   PORTA
                MOVLW   4H
                MOVWF   PORTB
                CALL    CLOCK
                MOVLW   1H              ;41H is code for A
                MOVWF   PORTB
                CALL    CLOCK           ;clock character onto display
                RETURN

3B              MOVLW   2               ;enable the display
                MOVWF   PORTA
                MOVLW   4H
                MOVWF   PORTB
                CALL    CLOCK
                MOVLW   2H              ;42H is code for B
                MOVWF   PORTB
                CALL    CLOCK           ;clock character onto display
                RETURN

:C              MOVLW   2               ;enable the display
                MOVWF   PORTA
                MOVLW   4H
                MOVWF   PORTB
                CALL    CLOCK
                MOVLW   3H              ;43H is code for C
                MOVWF   PORTB
                CALL    CLOCK           ;clock character onto display
                RETURN

)               MOVLW   2               ;enable the display
                MOVWF   PORTA
                MOVLW   4H
                MOVWF   PORTB
                CALL    CLOCK
                MOVLW   4H              ;44H is code for D
                MOVWF   PORTB
                CALL    CLOCK           ;clock character onto display
                RETURN

:               MOVLW   2               ;enable the display
                MOVWF   PORTA
                MOVLW   4H
                MOVWF   PORTB
                CALL    CLOCK
                MOVLW   5H              ;45H is code for E
                MOVWF   PORTB
                CALL    CLOCK           ;clock character onto display
```

```
        RETURN

'F      MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   4H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   6H              ;46H is code for A
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

;       MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   4H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   7H              ;47H is code for G
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

:       MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   4H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   8H              ;48H is code for H
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

        MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   4H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   9H              ;49H is code for I
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

        MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   4H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   0AH             ;4AH is code for J
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

:       MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   4H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   0BH             ;4BH is code for K
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

        MOVLW   2               ;enable the display
```

```
        MOVWF   PORTA
        MOVLW   4H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   0CH             ;4CH is code for L
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

        MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   4H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   0DH             ;4DH is code for M
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

        MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   4H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   0EH             ;4EH is code for N
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

        MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   4H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   0FH             ;4FH is code for O
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

        MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   5H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   0H              ;50H is code for P
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

        MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   5H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   1H              ;51H is code for Q
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

        MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   5H
        MOVWF   PORTB
```

```
        CALL    CLOCK
        MOVLW   2H              ;52H is code for R
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

        MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   5H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   3H              ;53H is code for S
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

        MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   5H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   4H              ;54H is code for T
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

        MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   5H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   5H              ;55H is code for U
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

        MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   5H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   6H              ;56H is code for V
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

W       MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   5H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   7H              ;57H is code for W
        MOVWF   PORTB
        CALL    CLOCK           ;clock character onto display
        RETURN

        MOVLW   2               ;enable the display
        MOVWF   PORTA
        MOVLW   5H
        MOVWF   PORTB
        CALL    CLOCK
        MOVLW   8H              ;58H is code for X
        MOVWF   PORTB
```

```
            CALL    CLOCK           ;clock character onto display
            RETURN

            MOVLW   2               ;enable the display
            MOVWF   PORTA
            MOVLW   5H
            MOVWF   PORTB
            CALL    CLOCK
            MOVLW   9H              ;59H is code for Y
            MOVWF   PORTB
            CALL    CLOCK           ;clock character onto display
            RETURN

Z           MOVLW   2               ;enable the display
            MOVWF   PORTA
            MOVLW   5H
            MOVWF   PORTB
            CALL    CLOCK
            MOVLW   0AH             ;5AH is code for Z
            MOVWF   PORTB
            CALL    CLOCK           ;clock character onto display
            RETURN

UM0         MOVLW   2               ;enable the display
            MOVWF   PORTA
            MOVLW   3H
            MOVWF   PORTB
            CALL    CLOCK
            MOVLW   0H              ;30H is code for NUM0
            MOVWF   PORTB
            CALL    CLOCK           ;clock character onto display
            RETURN

UM1         MOVLW   2               ;enable the display
            MOVWF   PORTA
            MOVLW   3H
            MOVWF   PORTB
            CALL    CLOCK
            MOVLW   1H              ;31H is code for NUM1
            MOVWF   PORTB
            CALL    CLOCK           ;clock character onto display
            RETURN

UM2         MOVLW   2               ;enable the display
            MOVWF   PORTA
            MOVLW   3H
            MOVWF   PORTB
            CALL    CLOCK
            MOVLW   2H              ;32H is code for NUM2
            MOVWF   PORTB
            CALL    CLOCK           ;clock character onto display
            RETURN

UM3         MOVLW   2               ;enable the display
            MOVWF   PORTA
            MOVLW   3H
            MOVWF   PORTB
            CALL    CLOCK
            MOVLW   3H              ;33H is code for NUM3
            MOVWF   PORTB
            CALL    CLOCK           ;clock character onto display
            RETURN
```

```
IUM4        MOVLW   2           ;enable the display
            MOVWF   PORTA
            MOVLW   3H
            MOVWF   PORTB
            CALL    CLOCK
            MOVLW   4H          ;34H is code for NUM4
            MOVWF   PORTB
            CALL    CLOCK       ;clock character onto display
            RETURN

IUM5        MOVLW   2           ;enable the display
            MOVWF   PORTA
            MOVLW   3H
            MOVWF   PORTB
            CALL    CLOCK
            MOVLW   5H          ;35H is code for NUM5
            MOVWF   PORTB
            CALL    CLOCK       ;clock character onto display
            RETURN

UM6         MOVLW   2           ;enable the display
            MOVWF   PORTA
            MOVLW   3H
            MOVWF   PORTB
            CALL    CLOCK
            MOVLW   6H          ;36H is code for NUM6
            MOVWF   PORTB
            CALL    CLOCK       ;clock character onto display
            RETURN

UM7         MOVLW   2           ;enable the display
            MOVWF   PORTA
            MOVLW   3H
            MOVWF   PORTB
            CALL    CLOCK
            MOVLW   7H          ;37H is code for NUM7
            MOVWF   PORTB
            CALL    CLOCK       ;clock character onto display
            RETURN

UM8         MOVLW   2           ;enable the display
            MOVWF   PORTA
            MOVLW   3H
            MOVWF   PORTB
            CALL    CLOCK
            MOVLW   8H          ;38H is code for NUM8
            MOVWF   PORTB
            CALL    CLOCK       ;clock character onto display
            RETURN

UM9         MOVLW   2           ;enable the display
            MOVWF   PORTA
            MOVLW   3H
            MOVWF   PORTB
            CALL    CLOCK
            MOVLW   9H          ;39H is code for NUM9
            MOVWF   PORTB
            CALL    CLOCK       ;clock character onto display
            RETURN

AP          MOVLW   2           ;enable the display
            MOVWF   PORTA
            MOVLW   2H
```

```
                MOVWF   PORTB
                CALL    CLOCK
                MOVLW   0H              ;20H is code for GAP
                MOVWF   PORTB
                CALL    CLOCK           ;clock character onto display
                RETURN

)OT             MOVLW   2               ;enable the display
                MOVWF   PORTA
                MOVLW   2H
                MOVWF   PORTB
                CALL    CLOCK
                MOVLW   0EH             ;2EH is code for DOT
                MOVWF   PORTB
                CALL    CLOCK           ;clock character onto display
                RETURN

'LRDISP         MOVLW   2               ;enable the display
                MOVWF   PORTA
                MOVLW   30
                MOVWF   PORTB
                CALL    CLOCK           ;clock character onto display
                MOVLW   1               ;1H is code for CLRDISP
                MOVWF   PORTB
                CALL    CLOCK
                CALL    DELAYP1
                RETURN

*************************************************************

CONFIGURATION SECTION

TART            BCF     STATUS,5        ;return to Bank0
                CLRF    PORTA           ;clears PORTA
                CLRF    PORTB           ;clears PORTB

                BSF     STATUS,5        ;turns to Bank1
                MOVLW   0x01            ;5bits of PORTA are input
                MOVWF   TRISA
                CLRF    TRISB           ;PORTB is output

                BCF     STATUS,5        ;Select Bank 0 for the rest of the
rogram

************************************************************

DISPLAY CONFIGURATION

                MOVLW   03H             ;FUNCTION SET
                MOVWF   PORTB           ;8 bit data (default)
                CALL    CLOCK

                CALL    DELAYP1         ;wait for display

                MOVLW   02H             ;FUNCTION SET
                MOVWF   PORTB           ;change to 4 bit
                CALL    CLOCK           ;clock in data

                CALL    DELAYP1         ;wait for display

                MOVLW   02H             ;FUNCTION SET
                MOVWF   PORTB           ;must repeat command
                CALL    CLOCK           ;clock in data
```

```
            CALL    DELAYP1          ;wait for display

            MOVLW   08H              ;4 bit micro
            MOVWF   PORTB            ;using 2 line display
            CALL    CLOCK            ;clock in data

            CALL    DELAYP1

            MOVLW   0H               ;Display on, cursor off
            MOVWF   PORTB            ;0CH
            CALL    CLOCK
            MOVLW   0CH
            MOVWF   PORTB
            CALL    CLOCK

            CALL    DELAYP1

            MOVLW   0H               ;Increment cursor, 06H
            MOVWF   PORTB
            CALL    CLOCK
            MOVLW   6H
            MOVWF   PORTB
            CALL    CLOCK

*****************************************************************

EGIN        BTFSS   PORTA,1          ;Test received bit (start bit?)
            GOTO    BEGIN
            BTFSS   PORTA,2
            GOTO    BEGIN
            BTFSS   PORTA,3
            GOTO    BEGIN
            BTFSS   PORTA,4
            GOTO    BEGIN

            CALL    CLRDISP
            CLRF    PORTA
            MOVLW   8H
            MOVWF   PORTB
            CALL    CLOCK
            MOVLW   0H
            MOVWF   PORTB
            CALL    CLOCK

            CALL    WW
            CALL    DELAYP1

            CALL    L
            CALL    DELAYP1

            CALL    BB
            CALL    DELAYP1

            CALL    GAP
            CALL    DELAYP1

            CALL    NUM6
            CALL    DELAYP1

            CALL    NUM1
            CALL    DELAYP1
```

```
CALL    NUM6
CALL    DELAYP1

CALL    NUM5
CALL    DELAY
CALL    DELAY
CALL    DELAY

GOTO    BEGIN

END
```

# APPENDIX C

MICROCHIP

# PIC16F8X

## 18-pin Flash/EEPROM 8-Bit Microcontrollers

### Devices Included in this Data Sheet:

- PIC16F83
- PIC16F84
- PIC16CR83
- PIC16CR84
- Extended voltage range devices available (PIC16**LF**8X, PIC16**LCR**8X)

### High Performance RISC CPU Features:

- Only 35 single word instructions to learn
- All instructions single cycle except for program branches which are two-cycle
- Operating speed: DC - 10 MHz clock input
  DC - 400 ns instruction cycle

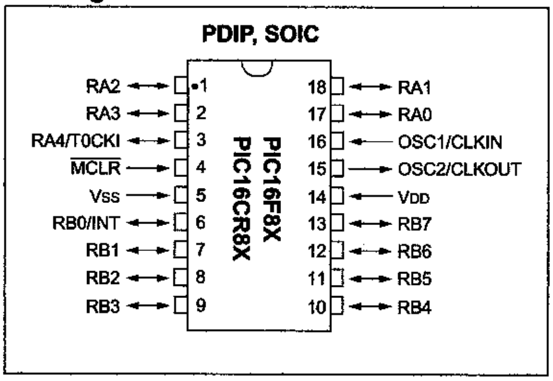
| Device | Program Memory (words) | Data RAM (bytes) | Data EEPROM (bytes) | Max. Freq (MHz) |
|---|---|---|---|---|
| PIC16F83 | 512 Flash | 36 | 64 | 10 |
| PIC16F84 | 1 K Flash | 68 | 64 | 10 |
| PIC16CR83 | 512 ROM | 36 | 64 | 10 |
| PIC16CR84 | 1 K ROM | 68 | 64 | 10 |

- 14-bit wide instructions
- 8-bit wide data path
- 15 special function hardware registers
- Eight-level deep hardware stack
- Direct, indirect and relative addressing modes
- Four interrupt sources:
  - External RB0/INT pin
  - TMR0 timer overflow
  - PORTB<7:4> interrupt on change
  - Data EEPROM write complete
- 1000 erase/write cycles Flash program memory
- 10,000,000 erase/write cycles EEPROM data memory
- EEPROM Data Retention > 40 years

### Peripheral Features:

- 13 I/O pins with individual direction control
- High current sink/source for direct LED drive
  - 25 mA sink max. per pin
  - 20 mA source max. per pin
- TMR0: 8-bit timer/counter with 8-bit programmable prescaler

### Pin Diagrams



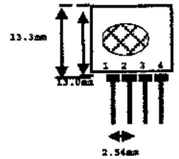

### Special Microcontroller Features:

- In-Circuit Serial Programming (ICSP™) - via two pins (ROM devices support only Data EEPROM programming)
- Power-on Reset (POR)
- Power-up Timer (PWRT)
- Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Code-protection
- Power saving SLEEP mode
- Selectable oscillator options

### CMOS Flash/EEPROM Technology:

- Low-power, high-speed technology
- Fully static design
- Wide operating voltage range:
  - Commercial: 2.0V to 6.0V
  - Industrial: 2.0V to 6.0V
- Low power consumption:
  - < 2 mA typical @ 5V, 4 MHz
  - 15 μA typical @ 2V, 32 kHz
  - < 1 μA typical standby current @ 2V

# Easy-Link Wireless




pin 1 : GND
pin 2 : Data In
pin 3 : Vcc
pin 4 : Antenna ( RF output )

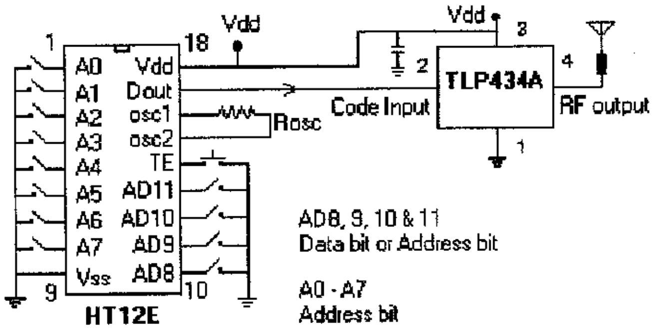
**Frequency 315, 418 and 433.92 Mhz**

**Modulation : ASK**
**Operation Voltage : 2 - 12 VDC**

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| Vcc | Operating supply voltage | | 2.0 | - | 12.0 | V |
| Icc 1 | Peak Current (2V) | | - | - | 1.64 | mA |
| Icc 2 | Peak Current (12V) | | - | - | 19.4 | mA |
| Vh | Input High Voltage | Idata= 100uA (High) | Vcc-0.5 | Vcc | Vcc+0.5 | V |
| Vl | Input Low Voltage | Idata= 0 uA (Low) | - | - | 0.3 | V |
| FO | Absolute Frequency | 315Mhz module | 314.8 | 315 | 315.2 | MHz |
| PO | RF Output Power- 50ohm | Vcc = 9V-12V | - | 16 | - | dBm |
| | | Vcc = 5V-6V | - | 14 | - | dBm |
| DR | Data Rate | External Encoding | 512 | 4.8K | 200K | bps |

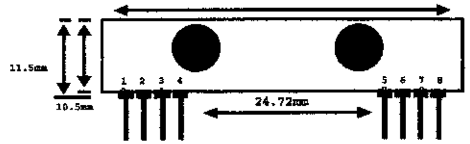
Notes : ( Case Temperature = 25°C +- 2°C , Test Load Impedance = 50 ohm )

Application Circuit :
Typical Key-chain Transmitter using HT12E-18DIP, a Binary 12 bit Encoder from Holtek Semiconductor Inc.




**Laipac Technology, Inc.**
105 West Beaver Creek Rd. Unit 207 Richmond Hill Ontario L4B 1C6 Canada
Tel: (905)762-1228 Fax: (905)763-1737 e-mail: info@laipac.com

LAIPAC
TECH




pin 2 : Digital Data Output
pin 3 : Linear Output /Test
pin 4 : Vcc
pin 5 : Vcc
pin 6 : Gnd
pin 7 : Gnd
pin 8 : Antenna

**Frequency 315, 418 and 433.92 Mhz**

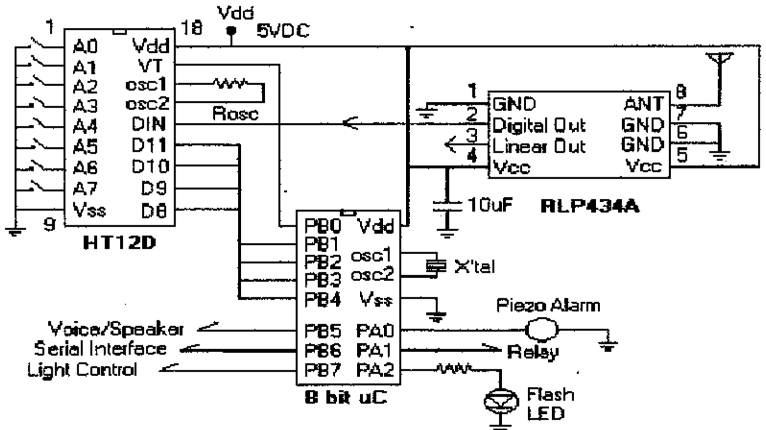
**Modulation : ASK**
**Supply Voltage : 3.3 - 6.0 VDC**
**Output : Digital & Linear**

| Symbol | Parameter | Conditions | Min | Typ | Max | |
|---|---|---|---|---|---|---|
| Vcc | Operating supply voltage | | 3.3 | 5.0V | 6.0 | V |
| Itot | Operating Current | | - | 4.5 | | mA |
| Vdata | Data Out | Idata = +200 uA ( High ) | Vcc-0.5 | - | Vcc | V |
| | | Idata = -10 uA ( Low ) | - | - | 0.3 | V |

Electrical Characteristics

| Characteristics | SYM | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Operation Radio Frequency | FC | 315, 418 and 433.92 | | | MHz |
| Sensitivity | Pref | | -110 | | dBm |
| Channel Width | | | +-500 | | Khz |
| Noise Equivalent BW | | | 4 | | Khz |
| Receiver Turn On Time | | | 5 | | ms |
| Operation Temperature | Top | -20 | - | 80 | C |
| Baseboard Data Rate | | | 4.8 | | KHz |

Application Circuit :
Typical RF Receiver using HT12D-18DIP, a Binary 12 bit Decoder with 8 bit uC HT48RXX from Holtek Semiconductor Inc.



HOLTEK

# HT12A/HT12E
# $2^{12}$ Series of Encoders

## Features

- Operating voltage
  - 2.4V~5V for the HT12A
  - 2.4V~12V for the HT12E
- Low power and high noise immunity CMOS technology
- Low standby current: 0.1μA (typ.) at $V_{DD}$=5V
- HT12A with a 38kHz carrier for infrared transmission medium
- Minimum transmission word
  - Four words for the HT12E
  - One word for the HT12A
- Built-in oscillator needs only 5% resistor
- Data code has positive polarity
- Minimal external components
- HT12A/E: 18-pin DIP/20-pin SOP package

## 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 $2^{12}$ encoders are a series of CMOS LSIs for remote control system applications. They are capable of encoding information which consists of N address bits and 12–N data bits. Each address/data input can be set to one of the two logic states. The programmed addresses/data are 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 $\overline{TE}$ trigger on the HT12E or a DATA trigger on the HT12A further enhances the application flexibility of the $2^{12}$ series of encoders. The HT12A additionally provides a 38kHz carrier for infrared systems.

## Selection Table

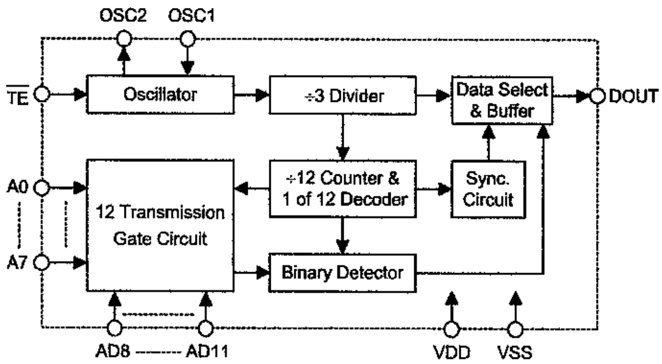
| Function / Part No. | Address No. | Address/ Data No. | Data No. | Oscillator | Trigger | Package | Carrier Output | Negative Polarity |
|---|---|---|---|---|---|---|---|---|
| HT12A | 8 | 0 | 4 | 455kHz resonator | D8~D11 | 18 DIP<br>20 SOP | 38kHz | No |
| HT12E | 8 | 4 | 0 | RC oscillator | $\overline{TE}$ | 18 DIP<br>20 SOP | No | No |

Note: Address/Data represents pins that can be address or data according to the decoder requirement.

## Block Diagram

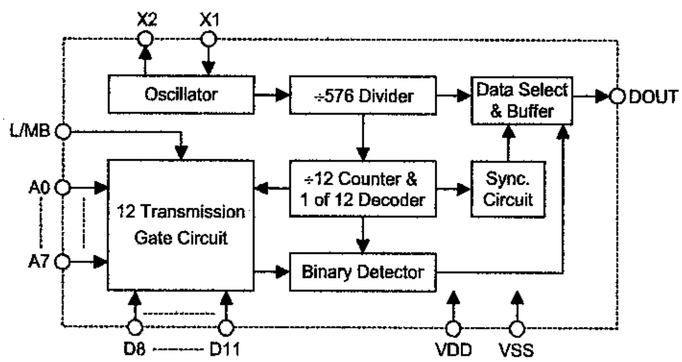
### $\overline{TE}$ trigger

HT12E




### DATA trigger

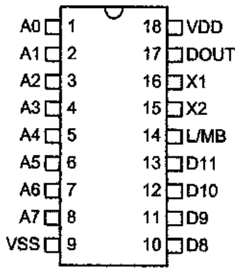
HT12A




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

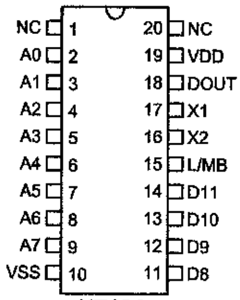
## Pin Assignment

8-Address
4-Data



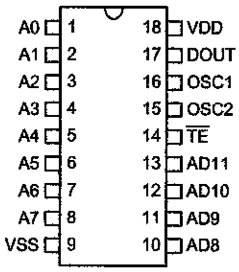

HT12A
– 18 DIP

8-Address
4-Data



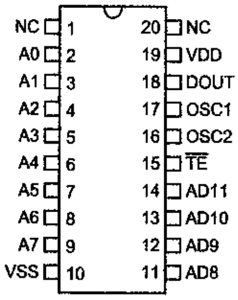

HT12A
– 20 SOP

8-Address
4-Address/Data




HT12E
– 18 DIP

8-Address
4-Address/Data




HT12E
– 20 SOP

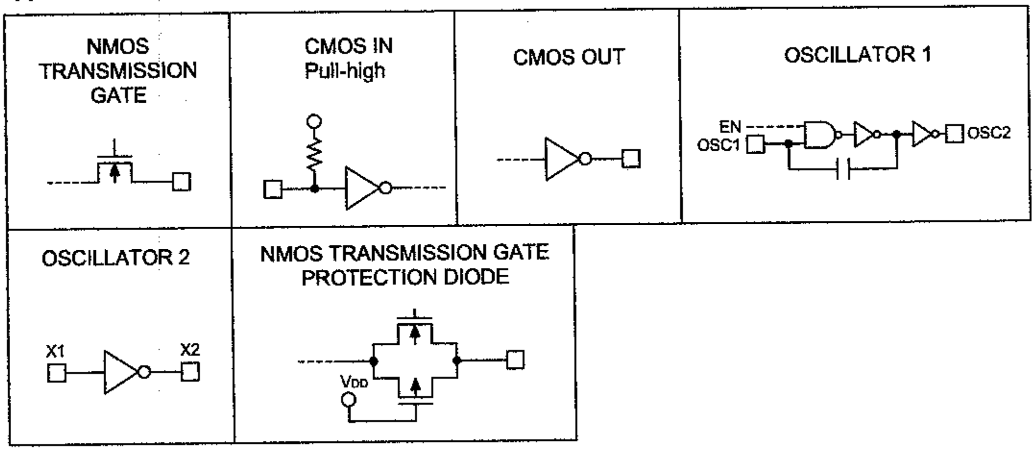
## Pin Description

| Pin Name | I/O | Internal Connection | Description |
|---|---|---|---|
| A0~A7 | I | CMOS IN Pull-high (HT12A)<br>NMOS TRANSMISSION GATE PROTECTION DIODE (HT12E) | Input pins for address A0~A7 setting<br>These pins can be externally set to VSS or left open |
| AD8~AD11 | I | NMOS TRANSMISSION GATE PROTECTION DIODE (HT12E) | Input pins for address/data AD8~AD11 setting<br>These pins can be externally set to VSS or left open |
| D8~D11 | I | CMOS IN Pull-high | Input pins for data D8~D11 setting and transmission enable, active low<br>These pins should be externally set to VSS or left open (see Note) |
| DOUT | O | CMOS OUT | Encoder data serial transmission output |
| L/MB | I | CMOS IN Pull-high | Latch/Momentary transmission format selection pin:<br>Latch: Floating or VDD<br>Momentary: VSS |

| Pin Name | I/O | Internal Connection | Description |
|---|---|---|---|
| $\overline{TE}$ | I | CMOS IN Pull-high | Transmission enable, active low (see Note) |
| OSC1 | I | OSCILLATOR 1 | Oscillator input pin |
| OSC2 | O | OSCILLATOR 1 | Oscillator output pin |
| X1 | I | OSCILLATOR 2 | 455kHz resonator oscillator input |
| X2 | O | OSCILLATOR 2 | 455kHz resonator oscillator output |
| VSS | I | — | Negative power supply, grounds |
| VDD | I | — | Positive power supply |

Note: D8~D11 are all data input and transmission enable pins of the HT12A.
$\overline{TE}$ is a transmission enable pin of the HT12E.

**Approximate internal connections**




## Absolute Maximum Ratings

Supply Voltage (HT12A) ..............–0.3V to 5.5V

Input Voltage....................$V_{SS}$–0.3 to $V_{DD}$+0.3V

Operating Temperature...............–20°C to 75°C

Supply Voltage (HT12E) ...............–0.3V to 13V

Storage Temperature.................–50°C to 125°C

Note: These are stress ratings only. Stresses exceeding the range specified under "Absolute Maximum Ratings" may cause substantial damage to the device. Functional operation of this device at other conditions beyond those listed in the specification is not implied and prolonged exposure to extreme conditions may affect device reliability.

## Electrical Characteristics

### HT12A

Ta=25°C

| Symbol | Parameter | Test Conditions $V_{DD}$ | Test Conditions Conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|---|
| $V_{DD}$ | Operating Voltage | — | — | 2.4 | 3 | 5 | V |
| $I_{STB}$ | Standby Current | 3V | Oscillator stops | — | 0.1 | 1 | µA |
| | | 5V | | — | 0.1 | 1 | µA |
| $I_{DD}$ | Operating Current | 3V | No load $f_{OSC}$=455kHz | — | 200 | 400 | µA |
| | | 5V | | — | 400 | 800 | µA |
| $I_{DOUT}$ | Output Drive Current | 5V | $V_{OH}=0.9V_{DD}$ (Source) | -1 | -1.6 | — | mA |
| | | | $V_{OL}=0.1V_{DD}$ (Sink) | 2 | 3.2 | — | mA |
| $V_{IH}$ | "H" Input Voltage | — | — | $0.8V_{DD}$ | — | $V_{DD}$ | V |
| $V_{IL}$ | "L" Input Voltage | — | — | 0 | — | $0.2V_{DD}$ | V |
| $R_{DATA}$ | D8~D11 Pull-high Resistance | 5V | $V_{DATA}$=0V | — | 150 | 300 | kΩ |

### HT12E

Ta=25°C

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

HOLTEK

# $2^{12}$ Series of Decoders

## Features

- Operating voltage: 2.4V~12V
- Low power and high noise immunity CMOS technology
- Low standby current
- Capable of decoding 12 bits of information
- Pair with Holtek's $2^{12}$ series of encoders
- Binary address setting
- Received codes are checked 3 times
- Address/Data number combination
  - HT12D: 8 address bits and 4 data bits
  - HT12F: 12 address bits only
- Built-in oscillator needs only 5% resistor
- Valid transmission indicator
- Easy 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 $2^{12}$ decoders are a series of CMOS LSIs for remote control system applications. They are paired with Holtek's $2^{12}$ series of encoders (refer to the encoder/decoder cross reference table). For proper operation, a pair of encoder/decoder with the same number of addresses and data format should be chosen.

The decoders receive serial addresses and data from a programmed $2^{12}$ series of encoders that are transmitted by a carrier using an RF or an IR transmission medium. They compare the serial input data three times continuously with their local addresses. If no error or unmatched codes are found, the input data codes are decoded and then transferred to the output pins. The VT pin also goes high to indicate a valid transmission.

The $2^{12}$ series of decoders are capable of decoding informations that consist of N bits of address and 12–N bits of data. Of this series, the HT12D is arranged to provide 8 address bits and 4 data bits, and HT12F is used to decode 12 bits of address information.

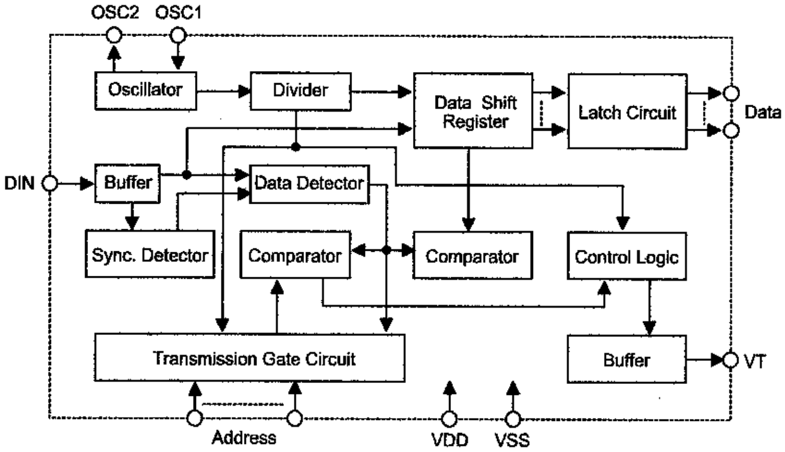
## Selection Table

| Function / Part No. | Address No. | Data No. | Data Type | VT | Oscillator | Trigger | Package |
|---|---|---|---|---|---|---|---|
| HT12D | 8 | 4 | L | √ | RC oscillator | DIN active "Hi" | 18 DIP/20 SOP |
| HT12F | 12 | 0 | — | √ | RC oscillator | DIN active "Hi" | 18 DIP/20 SOP |

Notes: Data type: L stands for latch type data output.

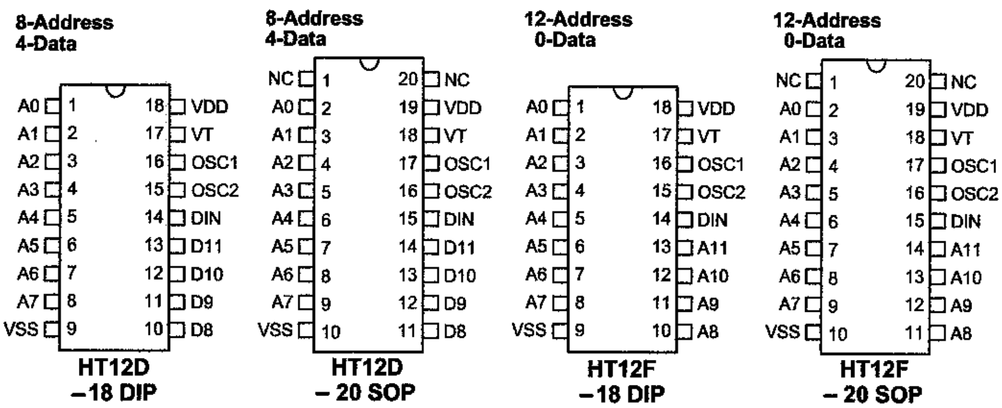
VT can be used as a momentary data output.

## Block Diagram




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

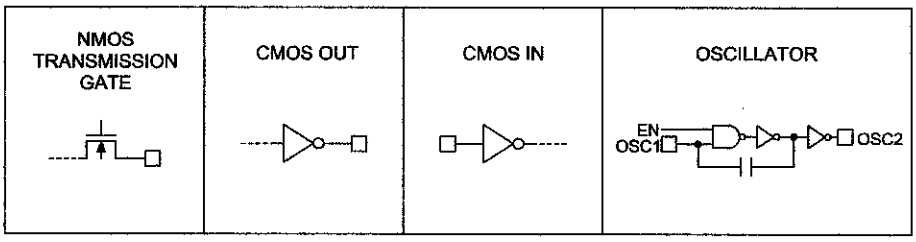
## Pin Assignment



## Pin Description

| Pin Name | I/O | Internal Connection | Description |
|---|---|---|---|
| A0~A11 | I | NMOS TRANSMISSION GATE | Input pins for address A0~A11 setting<br>They can be externally set to VDD or VSS. |
| D8~D11 | 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 |

### Approximate internal connection circuits




## Absolute Maximum Ratings

Supply Voltage................................–0.3V to 13V

Input Voltage.....................$V_{SS}$–0.3 to $V_{DD}$+0.3V

Storage Temperature.................–50°C to 125°C

Operating Temperature ..............–20°C to 75°C

Note: These are stress ratings only. Stresses exceeding the range specified under "Absolute Maximum Ratings" may cause substantial damage to the device. Functional operation of this device at other conditions beyond those listed in the specification is not implied and prolonged exposure to extreme conditions may affect device reliability.

## Electrical Characteristics

Ta=25°C

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

# HD44780U (LCD-II)

## (Dot Matrix Liquid Crystal Display Controller/Driver)

HITACHI

## Description

The HD44780U dot-matrix liquid crystal display controller and driver LSI displays alphanumerics, Japanese kana characters, and symbols. It can be configured to drive a dot-matrix liquid crystal display under the control of a 4- or 8-bit microprocessor. Since all the functions such as display RAM, character generator, and liquid crystal driver, required for driving a dot-matrix liquid crystal display are internally provided on one chip, a minimal system can be interfaced with this controller/driver.

A single HD44780U can display up to one 8-character line or two 8-character lines.

The HD44780U has pin function compatibility with the HD44780S which allows the user to easily replace an LCD-II with an HD44780U. The HD44780U character generator ROM is extended to generate 208 5 × 8 dot character fonts and 32 5 × 10 dot character fonts for a total of 240 different character fonts.

The low power supply (2.7V to 5.5V) of the HD44780U is suitable for any portable battery-driven product requiring low power dissipation.

## Features

- 5 × 8 and 5 × 10 dot matrix possible
- Low power operation support:
  - 2.7 to 5.5V
- Wide range of liquid crystal display driver power
  - 3.0 to 11V
- Liquid crystal drive waveform
  - A (One line frequency AC waveform)
- Correspond to high speed MPU bus interface
  - 2 MHz (when $V_{CC}$ = 5V)
- 4-bit or 8-bit MPU interface enabled
- 80 × 8-bit display RAM (80 characters max.)
- 9,920-bit character generator ROM for a total of 240 character fonts
  - 208 character fonts (5 × 8 dot)
  - 32 character fonts (5 × 10 dot)

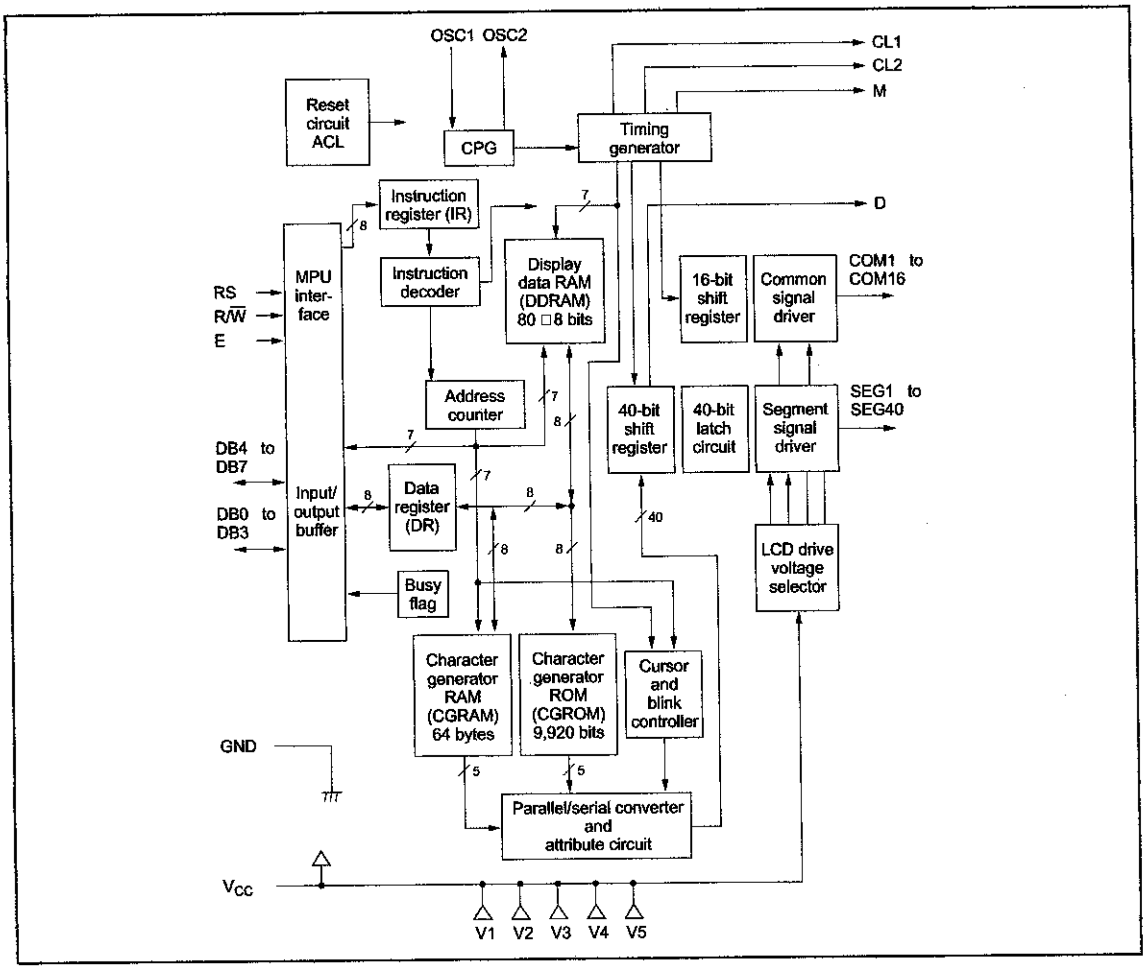
- 64 □8-bit character generator RAM
  - 8 character fonts (5 □8 dot)
  - 4 character fonts (5 □10 dot)
- 16-common □40-segment liquid crystal display driver
- Programmable duty cycles
  - 1/8 for one line of 5 □8 dots with cursor
  - 1/11 for one line of 5 □10 dots with cursor
  - 1/16 for two lines of 5 □8 dots with cursor
- Wide range of instruction functions:
  - Display clear, cursor home, display on/off, cursor on/off, display character blink, cursor shift, display shift
- Pin function compatibility with HD44780S
- Automatic reset circuit that initializes the controller/driver after power on
- Internal oscillator with external resistors
- Low power consumption

## Ordering Information

| Type No. | Package | CGROM |
|---|---|---|
| HD44780UA00FS | FP-80B | Japanese standard font |
| HCD44780UA00 | Chip | |
| HD44780UA00TF | TFP-80F | |
| HD44780UA02FS | FP-80B | European standard font |
| HCD44780UA02 | Chip | |
| HD44780UA02TF | TFP-80F | |
| HD44780UBxxFS | FP-80B | Custom font |
| HCD44780UBxx | Chip | |
| HD44780UBxxTF | TFP-80F | |

Note: xx: ROM code No.

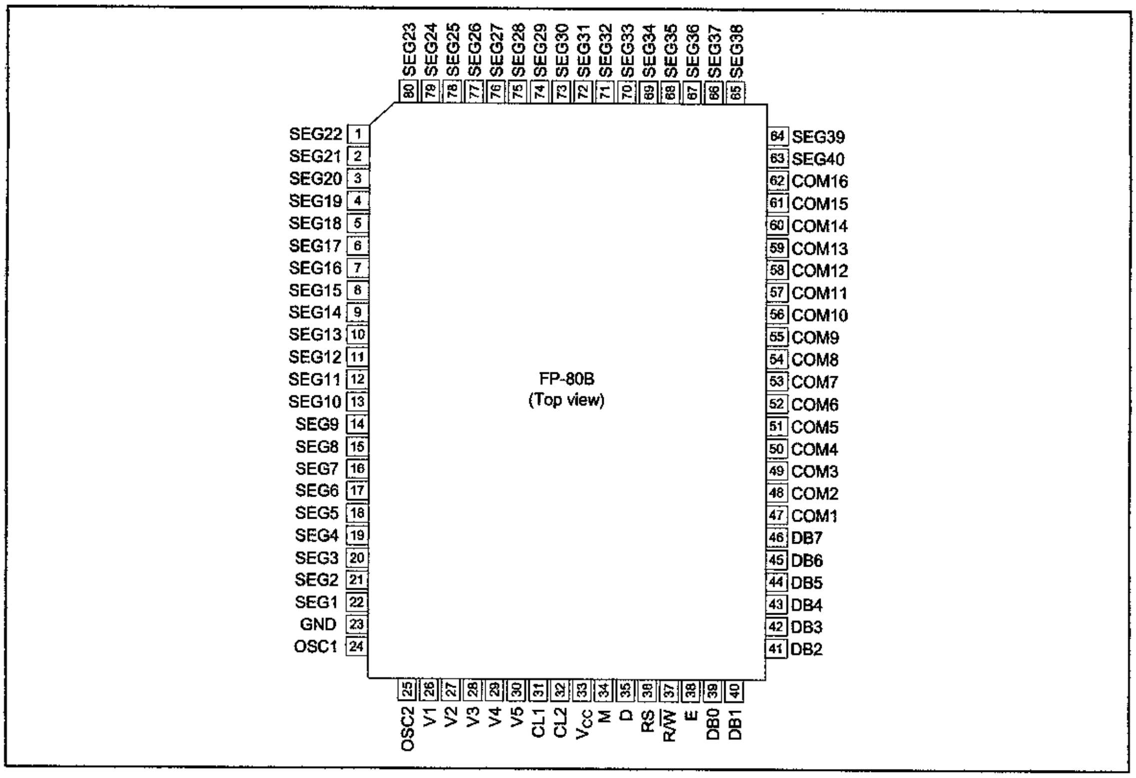
## HD44780U Block Diagram



## LCD-II Family Comparison

| Item | | HD44780S | HD44780U |
|---|---|---|---|
| Power supply voltage | | 5 V ±10% | 2.7 to 5.5 V |
| Liquid crystal drive voltage VLCD | 1/4 bias | 3.0 to 11.0V | 3.0 to 11.0V |
| | 1/5 bias | 4.6 to 11.0V | 3.0 to 11.0V |
| Maximum display digits per chip | | 16 digits (8 digits □ 2 lines) | 16 digits (8 digits □ 2 lines) |
| Display duty cycle | | 1/8, 1/11, and 1/16 | 1/8, 1/11, and 1/16 |
| CGROM | | 7,200 bits (160 character fonts for 5 □ 7 dot and 32 character fonts for 5 □ 10 dot) | 9,920 bits (208 character fonts for 5 □ 8 dot and 32 character fonts for 5 □ 10 dot) |
| CGRAM | | 64 bytes | 64 bytes |
| DDRAM | | 80 bytes | 80 bytes |
| Segment signals | | 40 | 40 |
| Common signals | | 16 | 16 |
| Liquid crystal drive waveform | | A | A |
| Oscillator | Clock source | External resistor, external ceramic filter, or external clock | External resistor or external clock |
| | $R_f$ oscillation frequency (frame frequency) | 270 kHz ±30% (59 to 110 Hz for 1/8 and 1/16 duty cycles; 43 to 80 Hz for 1/11 duty cycle) | 270 kHz ±30% (59 to 110 Hz for 1/8 and1/16 duty cycles; 43 to 80 Hz for 1/11 duty cycle) |
| | $R_f$ resistance | 91 k□ ±2% | 91 k□ ±2% (when $V_{CC}$ = 5V)<br>75 k□ ±2% (when $V_{CC}$ = 3V) |
| Instructions | | Fully compatible within the HD44780S | |
| CPU bus timing | | 1 MHz | 1 MHz (when $V_{CC}$ = 3V)<br>2 MHz (when $V_{CC}$ = 5V) |
| Package | | FP-80<br>FP-80A | FP-80B<br>TFP-80F |

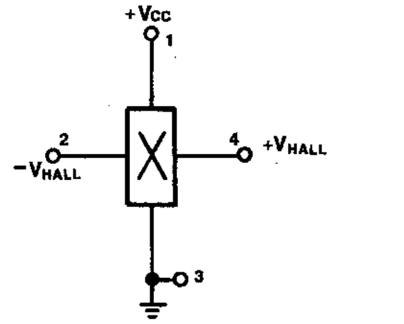
## HD44780U Pin Arrangement (FP-80B)



# APPLICATIONS INFORMATION

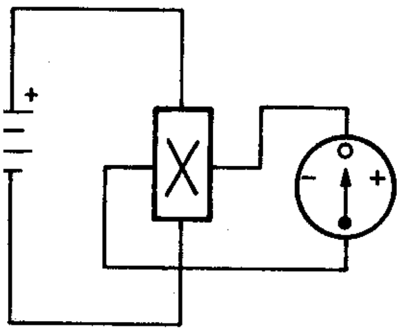
Application Note 27701B*

## *HALL-EFFECT IC APPLICATIONS GUIDE*



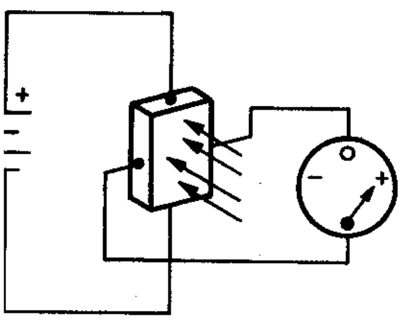

Dwg. No. A-13,100

**Figure 1**




Dwg. No. A-13,101

**Figure 2**




Dwg. No. A-13,102

**Figure 3**

Allegro MicroSystems uses the latest bipolar integrated circuit technology in combination with the century-old Hall effect to produce Hall-effect ICs. These are contactless, magnetically activated switches and sensors with the potential to simplify and improve systems.

### LOW-COST SIMPLIFIED SWITCHING

Simplified switching is a Hall sensor's strong point. Hall-effect IC switches combine Hall voltage generators, signal amplifiers, Schmitt trigger circuits, and transistor output circuits on single integrated circuit chips. Output is clean, fast, and switched without bounce—an inherent problem with mechanical contact switches. A Hall-effect switch typically operates at up to a 100 kHz repetition rate, and costs less than many common electromechanical switches.

### EFFICIENT, EFFECTIVE, LOW-COST LINEAR SENSORS

The linear Hall-effect sensor detects the motion, position, or change in field strength of an electromagnet, a permanent magnet, or a ferromagnetic material with an applied magnetic bias. Energy consumption is very low. The output is linear and temperature-stable. The sensor's frequency response is flat up to approximately 25 kHz.

A Hall-effect sensor is more efficient and effective than inductive or optoelectronic sensors, and at a lower cost.

### SENSITIVE CIRCUITS FOR RUGGED SERVICE

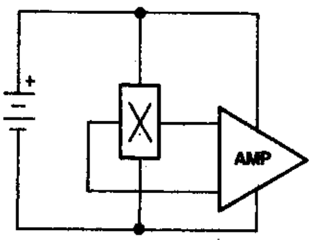
The Hall-effect sensor is virtually immune to environmental contaminants and is suitable for use under severe service conditions. The circuit is very sensitive and provides reliable, repetitive operation in close tolerance applications. The Hall-effect sensor can see precisely through dirt and darkness.

### CURRENT APPLICATIONS

Current applications for Hall-effect ICs include use in ignition systems, speed controls, security systems, alignment controls, micrometers, mechanical limit switches, computers, printers, disk drives, keyboards, machine tools, key switches, and pushbutton switches. They are also used as tachometer pickups, current limit switches, position detectors, selector switches, current sensors, linear potentiometers, and brushless dc motor commutators.

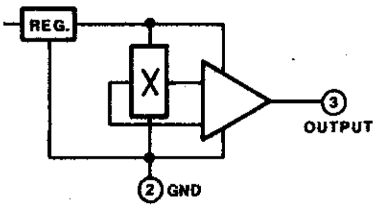
### THE HALL EFFECT SENSOR: HOW DOES IT WORK?

The basic Hall sensor is a small sheet of semiconductor material represented by figure 1.



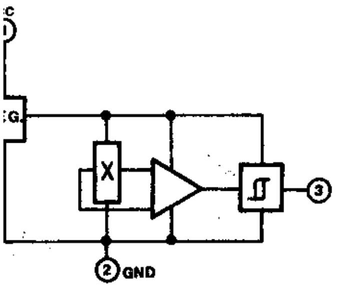

Dwg. No. A-13,103

**Figure 4**



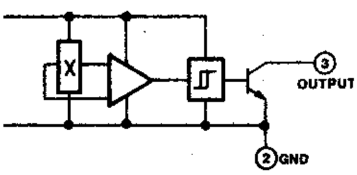

Dwg. No. A-13,104

**Figure 5**




Dwg. No. A-13,105

**Figure 6**




Dwg. No. A-13,106

**Figure 7**

A constant voltage source, as shown in figure 2, will force a constant bias current to flow in the semiconductor sheet. The output will take the form of a voltage measured across the width of the sheet that will have negligible value in the absence of a magnetic field.

If the biased Hall sensor is placed in a magnetic field with flux lines at right angles to the Hall current (figure 3), the voltage output is directly proportional to the strength of the magnetic field. This is the Hall effect, discovered by E. F. Hall in 1879.

## LINEAR OUTPUT HALL-EFFECT DEVICES

The output voltage of the basic Hall-effect sensor (Hall element) is quite small. This can present problems, especially in an electrically noisy environment. Addition of a stable, high-quality dc amplifier and voltage regulator to the circuit (figures 4 and 5) improves the transducer's output and allows it to operate over a wide range of supply voltages. The modified device provides an easy-to-use analog output that is linear and proportional to the applied magnetic flux density.

The UGN3503 is this type of linear output device. The A3506/07/08 and A3515/16 have improved sensitivity and temperature-stable characteristics. The output is ratiometric; that is, its output is proportional to its supply voltage.

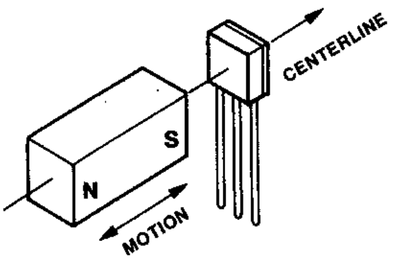
## DIGITAL OUTPUT HALL-EFFECT SWITCHES

The addition of a Schmitt-trigger threshold detector with built-in hysteresis, as shown in figure 6, gives the Hall-effect circuit digital output capabilities. When the applied magnetic flux density exceeds a certain limit, the trigger provides a clean transition from OFF to ON without contact bounce. Built-in hysteresis eliminates oscillation (spurious switching of the output) by introducing a magnetic dead zone in which switch action is disabled after the threshold value is passed.

An open-collector NPN output transistor added to the circuit (figure 7) gives the switch digital logic compatibility. The transistor is a saturated switch that shorts the output terminal to ground wherever the applied flux density is higher than the ON trip point of the device. The switch is compatible with all digital families. The output transistor can sink enough current to directly drive many loads, including relays, triacs, SCRs, LEDs, and lamps.

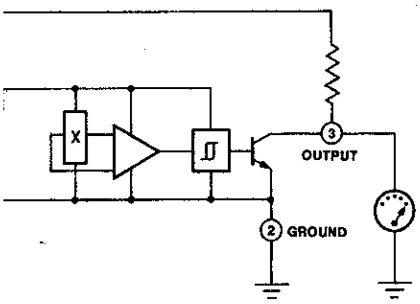
The circuit elements in figure 7, fabricated on a monolithic silicon chip and encapsulated in a small epoxy or ceramic package, are common to all Hall-effect digital switches. Differences between device types are generally found in specifications such as magnetic parameters, operating temperature ranges, and temperature coefficients.


115 Northeast Cutoff, Box 15036
Worcester, Massachusetts 01615-0036 (508) 853-5000



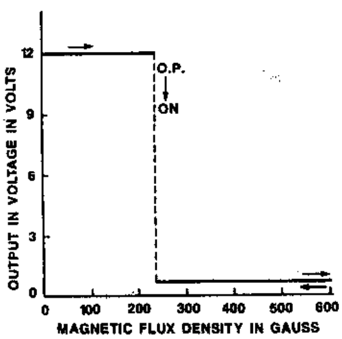

Dwg. No. A-13,107

**Figure 8**




Dwg. No. A-13,108

**Figure 9**




Dwg. No. A-13,109

**Figure 10**

## OPERATION

All Hall-effect devices are activated by a magnetic field. A mount for the the devices, and electrical connections, must be provided; Parameters such as load current, environmental conditions, and supply voltage must fall within the specific limits shown in the appropriate documentation.

Magnetic fields have two important characteristics—flux density and polarity (or orientation). In the absence of any magnetic field, most Hall-effect digital switches are designed to be OFF (open circuit at output). They will turn ON only if subjected to a magnetic field that has both sufficient density and the correct orientation.

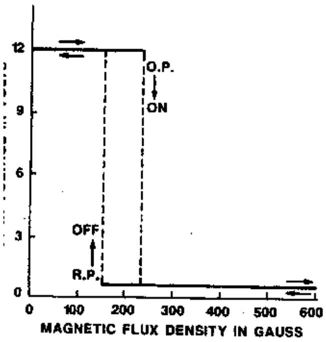
Hall switches have an active area that is closer to one face of the package (the face with the lettering, the branded face). To operate the switch, the magnetic flux lines must be perpendicular to this face of the package, and must have the correct polarity. If an approaching south pole would cause switching action, a north pole would have no effect. In practice, a close approach to the branded face of a Hall switch by the south pole of a small permanent magnet will cause the output transistor to turn ON (figure 8).

A Transfer Characteristics Graph (figures 10 and 11) plots this information. It is a graph of output as a function of magnetic flux density (measured in gauss; 1 G = 0.1 mT) presented to the Hall cell. The magnetic flux density is shown on the horizontal axis. The digital output of the Hall switch is shown along the vertical axis.

To acquire data for this graph, add a power supply and a pull-up resistor that will limit current through the output transistor and enable the value of the output voltage to approach zero (figure 9).

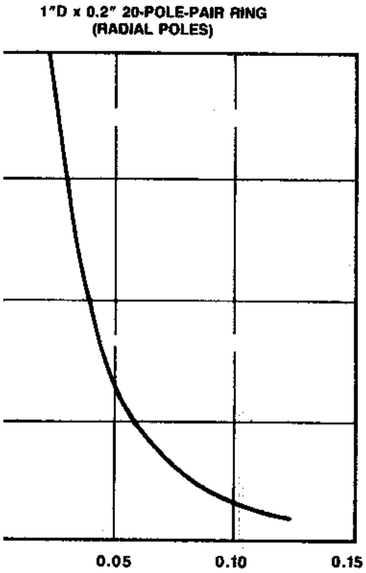
In the absence of an applied magnetic field (0 G), the switch is OFF, and the output voltage equals the power supply (12 V).
A permanent magnet's south pole is then moved perpendicularly toward the active area of the device. As the magnet's south pole approaches the branded face of the switch, the Hall cell is exposed to increasing magnetic flux density. At some point (240 G in this case), the output transistor turns ON and the output voltage approaches zero (figure 10). That value of flux density is called the operate point. If we continue to increase the field's strength, say to 600 G, nothing more happens. The switch turns ON once and stays ON.

To turn the switch OFF, the magnetic flux density must fall to a value far lower than the 240 G "operate point" because of the built-in hysteresis. For this example we use 90 G hysteresis, which means the device turns OFF when flux density decreases to 150 G (figure 11). That value of flux density is called the "release point".




Dwg. No. A-13,110

**Figure 11**




Dwg. No. A-13,126

**Figure 12A**

## CHARACTERISTICS AND TOLERANCES

The exact magnetic flux density values required to turn Hall switches ON and OFF differ for several reasons, including design criteria and manufacturing tolerances. Extremes in temperature will also somewhat affect the operate and release points.

For each device type, worst-case magnetic characteristics for the operate value, the release value, and hysteresis are provided.

All switches are guaranteed to turn ON at or below the maximum operate point flux density. When the magnetic field is reduced, all devices will turn OFF before the flux density drops below the minimum release point value. Each device is guaranteed to have at least the minimum amount of hysteresis to ensure clean switching action. This hysteresis ensures that, even if mechanical vibration or electrical noise is present, the switch output is fast, clean, and occurs only once per threshold crossing.

## GETTING STARTED

Because the electrical interface is usually straightforward, the design of a Hall-effect system should begin with the physical aspects. In position-sensing or motion-sensing applications, the following questions should be answered:

How much and what type of motion is there?

What angular or positional accuracy is required?

How much space is available for mounting the sensing device and activating magnet?

How much play is there in the moving assembly?

How much mechanical wear can be expected over the lifetime of the machine?

Will the product be a mass-produced assembly, or a limited number of machines that can be individually adjusted and calibrated?

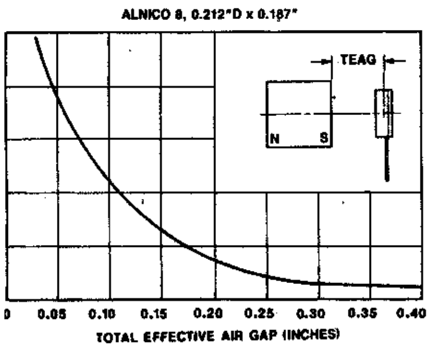
What temperature extremes are expected?

A careful analysis will pay big dividends in the long term.

## THE ANALYSIS

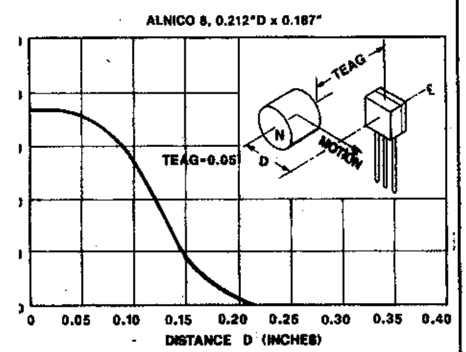
The field strength of the magnet should be investigated. The strength of the field will be the greatest at the pole face, and will decrease with increasing distance from the magnet. The strength of the magnetic field can be measured with a gaussmeter or a calibrated linear Hall sensor.

A plot of field strength (magnetic flux density) is a function of distance along the intended line of travel of the magnet. Hall device



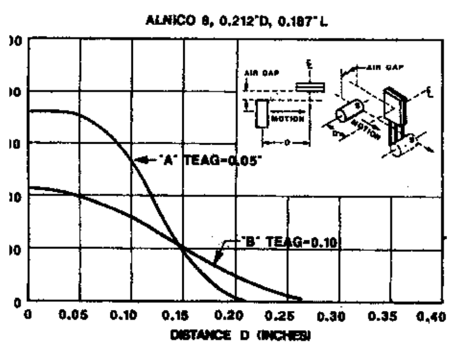

Dwg. No. A-13,112

**Figure 12B**




Dwg. No. A-13,113

**Figure 13**




Dwg. No. A-13,114

**Figure 14**

specifications (sensitivity in mV/G for a linear device, or operate and release points in gauss for a digital device) can be used to determine the critical distances for a particular magnet and type of motion. Note that these field strength plots are not linear, and that the shape of the flux density curve depends greatly upon magnet shape, the magnetic circuit, and the path traveled by the magnet.

### TOTAL EFFECTIVE AIR GAP (TEAG)

Total effective air gap, or TEAG, is the sum of active area depth and the distance between the package's surface and the magnet's surface. A graph of flux density as a function of total effective air gap (figure 12A) illustrates the considerable increase in flux density at the sensor provided by a thinner package. The actual gain depends on the characteristic slope of flux density for a particular magnet.

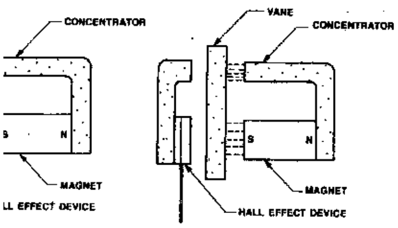
### MODES OF OPERATION

Even with a simple bar or rod magnet, there are several possible paths for motion. The magnetic pole could move perpendicularly straight at the active face of the Hall device. This is called the head-on mode of operation. The curve of figure 12B illustrates typical flux density (in gauss) as a function of TEAG for a cylindrical magnet.

The head-on mode is simple, works well, and is relatively insensitive to lateral motion. The designer should be aware that overextension of the mechanism could cause physical damage to the epoxy package of the Hall device.

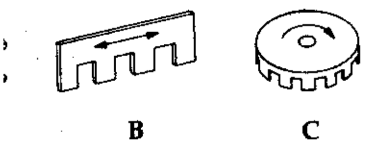
A second possibility would be to move the magnet in from the side of the Hall device in the slide-by mode of operation, as illustrated in figure 13. Note that now the distance plotted is not total effective air gap, but rather the perpendicular distance from the centerline of the magnet to the centerline of the package. Air gap is specified because of its obvious mechanical importance, but bear in mind that to do any calculations involving flux density, the "package contribution must be added and the TEAG used, as before. The slide-by mode is commonly used to avoid contact if overextension of the mechanism is likely. The use of strong magnets and/or ferrous flux concentrators in well-designed slide-by magnetic circuits will allow better sensing precision with smaller magnet travel than the head-on mode.

Magnet manufacturers generally can provide head-on flux density curves for their magnets, but they often do not characterize them for slide-by operation, possibly because different air gap choices lead to an infinite number of these curves; however, once an air gap is chosen, the readily available head-on magnet curves can be used to find the peak flux density (a single point) in the slide-by application by noting the value at the total effective air gap.



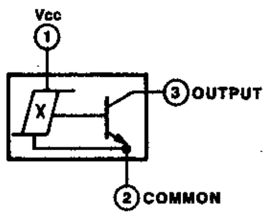

Dwg. No. A-13,115

**Figure 15**



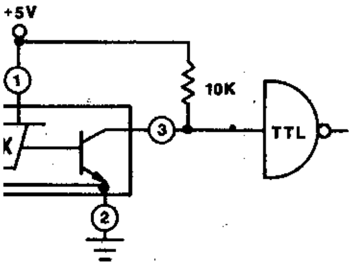

Dwg. No. A-13,116

**Figure 16**




Dwg. No. A-13,117

**Figure 17**




Dwg. No. A-13,118

**Figure 18A**

### STEEP SLOPES—HIGH FLUX DENSITIES

For linear Hall devices, greater flux changes for a given displacement give greater outputs, clearly an advantage. The same property is desirable for digital Hall devices, but for more subtle reasons. To achieve consistent switching action in a given application, the Hall device must switch ON and OFF at the same positions relative to the magnet.

To illustrate this concept, consider the flux density curves from two different magnet configurations in figure 14. With an operate-point flux density of 200 G, a digital Hall-effect device would turn ON at a distance of approximately 0.14 inches in either case. If manufacturing tolerances or temperature effects shifted the operate point to 300 G, notice that for curve A (steep slope) there is very little change in the distance at which switching occurs. In the case of curve B, the change is considerable. The release point (not shown) would be affected in much the same way. The basic principles illustrated in this example can be modified to include mechanism and device specification tolerances and can be used for worse-case design analysis. Examples of this procedure are shown in later sections.

### VANE-INTERRUPTER SWITCHING

In this mode, the activating magnet and the Hall device are mounted on a single rigid assembly with a small air gap between them. In this position, the Hall device is held in the ON state by the activating magnet. If a ferromagnetic plate, or vane is placed between the magnet and the Hall device, as shown in figure 15, the vane forms a magnetic shunt that distorts the flux field away from the Hall device.

Use of a movable vane is a practical way to switch a Hall device. The Hall device and magnet can be molded together as a unit, thereby eliminating alignment problems, to produce an extremely rugged switching assembly. The ferrous vane or vanes that interrupt the flux could have linear motion, or rotational motion, as in an automotive distributor. Ferrous vane assemblies, due to the steep flux density/distance curves that can be achieved, are often used where precision switching over a large temperature range is required.

The ferrous vane can be made in many configurations, as shown in figure 16. With a linear vane similar to that of figure 16B, it is possible to repeatedly sense position within 0.002 inch over a 125°C temperature range.

## ELECTRICAL INTERFACE FOR DIGITAL HALL DEVICES

The output stage of a digital Hall switch is simply an open-collector npn transistor. The rules for use are the same as those for any similar switching transistor.

Allegro MicroSystems, Inc.
115 Northeast Cutoff, Box 15036
Worcester, Massachusetts 01615-0036 (508) 853-5000

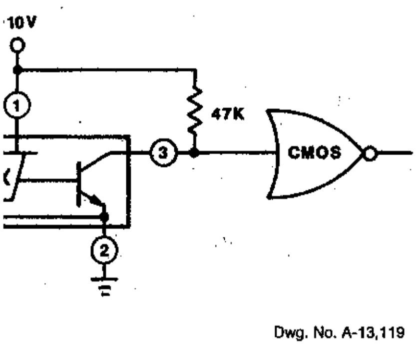



**Figure 18B**

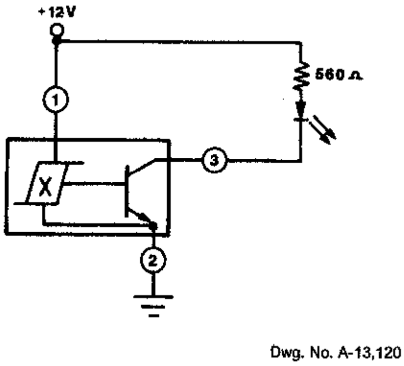



**Figure 19**

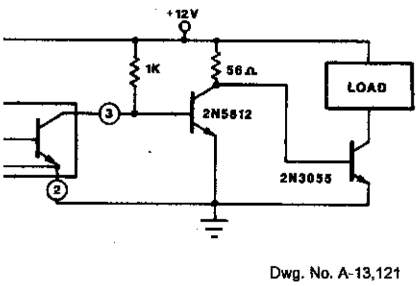



**Figure 20**

When the transistor is OFF, there is a small output leakage current (typically a few nanoamperes) that usually can be ignored, and a maximum (breakdown) output voltage (usually 24 V), which must not be exceeded.

When the transistor is ON, the output is shorted to the circuit common. The current flowing through the switch must be externally limited to less than a maximum value (usually 20 mA) to prevent damage. The voltage drop across the switch ($V_{CE(sat)}$) will increase for higher values of output current. You must make certain this voltage is compatible with the OFF, or "logic zero," voltage of the circuit you wish to control.

Hall devices switch very rapidly, with typical rise and fall times in the 400 ns range. This is rarely significant, because switching times are almost universally controlled by much slower mechanical parts.

## COMMON INTERFACE CIRCUITS

Figure 17 illustrates a simplified schematic symbol for Hall digital switches. It will make further explanation easier to follow.

Interface for digital logic integrated circuits usually requires only an appropriate power supply and pull-up resistor.

With current-sinking logic families, such as DTL or the popular 7400 TTL series (figure 18A), the Hall switch has only to sink one unit-load of current to the circuit common when it turns ON (1.6 mA maximum for TTL). In the case of CMOS gates (figure 18B), with the exception of switching transients, the only current that flows is through the pull-up resistor (about 0.2 mA in this case).

Loads that require sinking currents up to 20 mA can be driven directly by the Hall switch.

A good example is a light-emitting diode (LED) indicator that requires only a resistor to limit current to an appropriate value. If the LED drops 1.4 V at a current of 20 mA, the resistor required for use with a 12 V power supply can be calculated as:

$$\frac{12\ \text{V} - 1.4\ \text{V}}{0.02\ \text{A}} = 530\ \Omega$$

The nearest standard value is 560 Ω, resulting in the circuit of figure 19.

Sinking more current than 20 mA requires a current amplifier. For example, if a certain load to be switched requires 4 A and must turn ON when the activating magnet approaches, the circuit shown in figure 20 could be used.

the Hall switch is OFF (insufficient flux to operate), about 12 mA of ent flows through the 1 kΩ resistor 5812 transistor, thereby saturating it ng the base of the 2N3055 to hich keeps the load OFF. When a brought near the Hall switch, it shorting the base of the 2N5812 to id turning it OFF. This allows:

$$\frac{12\ \text{V}}{56\ \Omega} = 210\ \text{mA}$$

irrent to flow to the 2N3055, which to saturate it for any load current of s.

The Hall switch cannot source current to a load in its OFF state, but it is no problem to add a transistor that can. For example, consider using a 40669 triac to turn ON a 115 V or 230 V ac load. This triac would require about 80 mA of gate current to trigger it to the ON condition. This could be done with a 2N5811 PNP transistor, as shown below in figure 21.

When the Hall switch is turned ON, 9 mA of base current flows into the 2N5811, thereby saturating it and allowing it to supply 80 mA of current to trigger the triac. When the Hall switch is OFF, no base current flows in the 2N5811, which turns it OFF and allows no gate current to pass to the triac. The 4.7 kΩ and the 1 kΩ resistors were added as a safeguard against accidental turn-on by leakage currents, particularly at elevated temperatures.

**Note that the +12 V supply common is connected to the low side of the ac line, and in the event of a mixup, the Hall switch and associated low-voltage circuitry would be 115 V above ground. *Be careful!***

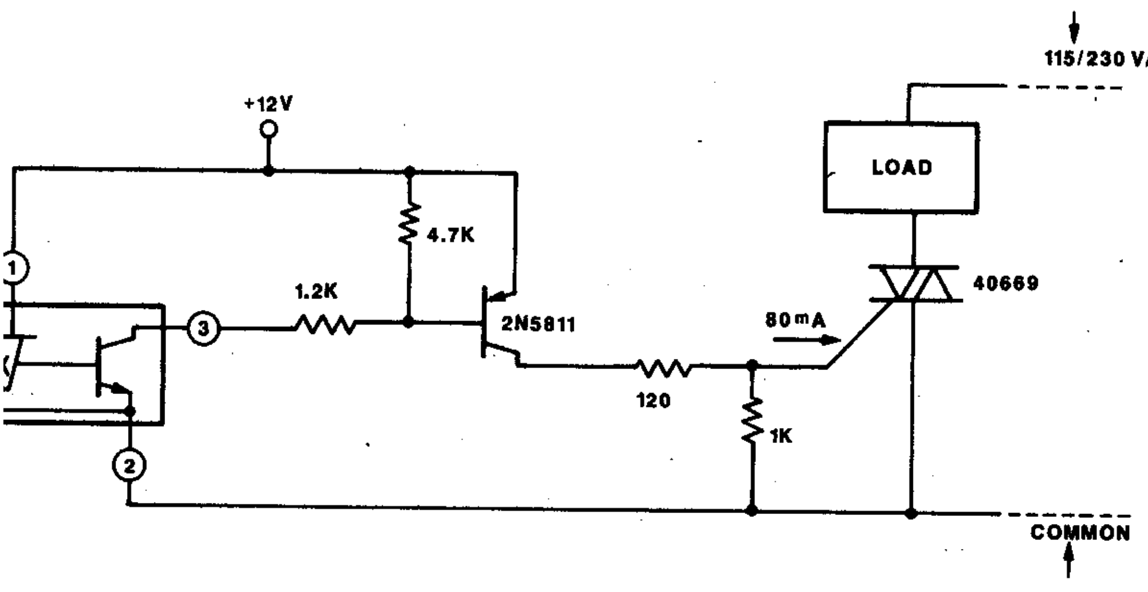



Figure 21