# Microprocessor-Based Automatic Door Opener

Jitendra Chaudhary
UG201110012
CSE, IIT Jodhpur

Hemraj Kumawat
*UG201110011*
CSE, IIT Jodhpur

***Abstract***— *A microprocessor controlled automatic door opener including means for detecting the velocity and direction of travel of the door. The micro-controller evaluates detected changes in either the velocity or direction of travel to determine the cause, and either reverses door travel direction or ignores the detection. The settings of the door are stored in the memory of the microprocessor. The microcontroller will regulate the opening and closing speed and the direction of door travel depending upon a preprogrammed sequence. The door control mechanism is able to differentiate between these internal factors and external obstructions.*

(Keywords: Automatic gate, Microprocessor, Programmable Input /Output Controller (PIO), light dependent resistor(LDR))

## 1. Introduction

The need for automatic gates has been on the increase in recent times. The system described in the project incorporates the use of a microprocessor as a controller in achieving the aims of this project.

It uses a remote control convenience to avoid the stress of manually opening and closing the gate. The technology used eliminates gate monitoring and manning by human beings. The gate uses a state-of-the-art entry system. The gates have to perform gyrations – open, auto-reverse, stop, fully close and fully stop. The system senses, opens and closes the gate, counts, and registers. The automatic gate system comprises a sensor unit, a trigger circuitry, CPU module, memory module, display unit, gate control unit and the power supply unit as shown in the block diagram below. The automatic gate system comprises a sensor unit, a trigger circuitry, CPU module, memory module, display unit, gate control unit and the power supply unit. As a monitoring and control system, the microprocessor was used to read in data values from the input device and interact with the outside world. The automatic gate developed in this project is unique in that it is controlled by software, which can be modified any time the system demands a change.

The automatic gate is not a security device and should not be construed as one. It provides convenient access and intelligent features that makes it distinct from all other gates which bring it so close to a security device.

## 2. Theory and Design

Certain specifications, parameters, and methods of implementation must be considered in system design and construction in order to give the expected result. The implementation of the design involves segmenting the overall system design into subsystems/modules/units, which are individually designed and tested before the integration of the various subsystems. The system design is divided into:

2.1) Hardware Design consisting of:

- Sensor Unit
- Trigger Circuit
- CPU Module
- Memory Module
- Gate Control Unit

2.2) Software

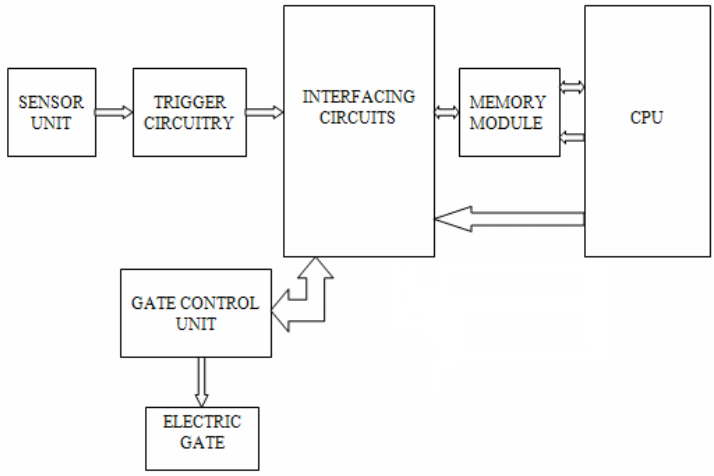


Figure: Block Diagram of the system

## I. SENSOR

The sensor provides an input signal to the system. This module makes use of an optical sensor, specifically a light dependent resistor (photo conductive cell, LDR), whose resistance changes with the intensity of light. When light rays are focused on the LDR, the resistance becomes very low (0-500Ω) but when the rays are interrupted, the resistance increases to its dark resistance. The variable resistor is used to vary the sensitivity of the LDR. It is otherwise called Dark Activated Sensor.

Two pairs of sensors (4 total) were used for the entire system; each pair for the entrance and exit gates and the outputs from the sensor units and is part of the trigger circuitry. The sensor unit is arranged in such a way that it consists of two pairs of LDRs to provide signals for the trigger circuitry whenever there is an obstruction through the entrance or exit gate For the design, two conditions are considered: first, when light rays are focused on the sensor resistor, and secondly, when the rays are being interrupted.

The circuit has the ability to detect only the passage of a person or other moving objects. Each pair of sensor is separated by a reasonable distance such that it can cover the complete gate. When a person or other moving objects obstruct the sensor, one sensing unit (or may be two) is activated and is processed by the trigger circuitry.

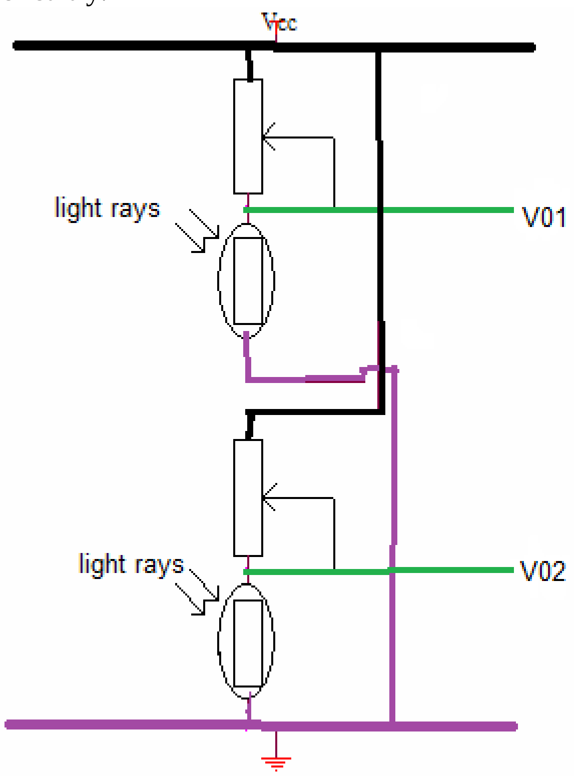


Figure: Sensor unit

When light rays of great intensity are focused on the sensor resistor, the output voltages, V01 and V02 are low (approximately 0V). When the light beams are interrupted, the output voltages increase to 5V approximately.

## II. TRIGGER CIRCUIT

The trigger circuitry serves as an Analog–to–Digital converter (ADC), which produces a HIGH signal when the beam is interrupted. This is made up of a Schmitt-trigger. It accepts the output of the two sensor units. Its function is to NOR the two inputs from the sensing unit, clip, and shape the pulse into square waves. It is configured in such a way that when there is an output from any of the two sensing units will the trigger circuitry go HIGH, else it remains at LOW level.

The trigger circuitry sends a signal to the interface unit, which is made up of Programmable Input /Output Controller (PIO). The software causes the microprocessor to be check the input port of the interface unit for the sensor status information (the outputs of the trigger circuitry). A HIGH value causes the microprocessor to send a signal to the output port of the interface unit in order to activate the DC motor to control the gate (open and close).

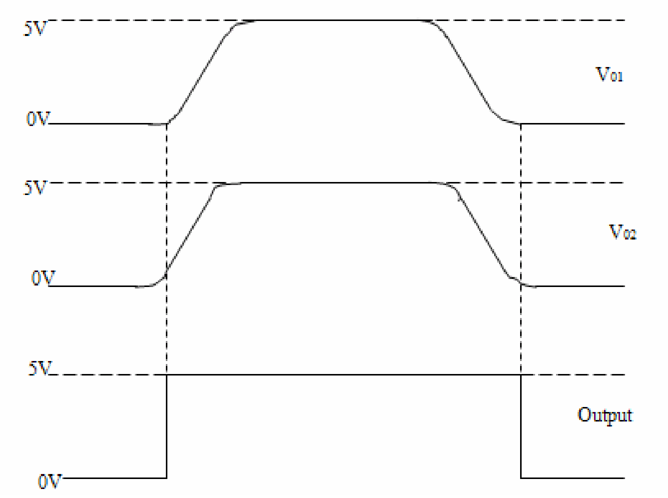


Figure: Waveform Schmitt-Trigger

When obstruct interrupts the light beams of the sensor units, the voltage at the output of the Schmitt-trigger goes HIGH. The circuit resets to a LOW voltage level when the interrupt is removed.

**Table 1:** Truth Table for Schmitt-Trigger

| Inputs | | Outputs |
|---|---|---|
| V01 | V02 | F |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

## III. **CPU MODULE**

This module provides the system clock, reset, and access provision to address, data, and control buses architecture. Additional circuitries are needed for the Z80 CPU to perform its functions. These circuitries, with their design, are discussed as follows.

### CLOCK CIRCUITRY

The Z80 requires a single-phase 0V to 5V clock signal. This can be generated with the use of a crystal oscillator Because the typical high level output voltage of a TTL gate is only 3.3V or 3.4V and the minimum high level required at the Z80 clock input is 4.4V, a 330Ω pull up resistor is required.

In this design, a Z80 CPU was used, which has a clock cycle of 2.5MHz. Thus, a crystal oscillator of 5MHz was used and passed through a D flip-flop to give 2.5MHz (divide by 2), which is able to drive the Z80 clock rate.

### RESET CIRCUITRY

This circuit functions to initialize the Z80 after power is supplied and also to revitalize it if a “HALT” occurs or after some other catastrophic event. When the Z80 is reset, it begins execution from memory location 0000H because the program counter is cleared to zero and forced to begin at location 0000H. It also disables interrupts and clears the interrupt enabled flip-flop.

### MEMORY MODULE

The memory devices interfaced to the CPU module consists of:

a. 2K X 8 ROM

b. 2K X 8 RAM

In order to interface a microprocessor to ROM/RAM chips, two address decoding techniques are usually employed.

- Linear select decoding technique
- Fully decoding technique

In linear select decoding, each bit on the address bus can select a device. This is employed in very small systems with few devices. It does not require any decoding hardware but wastes a large amount of address space and sometimes causes bus conflict.

Fully decoding memory addressing requires the use of a decoder to select a memory device. In the hardware design implementation, the fully decoding technique was employed for memory decoding. Address bus lines A0-A7 were used for the input/output mapping.

This indicates that a particular area of memory is being addressed or pointed to by the microprocessor. This can be realized by use of combinational circuits. The most useful features of this device are the multiple enable inputs. The decoder has an active low output that only becomes active when all of the enables are at their active levels.

Whenever this device is enabled, the 3-bit binary number present at the address inputs causes one of the output pins to become active.

## IV. MEMORY MAPPING

The memory mapping of the system design is shown in figure 3.Interfacing both memory devices to the CPU does not require any additional circuitry to generate wait states. The reason being that both devices meet the memory read/write timing specification of the Z80 CPU.

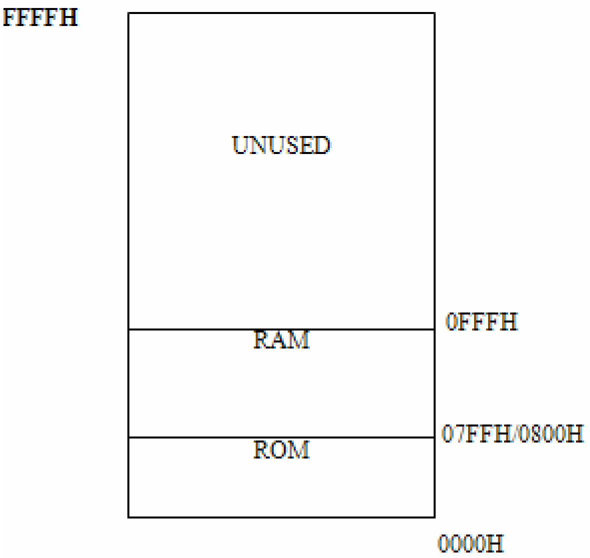


Figure: Memory Mapping

## V. GATE CONTROL UNIT

The gate control unit is made up of:

- PNP and NPN transistors

- Diodes
- Motor

The PNP and NPN transistors are arranged in such a way that a pair (PNP and NPN) controls the opening of the gate through the motor and the other pair reverse the polarity of the motor by rotating it in the opposite direction to close the gate. There is a time interval of 10 seconds between the opening and closing of the gate. The software varies this. The arrangement of the diodes serves to protect the transistors from reverse-bias polarity and the resistors serve to improve switching times.

The motor is used to control the opening and closing of gate. The electric (DC) motor used is one that has the ability to rotate in both directions simply by reversing the polarity.

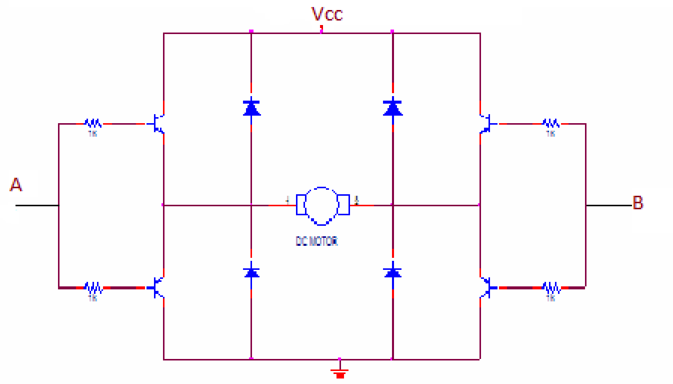


Figure: Gate Control Unit

LOW signal output from a transistor buffer through the Z80 PIO applied to point A bases the NPN and PNP transistors and these cause the motor to rotate in a particular direction. Similarly, a LOW signal applied to point B reverse (change) the rotation of the motor in the opposite direction. The control circuit is used for both entrance and exit gates.

The power supply unit supplies the required DC voltage needed by the entire microcomputer system. A microprocessor based system design has to be activated with a clean power supply of good regulation characteristics. A transient on the power line could send the microprocessor wandering, resulting in system failure. Z80 operates on a voltage VCC = 5V ±10% and as a result of this, the power supply unit designed is 5V DC and is not affected by variation in the AC voltage serving as input to the transformer.

### SOFTWARE DESIGN

In the development cycle of a microprocessor-based system, decisions are made on the parts of the system to be realized in the hardware design and the parts to be implemented in software. The software is decomposed into modules so that each module can be individually tested as a unit and debugged before the modules are integrated and tested as a software system in order to ensure that the software design meets its specification.

## 3. IMPLEMENTATION

### SOFTWARE DEVELOPMENT PROCEDURE

The program for the system is written in assembly language. Assembly code represents halfway position between machine code and a high level language. The assembly code is usually a mnemonic derived from the instruction itself, i.e. LDA means Load the Accumulator. Assembly code is thus very easy to remember and use when writing programs.

When entering an assembly program into a microprocessor, the assembly code is first to be converted into machine code. For short programs, of a few lines, this is relatively easy and usually requires that the programmer construct a table which contains the assembly mnemonics and the equivalent machine code and the values of data in registers and pointer stack.

For longer programs, a separate program called an assembler program, is used to convert the assembly code into machine code, which is placed directly into the microprocessor memory. The program modules are segmented into:

a. Main program
b. Sensor subroutine
c. Delay subroutine
d. Output (Gate Control) subroutine

The software was designed using the following steps:

1. Algorithm
2. Flowchart
3. Assembly Language Codes.

### ALGORITHM

The algorithm used to implement the program for the system described in this paper is as follows:

START
1. Initialize the Microprocessor
2. Fetch the status of the sensor bit

3. Compare the status of the sensor bit with entrance code
a. If status = entrance code then step 5
b. Else step 4

4. Go to step 2

5. Gate open, wait and close

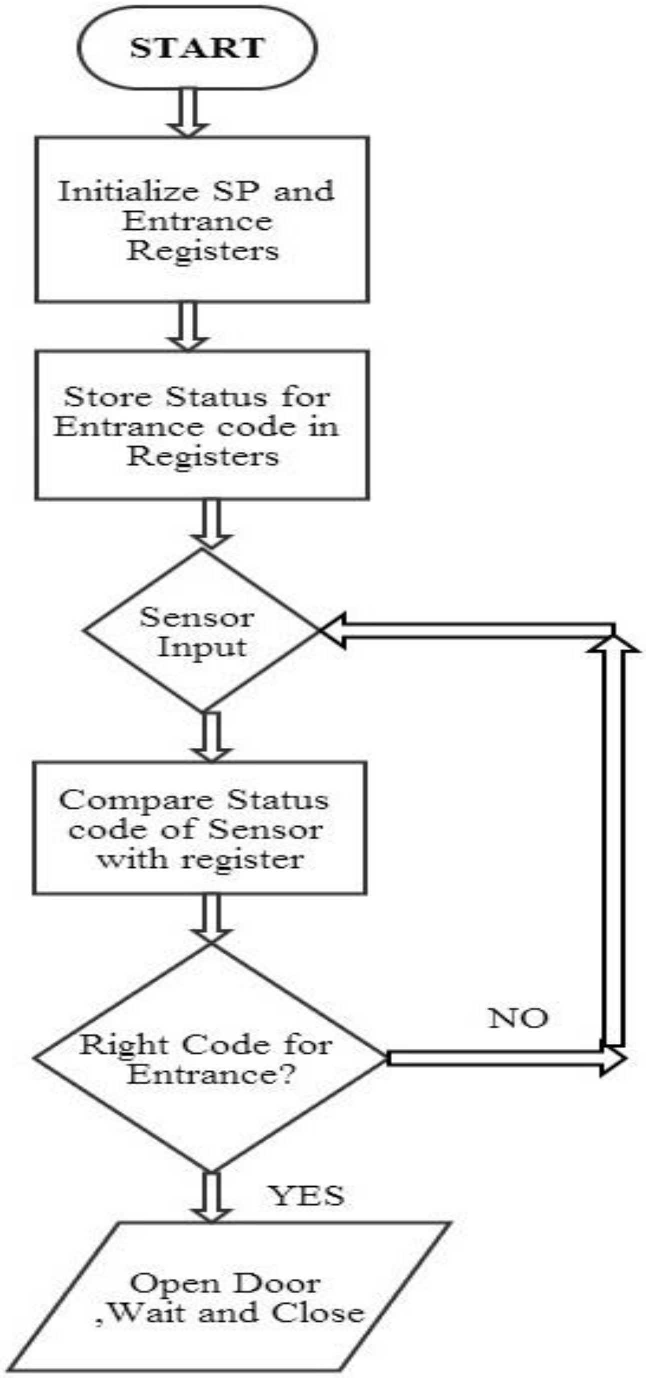


Fig. Flow Chart Diagram for Automatic Door Opener

**THE ASSEMBLY LANGUAGE**

**INITIALIZATION**

LD SP, 0FFFH //Load Value to Stack Pointer.
LD A, 3FH //Load accumulator with 3FH value
OUT 03H //Output data from accumulator to 1st sensor port
OUT 06H //Output data from accumulator to 2nd sensor port
OUT 07H // Output data from accumulator to motor port

**MAIN PROGRAM**

**START**

//These hexadecimal locations are respectively for accessing data from memory to port and back to memory.
LD B, 80H
LD C, 01H
LD D, 14H
LD A, 00H
OUT (01H)
OUT (04H)
OUT (05H)

**MAIN**

IN A, (00H) //Input data to accumulator from Sensor
CP 00H //checking condition of entrance
JP Z, INPUT //JUMP to Input
LD E, A
LD A, B
JP Z, GATE // Jump to gate

**ENTRANCE**

LD A, C
JP Z, GATE
NOP //No Operation
NOP
JP START

## 4. TESTING AND RESULTS

This Assembly language code was tested by 8085 Microprocessor Simulator Program:
Test case: 1 When first sensor rays were blocked by an Obstacle then the value of stored register conditioned to rotate the motor from Point A, the gate opened, wait and Closed.
Test Case 2: When 2nd Sensor rays were blocked by An Obstacle then the Combination of registers cause low voltage at Point A and then the Door opened, wait for some time and Closed.

## 5. CONCLUSION

The construction of a microprocessor based system had been achieved in this project. This design can be easily adapted to any electric gate and any form of control which requires the use of sensors. The Design includes the basic sensor characteristics, microprocessor input and output interfacing, and assembly language principles. Sensors for vehicle detection while the programming language is fundamental to software design based on the system requirements, specifications, and planned operation of the system. There is total agreement between the system designed and the required operation of the system.
Every good project has limitations; the limitation of this design lies in the effectiveness of the sensor. The sensor will work most effectively if operated under high intensity light

## 6. FURTHER IMPROVEMENTS

For an improved, effective system to be implemented and achieved, the following suggestions should be considered for further work.
1. There can be a Display Unit for showing number of persons entered in a particular room.
2. A better sensor is recommended to achieve new functionality. For instance, a suitable sensor such as radar sensor that could detect contraband goods in any vehicle.
3. To achieve full automation, a real time system should be employed and a Closed Circuit Television (CCTV) system provided for proper monitoring and security purposes. This can helpful in detecting the presence of vehicles before the system is activated.
4. Upgrading the system using a higher bit microprocessors for speed optimization.
5. Along with this system we can use Face-detection through Camera for Automated Attendance System can be used.

## 7. ACKNOWLEDGMENT

We would like to take the opportunity to thank Prof K.R.Chowdhary, Head of Department Computer Science MBM Jodhpur. We would like to thank to our colleagues who helped to make our project successful.

## 8. REFERENCES

[1] Hall, Douglas V. 1991. *Microprocessors and Interfacing Programming and Hardware. 2nd edition.* Gregg College Division: New York, NY.

[2] Leventhal, Lance A. 1978. *8080A, 8085 Assembly Language Programming*. McGraw-Hill, Inc.: New York, NY.

[3] Private Door Openers. 2006. “Private Door Information”. Lombard, IL. http://www.privatedoor.com.

[4] McGlynn, Daniel R. 1976. *Microprocessor Technology and Application*. John Wiley & Sons, Inc: New York, NY.

[5] Digital Systems; Principles and Application. 7th Edition. Ronald J. Tocci & Neil S. Widmer. 1998. Printice-Hall International: Princeton, NJ.

[6] Microprocessor and Logic Design, Krutz, R.L. 198, John Wiley & Sons, Inc.: New York, NY.

[7] “Private Door Information”. Private Door Openers. 2006. Lombard, IL. http://www.privatedoor.com.