A D PATEL INSTITUTE OF TECHNOLOGY
NEW V V NAGAR

(Affiliated to Gujarat Technological University)

ELECTRONICS AND COMMUNICATION ENGINEERING (ECE) DEPARTMENT

PROJECT-I
REPORT

Subject Code:170001

"Autonomous Car Space Allocation System"

Submitted By:

Pipalia Jaydeep P (090010111035)
Patel Jay R (090010111027)
Mayani Ridham S (090010111017)

Guided By:

Ami Shah
Assistant Professor,
ADIT

Acknowledgements:

We are fortunate to have many useful suggestions for our project from faculty Members and student friends, which has helped in improving performance of project. We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend our sincere thanks to all of them.

We are highly thankful to Prof. AMI SHAH for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing full the project.

We wish to express my sincere thanks to our team members as well as our college faculty for their technical support which helped us to enhance our project performance.

We feel extremely happy to come out with this project, with support of many individuals who have willingly helped us out with their abilities and we are sincerely thankful to them.

Jay Patel	Mayani Ridham	Pipalia jaydeep
(090010111027)	(090010111017)	(090010111035)

Abstract:

This project is basically a parking system that can be used in residential areas, malls etc. It allows the user to know whether the parking is full or not. If the parking has empty space then the empty slots are displayed on LCD. This makes it easier for the user to guide the car directly to the empty slot. The status is displayed on LCD and thus doesn't require manpower to monitor the system.

There is a sensor at the entry gate that detects the entry of the car and increments the count by one. There is another sensor at the exit gate that detects the exit of the car and decrements the count by one.

There are ultra sonic sensors in every parking slot that detects the presence of the car and displays the status of the parking on LCD.

The system doesn't require any manpower for monitoring this makes the system autonomous.

INDEX

List of figures and tables

Chapter 1

Figures

Chapter 2

Table

Figures

Chapter 3

Figures

CHAPTER 1

INTRODUCTION

1.1 Overview:

Our project is based on 8051 micro controller. It is a parking system which can be used in various malls and residential areas like flats.

1.2 Major work area of our project:

- We have designed the loader circuit of 8051 for programming the chip using UART. The chip use is 89v51rd2 of Phillips (NXP).
- LCD module interface with 89v51rd2.
- Sensor interface with 89v51rd2. (IR sensor and ultrasonic sensor)
- Seven segment interface with 89v51rd2.

1.3 Detailed description of system:

System is required to display the available parking slots on an LCD out of the total slots. As per the requirements two IR sensors have been set up at the entry and exit gate of the parking. An LCD has been initialized to zero count when the parking is empty. The sensor at the entry gate increments the counter when the car enters the parking, the other sensor at the exit gate decrements the counter when the car leaves the parking. This gives the user an idea whether parking is available or not.

System is now required to display which parking slots are empty if the parking is available. Suppose the parking area has 8 slots of parking and if any slot is empty the LCD displays the slot number which is empty else it displays that the parking is full.

1.4 Block Diagram:

Counter

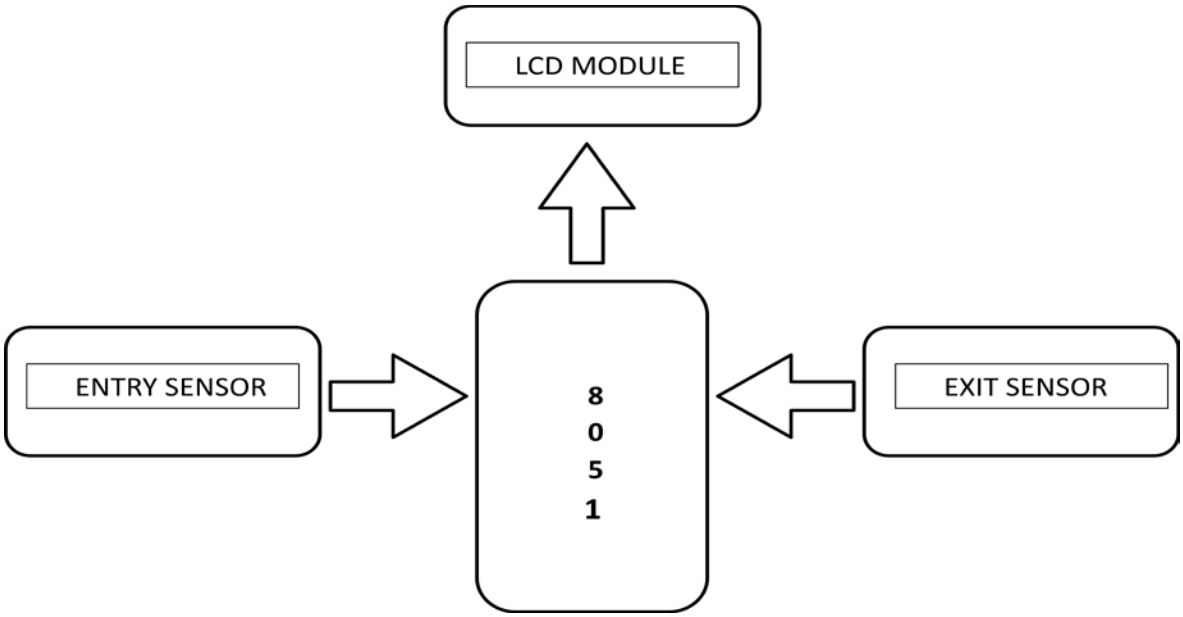


Fig: 1.4.1

Space allocation block diagram:

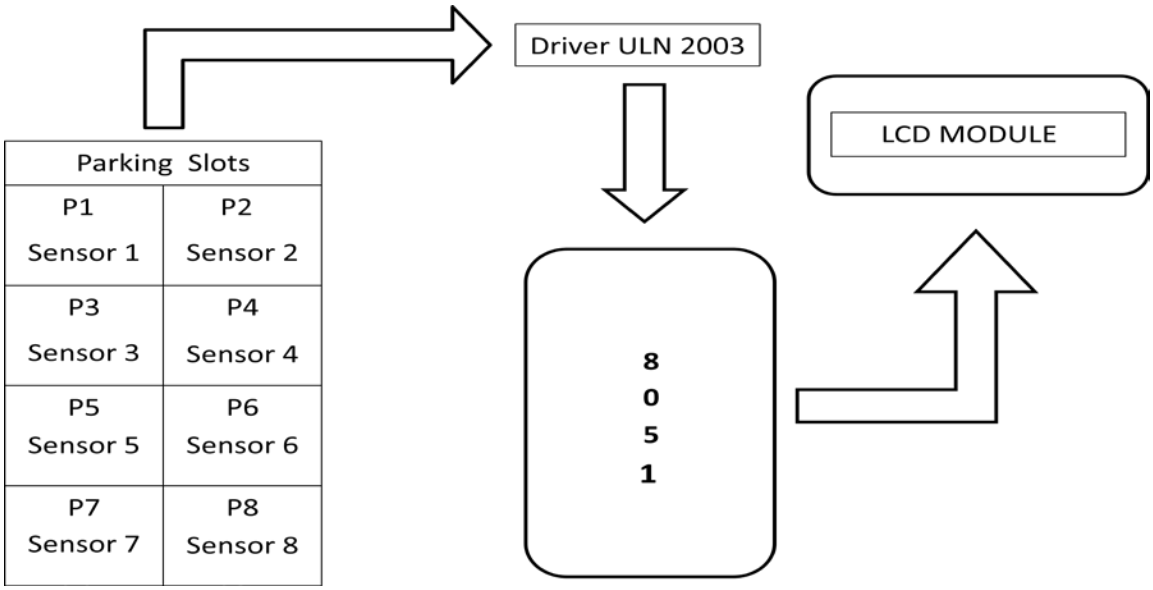


Fig: 1.4.2

CHAPTER 2

Integrated circuit details

2.1 MOCROCONTROLLER 89V51RD2

General description:

The P89V51RD2 is an 80C51 microcontroller with 64 KB Flash and 1024 bytes of data RAM. A key feature of the P89V51RD2 is its X2 mode option. The design engineer can choose to run the application with the conventional 80C51 clock rate (12 clocks per machine cycle) or select the X2 mode (6 clocks per machine cycle) to achieve twice the throughput at the same clock frequency. Another way to benefit from this feature is to keep the same performance by reducing the clock frequency by half, thus dramatically reducing the EMI.

The Flash program memory supports both parallel programming and in serial In-System Programming (ISP). Parallel programming mode offers gang-programming at high speed, reducing programming costs and time to market. ISP allows a device to be reprogrammed in the end product under software control. The capability to field/update the application firmware makes a wide range of applications possible.

The P89V51RD2 is also In-Application Programmable (IAP), allowing the Flash program memory to be configured even while the application is running.

Capabilities of 8051:

- Internal ROM & RAM
- I/O ports with programmable pins
- Timers & Counters
- Serial data communication

Features:

- 80C51 Central Processing Unit
- 5 V Operating voltage from 0 to 40 MHz
- 64 KB of on-chip Flash program memory with ISP (In-System Programming) and
- IAP (In-Application Programming)
- Supports 12-clock (default) or 6-clock mode selection via software or ISP
- SPI (Serial Peripheral Interface) and enhanced UART
- PCA (Programmable Counter Array) with PWM and Capture/Compare functions
- Four 8-bit I/O ports with three high-current Port 1 pins (16 mA each)
- Three 16-bit timers/counters
- Programmable Watchdog timer (WDT)
- Eight interrupt sources with four priority levels
- Second DPTR register

- Low EMI mode (ALE inhibit)
- TTL- and CMOS-compatible logic levels

Block diagram:

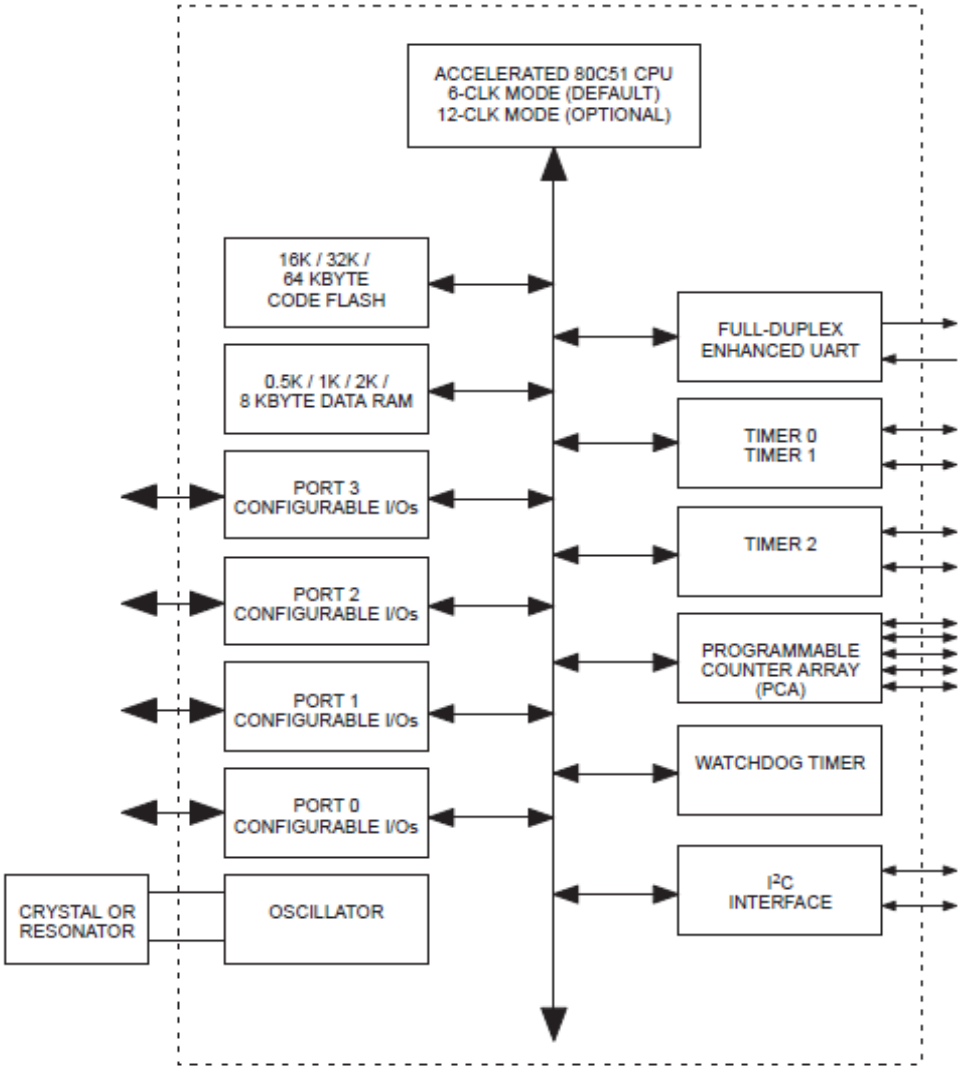


Fig: 2.1.1

Hardware description:

Pin diagram:

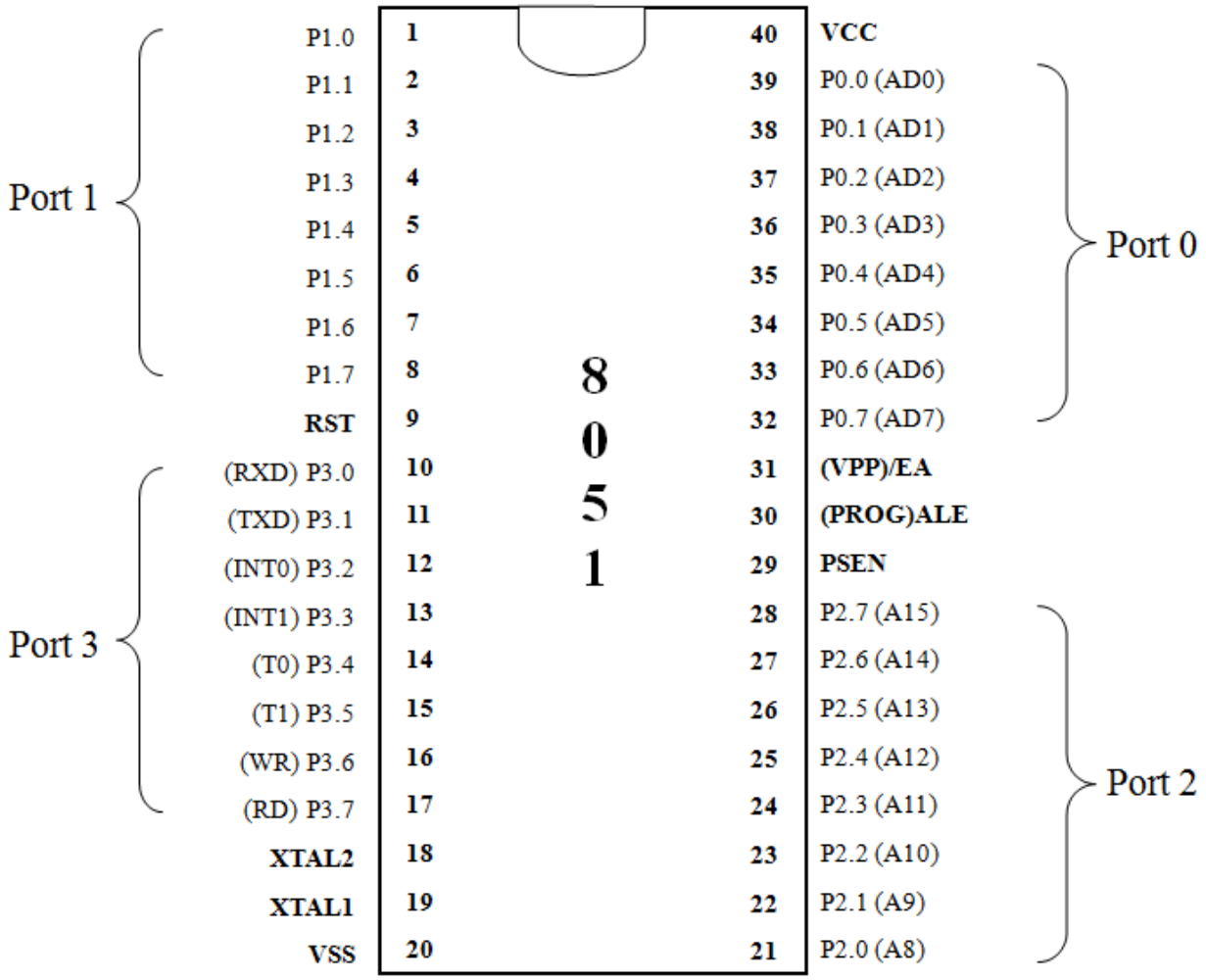


Fig: 2.1.2

Pin description:

PINS	TYPE	DESCRIPTION
P0.0 to P0.7	I/O	**Port 0:** Port 0 is an 8-bit open drain bi-directional I/O port. Port 0 pins that have '1's written to them float, and in this state can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external code and data memory. In this application, it uses strong internal pull-ups when transitioning to '1's. Port 0 also receives the code bytes during the external host mode programming, and outputs the code bytes during the external host mode verification. External pull-ups are required during program verification or as a general purpose I/O port.
P1.0 to P1.7	I/O with internal pull-up	**Port 1:** Port 1 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 1 pins are pulled high by the internal pull-ups when '1's are written to them and can be used as inputs in this state. As inputs, Port 1 pins that are externally pulled LOW will source current (IIL) because of the internal pull-ups. P1.5, P1.6, P1.7 have high current drive of 16 mA. Port 1 also receives the low-order address bytes during the external host mode programming and verification.
P1.0	I/O	**T2:** External count input to Timer/Counter 2 or Clock-out from Timer/Counter 2
P1.1	I	**T2EX**: Timer/Counter 2 capture/reload trigger and direction control
P1.2	I	**ECI**: External clock input. This signal is the external clock input for the PCA
P1.3	I/O	**CEX0**: Capture/compare external I/O for PCA Module 0. Each capture/compare module connects to a Port 1 pin for external I/O. When not used by the PCA, this pin can handle standard I/O.

P1.4	I/O	**SS**: Slave port select input for SPI **CEX1**: Capture/compare external I/O for PCA Module 1
P1.5	I/O	**MOSI**: Master Output Slave Input for SPI **CEX2**: Capture/compare external I/O for PCA Module 2
P1.6	I/O	**MISO**: Master Input Slave Output for SPI **CEX3**: Capture/compare external I/O for PCA Module 3
P1.7	I/O	**SCK**: Master Output Slave Input for SPI **CEX4**: Capture/compare external I/O for PCA Module 4
P2.0 to P2.7	I/O with internal pull-up	**Port 2**: Port 2 is an 8-bit bi-directional I/O port with internal pull-ups. Port 2 pins are pulled HIGH by the internal pull-ups when '1's are written to them and can be used as inputs in this state. As inputs, Port 2 pins that are externally pulled LOW will source current (IIL) because of the internal pull-ups. Port 2 sends the high-order address byte during fetches from external program memory and during accesses to external Data Memory that use 16-bit address (MOVX@DPTR). In this application, it uses strong internal pull-ups when transitioning to '1's. Port 2 also receives some control signals and a partial of high-order address bits during the external host mode programming and verification
P3.0 to P3.7	I/O with internal pull-up	**Port 3**: Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins are pulled HIGH by the internal pull-ups when '1's are written to them and can be used as inputs in this state. As inputs, Port 3 pins that are externally pulled LOW will source current (IIL) because of the internal pull-ups. Port 3 also receives some control signals and a partial of high-order address bits during the external host mode programming and verification.
P3.0	I	**RXD**: serial input port
P3.1	O	**TXD**: serial output port

P3.2	I	**INT0**: external interrupt 0 input
P3.3	I	**INT1**: external interrupt 1 input
P3.4	I	**T0**: external count input to Timer/Counter 0
P3.5	I	**T1**: external count input to Timer/Counter 1
P3.6	O	**WR**: external data memory write strobe
P3.7	O	**RD**: external data memory read strobe
PSEN	I/O	**Program Store Enable**: PSEN is the read strobe for external program memory. When the device is executing from internal program memory, PSEN is inactive (HIGH). When the device is executing code from external program memory, PSEN is activated twice each machine cycle, except that two PSEN activations are skipped during each access to external data memory. A forced HIGH-to-LOW input transition on the PSEN pin while the RST input is continually held HIGH for more than 10 machine cycles will cause the device to enter external host mode programming.
RST	I	**Reset**: While the oscillator is running, a HIGH logic state on this pin for two machine cycles will reset the device. If the PSEN pin is driven by a HIGH-to-LOW input transition while the RST input pin is held HIGH, the device will enter the external host mode; otherwise the device will enter the normal operation mode.
EA	I	**External Access Enable**: EA must be connected to VSS in order to enable the device to fetch code from the external program memory. EA must be strapped to VDD for internal program execution. However, Security lock level 4 will disable EA, and program execution is only possible from

		internal program memory. The EA pin can tolerate a high voltage of 12 V.
ALE/ PROG	I/O	**Address Latch Enable:** ALE is the output signal for latching the low byte of the address during an access to external memory. This pin is also the programming pulse input (PROG) for flash programming. Normally the ALE is emitted at a constant rate of 1 for external timing and clocking. One ALE pulse is skipped during each access to external data memory. However, if AO is set to '1', ALE is disabled.
XTAL1	I	**Crystal 1**: Input to the inverting oscillator amplifier and input to the internal clock generator circuits.
XTAL2	O	**Crystal 2:** Output from the inverting oscillator amplifier.
VDD	I	**Power supply**
VSS	I	**Ground**

Table: 2.1.1

Internal data memory

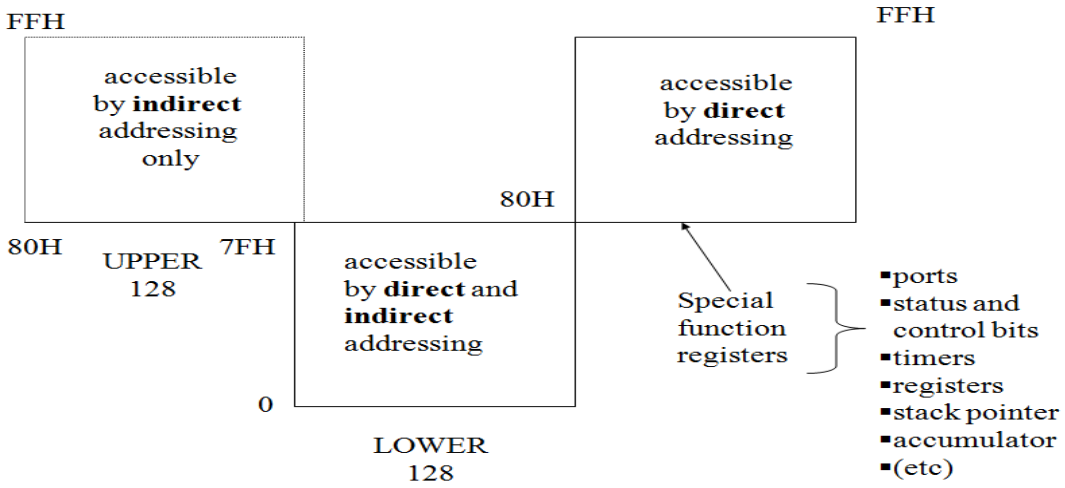


Fig: 2.1.3

The lower 128 bytes of Internal RAM

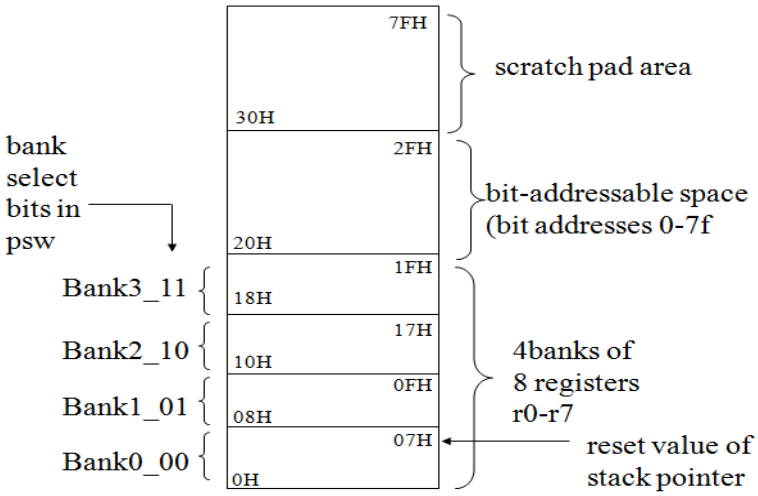


Fig: 2.1.4

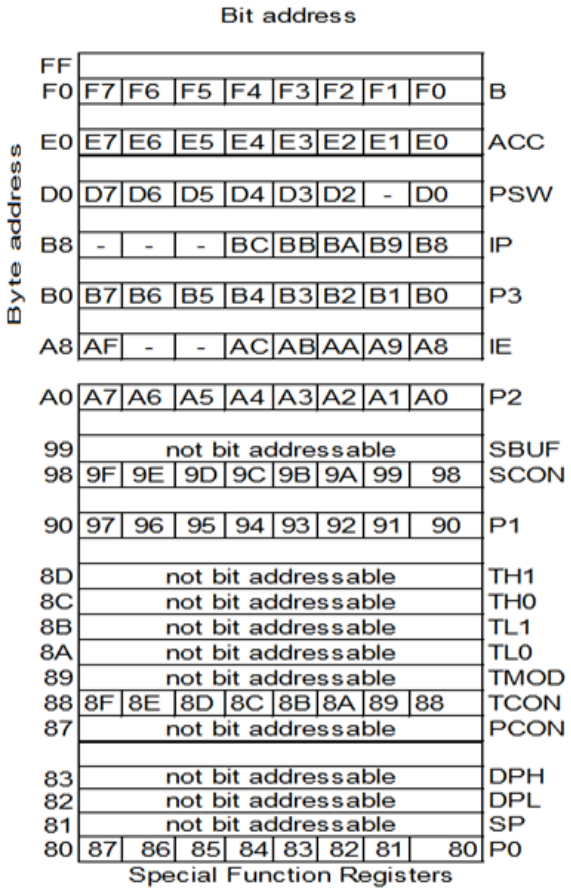


Fig: 2.1.5

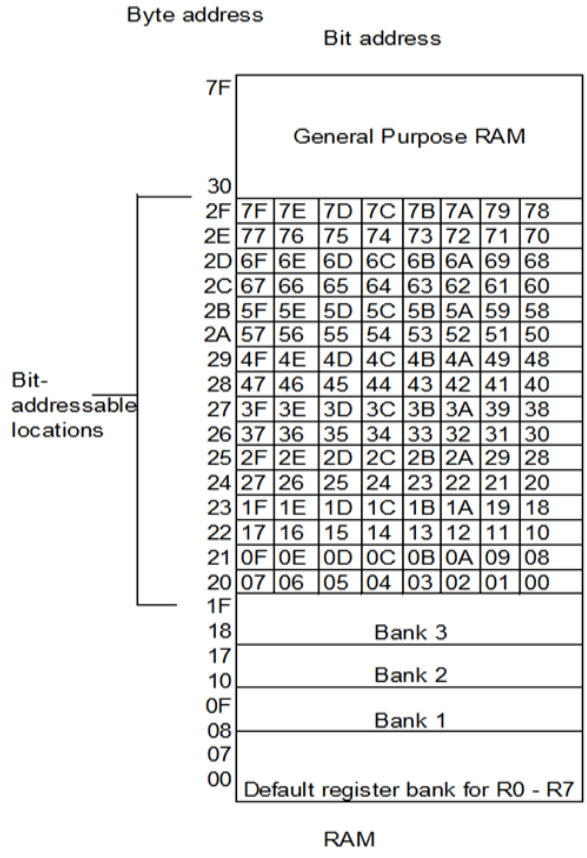


Fig: 2.1.6

Using the on chip oscillator

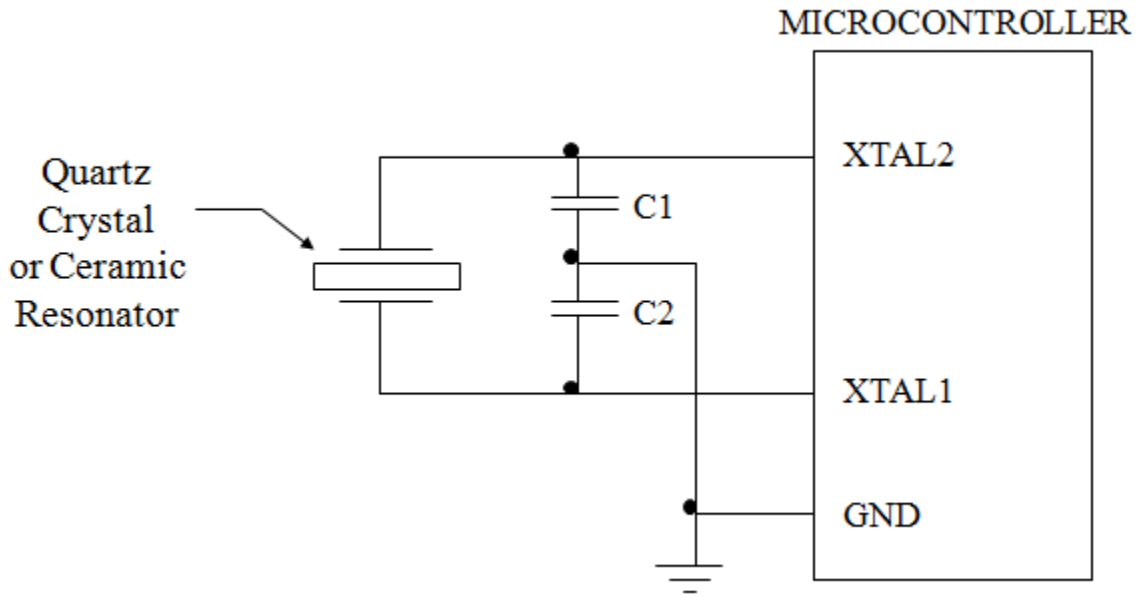


Fig: 2.1.7

External clock drive configuration

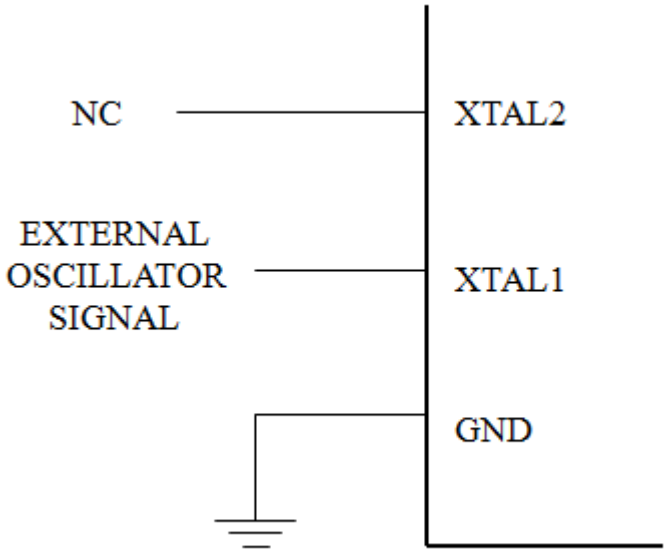


Fig: 2.1.8

Power on reset

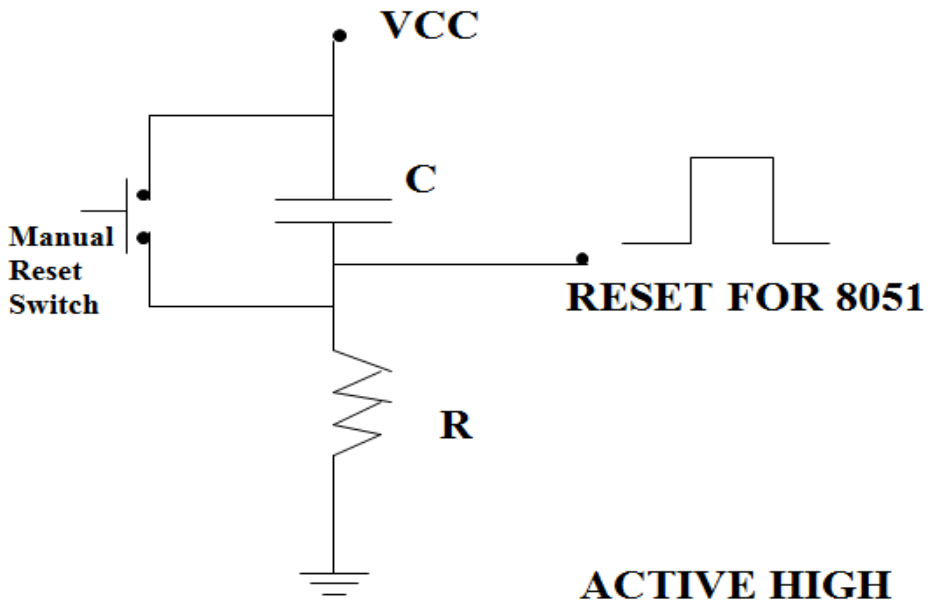


Fig: 2.1.9

Regulated power supply using 7805 voltage regulator

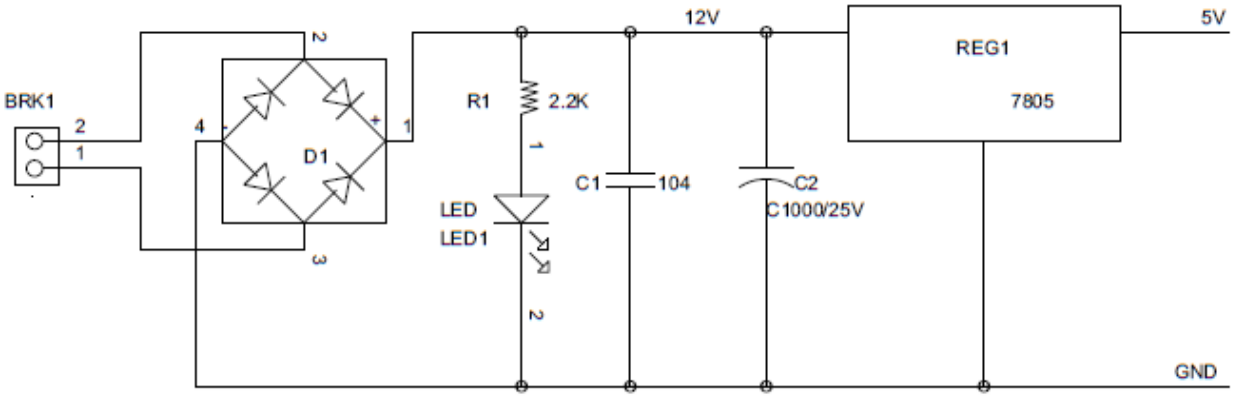


Fig: 2.1.10

Basic circuit diagram:

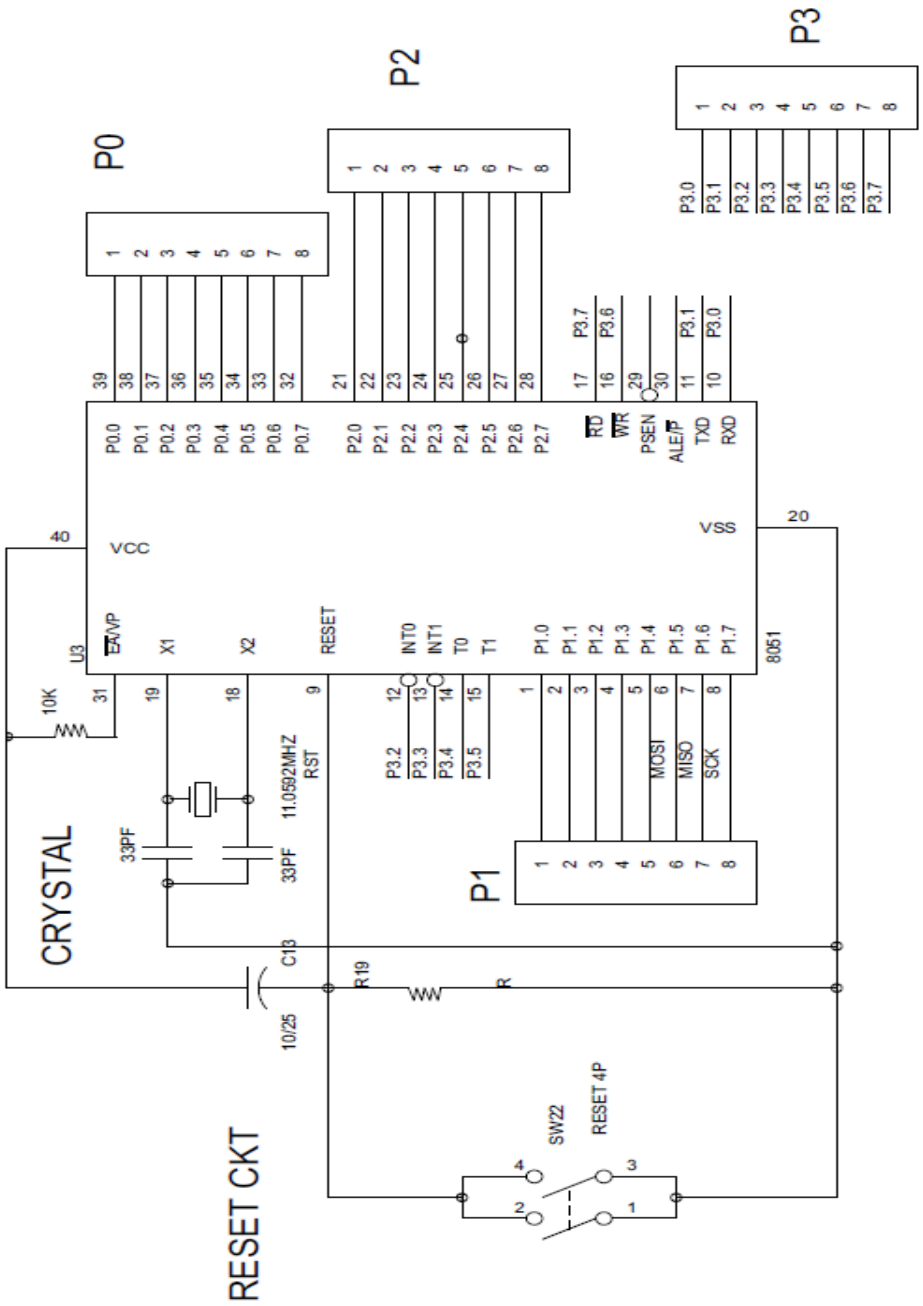


Fig: 2.1.11

Basic configuration required for the working of 89v51rd2. The configuration has on board oscillator and reset circuitry. Power supply to the chip is obtained through voltage regulator 7805. The IC can now be programmed using UART, ISP or external programmer like super-prog or universal programmer. The peripherals can be interfaced and applications can me designed.

2.2 7805 -Positive Voltage Regulator

Description:

The L7800 series of three-terminal positive regulators is available in TO-220, TO-220FP, TO-220FM, TO-3 and D2PAK packages and several fixed output voltages, making it useful in a wide range of applications. These regulators can provide local on-card regulation, eliminating the distribution problems associated with single point regulation. Each type employs internal current limiting, thermal shut-down and safe area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltage and currents.

Features:

- Output current to 1.5a
- Output voltages of 5; 5.2; 6; 8; 8.5; 9;10; 12; 15; 18; 24v
- Thermal overload protection
- Short circuit protection
- Output transition SOA protection

Packages:

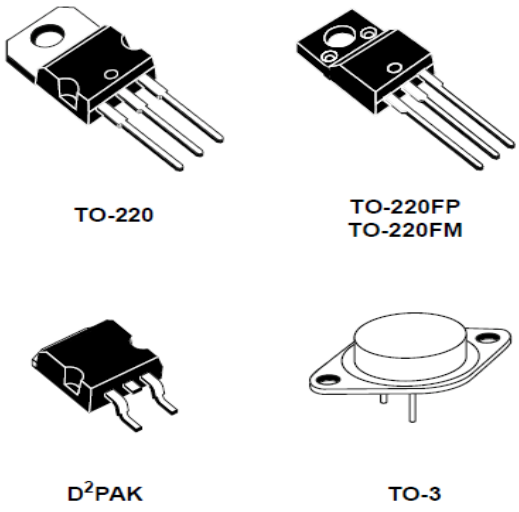


Fig: 2.2.1

Schematic Diagram:

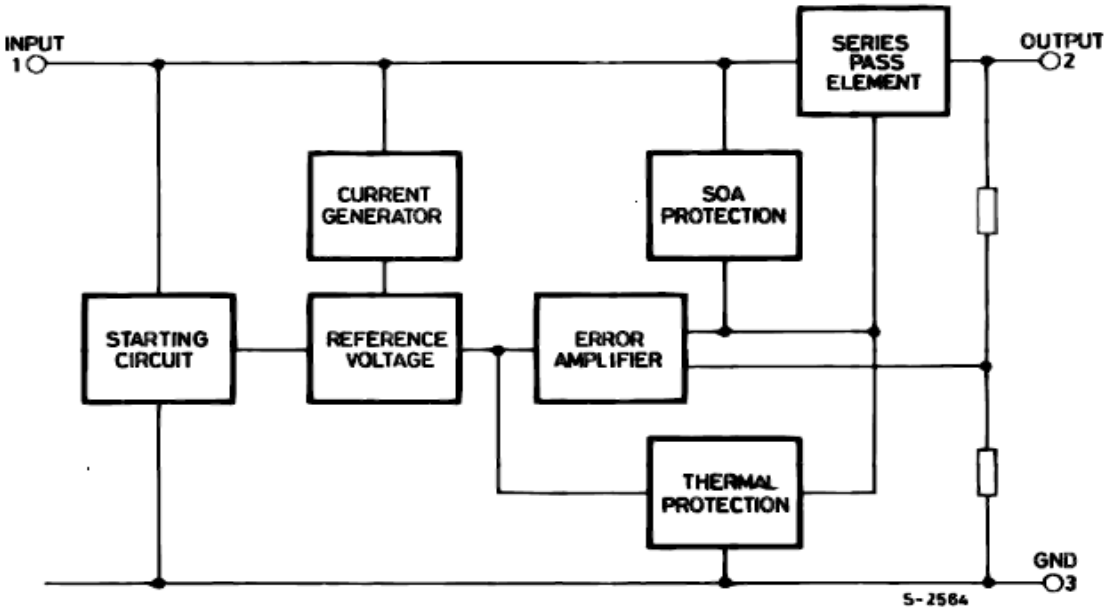


Fig: 2.2.2

Regulated power supply using 7805 voltage regulator

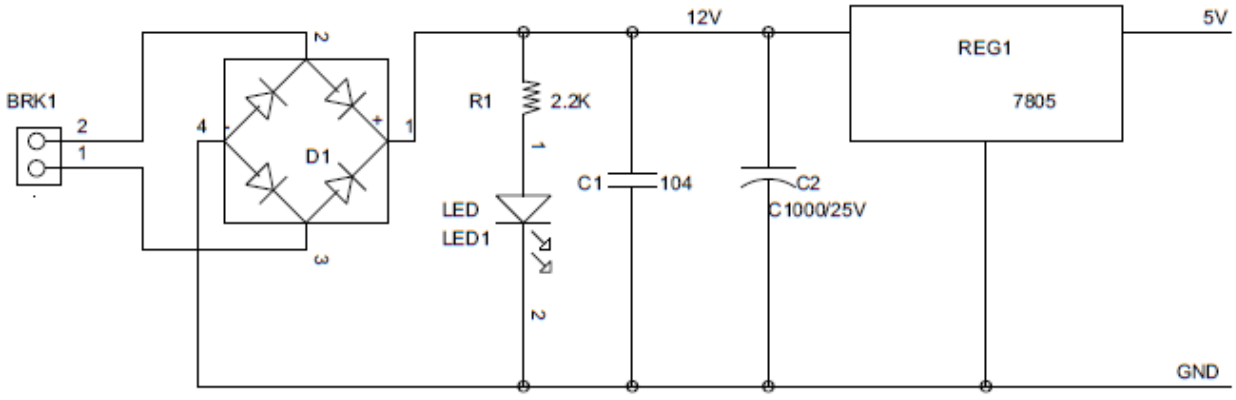


Fig: 2.2.3

2.3 ULN 2003 Driver IC

High voltage and high current Darlington transistor array description:

The ULN2003 is a monolithic high voltage and high current Darlington transistor arrays. It consists of seven NPN Darlington pairs that feature high-voltage outputs with common-cathode clamp diode for switching inductive loads. The collector-current rating of a single Darlington pair is 500mA. The Darlington pairs may be paralleled for higher current capability. Applications include relay drivers, hammer drivers, lamp drivers, display drivers (LED gas discharge), line drivers, and logic buffers. The ULN2003 has a 2.7kW series base resistor for each Darlington pair for operation directly with TTL or 5V CMOS devices.

Features:

- 500mA rated collector current(Single output)
- High-voltage outputs: 50V
- Inputs compatible with various types of logic.
- Relay driver application_

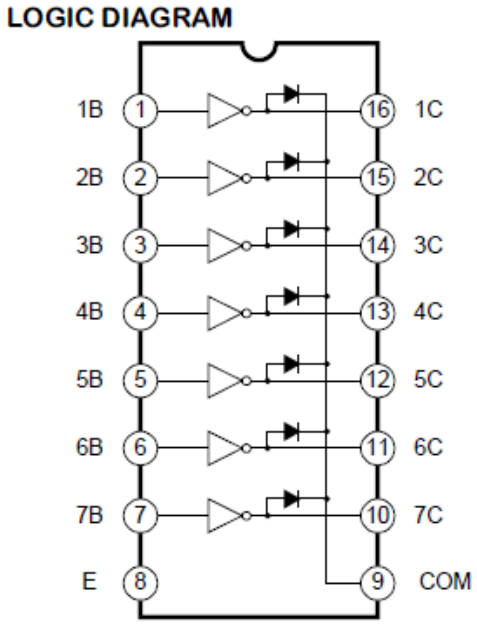


Fig: 2.3.1

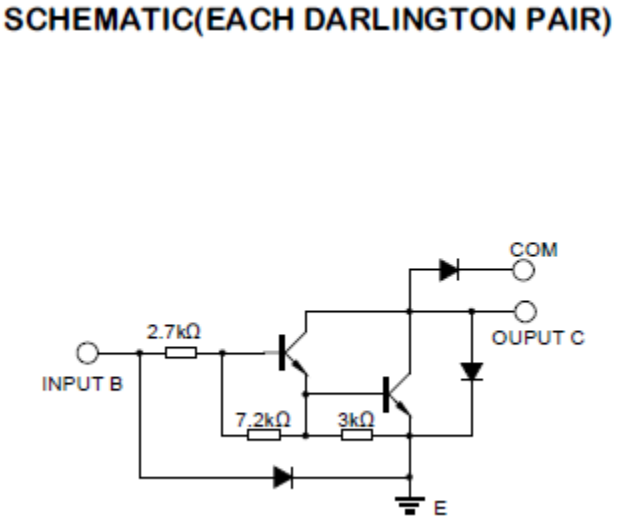


Fig: 2.3.2

2.4 MAX 232

Description:

The MAX232 device is a dual driver/receiver that includes a capacitive voltage generator to supply EIA-232 voltage levels from a single 5-V supply. Each receiver converts EIA-232 inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V and a typical hysteresis of 0.5 V, and can accept ±30-V inputs. Each driver converts TTL/CMOS input levels into EIA-232 levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments Lin-ASICE library. The MAX232 is characterized for operation from 0°C to 70°C. The MAX232I is characterized for operation from –40°C to 85°C.

Features:

- Operates With Single 5-V Power Supply
- LinBiCMOSE Process Technology
- Two Drivers and Two Receivers
- ±30-V Input Levels
- Low Supply Current . . . 8 mA Typical
- Meets or Exceeds TIA/EIA-232-F and ITU Recommendation V.28
- Designed to be Interchangeable With Maxim MAX232

Applications:

- TIA/EIA-232-F Battery-Powered Systems
- Terminals
- Modems
- Computers

Packages:

Package Options Include Plastic Small-Outline (D, DW) Packages and Standard Plastic (N) DIPs

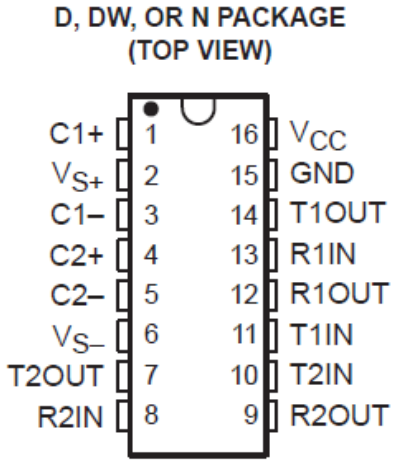
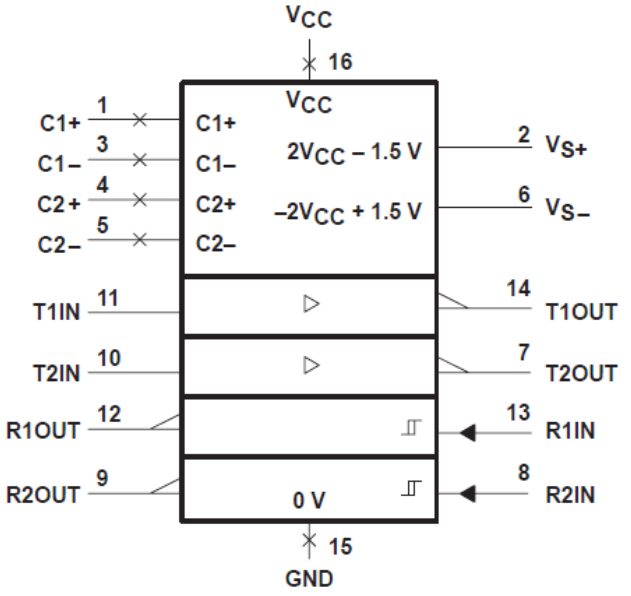


Fig: 2.4.1

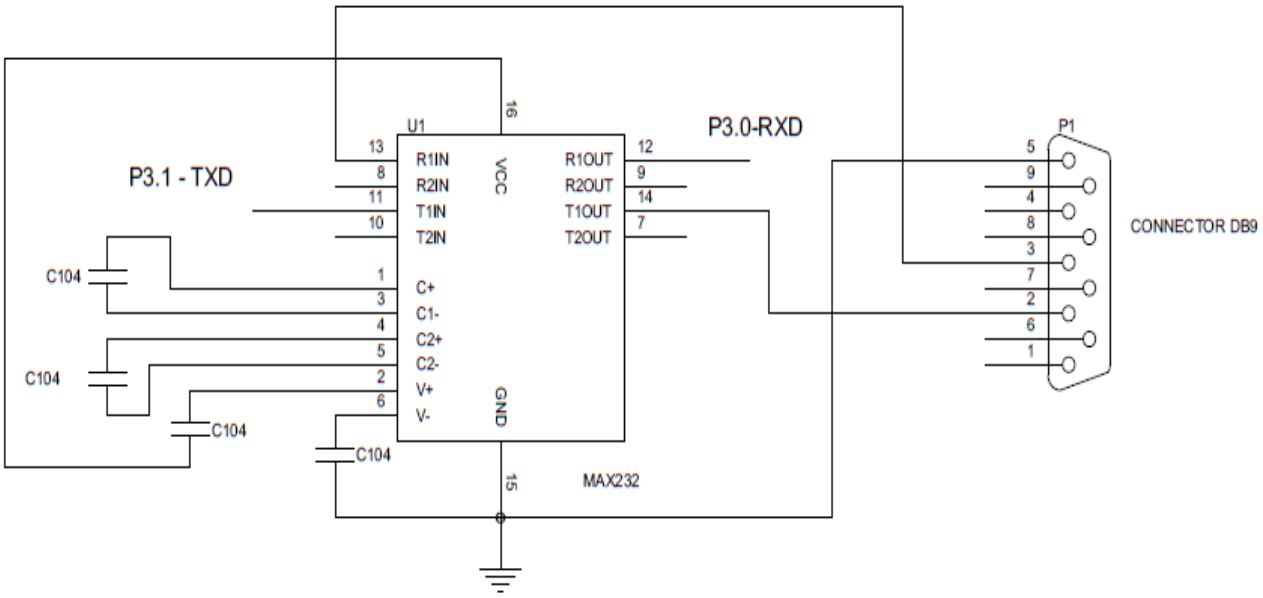
logic symbol†



† This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.

Fig: 2.4.2

MAX 232 Connections:



RS232 SECTION

Fig: 2.4.3

2.5 Infrared sensor:

Infrared sensor works on infrared light. There is an infrared transmitter and an infrared detector at the opposite ends. The object is detected when the sensor gets cut.

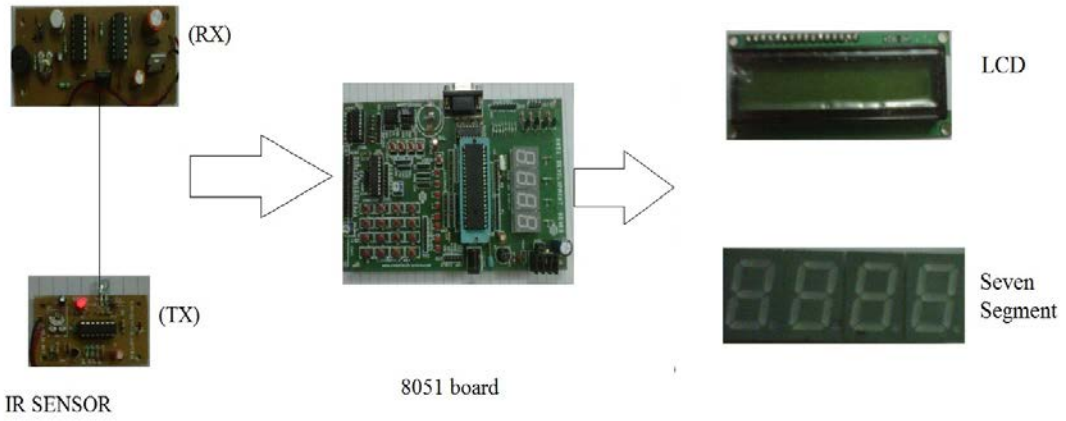
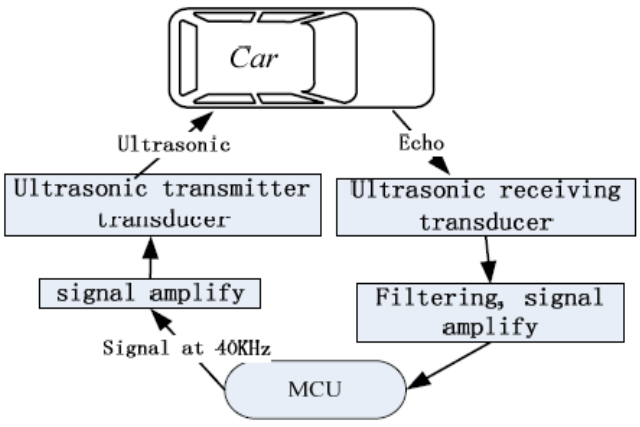


Fig: 2.5.1

2.6 Ultrasonic sensor:

This sensor works on ultra sonic sound. It has a transmitter that generates ultra sonic sound and a receiver that detects the reflected sound. The sound gets reflected from the near by object and the receiver detects the reflected sound the objects can thus be detected.



Design schematic of ultrasonic sensor

Fig: 2.6.1

The actual photograph of ultrasonic sensor module

Fig: 2.6.2

CHAPTER 3

Result and analysis

3.1 Implementing car counter on seven segment display:

Cars counter application:

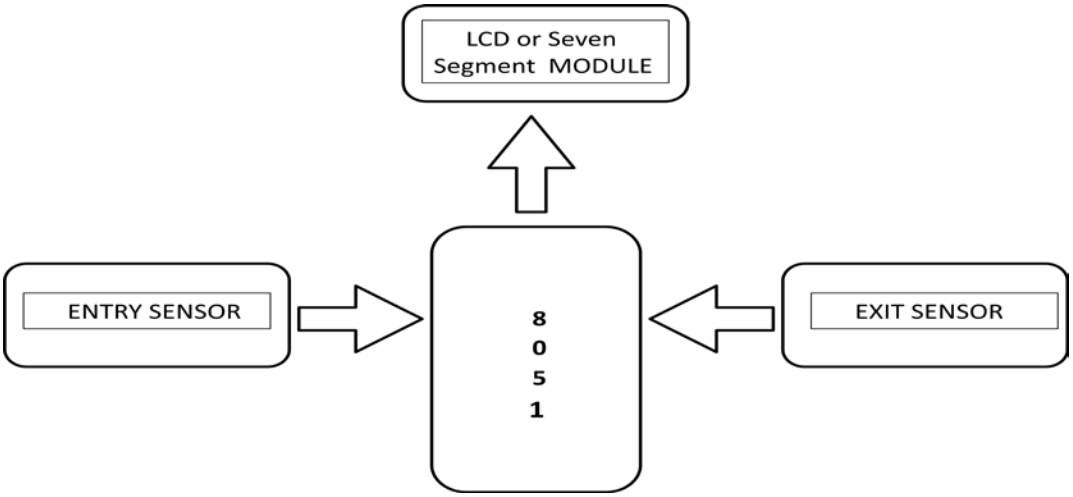
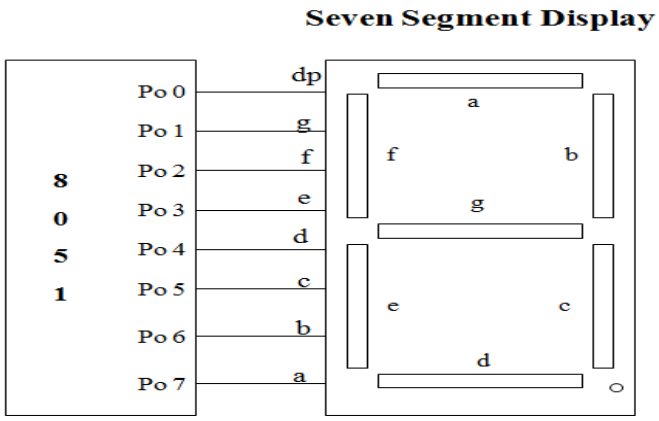


Fig: 3.1.1

Seven segment display




```
;               MOVC A,@A+DPTR
;---------------------
                MOV     A,#06
                MOV     DPTR,#DISP_LUT

                MOVC    A,@A+DPTR
                MOV     P0,A
                JMP     $
DISP_LUT:
;                       -abcdefgd-

                DB      11111100b   ;0
                DB      01100000b   ;1
                DB      11011010b   ;2
                DB      11110010b   ;3
                DB      01100110b   ;4
                DB      10110110b   ;5
                DB      10111110b   ;6
                DB      11100000b   ;7
                DB      11111110b   ;8
                DB      11100110b   ;9
```

Fig: 3.1.2

Interfacing of multiple Seven segment display

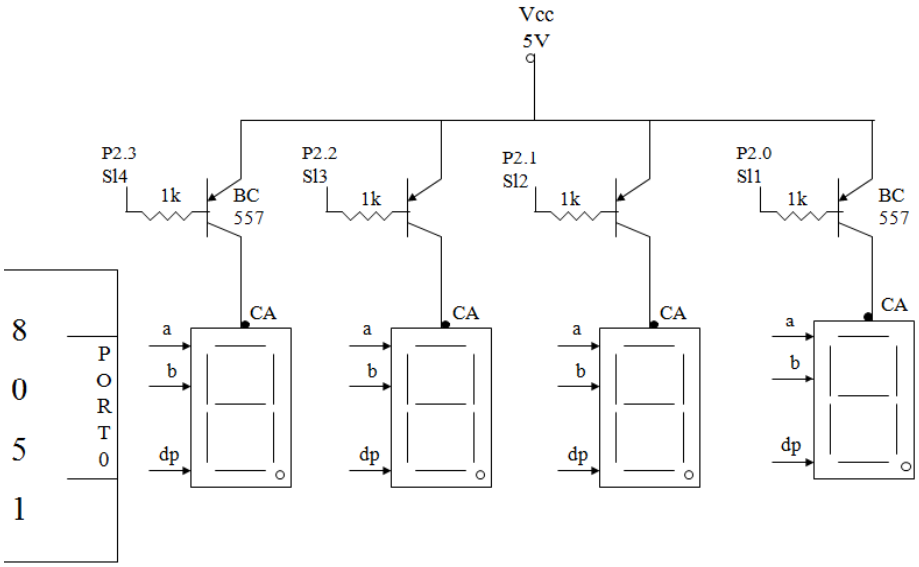


Fig: 3.1.3

Configuration:

- Seven segment is connected on port P0.
- Sensors have been interfaced on INT0 andINT1 pins.
- Counter has been initialized to 0.
- Interrupts have been enabled.
- Interrupts have been configured in edge triggering mode.

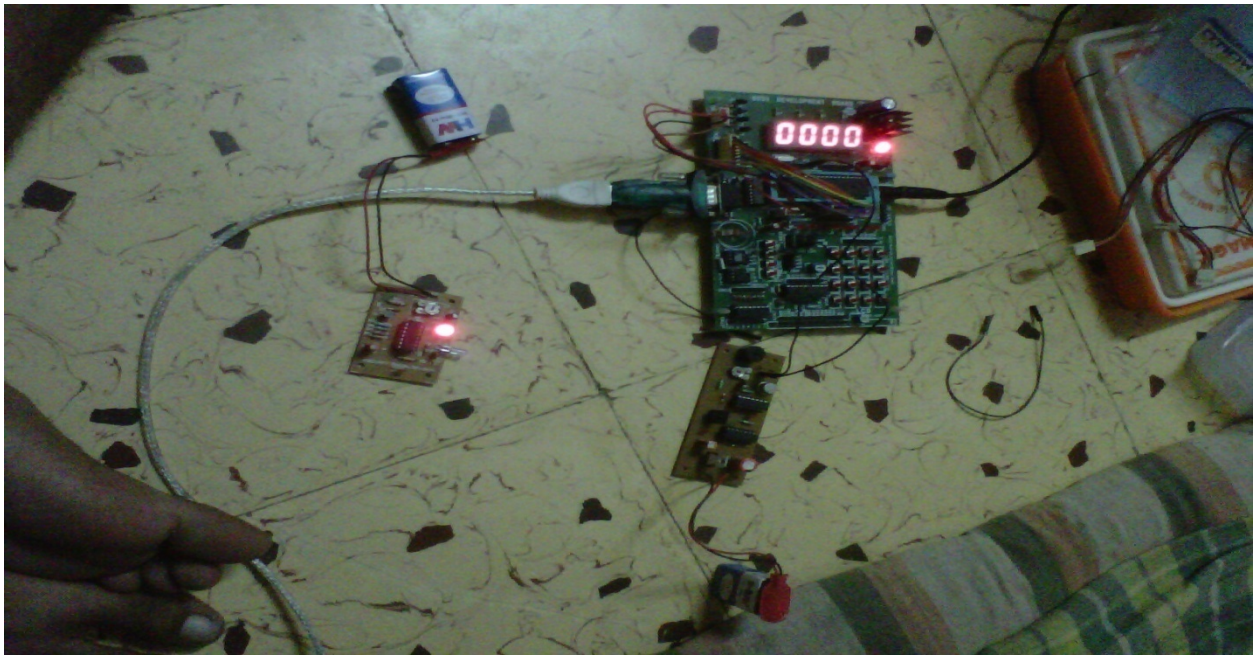
Working:

Initially the counter has been initialized to zero count. The external interrupts have been enabled in negative edge triggering mode. INT0 interrupt when called increments the count by 1. INT1 interrupt when called decrements the count by 1. Initially on reset INT1 interrupt has been disabled in order to prevent the reverse count. The INT1 gets enabled only when once INT0 interrupt is called.

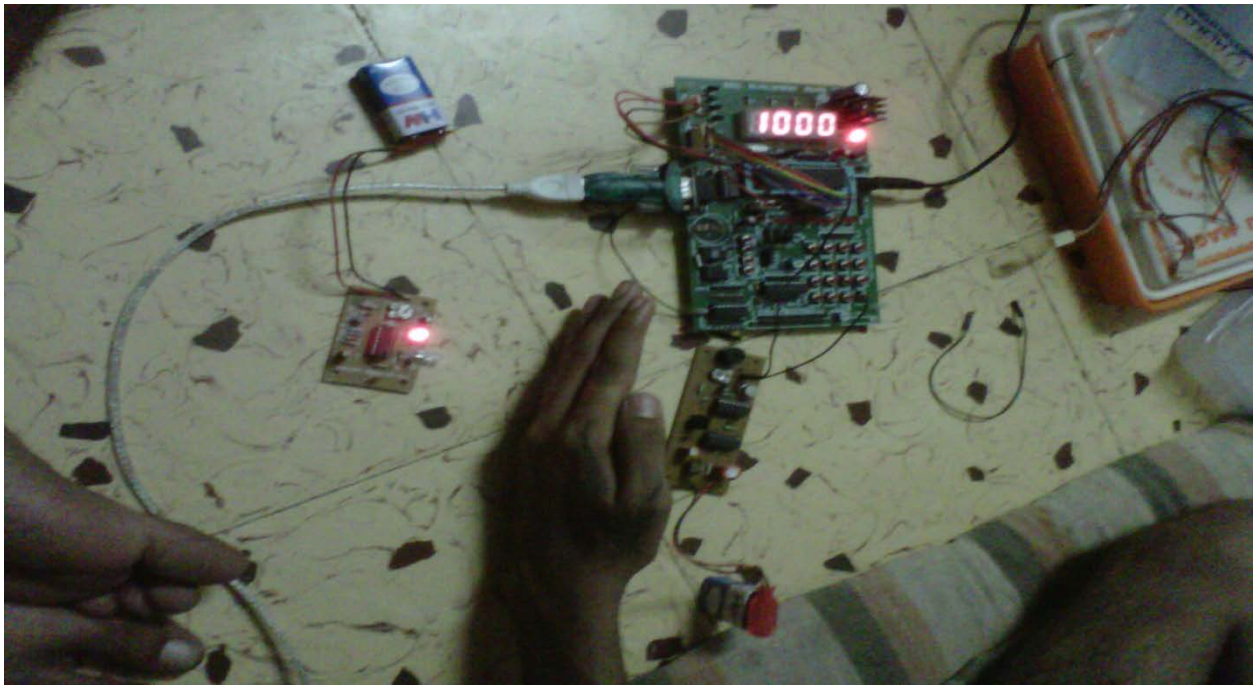
When car enters the entry gate the sensor gets cut and the counter increments. At the exit gate when the car leave counter decrements. The count is displayed in seven segment and this gives user an idea of how many cars have been parked in the parking.

Output:

1. Initially system is set up and the counter is initialized to zero, seven segment display shows initial count zero.



2. When the sensor gets cut counter increments to the next value and displays the value on seven segment display.



3.2 Implementing car counter on LCD display

Interfacing LCD on 8051, pin configuration:

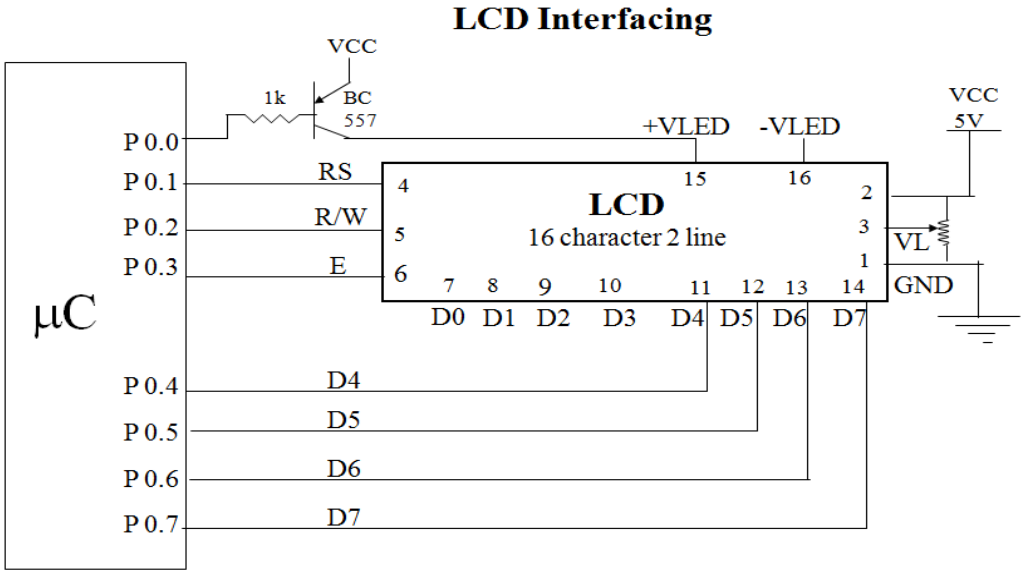
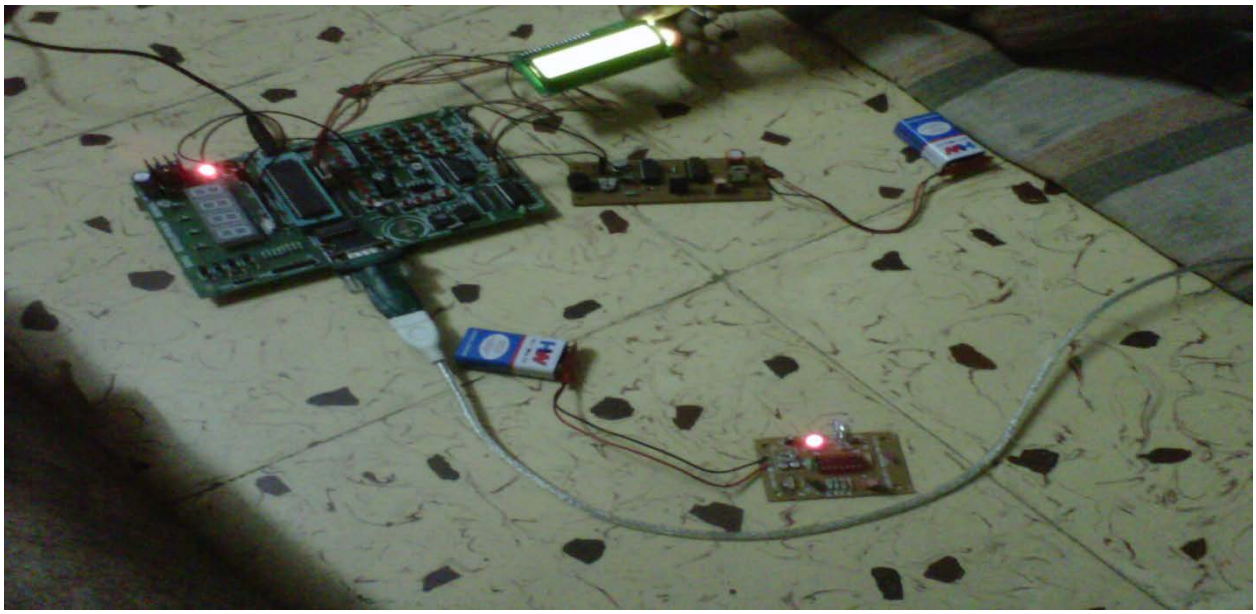


Fig: 3.2.1

The above counter can also be implemented on an LCD display instead of seven segment. The working of the system remains the same only the program needs to be modified. Here the LCD has been interfaced in port P0 as shown above.

Output:

1. Initially system is set up and the counter is initialized to zero, LCD displays zero initially.



2. The counter increments to the next value when the sensor gets cut and displays the value on LCD.

3.3 Implementing Space allocation:

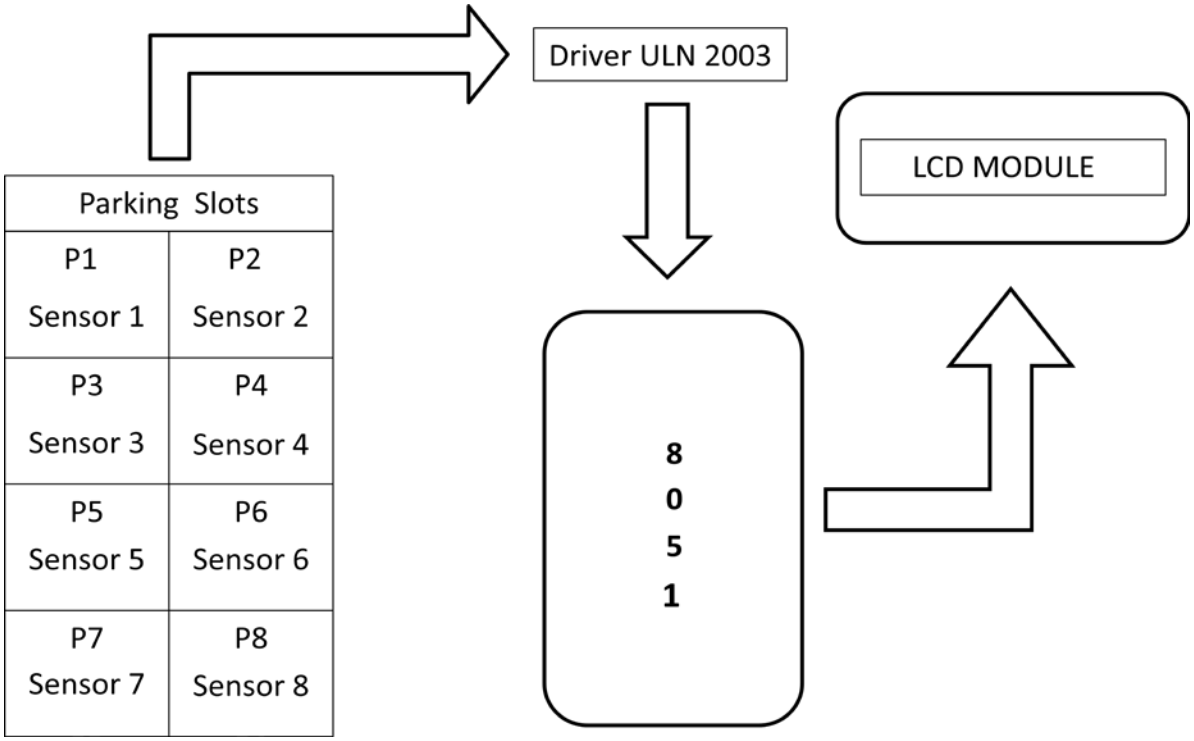


Fig: 3.3.1

Objective:

Now we need to display specifically which parking slots are empty on LCD module so that the user can guide the car to that particular slot.

Configuration:

- LCD module as previously seen has been connected on port P0.
- There are 8 parking slots and every slot has an ultra sonic sensor.
- These sensors have been connected to port P1 using driver ULN2003.

Working:

The ultra sonic sensor in each parking slot detects the presence of the car and continuously sends the information to the port P1 of the micro controller. The microcontroller displays the result on LCD. The empty parking slots are displayed on LCD and if the parking is full then the LCD displays “no space”.

ULN 2003 pin configuration:

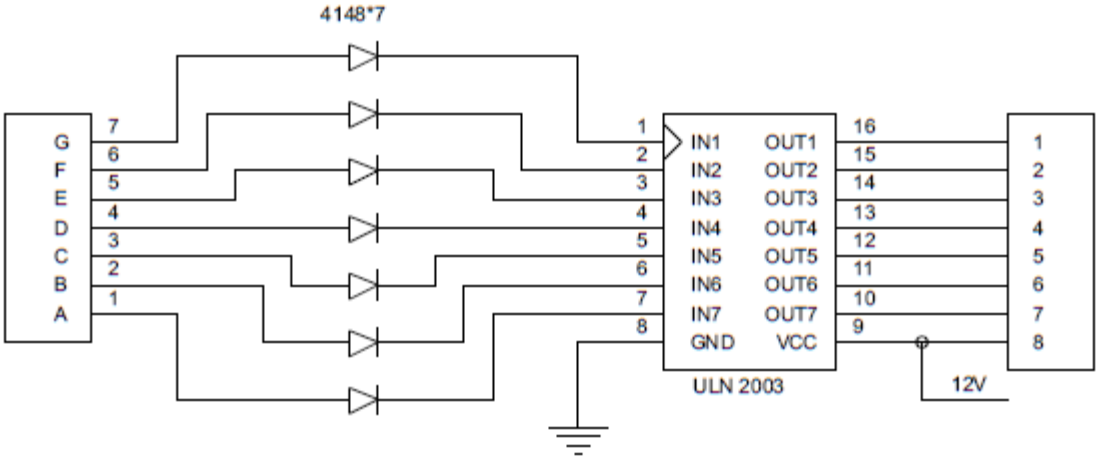
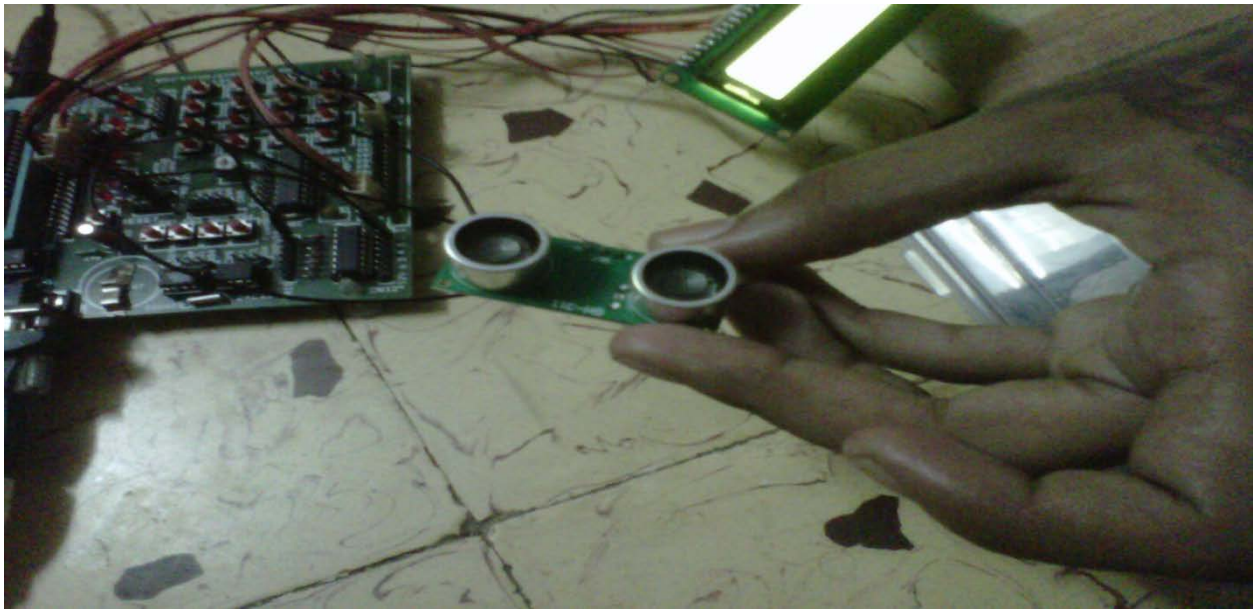


Fig: 3.3.2

Output:

1. Initially on reset system displays the parking slots that are empty and available for parking here slots a, b, c, d, e, f, g and h are empty and so are displayed on LCD.



2. Evert slot has ultrasonic sensor that detects the presence of the car. Now here the slot a has car parked in it and is not available for parking hence not displayed on LCD.

3.4 Monitoring and Security Using Malab

It is most important to monitor and secure car parking area. That's why here we are monitoring and saving all data in some storage with time and date.
Here we are using camera to capture image data and Matlab pc software to process it and storing it in pc or laptop.

Block diagram:

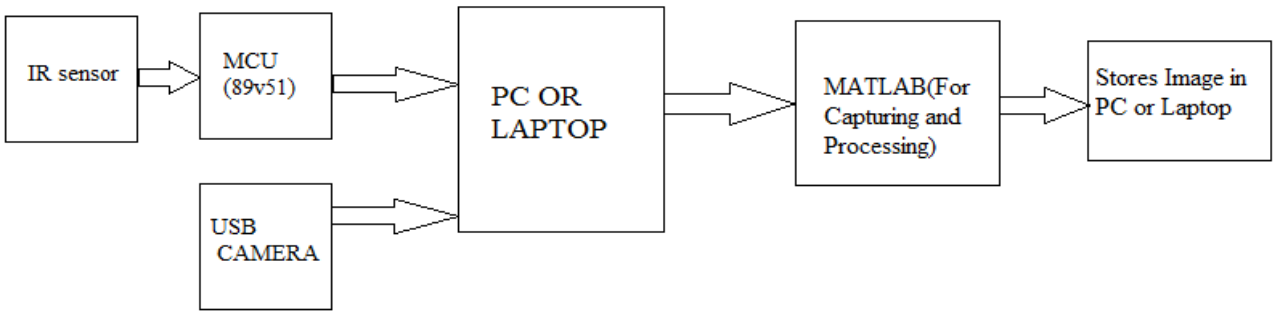


Fig: 3.4.1

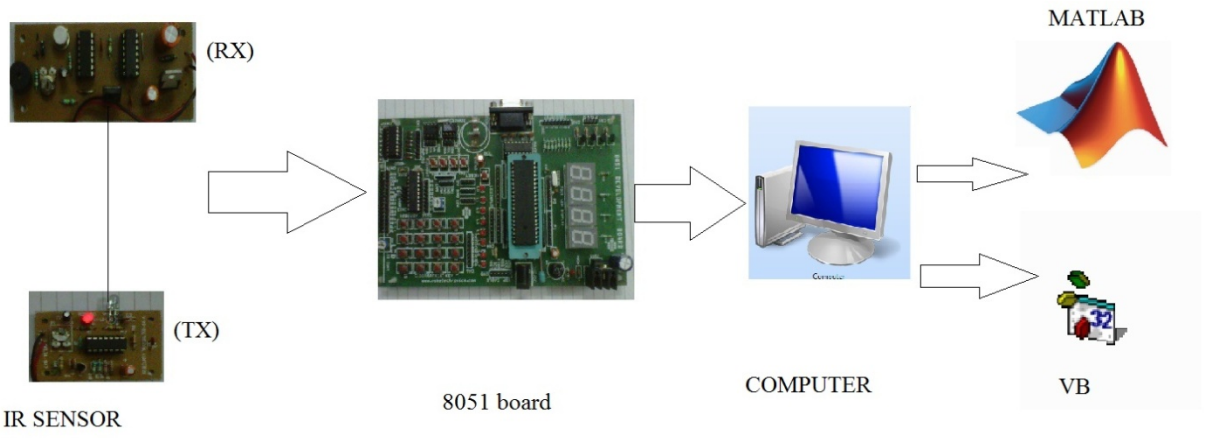
In this block diagram we can see that IR Sensor gives the high pulse from detection of car to 89v51 microcontroller.
Here USB camera takes continuous video frame. Now here MCU and USB camera is processed using PC which uses some software for capturing storing information of image into PC or Laptop.
Here MATLAB software is used for capturing image and processing it and stores that into some allocation area.
Here all information of all cars will store in PC or Laptop with its timing and car number plate. Then we can see whole information when we need.

Benefits of Matlab:

(1) Easy to understand and working
(2) Large set of function
(3) Easy for image processing and capturing
(4) Easy for color recognition

Disadvantages of Matlab:

(1) It is commercial version. We can not use it widely
(2) It is so much costly.
(3) It is slow in com port interfacing.
(4) It is slow in video processing
(5) Not application specific



Monitoring and Security Using Visual Basic

Visual Basic (VB) is a third-generation event-driven programming language and integrated development environment (IDE) from Microsoft for its COM programming.
Due to some demits of Matlab we here decided to use visual basic for image capturing and storing in database of PC or Laptop.

Block diagram:

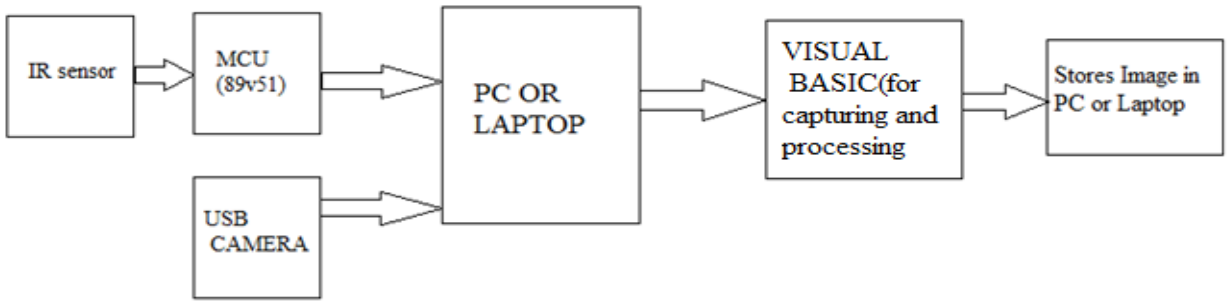


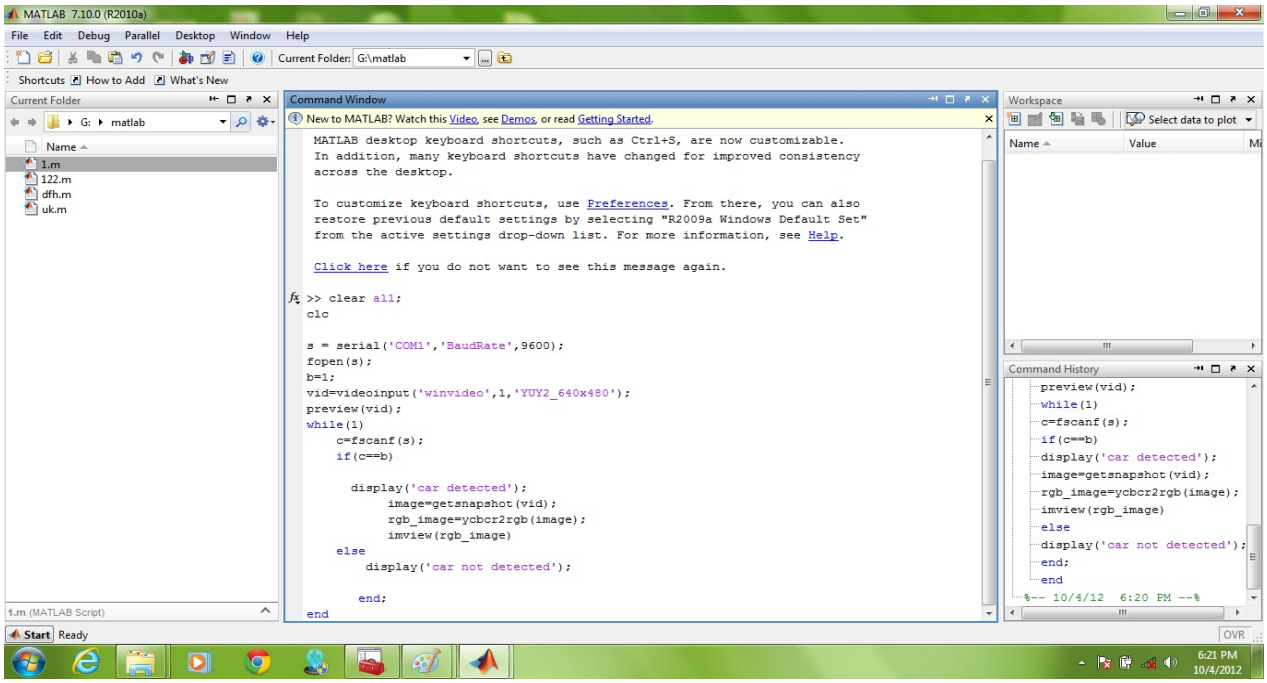
Fig: 3.4.2

Advantages of Visual Basic:

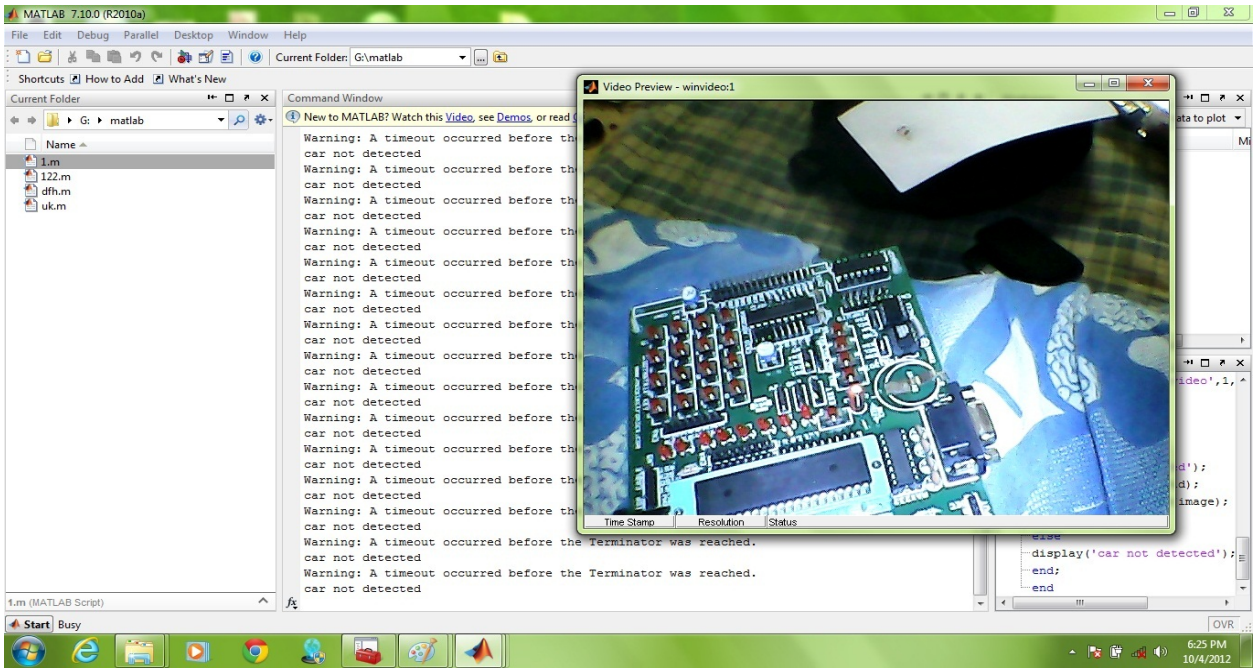
(1) Easy to understand and design
(2) Fast in processing with com port rather than Matlab
(3) Less cost
(4) It is application specific software

Matlab Output:

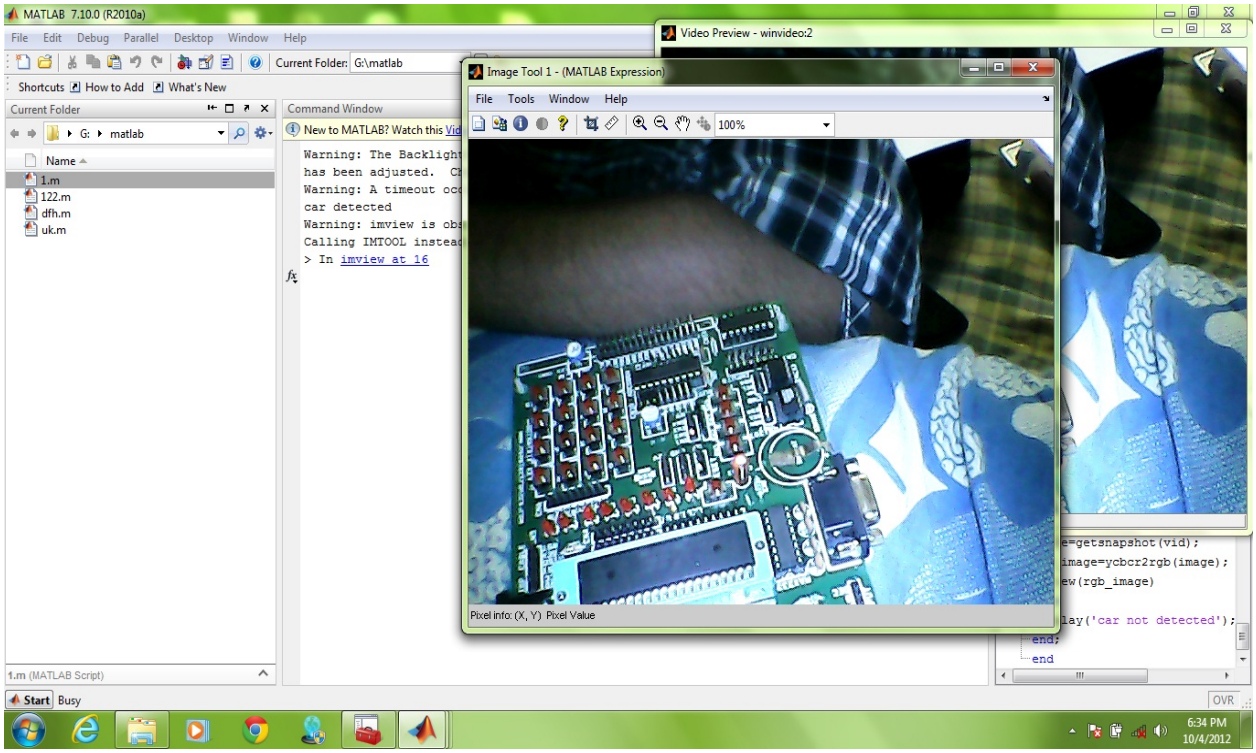
1. Code typed on Matlab window



2. Video output on Matlab window.

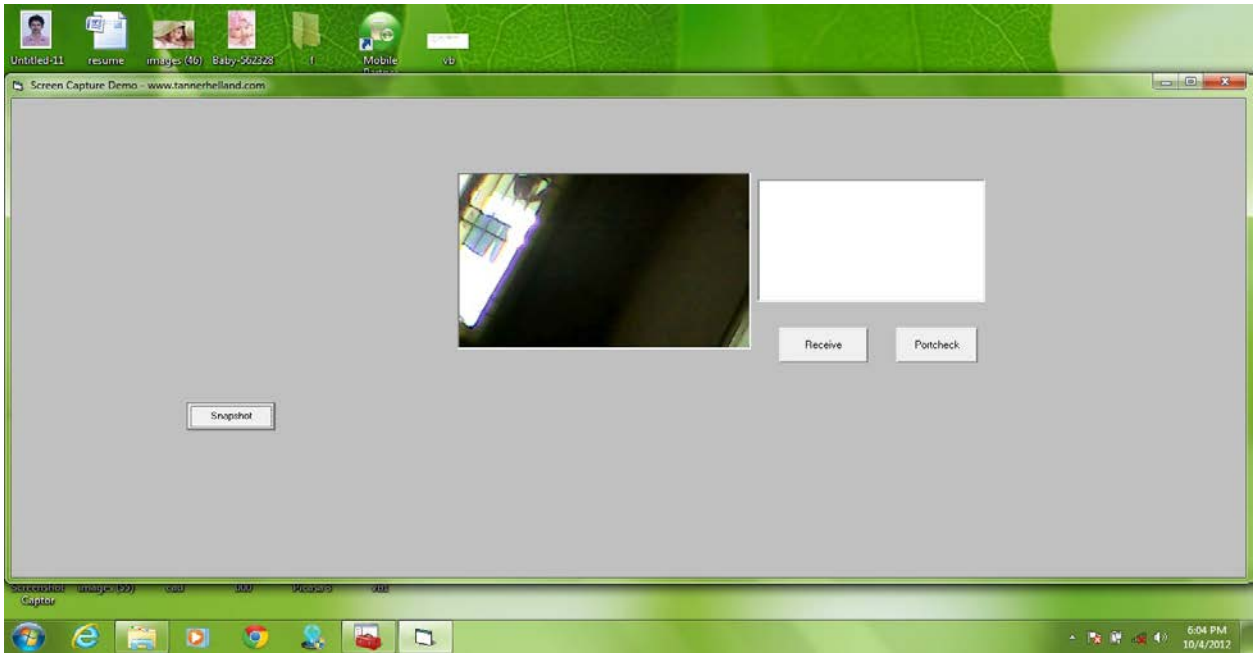


3. Snapshot taken by Matlab on detection of high pulse transmitted serially.

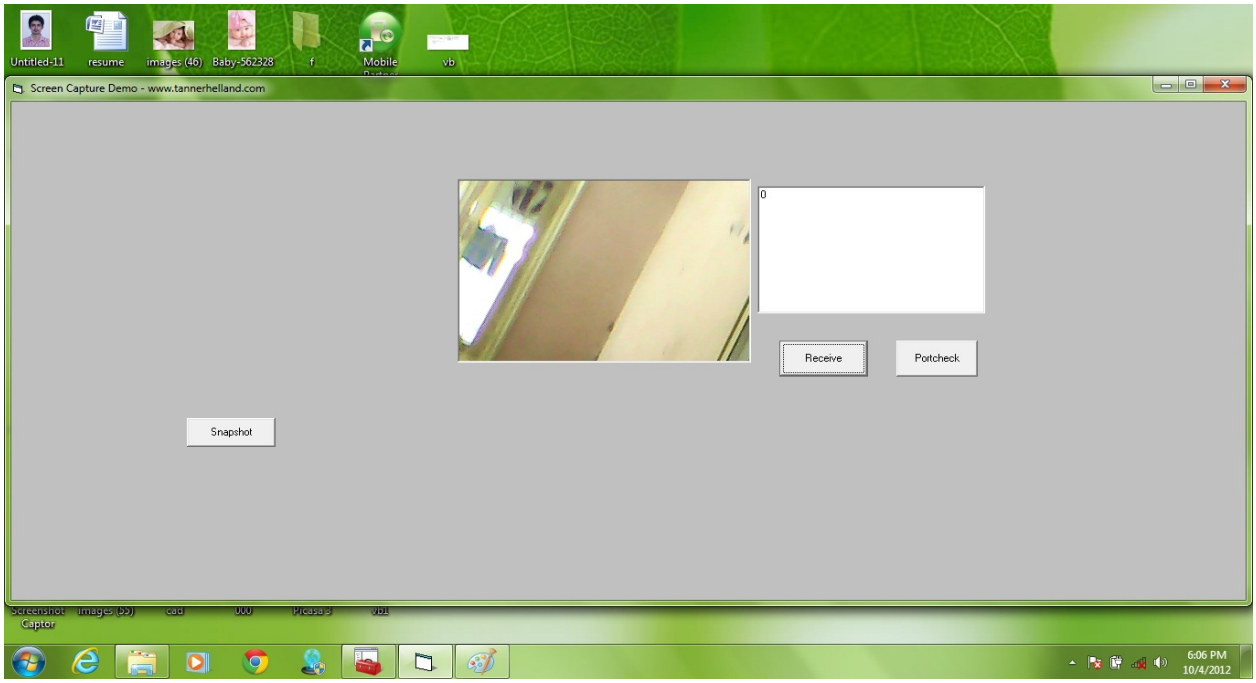


Visual Basic Output:

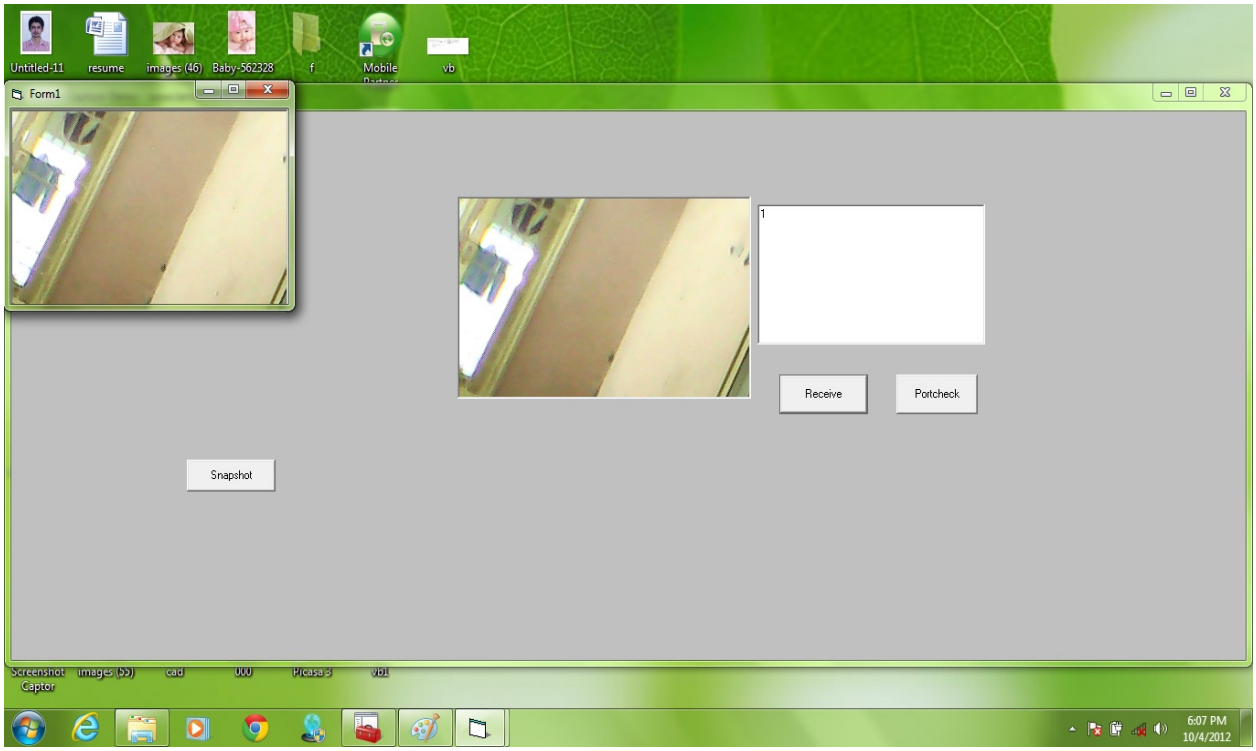
1. Camera started.



2. No high pulse detected yet.



3. High pulse detected and snapshot taken.



Future work

- Making system more compact.
- Implementing multi level parking.
- Interfacing RFID and barrier in this parking system.
- Implementing RTC to show time.
- Using VB coding to take snapshots of cars number plate and saving the image in database.

Conclusion

The above system has been implemented and gives error free results. Out of the different sensors IR sensor and the proximity ultrasonic sensor work well in real time systems under external environmental conditions. The results are displayed on LCD or seven segment. The outputs of the system on real time basis have also been tested and the results noted are error free.

References

1. The 8051 Microcontroller and Embedded Systems Using Assembly and C by Muhammad Ali Mazidi, Janice Gillispie Mazidi and Rolin McKinlay (Second Edition, Pearson Education).

2. The 8051 Microcontroller & Embedded Systems using Assembly and C By K. J. Ayala, D. V. Gadre (Cengage Learning , India Edition).

3. VBA programming by Kogent Learning Solutions Inc. (Dreamtech Press)

Appendix

- Datasheet of 89v51rd2
- Datasheet of max 232
- Datasheet of ULN 2003