AHMEDABAD UNIVERSITY

Embedded System Design

Project Title:

Wirobot

Prepared By:

Ashutosh Kakadiya 1401075

Deep Talati 1401085

Harsh Mehta 1401086

KishanRaval 1401117

Assisted By:

Prof. Anurag Lakhalani

Mentors:

MS. Bhavika patel

MS. Kavita Anjariya

Index:

Introduction……03

Block Diagram……04

Selection Criteria for Micro controller……05

Coasting table……06

Problems were Faced……06

Circuit Diagram…… 07

Snapshots……08

Flowcharts ……09

Code……10

Conclusion ……14

Time Line……...15

References……..16

Introduction:

In this exposed world with the increase in threat and security risks, the need of surveillance and control has also been increased. Which encouraged us to design a robot car which can be used wirelessly in any area having internet connection and can be controlled remotely. This kind of application can be adopted for surveillance at the border area. The same can be used for remotely observing house or office through the robot.

WiRobot (Wireless Robot) is a robot which can be used remotely. The WiRobot gets instructions from the user through internet over Wi-Fi network and acts as per the instruction given by the user. Along with this, WiRobot is also gathering the information about the surroundings and will send them to the user through internet.

Block Diagram:

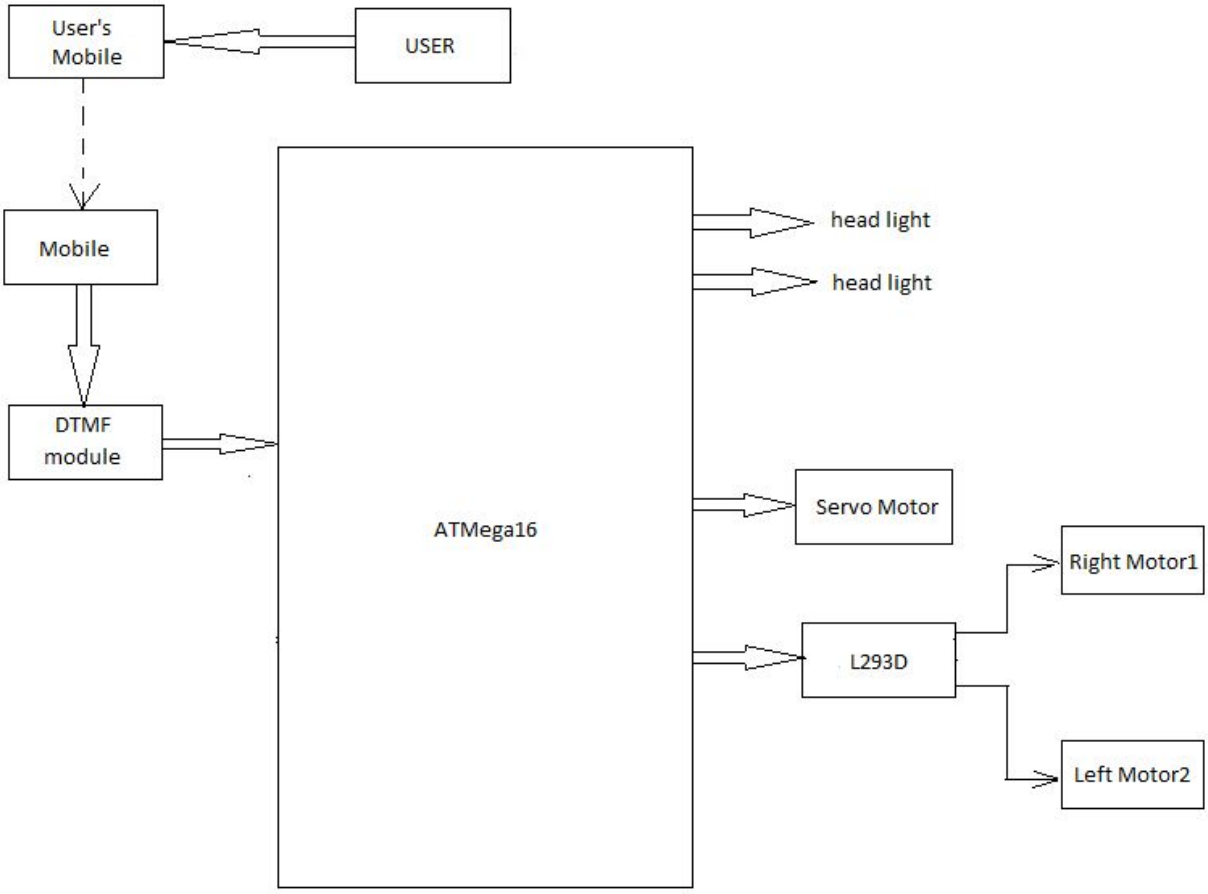


Figure .I Block Diagram

Selection Criteria for major components:

1. **Micro controller:**

Name of uc	**ATmega 328p**	**ATmega 16**	**ATmega 32**
CodeROM	8K	16K	32K
DataRAM	1K	1K	2K
DataEEPROM	0.5K	0.5K	1K
IO pins	32	32	32
ADC	08	08	08
Timer	03	03	03

Costing table:

Component	Quantity	Cost per unit	Total Cost
Development Board	1	500	500
DTMF Module	1	400	400
Programmer	1	530	530
Servo Motor	1	450	450
9V battery	2	15	30
DC Motor	2	125	250
Car Tyre	4	20	80
Connectors	40	3	120
Chases	1	100	100
LED	2	5	10
Motor Driver	1	120	120
		Total	2590

Problems were Faced :

- Data transfer using Wi-Fi module was not working properly.
- DC & Servo motor interfacing .
- In programming of changing angle of servo motor
- Infra – Red sensor configuration with ATMEGA 32
- Data transfer through DTMF module

Circuit Diagram:

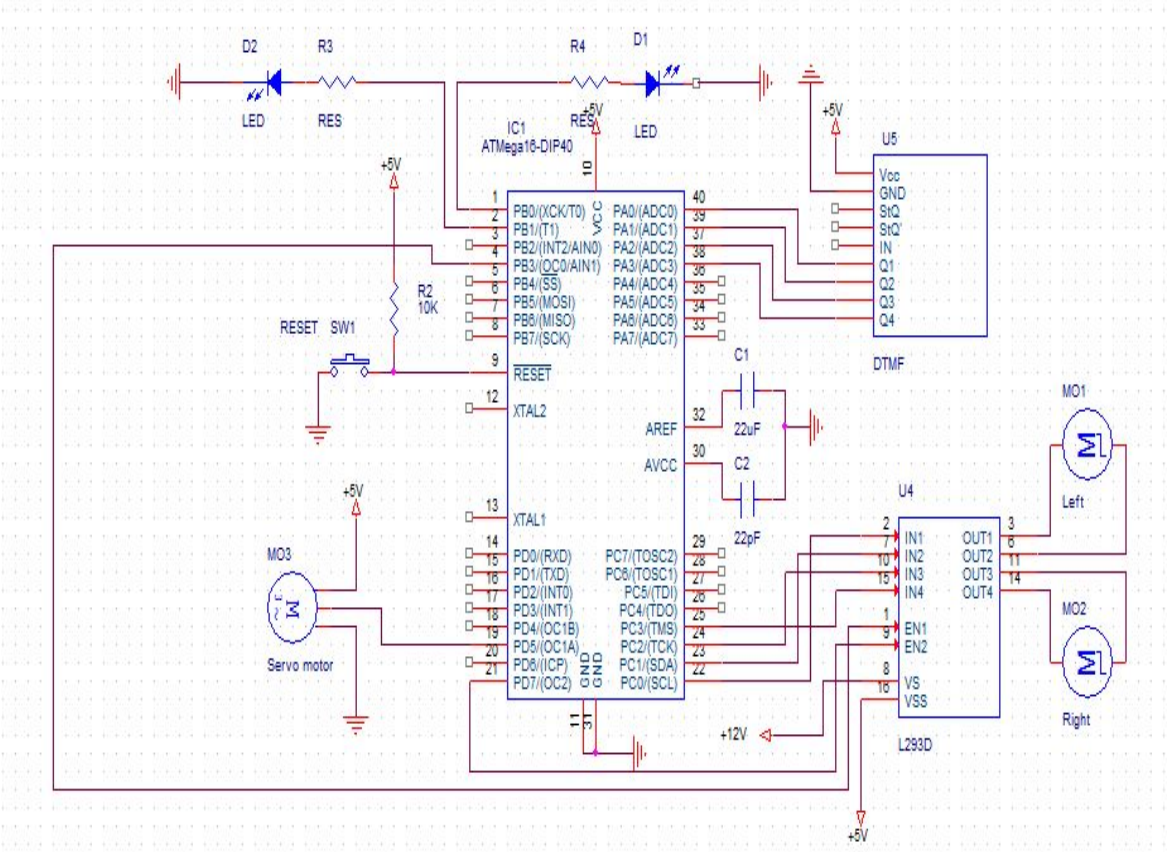
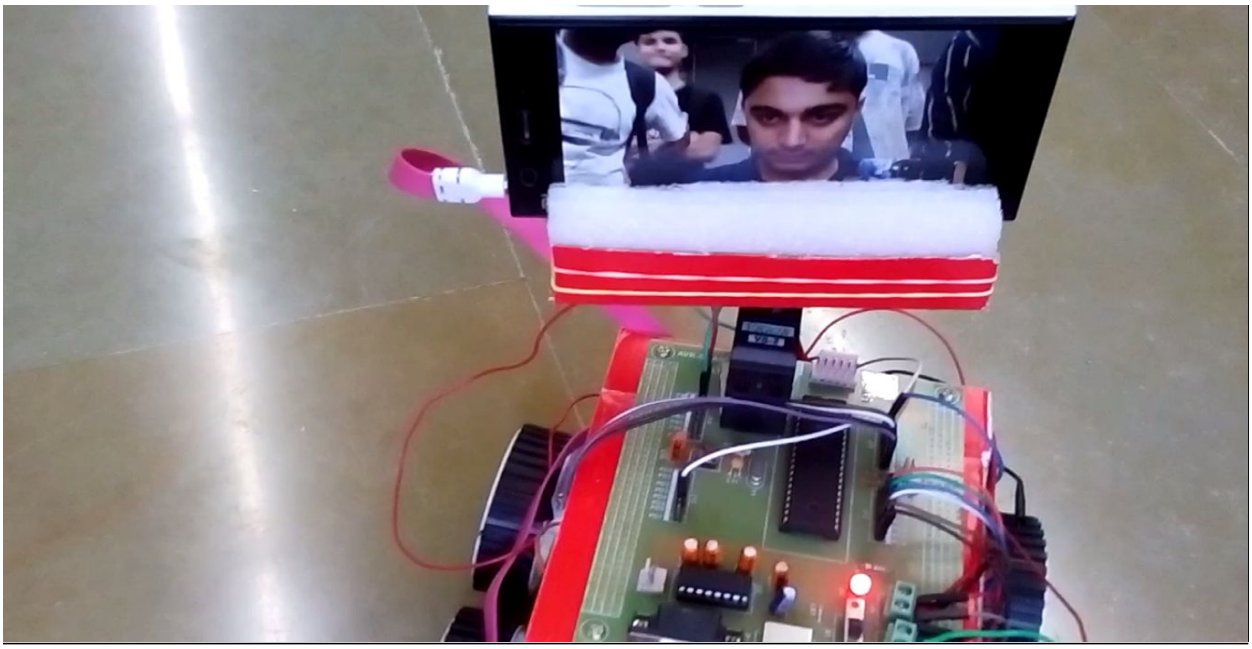
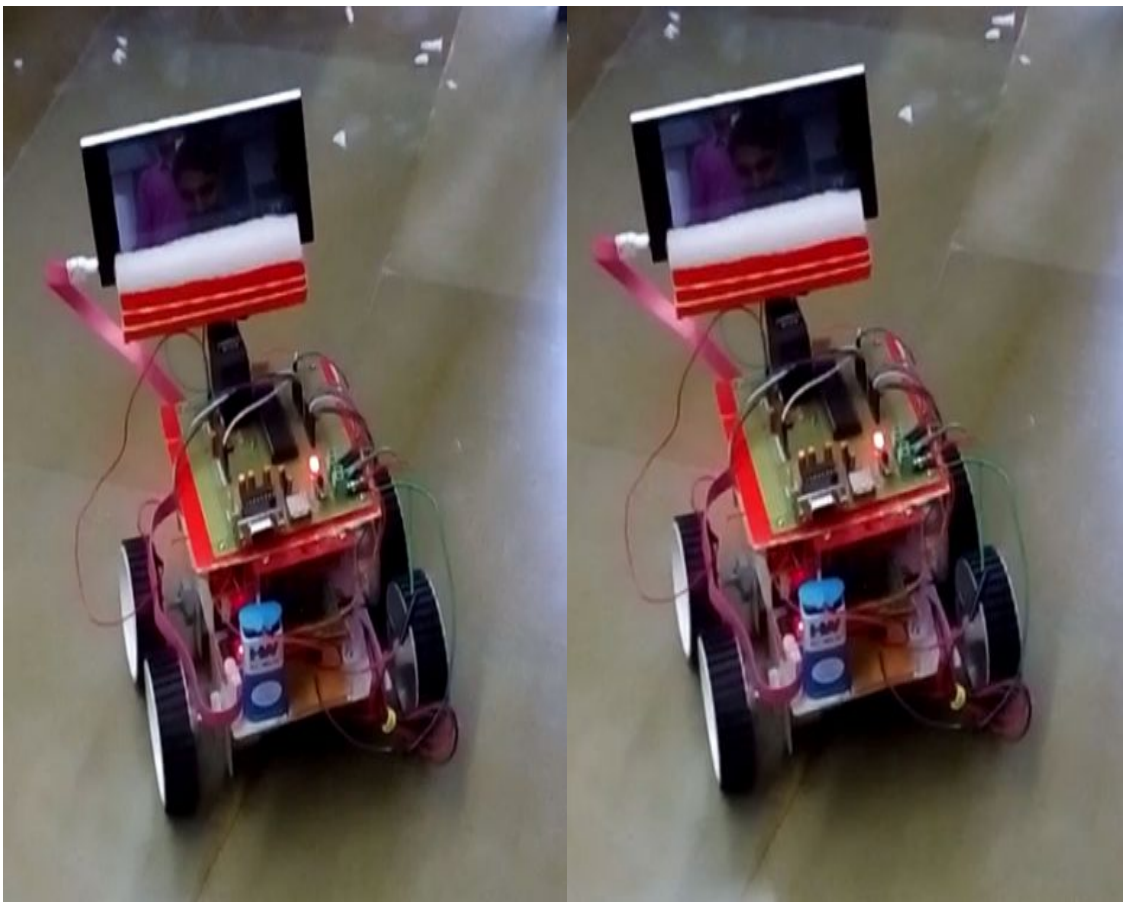


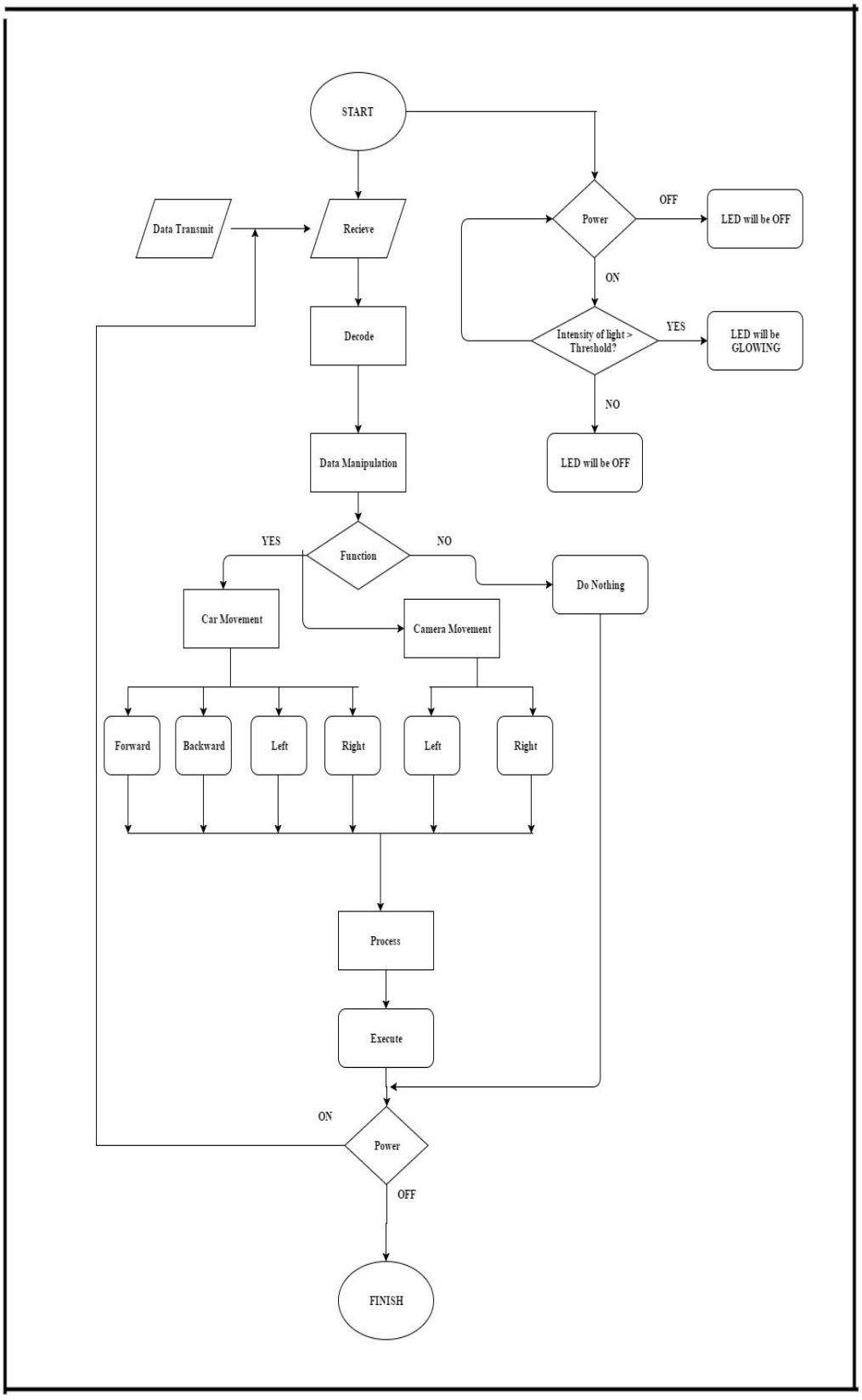
Figure .II Circuit Diagram

Snap Shots of working model:





Flow Chart:


START
Data Transmit
Recieve
Power
OFF
LED will be OFF
ON
Decode
Intensity of light > Threshold?
YES
LED will be GLOWING
NO
Data Manipulation
LED will be OFF
YES
Function
NO
Do Nothing
Car Movement
Camera Movement
Forward
Backward
Left
Right
Left
Right
Process
Execute
ON
Power
OFF
FINISH

Code:

```
/*
 * Wirobot.c
 *
 * Created: 25-04-2016 19:01:53
 * Author : WIROBOT
 */
#define F_CPU 16000000
#include <avr/io.h>
#include <util/delay.h>
#include<avr/interrupt.h>
#define servo_min 0.600       //period in ms
#define servo_max 2.400       //period in ms


float D; //declaring global variable for change the degree of servo motor


void servo(float degree) {
int cmpMatch; // for calculating the compare match value for OCR1A


cmpMatch=  (int)(round( ( (float) (degree * (float) (servo_max - servo_min) /
(float) 180.0) + servo_min) * 125) );


OCR1A= cmpMatch;
}


void changeDegree(unsigned char f)
{
if(f == 0){ // if flag is zero, motor will rotate 20 degree to the left
D = D + 20;
if(D > 180) // max degree is 180
{
D = 180;
}
}
else // if flag is not zero, motor will rotate 20 degree to the right
{
```

```
D = D - 20;
if(D <= 0)
{
D = 0; // minmum degree is 0
}
}
}


ISR (ADC_vect)// org 0x020
{
unsigned char low,high;
int num;
low = ADCL; //Diplay ADCL value @ PORTC
high = ADCH; //Diplay ADCH value @ PORTB


num = high;
num = num << 8;
num = num | low;
num = num & 0X03FF;
num = num * 4882.8;                    //(5/1024*1000) = 4.8828
//for converting from scale 0-1024 to 0-5 with precition of 2, we multiply the
number by 4.8828
if(num < 3300)
{
PORTB = 0XFF;
}
else
{
PORTB = 0X00;
}
ADCSRA |= (1<<ADSC); // After completion of earlier conversion,
start              //new conversion
}


int main(void) {
DDRA = 0x00; // PORT A as input
DDRC = 0xFF; // PORT C as output
DDRB = 0xFF; // PORT B as output
```

```
D = 90; // intial position for servo motor is 90 degree
TCCR1A|=(1<<COM1A1)|(1<<COM1B1)|(1<<WGM11);        //NON Inverted
PWM
TCCR1B|=(1<<WGM13)|(1<<WGM12)|(1<<CS11)|(1<<CS10);
//PRESCALER=64 MODE 14(FAST PWM)
ICR1=4999;  //fPWM=50Hz
DDRD|=(1<<PD4)|(1<<PD5);       //PWM Pins as Output
ADMUX=0xC0;      //Reference Voltage VCC, Left Adjust, ADC0
ADCSRA=0b10001111;     //ADC enable,ADC interrupt enable, PRESCALER
128
sei();
ADCSRA |= (1<<ADSC);  //Start ADC conversion
servo(D); // calling servo function to set servo motor at 90 degree
//PORTC.4 and PORTC.5 is 1 for enable the motor driver
// table for motor direction
// PORTC.0  PORTC.1  PORTC.2  PORTC.3
// 0       0       0       0        stop
// 1       0       1       0        forward direction
// 0       1       0       1        backward direction
// 0       0       1       0        right turn
// 1       0       0       0        left turn
// table for user
// PORTA.0  PORTA.1  PORTA.2  PORTA.3
// 0       0       1       0        for forward direction
// 1       0       0       0        for backward direction
// 0       1       1       0        for right turn
// 0       1       0       0        for left turn
// 0       0       0       0        to stop
// 0       0       0       1        to rotate motor in left direction
// 0       0       1       1        to rotate motor in right direction
while(1)
{
PORTA =  PINA & 0xF0; // considering 4 MSB as input


if(PORTA == 0x20) // if 2 is send by the user, car will run in forward direction
{
PORTC = 0x3A;
}
else if(PORTA == 0x80) // if 8 is sent by the user, car will run in backward
direction
```

```
{
PORTC = 0x35;
}
else if(PORTA == 0x60) // if 6 is sent by the user, car will take right turn
{
PORTC = 0x32;
}
else if(PORTA == 0x40) // if 4 is sent by the user, car will take left turn
{
PORTC = 0x38;
}
else if(PORTA == 0xA0) // if 0 is sent by the user, car will stop
{
PORTC = 0x00;
}
else if(PORTA == 0x10) // if 1 is sent by the user, servo motor will rotate in left
direction
{
changeDegree(0);
_delay_ms(1000);
servo(D);
}
else if(PORTA == 0x30) // if 3 is sent by the user, servo motor will rotate in right
direction.
{
changeDegree(1);
_delay_ms(1000);
servo(D);
}
}
}
```

Conclusion:

Motivation behind the project was to have a machine or robot which can do surveillance from a remote place and sends the data to the user. To control the robot, first we used WiFi Module (ESP8266), but because of some problems, we shifted to DTMF module which serves the same purpose. Also, headlights of the robot goes on whenever it detects low brightness around using LDR sensor.

The robot was controlled through Skype and surveillance is captured through google hangouts.

We learned to use micro-controller efficiently, tackled some of voltage and power supply problem, using serial transmission, remote controlling through DTMF, to use sensors, to work eff and many miscellaneous things. Overall, it was great experience working with our peers and playing with micro-controllers

Timeline:

	23/03/16	04/04/16	11/04/16	18/04/16	25/04/16
Assignment 1	✓				
Gathering all the components and basic connection		✓			
Final submit of Circuit Diagram			✓		
Testing with Static code			✓		
Assembling Components(Make Robot)			✓		
Program Complete				✓	
Testing Demo					✓

References:

[1]Muhammad Ali Mazidi and Sarmad Naimi and Sepehr Naimi, ”ADC and Sensor interfacing” in the avr microcontroller and embedded system, pp 464-483.

[2]http://www.ablab.in/ll293d-driver-interfacing-with-avr-atmega32-microcontroller/