Islamic University of Technology (IUT)
Organization of Islamic Cooperation (OIC)
Department of Electrical and Electronic Engineering (EEE)

Course : EEE 4606 (Microcontroller Based System Design Lab)

Project No. : 06 (Individual Project)
Project Name: Bluetooth Interfacing.

Objective:
1. Familiarization with the AT89C52 Microcontroller.
2. Familiarization with the HC-05 Bluetooth Module.
3. Familiarization with the DOT MATRIX Display.

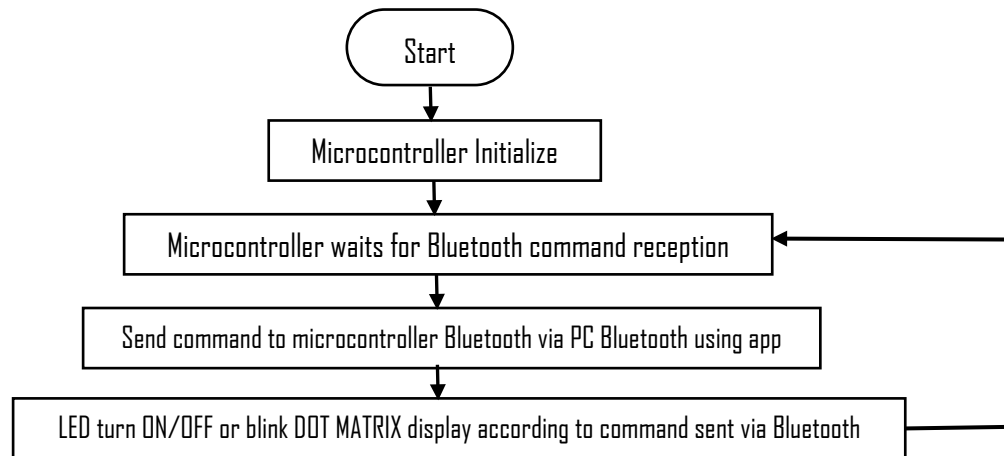|		
Name	:	Md Mohi Uddin Khan
Student ID	:	160021163
Section	:	C1
Course Name	:	Microcontroller Based System Design Lab
Course No.	:	EEE 4606

OPERATION

- Upon selection of 'ON' knob of the sender app, PC Bluetooth will connect with Microcontroller Bluetooth.
- Switching control via that app will send command to microcontroller via Bluetooth to control LED & DOT MATRIX display.
- Total LED that are ON/OFF will be shown on the app panel via colored circle & Slider.
- Upon selection of 'Reset' knob of the sender app, all LED's will be turned off but Bluetooth will remain connected.
- Upon selection of 'OFF' knob of the sender app, all LED's will be turned off & Bluetooth will be disconnected.

ALGORITHM

The program is designed according to the following Flow Chart :

```
                    ┌─────────────┐
                    │    Start    │
                    └──────┬──────┘
                           │
                           ▼
              ┌────────────────────────────┐
              │  Microcontroller Initialize │
              └────────────┬───────────────┘
                           │
                           ▼
    ┌──────────────────────────────────────────────┐
    │ Microcontroller waits for Bluetooth command   │◄───┐
    │ reception                                      │    │
    └────────────────────┬─────────────────────────┘    │
                         │                                │
                         ▼                                │
    ┌──────────────────────────────────────────────────┐ │
    │ Send command to microcontroller Bluetooth via PC  │ │
    │ Bluetooth using app                               │ │
    └────────────────────┬─────────────────────────────┘ │
                         │                                │
                         ▼                                │
    ┌───────────────────────────────────────────────────────┐
    │ LED turn ON/OFF or blink DOT MATRIX display according  │─┘
    │ to command sent via Bluetooth                          │
    └───────────────────────────────────────────────────────┘
```

Components Used

- AT89C52 microcontroller
- Resistors
- 8 LEDs
- 8x8 DOT MATRIX Display

Sender App Designed using MATLAB Appdesigner
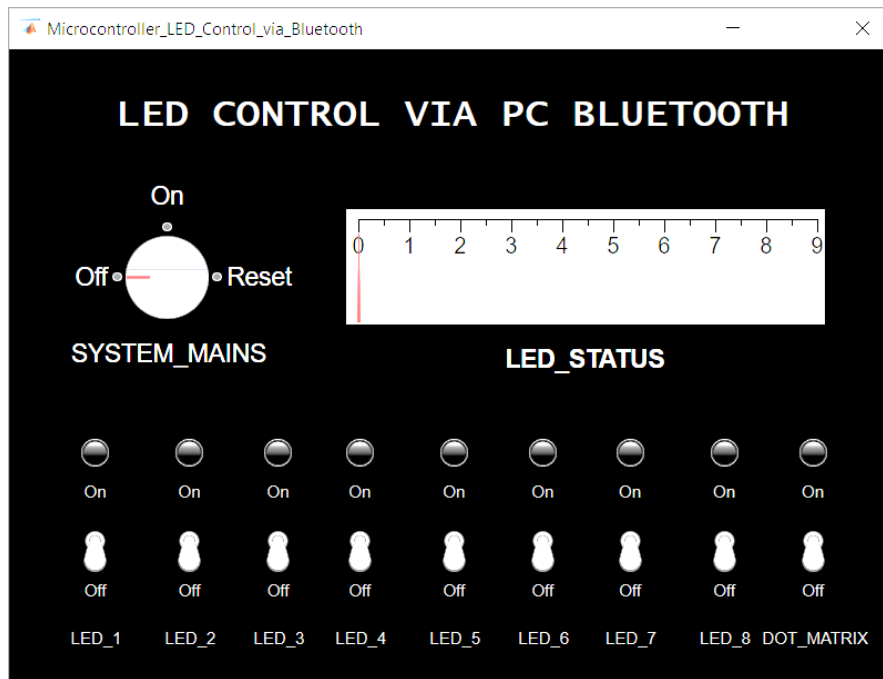
Circuit Diagram in PROTEUS
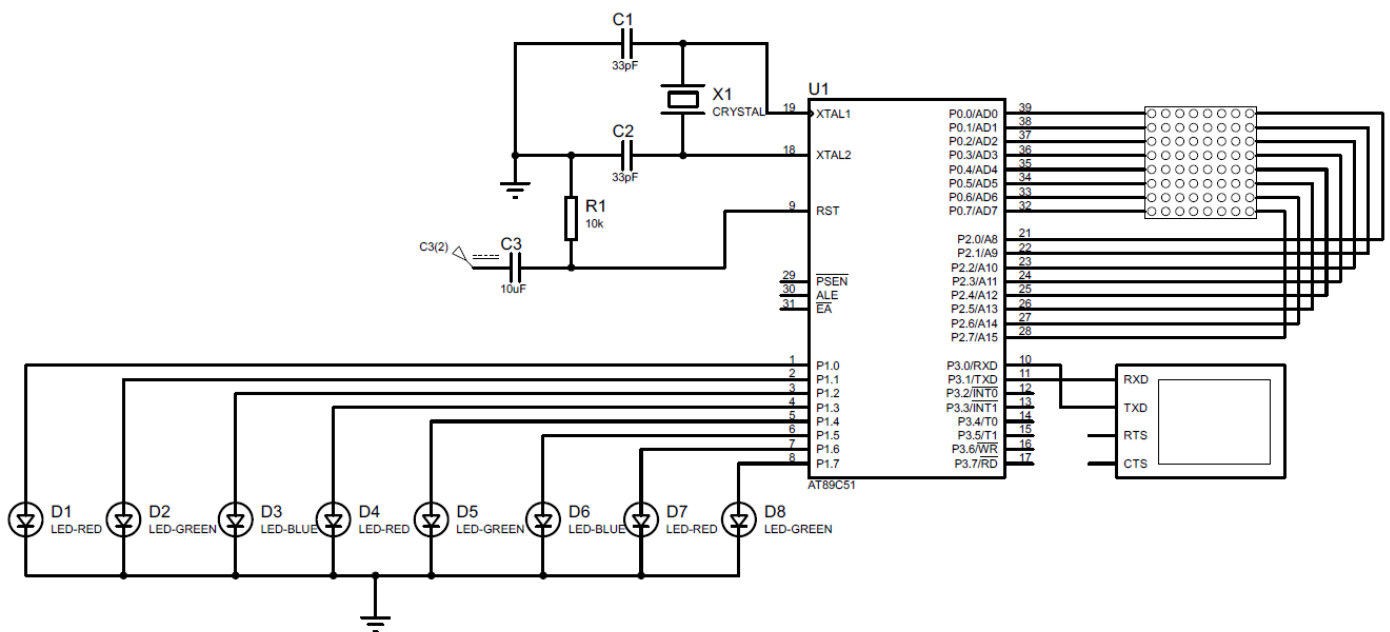
Sender App Design MATLAB CODE

```matlab
% Value changed function: SYSTEM_MAINSKnob
function SYSTEM_MAINSKnobValueChanged(app, event)
    global bt  on_flag
    value = app.SYSTEM_MAINSKnob.Value;
    if isequal(value,'On')
        if isempty(on_flag)
            instrhwinfo('Bluetooth')
            ans.RemoteNames
            instrhwinfo('Bluetooth','HC-05')
            bt = Bluetooth('HC-05', 1)
            fopen(bt)                  % Connects PC bluetooth with HC-05 bluetooth
            on_flag = 1;
        end
    elseif isequal(value,'Reset')
        fwrite(bt,0);
        app.LED_STATUSGauge.Value = 0;
        app.LED_XSwitch.Value = 'Off';   % here, X = 1,2,3,4,5,6,7,8
        app.DOT_MATRIXSwitch.Value = 'Off';
        app.LXLamp.Color = 'k';          % here, X = 1,2,3,4,5,6,7,8
        app.DLamp.Color = 'k';
    elseif  isequal(value,'Off')
        fwrite(bt,0);
        app.LED_STATUSGauge.Value = 0;
        app.LED_XSwitch.Value = 'Off';   % here, X = 1,2,3,4,5,6,7,8
        app.DOT_MATRIXSwitch.Value = 'Off';
        app.LXLamp.Color = 'k';          % here, X = 1,2,3,4,5,6,7,8
        app.DLamp.Color = 'k';
        fclose(bt);                 % Disconnects bluetooth
        evalin('base','clear all');
    end end
% Value changed function: LED_XSwitch            % here, X = 1,2,3,4,5,6,7,8
function LED_XSwitchValueChanged(app, event)      % here, X = 1,2,3,4,5,6,7,8
    global bt
    if isequal(app.SYSTEM_MAINSKnob.Value,'On')
        value = app.LED_XSwitch.Value;            % here, X = 1,2,3,4,5,6,7,8
        if isequal(value,'On')
            app.LXLamp.Color = 'g';               % here, X = 1,2,3,4,5,6,7,8
            app.LED_STATUSGauge.Value = app.LED_STATUSGauge.Value + 1;
            fwrite(bt,X)              % Value sending to HC-05 from PC bletooth. here, X = 1,2,3,4,5,6,7,8
        else
            app.LXLamp.Color = 'k';               % here, X = 1,2,3,4,5,6,7,8
            app.LED_STATUSGauge.Value = app.LED_STATUSGauge.Value - 1;
            fwrite(bt,X)              % Value sending to HC-05 from PC bletooth. here, X = 1,2,3,4,5,6,7,8
        end end  end
% Value changed function: DOT_MATRIXSwitch
function DOT_MATRIXSwitchValueChanged(app, event)
    global bt
    if isequal(app.SYSTEM_MAINSKnob.Value,'On')
        value = app.DOT_MATRIXSwitch.Value;
        if isequal(value,'On')
            app.DLamp.Color = 'g';
            app.LED_STATUSGauge.Value = app.LED_STATUSGauge.Value + 1;
            fwrite(bt,9)             % Value sending to HC-05 from PC bletooth
            pause(20)
            app.DLamp.Color = 'k';
            app.DOT_MATRIXSwitch.Value = 'Off';
            app.LED_STATUSGauge.Value = app.LED_STATUSGauge.Value - 1;
        end end end
```

Microcontroller, Bluetooth & LED Control ASSEMBLY CODE

```
ORG 00H

bluetooth:
MOV TMOD,#20H
;TIMER 1 MODE 2   8 BIT MODE
MOV TH1,#-3
;9600 BAUD RATE
MOV SCON,#50H
;SERIAL MODE 1; 10 BIT TOTAL [1
START BIT + 8 DATA BIT + 1 STOP
BIT]]
CLR TI
; MAKE TI REGISTER ZERO
CLR RI
; MAKE RI REGISTER ZERO
SETB TR1; START TIMER-1

MOV P1,#000H

AGAIN:
REPEAT: JNB RI,REPEAT
        MOV A,SBUF
; MOVE CONTENT OF SBUF TO
ACCUMULATOR

        CJNE A,#1,CHECKNEXT1
        CPL P1.0
        CLR RI
        CLR TI
        LJMP AGAIN

CHECKNEXT1:
        CJNE A,#2,CHECKNEXT2
        CPL P1.1
        CLR RI
        CLR TI
        LJMP AGAIN

CHECKNEXT2:
        CJNE A,#3,CHECKNEXT3
        CPL P1.2
        CLR RI
        CLR TI
        LJMP AGAIN

CHECKNEXT3:
        CJNE A,#4,CHECKNEXT4
        CPL P1.3
        CLR RI
        CLR TI
        LJMP AGAIN

CHECKNEXT4:
        CJNE A,#5,CHECKNEXT5
        CPL P1.4
        CLR RI
        CLR TI
        LJMP AGAIN

CHECKNEXT5:
        CJNE A,#6,CHECKNEXT6
        CPL P1.5
        CLR RI
        CLR TI
        LJMP AGAIN

CHECKNEXT6:
        CJNE A,#7,CHECKNEXT7
        CPL P1.6
        CLR RI
        CLR TI
        LJMP AGAIN
```

```
CHECKNEXT7:
        CJNE A,#8,CHECKNEXT8
        CPL P1.7
        CLR RI
        CLR TI
        LJMP AGAIN

CHECKNEXT8:
        CJNE A,#9,CHECKNEXT9
        MOV P0,#0000H;connected to
DOT_MATRIX DISPLAY
        MOV P2,#0000H;connected to
DOT_MATRIX DISPLAY
        MOV R5,#00000000B
        MOV R6,#11111111B
        MOV 50H,#20D    ;COUNTER
FOR BLINKING 20 TIMES
        DOT_MATRIX:  LCALL
DOT_MATRIX_BLINK
        DJNZ 50H,DOT_MATRIX
        CLR RI
        CLR TI
        LJMP AGAIN

CHECKNEXT9:
        CJNE A,#0,GO_BACK
        ;RESET ALL
        MOV P1,#00H
        CLR RI
        CLR TI
        LJMP bluetooth

GO_BACK: LJMP AGAIN

DOT_MATRIX_BLINK:
;GLOW
CLR P0.0
MOV P2,R6
ACALL DELAY1

;DIM
SETB P0.0
MOV P2,R5
ACALL DELAY1

;GLOW
CLR P0.1
MOV P2,R6
ACALL DELAY1

;DIM
SETB P0.1
MOV P2,R5
ACALL DELAY1

;GLOW
CLR P0.2
MOV P2,R6
ACALL DELAY1

;DIM
SETB P0.2
MOV P2,R5
ACALL DELAY1

;GLOW
CLR P0.3
MOV P2,R6
ACALL DELAY1
```

```
;DIM
SETB P0.3
MOV P2,R5
ACALL DELAY1

;GLOW
CLR P0.4
MOV P2,R6
ACALL DELAY1

;DIM
SETB P0.4
MOV P2,R5
ACALL DELAY1

;GLOW
CLR P0.5
MOV P2,R6
ACALL DELAY1

;DIM
SETB P0.5
MOV P2,R5
ACALL DELAY1

;GLOW
CLR P0.6
MOV P2,R6
ACALL DELAY1

;DIM
SETB P0.6
MOV P2,R5
ACALL DELAY1

;GLOW
CLR P0.7
MOV P2,R6
ACALL DELAY1

;DIM
SETB P0.7
MOV P2,R5
ACALL DELAY1

RET

DELAY:   MOV R3,#50
AGAIN_2: MOV R4,#255
AGAIN_3:   DJNZ R4,AGAIN_3
        DJNZ R3,AGAIN_2
        RET

DELAY1:
              MOV R0,#3
        L1:   MOV R1,#100
        L2:   MOV R2,#100
        L3:   DJNZ R2,L3
              DJNZ R1,L2
              DJNZ R0,L1
              RET

        END
```