



DIGITAL LOGIC DESIGN LABORATORY REPORT

Project : Basketball Bluetooth Scoreboard

Description: Design and implement the scoreboard used in sport (Basketball) from Arduino, Android Smartphone, LCD Display and MIT App Inventor.



Name: Pham Le Trung
ID: ITITIU14102

Date: December 12th, 2016

1. Device and operations

a) Abstract

- Arduino UNO R3.
- LCD 16x2.
- I2C Module for LCD.
- Bluetooth Module HC-05.
- Breadboard, Jumperwires, USB-cable .

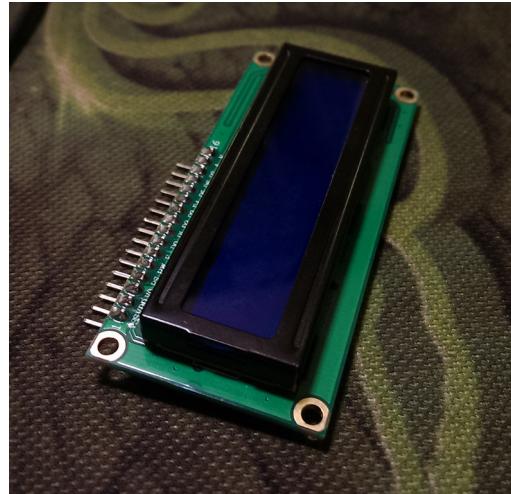
b) Detail

Arduino UNO R3:



- Arduino/Genuino Uno is a microcontroller board based on the ATmega328P.
- It has:
 - + 14 digital input/output pins (6 of them can be used as PWM outputs)
 - + 6 analog inputs
 - + 16 MHz quartz crystal
 - + USB connection
 - + Power jack
 - + ICSP header
 - + Reset button.

LCD 16x02

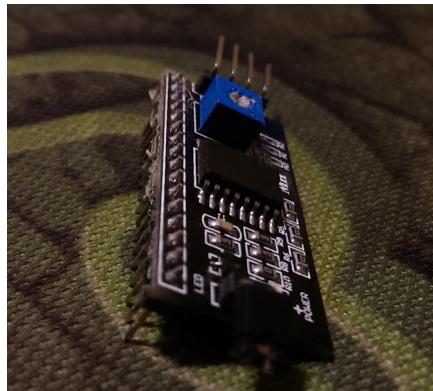


- LCD 16x02 is developed for Arduino compatible boards. It includes a 16x2 white character blue backlight LCD. Pin List:

Pin No.	Function	Name
1	Ground (0V)	Ground
2	Supply voltage; 5V (4.7V – 5.3V)	Vcc
3	Contrast adjustment; through a variable resistor	V _{EE}
4	Selects command register when low; and data register when high	Register Select
5	Low to write to the register; High to read from the register	Read/write
6	Sends data to data pins when a high to low pulse is given	Enable
7	8-bit data pins	DB0
8		DB1
9		DB2
10		DB3
11		DB4
12		DB5
13		DB6
14		DB7
15	Backlight V _{CC} (5V)	Led+
16	Backlight Ground (0V)	Led-

Note: In the project, we will connect the LCD with the I2C module to minimize the ports used when we connect them together with the Arduino.

I2C Module for LCD



- There are 16 ports in I2C module to match exactly with the numbers of ports in LCD. The main purpose of the I2C is to minimize the ports used in LCD from 16 to 4 apparently. Including :

- + **VCC** : connect with the power 5V .
- + **GND** : connect with the GND (ground) in Arduino UNO board.
- + **SDA** : connect to A4 (of the Arduino).
- + **SCL** : connect to A5 (of the Arduino).

Bluetooth HC-05

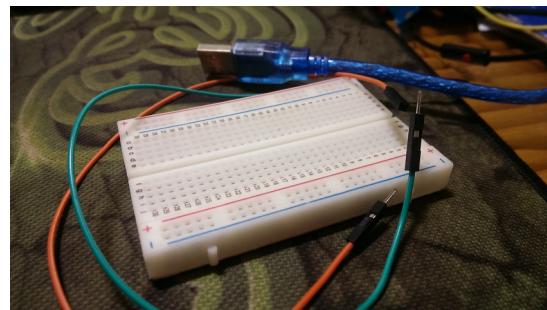


- A special device to transfer data from Android smartphone to Arduino and via Bluetooth connection.

-We use the total of 4 ports of the device:

- +**VCC** : connect to the 3.3V.
- +**GND** : connect to the GND of the Arduino.
- +**TX** : connect to Digital Pin 0 (of the Arduino, → RX).
- +**RX** : connect to Digital Pin 1 (of the Arduino, TX →).

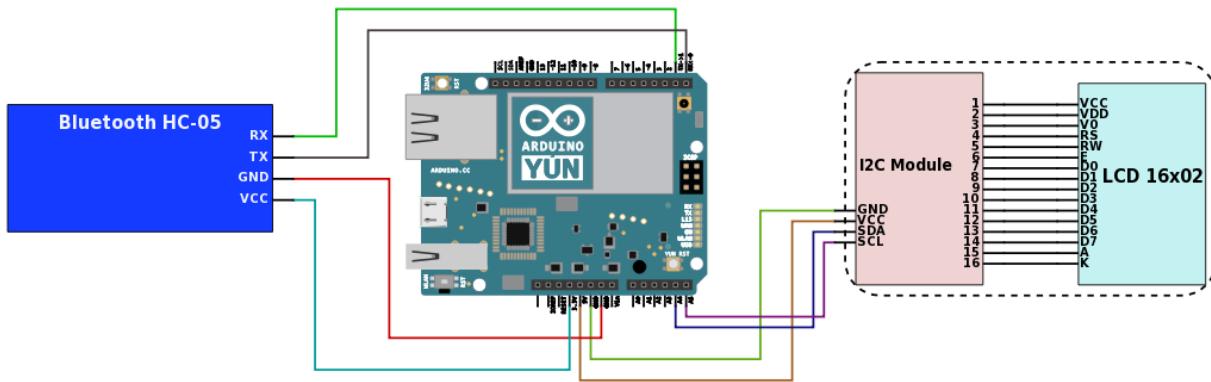
Breadboard, Jump wires, USB cable



Android Smartphone

-A smartphone with the “**SporTrig**” application installed and connect to Arduino via Bluetooth.

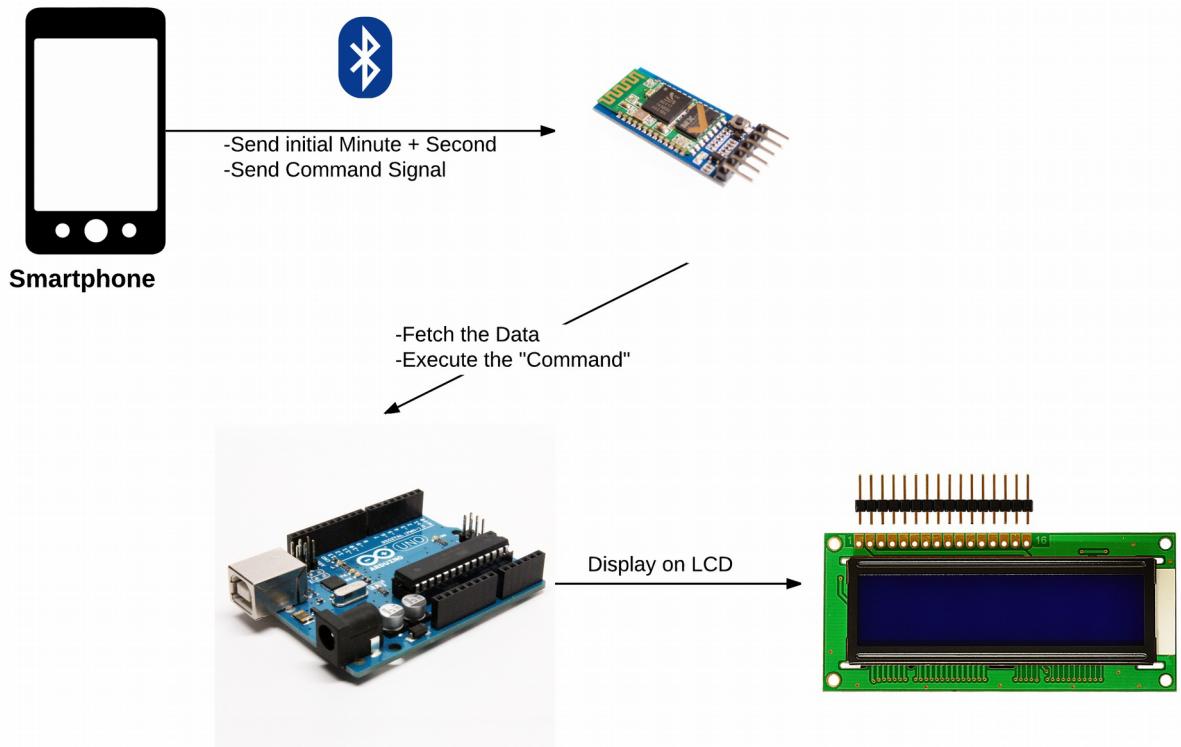
2.Schematic



*Note:

- +Bluetooth HC-05 Connect to 3.3V
- +I2C with LCD Connect to 5V

3. Function explanation



Description : There will be THREE stages for the program to be launched:

-1st Stage : Input the Minute from the Smartphone

+The Smartphone will send directly the initial Minute to the Arduino and will be displayed on the LCD.

-2nd Stage: Input the Second from the Smartphone

+The Smartphone will send directly the initial Second to the Arduino and will be displayed on the LCD.

-3rd Stage: The game Begin

+The Smartphone will send “Command Signal” (from 1 to 6) as “Start, PlusA, MinusA, PlusB, MinusB, Pause” in the Arduino.

+The Arduino then proceed the function as it receive the signal from the smartphone.

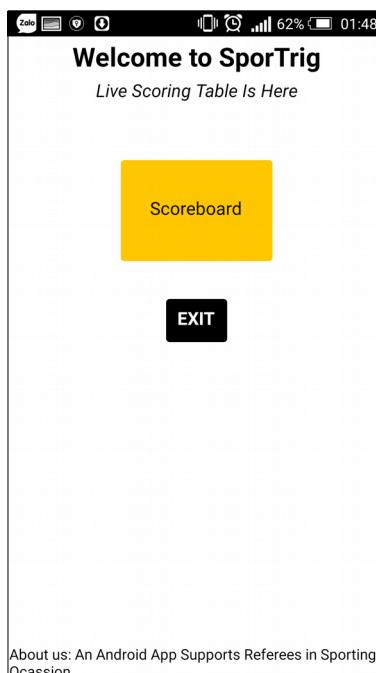
+Display the form of information on LCD.

4. Implementation

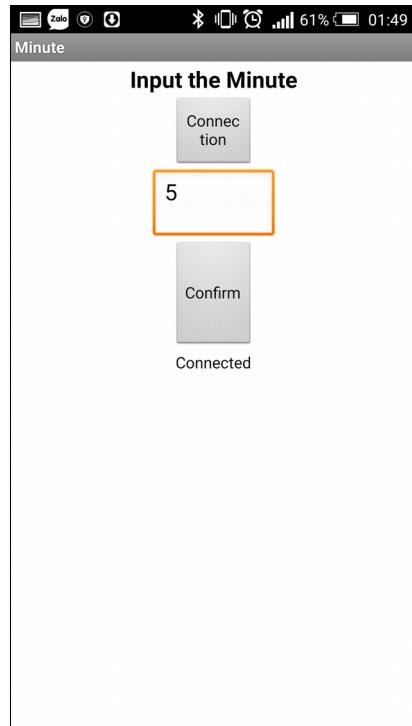
Step 1 : Connect the Arduino + I2C + LCD + Bluetooth Module as Schematic and Upload the Code from IDE.



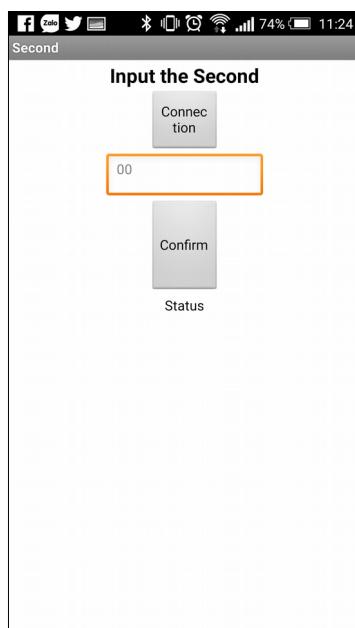
Step 2 : Download “SporTrig” Application to your smartphone and turn on the Bluetooth.



Step 3 : Choose Pair Device (HC-05) from your smartphone, input initial Minute and press “Confirm”



Step 4 : Choose Pair Device (HC-05) from your smartphone, input initial Second and press “Confirm”



By the end of Step 4, you should see the initial Time on the LCD.

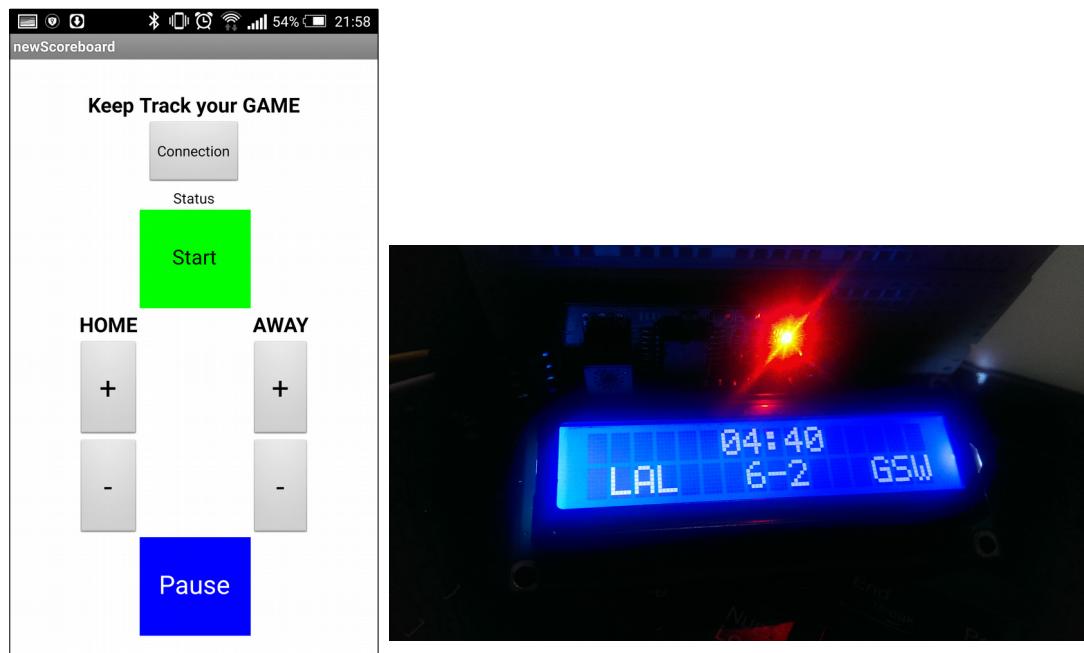


Step 5 : Choose Pair Device (HC-05) from your smartphone, press "Start" to begin the game.

Result : The clock start counts down.

-During the game, you can press "+" or "-" from both side to add or subtract point of two teams.

-Or you can press "Pause" to temporarily stop the game, press "Start" to resume (continue) the game.



Step 6 : When the clock finishes counting down, the game is finished.

Function on Arduino :

1/Initialize lcd

```
LiquidCrystal_I2C lcd(0x3f, 2, 1, 0, 4, 5, 6, 7); // Set the LCD I2C address
```

2/Set teamname on LCD

```
13 //-----SET TeamName -----
14 String teamA_name = "LAL";
15 String teamB_name = "GSW";
16
```

3/Declare Variables used

```
int i = 0;

int minute_1;
int minute_2;
int minute;

int second_1;
int second_2;
int second;

int score_A = 0;
int score_B = 0;

char start;
char sign;
```

4/Initialize Arduino , with bluetooth Serial and lcd(Size, Backlight).

```
//Set Initialize
void setup()
{
    Serial.begin(9600);
    lcd.begin(16, 2);           // initialize the lcd
    // Switch on the backlight

    lcd.setBacklightPin(3, POSITIVE);
    lcd.setBacklight(HIGH);

    lcd.clear();

    lcd.home ();
}
```

5/ Begin the Loop , starting with “auto-update” display on LCD

```
//-----TeamA Name-----
lcd.setCursor(1, 1);
lcd.print(teamA_name);
```

lcd.setCursor : Assign the position of the cursor on LCD to print.

Lcd.print : Print out on the LCD.

6/ Receive data from bluetooth and display.

```
//-----RECEIVE 1 byte of MIN-----
if(Serial.available() == 1) {
    minute = Serial.read();
    minute = minute -48;
}
```

Serial.available() : check to see whether there is data received with bluetooth.
Serial.read(): read the data and put it into variable.

Appendix: Full Source Codes:

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

#define BACKLIGHT_PIN    13

LiquidCrystal_I2C lcd(0x3f, 2, 1, 0, 4, 5, 6, 7); // Set the LCD I2C address

//LiquidCrystal_I2C lcd(0x38, BACKLIGHT_PIN, POSITIVE); // Set the
LCD I2C address

//-----SET TeamName -----
String teamA_name = "LAL";
String teamB_name = "GSW";

int i = 0;

int minute_1;
int minute_2;
int minute;

int second_1;
int second_2;
int second;

int score_A = 0;
int score_B = 0;

char start;
char sign;

//Set Initialize
void setup()
```

```
{  
    Serial.begin(9600);  
    lcd.begin(16, 2);           // initialize the lcd  
    // Switch on the backlight  
  
    lcd.setBacklightPin(3, POSITIVE);  
    lcd.setBacklight(HIGH);  
  
    lcd.clear();  
  
    lcd.home ();  
}  
  
//-----Begin of the LOOP-----  
void loop()  
{  
  
    //-----Begin-----  
    lcd.clear();  
  
    //-----TeamA Name-----  
    lcd.setCursor(1, 1);  
    lcd.print(teamA_name);  
  
    //-----TeamB Name-----  
    lcd.setCursor(13, 1);  
    lcd.print(teamB_name);  
  
    //-----MINUTE Print-----  
    //-----If Min < 10-----  
    if (minute < 10)  
    {  
        lcd.setCursor(6, 0);  
        lcd.print("0");  
        lcd.setCursor(7, 0);  
        lcd.print(minute);  
    }  
  
    //-----If 10<= MIN <= 100-----  
    else if ((minute >= 10) && (minute < 100))
```

```
{  
    lcd.setCursor(6, 0);  
    lcd.print(minute);  
}  
  
//----- : -----  
lcd.setCursor(8, 0);  
lcd.print(":");  
  
//-----SECOND Print-----  
if (second < 10)  
{  
    lcd.setCursor(9, 0);  
    lcd.print("0");  
    lcd.setCursor(10, 0);  
    lcd.print(second);  
}  
  
else if ((second >= 10) && (second < 100))  
{  
    lcd.setCursor(9, 0);  
    lcd.print(second);  
}  
  
//-----Score A Display-----  
if (score_A < 10)  
{  
    lcd.setCursor(7, 1);  
    lcd.print(score_A);  
}  
  
else if ((score_A >= 10) && (score_A < 100))  
{  
    lcd.setCursor(6, 1);  
    lcd.print(score_A);  
}  
  
else  
    if (score_A >= 100)  
    {
```

```
lcd.setCursor(5, 1);
lcd.print(score_A);
}

//-----
lcd.setCursor(8, 1);
lcd.print("-");

//-----Score B Display -----
lcd.setCursor(9, 1);
lcd.print(score_B);

//gerdataBT

//Start of Control

//-----STAGE 1 : SET INITIAL MINUTE-----
if ((i == 0) && (Serial.available() > 0))
{
    //-----RECEIVE 1 byte of MIN-----
    if(Serial.available() == 1) {
        minute = Serial.read();
        minute = minute -48;
    }

    //-----RECEIVE 2 bytes of MIN-----
    else
        if(Serial.available() > 1)
        {
            minute_1 = Serial.read();
            minute_2 = Serial.read();

            minute_1 = minute_1 - 48;
            minute_2 = minute_2 - 48;

            minute = minute_1*10 + minute_2;
        }
}
```

```

//-----Print MINUTE-----
if (minute < 10)
{
    lcd.setCursor(6, 0);
    lcd.print("0");
    lcd.setCursor(7, 0);
    lcd.print(minute);
}

else if ((minute >= 10) && (minute < 100))
{
    lcd.setCursor(6, 0);
    lcd.print(minute);
}

i = i + 1; //Change from Stage 1 to Stage 2

} //End of i=1-----


else
//-----STAGE 2 : SET INITIAL SECOND-----
if ((i == 1) && (Serial.available() > 0))
{

//----RECEIVE 1 byte of SECOND-----
if(Serial.available() == 1) {
    second = Serial.read();
    second = second -48;
}

//----RECEIVE 2 byte of SECOND-----
else
if(Serial.available() > 1)
{
    second_1 = Serial.read();
    second_2 = Serial.read();

    second_1 = second_1 - 48;
}
}

```

```

second_2 = second_2 - 48;

second = second_1*10 + second_2;
}

//----- Print SECOND -----
if (second < 10)
{
    lcd.setCursor(9, 0);
    lcd.print("0");
    lcd.setCursor(10, 0);
    lcd.print(second);
}

else
if ((second >= 10) && (second < 100))
{
    lcd.setCursor(9, 0);
    lcd.print(second);
}

i = i + 1;
} //End of STAGE 2-----


else
//-----STAGE 3 : Prepare TO BEGIN-----
if ((i == 2) && (Serial.available() > 0))
{
    start = Serial.read();
    while (((minute != 0) || (second != 0)) && (start == '1'))
    {
        if ((second == 0) && (minute != 0))
        {
            second = 60;
            minute = minute - 1;

            if (minute < 10)
            {

```

```
lcd.setCursor(6, 0);
lcd.print("0");
lcd.setCursor(7, 0);
lcd.print(minute);
}

else if ((minute >= 10) && (minute < 100))
{
    lcd.setCursor(6, 0);
    lcd.print(minute);
}

if (second > 0)
{
    second = second - 1;

    if (second < 10)
    {

        lcd.setCursor(9, 0);
        lcd.print("0");
        lcd.setCursor(10, 0);
        lcd.print(second);
    }

    else if ((second >= 10) && (second < 100))
    {
        lcd.setCursor(9, 0);
        lcd.print(second);
    }

    delay(950);
}

sign = Serial.read();

//-----PLUS A-----
if (sign == '2')
{
```

```
score_A = score_A + 1;

if (score_A < 10)
{
    lcd.setCursor(7, 1);
    lcd.print(score_A);
}

else
if ((score_A >= 10) && (score_A < 100))
{
    lcd.setCursor(6, 1);
    lcd.print(score_A);
}

else
if (score_A >= 100)
{
    lcd.setCursor(5, 1);
    lcd.print(score_A);
}

else
//-----MINUS A-----
if (sign == '3')
{
    if (score_A > 0) {
        score_A = score_A - 1;

        if (score_A < 10)
        {
            lcd.setCursor(7, 1);
            lcd.print(score_A);
        }

        else
if ((score_A >= 10) && (score_A < 100))
{
    lcd.setCursor(6, 1);
    lcd.print(score_A);
}
```

```
    }

    else
        if (score_A >= 100)
        {
            lcd.setCursor(5, 1);
            lcd.print(score_A);
        }

    }

else
    //-----PLUS B-----
    if (sign == '4')
    {
        score_B = score_B + 1;

        lcd.setCursor(9, 1);
        lcd.print(score_B);
    }

else
    //-----MINUS B-----
    if (sign == '5')
    {

        if (score_B > 0) {
            score_B = score_B - 1;

            lcd.setCursor(9, 1);
            lcd.print(score_B);
        }
    }

else
    //-----PAUSE / RESUME -----
    if (sign == '6')
    {

        if (start == '1')
```

```
    start = '0';  
  
    else start ='1';  
}  
}  
  
delay(1000);  
}
```

THE END.