

NAME : SAUARBH SALVI

ROLL NO.: ME22B1033

TAD
ASSIGNMENT
(MISC.)



Code Explanation :

I have divided the code into 3 sections:

- Initialization
- Void setup
- Void loop

Initialization :

```
7  #include <Servo.h>
8
9  #include <LiquidCrystal.h>
10
11  const int rs = 13, enLCD = 11, d4 = 4, d5 = 5, d6 = 7, d7 = 2;
12  LiquidCrystal lcd(rs, enLCD, d4, d5, d6, d7);
13
14  Servo rudder;
15  Servo aeliron;
16
17  int potValProp; //Propeller
18  int potValRud;
19  int potValAeli;
20  int pos;
21  int pos1;
22  int in3 = 9;    // for DC motor control
23  int in4 = 8;
24  int en = 10; // enable pin
25
```

First, I have included the libraries, `#include<Servo.h>` for controlling servo motors and `#include<LiquidCrystal.h>` for the LCD display.

After that, I have done the variable declaration for the purpose of LCD, i.e. all the pins that are connected to the LCD.

I am naming my two servos as rudder and aeliron as both are supposed to do the respective work.

- ❖ Rudder will take care of the 'yaw' motion
- ❖ Aelirons will take care of 'roll'

After that, I have declared some variables which will be used to do analog Reading from analog pins with the help of potentiometers.

'pos' and 'pos1' are actually used to show angle rotated by the servo actually.

in3, in4 and en are the variables for the DC motor controlling purpose via L293D motor driver.

Note: enable pin is to be connected to one of the PWM pins on Arduino.

Void Setup :

```
26 void setup()
27 {
28
29     rudder.attach(6);
30     aeliron.attach(3);
31
32     pinMode(en, OUTPUT);
33     pinMode(A0, INPUT); //rudder
34     pinMode(A1, INPUT); //aeliron
35     pinMode(A3, INPUT); //propeller
36     pinMode(A5, INPUT);
37
38     Serial.begin(9600);
39
40     // set up the LCD's number of columns and rows:
41     lcd.begin(16, 2);
42     // Print a message to the LCD.
43     lcd.print("TAD Club");
44     delay(3000);
45     for (int Counter = 0; Counter < 16; Counter++) {
46         // scroll one position right:
47         lcd.scrollDisplayRight();
48         // wait a bit:
49         delay(150);
50     }
51     lcd.clear();
52     lcd.setCursor(0, 0);
53     lcd.print("RudderAngle:");
54     // print the number of seconds since reset:
55 }
56
```

I first, give the pins to which my servo's(rudder and aeliron) are attached and those pins should be PWM pins as well.

Next, I had set the enable pin as OUTPUT, and rest as INPUT as they are mostly potentiometers which will be acting as inputs basically.

Then, I establish Serial communication.

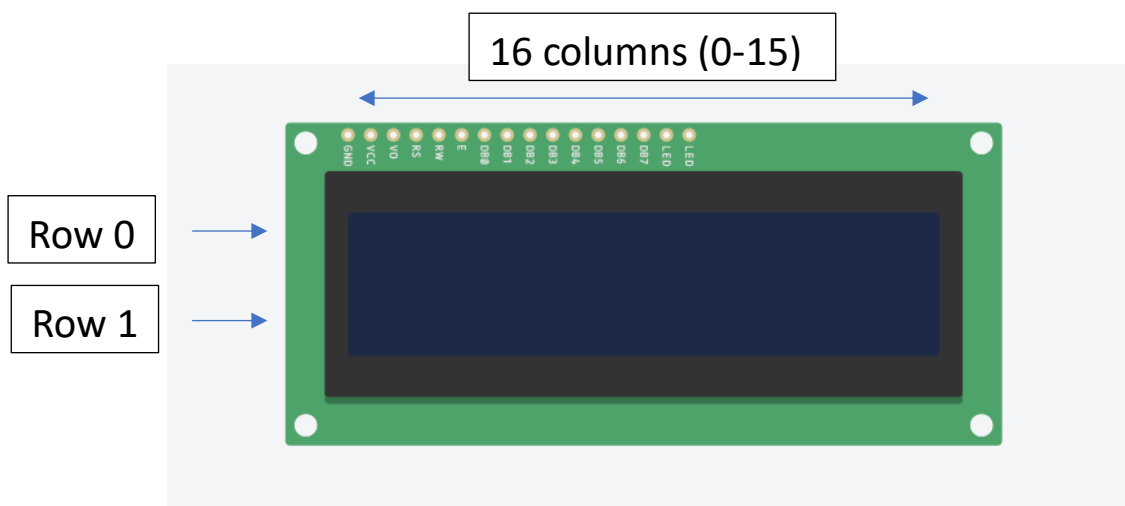
I now set up the LCD screen 16 x 2 by `lcd.begin(16, 2)`

Now, we print "TAD Club" on LCD and make it scroll till right and we start that by setting cursor starting from 0 and going till 16. That will happen till 3sec.

Now, we clear our LCD display.

For setting cursor on LCD display we use :

```
lcd.setCursor(column, row);
```

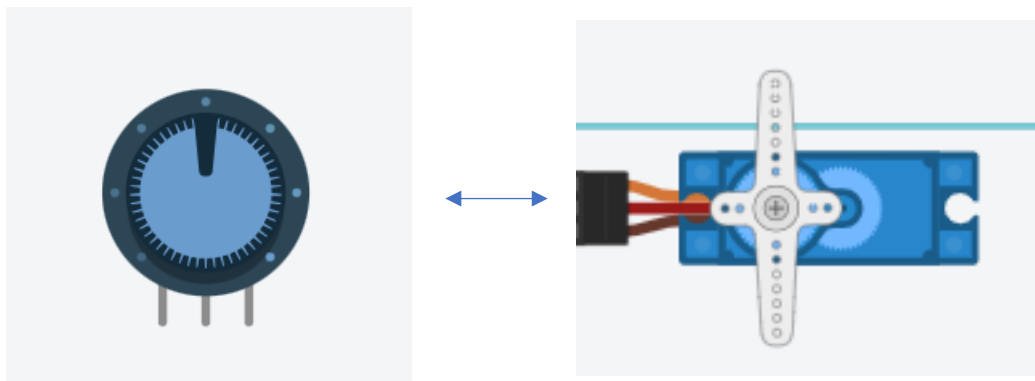


Void loop :

```
57 void loop()
58 {
59   int m= pos-90;
60   // lcd.setCursor(0, 1);
61   // + ACW
62   //lcd.print(m);
63   if(m>=0){
64     lcd.setCursor(0, 1);
65     lcd.print(m);
66     lcd.setCursor(3, 1);
67     lcd.print("deg");
68     lcd.setCursor(7, 1);
69     lcd.print("ACW");
70   }
71   else if(m<0){
72     lcd.setCursor(0, 1);
73     lcd.print(-m);
74     lcd.setCursor(3, 1);
75     lcd.print("deg");
76     lcd.setCursor(7, 1);
77     lcd.print("CW");
78   }
79 }
```

This set of code is to print the value of angle rotated by servo from the 90 deg mark.**

I have tried to make the potentiometer act like my controller so initially it will be at 90deg angle (actually). So that we get a controller like thing, **turning on left side, turns servo arm clockwise and turning to right, arm turns anticlockwise



```

87 //-----
88 /**for rudder servo**
89 potValRud = analogRead(A0);
90
91
92 pos = map(potValRud, 0, 1023, 0, 180);
93 Serial.print("Angle by rudder : ");
94
95 Serial.println(m);
96
97 //Problem is coming regarding the continuous looping of
98 //if(pos>=90)
99 if(pos >= 90){
100 for(int i = 90; i<=pos; i++){
101     rudder.write(i);
102     delay(100);
103 }
104 }
105 }
106 //perfect below code !!
107 else if(pos < 90){
108     for(int i = 90; i>=pos; i--){
109         rudder.write(pos);
110         delay(100);
111     }
112 }
113 //-----

```

I map the values coming from analog pin A0,

0 -> 0

1023 -> 180

'pos' is nothing but the angle rotated by servo is obtuse actually, then bring it to 90 and then move pos-90 angle to complete, but it actually looks like as if it just moved pos-90 angle.

```

113 //-----
114 /**for aeliron servo**
115 potValAeli = analogRead(A1);
116
117
118 pos1 = map(potValAeli, 0, 1023, 0, 180);
119 Serial.print("Angle by aeliron : ");
120 Serial.println(pos1-90);
121
122 //Problem is coming regarding the continuous looping of
123 //if(pos>=90)
124 if(pos1 >= 90){
125   for(int i = 90; i<=pos1; i++){
126     aeliron.write(i);
127     delay(100);
128   }
129 }
130 }
131 //perfect below code !!
132 else if(pos1 < 90){
133   for(int i = 90; i>=pos1; i--){
134     aeliron.write(pos1);
135     delay(100);
136   }
137 }
138 }
139 //-----

```

Similar code for controlling aeliron to rotate down and up so correspondingly for servo, it will be clockwise and anticlockwise.

Here, potValAeli will be the variable storing the values which will be obtained from analog pin A1 and signal to analog pin A1 is given by potentiometer

```

139 //-----
140 /**for propeller**
141
142 potValProp = analogRead(A3);
143 int final = map(potValProp, 0, 1023, 0, 255);
144 //if (final < 127){
145
146   analogWrite(en, 255);
147   digitalWrite(in3, HIGH);
148   digitalWrite(in4, LOW);
149   delay(3000);
150
151   digitalWrite(in3, LOW);
152   digitalWrite(in4, LOW);
153   //}
154
155
156 /*
157   else if (final>127){
158
159
160     analogWrite(en, potValProp/4);
161     digitalWrite(in3, LOW);
162     digitalWrite(in4, HIGH);
163
164
165   }
166   */
167
168 //-----
169
170 }

```

I set the motor driver's enable pin which is responsible to control propeller motor to 255 in analogWrite motion with in3 HIGH and in4 LOW which makes motor rotate in clockwise at full speed

And then, I turn in3 and in4 LOW which makes the motor turn OFF.
