

```

1: //Header Files
2: #include<LiquidCrystal.h>
3: #include<Q2HX711.h>
4: #include <Servo.h>
5: Servo myservo;
6:
7: int sonarread();
8: int Loadread();
9: void motoron();
10: void motoroff();
11:
12: //Ideal parameters;
13: int id_dist_a = 3;
14: int id_wt_a = 100;
15: int id_dist_b = 2.5;
16: int id_wt_b = 70;
17: int Ano,Bno;
18:
19: //LCD
20: const int rs=36,en=37,d4=38,d5=39,d6=40,d7=41;
21: LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
22:
23: //IR SENSOR
24: int ir1 = 22;
25: int ir2 = 23;
26:
27: //SONAR
28: int echo = 24;
29: int trig = 25;
30:
31: //STATUS LED
32: int gLED = 28;
33: int rLED = 29;
34: int bLED = 30;
35: int stLED = 31;
36:
37: //Load Cell
38: #define SCK A1
39: #define DT A0
40: Q2HX711 cell(DT,SCK);
41: long int values = 0;
42: int load_data;
43:
44: long val = 0;
45: float count = 0;
46:
47: //DC MOTOR
48: int enb = 32;
49: int op1 = 33;
50: int op2 = 34;
51:
52: //ESP8266
53:
54: #define prod_a 42
55: #define prod_b 43

```

```

56: int a = 0;
57: int b = 0;
58: int a1 = 44;
59: int a2 = 45;
60: int a3 = 46;
61: int b1 = 47;
62: int b2 = 48;
63: int b3 = 49;
64:
65: void setup()
66: {
67:     // put your setup code here, to run once:
68:
69:     //Servo
70:     myservo.attach(26);
71:     //IR
72:     pinMode(ir1, INPUT);
73:     pinMode(ir2, INPUT);
74:
75:     //SONAR
76:     pinMode(echo, INPUT);
77:     pinMode(trig, OUTPUT);
78:
79:     //STATUS LED
80:     pinMode(gLED, OUTPUT);
81:     pinMode(rLED, OUTPUT);
82:     pinMode(bLED, OUTPUT);
83:     pinMode(stLED, OUTPUT);
84:
85:     //Load cell
86:     pinMode(DT, INPUT);
87:     pinMode(SCK, INPUT);
88:
89:     //DC MOTOR
90:     pinMode(enb, OUTPUT);
91:     pinMode(op1, OUTPUT);
92:     pinMode(op2, OUTPUT);
93:
94:     //esp8266
95:     pinMode(prod_a, INPUT);
96:     pinMode(prod_b, INPUT);
97:     pinMode(a1, INPUT);
98:     pinMode(a2, INPUT);
99:     pinMode(a3, INPUT);
100:    pinMode(b1, INPUT);
101:    pinMode(b2, INPUT);
102:    pinMode(b3, INPUT);
103:    //a and b are the indication of prod A and prod B
104:
105:    //LCD
106:    lcd.begin(16, 2);
107:    lcd.println("Randiyarasan");
108:
109:    Serial.begin(9600);
110: }

```

```

111:
112: void loop()
113: {
114:     // put your main code here
115:     while( a == 0 || b == 0)
116:     {
117:         lcd.clear();
118:         lcd.setCursor(0,0);
119:         lcd.println("Plz enter prod");
120:         lcd.setCursor(0,1);
121:         lcd.println("A and/or B");
122:         a = digitalRead(prod_a);
123:         while(a == 1)
124:         {
125:
126:             int A1 = digitalRead(a1);
127:             int A2 = digitalRead(a2);
128:             int A3 = digitalRead(a3);
129:
130:             if(A1 == 1 || A2 == 1 || A3 == 1)
131:             {
132:                 Ano = A1*0+A2*1+A3*2;
133:                 break;
134:             }
135:         }
136:         b = digitalRead(prod_b);
137:         while(b == 1)
138:         {
139:             int B4 = digitalRead(b1);
140:             int B2 = digitalRead(b2);
141:             int B3 = digitalRead(b3);
142:
143:             if(B4 == 1 || B2 == 1 || B3 == 1)
144:             {
145:                 Bno = B4*0+B2*1+B3*2;
146:                 break;
147:             }
148:         }
149:     }
150:
151:     lcd.clear();
152:     lcd.setCursor(0,0);
153:     lcd.println("A = %d");
154:     lcd.setCursor(0,1);
155:     lcd.println("B = %d");
156:
157:
158:     //Starting the process
159:     digitalWrite(stLED,1);
160:
161:     //Process for product A
162:     for(int delta = 0; delta < Ano; delta++)
163:     {
164:         motoroff();
165:         while( digitalRead(ir1) != 1 ){

```

```

166:         lcd.clear();
167:         lcd.println("Place product A");
168:     }
169:
170:     while(digitalRead(ir2)!=0)
171:     {
172:         motoron();
173:     }
174:
175:     motoroff();
176:
177:     int dist = sonarread();
178:     int weight = Loadread();
179:     if(weight>id_wt_a+10 || weight<id_wt_a-10 || dist>id_dist_a+1 || dist<id_dist_a-1)
180:     {
181:         delta = delta-1;
182:         lcd.clear();
183:         lcd.println("Faulty product A");
184:         digitalWrite(rLED,1);
185:         myservo.write(90);
186:         delay(100);
187:         myservo.write(0);
188:     }
189:     else{
190:         motoron();
191:         delay(2000);
192:         lcd.clear();
193:         lcd.println("Success");
194:     }
195: }
196:
197:
198:
199: //Process for product b
200: for(int delta = 0;delta < Bno; delta++)
201: {
202:     motoroff();
203:     while(digitalRead(ir1) != 1)
204:     {
205:         lcd.clear();
206:         lcd.println("Place product B");
207:     }
208:
209:     while(digitalRead(ir2)!=0)
210:     {
211:         motoron();
212:     }
213:
214:     motoroff();
215:
216:     int dist = sonarread();
217:     int weight = Loadread();
218:     if(weight>id_wt_b+10 || weight<id_wt_b-10 || dist>id_dist_b+1 || dist<id_dist_b-1)
219:     {
220:         delta = delta-1;

```

```

221:         lcd.clear();
222:         lcd.println("Faulty product B");
223:         digitalWrite(rLED,1);
224:         myservo.write(90);
225:         delay(100);
226:         myservo.write(0);
227:     }
228:     else
229:     {
230:         motoron();
231:         delay(2000);
232:         lcd.clear();
233:         lcd.println("Success");
234:     }
235: }
236: }
237:
238: //distance function
239: int sonarread()
240: {
241:     digitalWrite(trig,0);
242:     delayMicroseconds(2);
243:
244:     digitalWrite(trig,1);
245:     delayMicroseconds(10);
246:     digitalWrite(trig,0);
247:
248:     int duration = pulseIn(12,1);
249:     int distance = duration*0.034/2;
250:
251:     return distance;
252: }
253:
254: //Weight function
255: int Loadread()
256: {
257:     count = count +1;
258:     for(int i=0;i<15;i++)
259:     {
260:         val = 0.5*val+0.5*cell.read();
261:         load_data = -((val-8547083)/450);
262:     }
263:     return load_data;
264: }
265:
266: //motor functions
267: void motoron()
268: {
269:     digitalWrite(enb,1);
270:     digitalWrite(op1,1);
271:     digitalWrite(op2,0);
272: }
273: void motoroff()
274: {
275:     digitalWrite(enb,0);

```

```
276:     digitalWrite(op1,0);
277:     digitalWrite(op2,0);
278: }
```