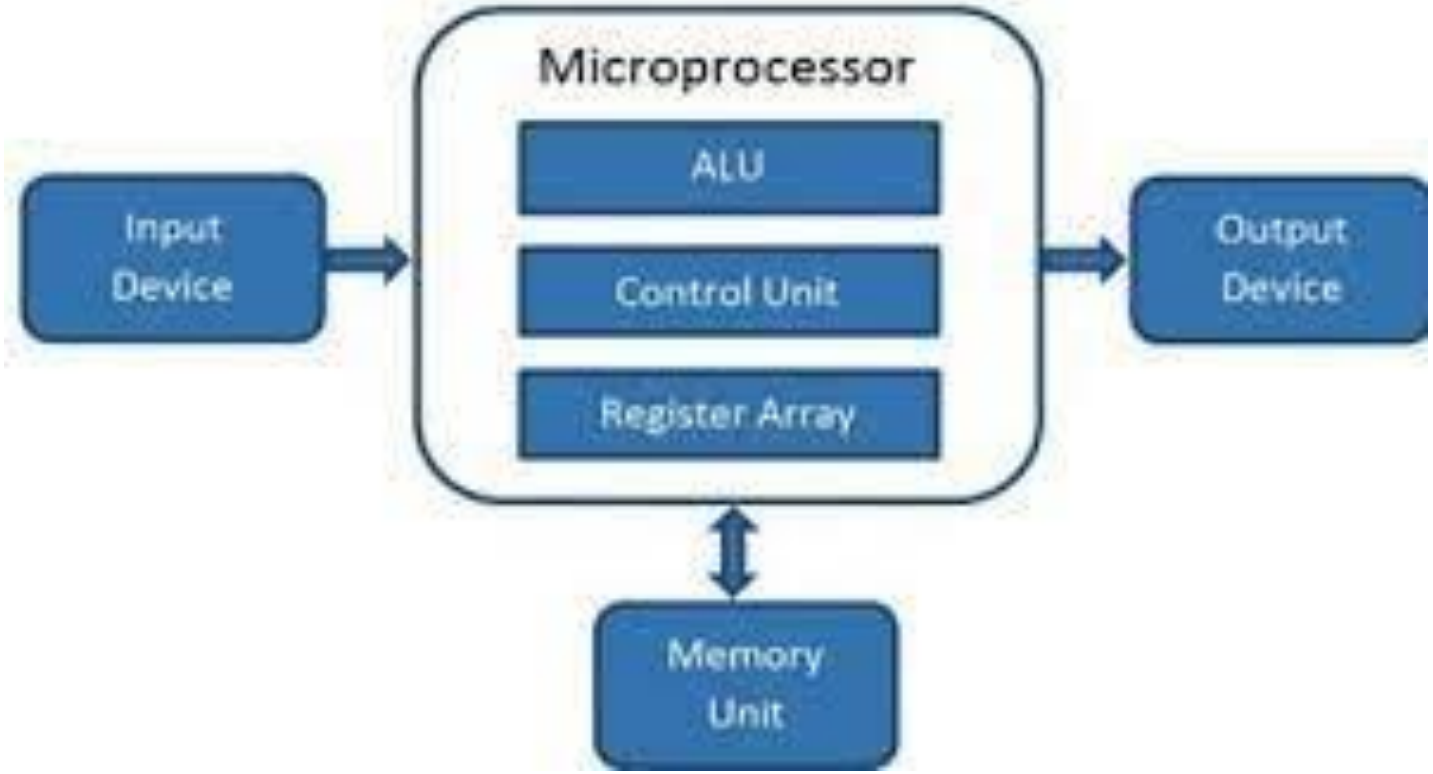
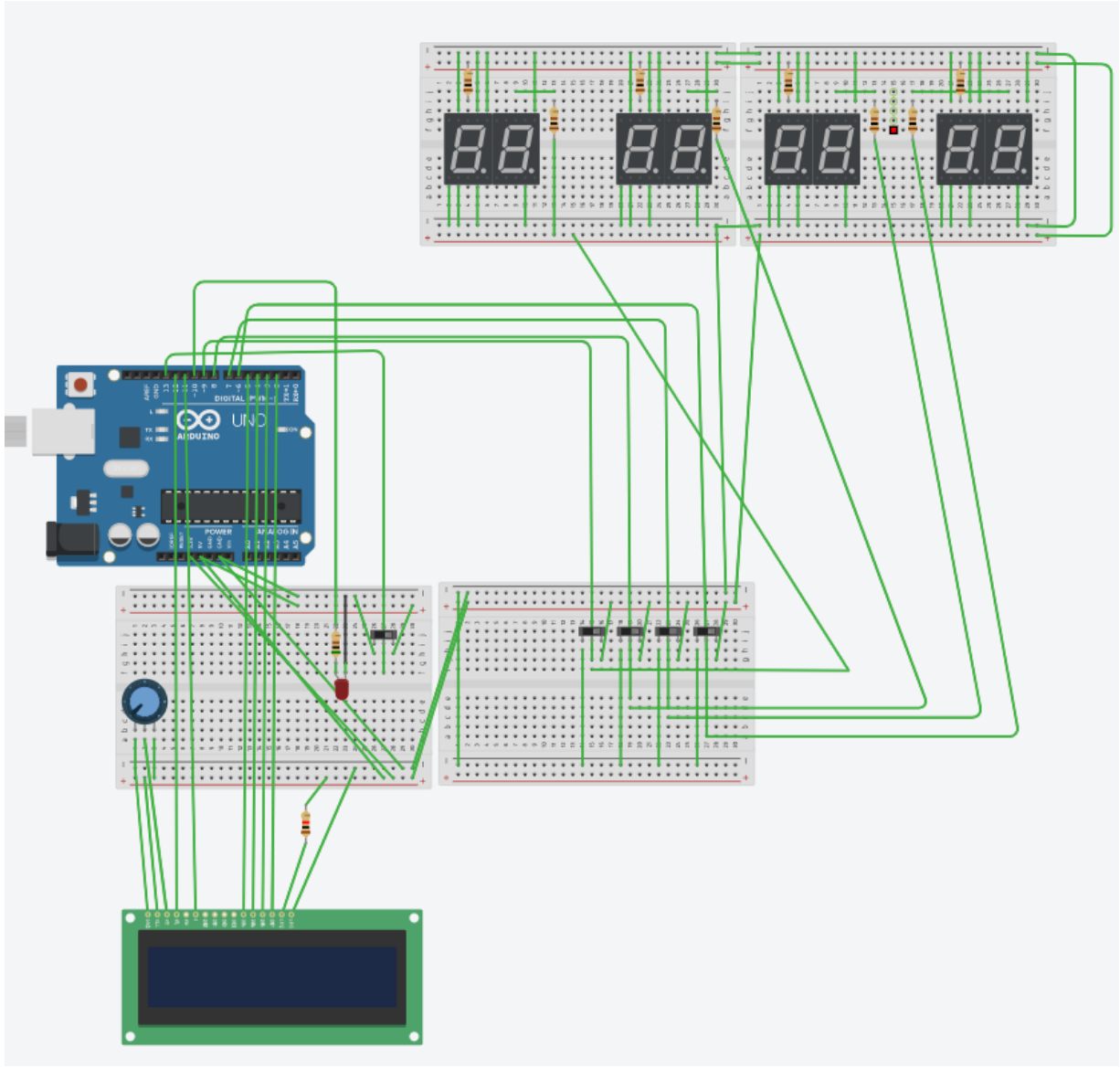


Microprocessors

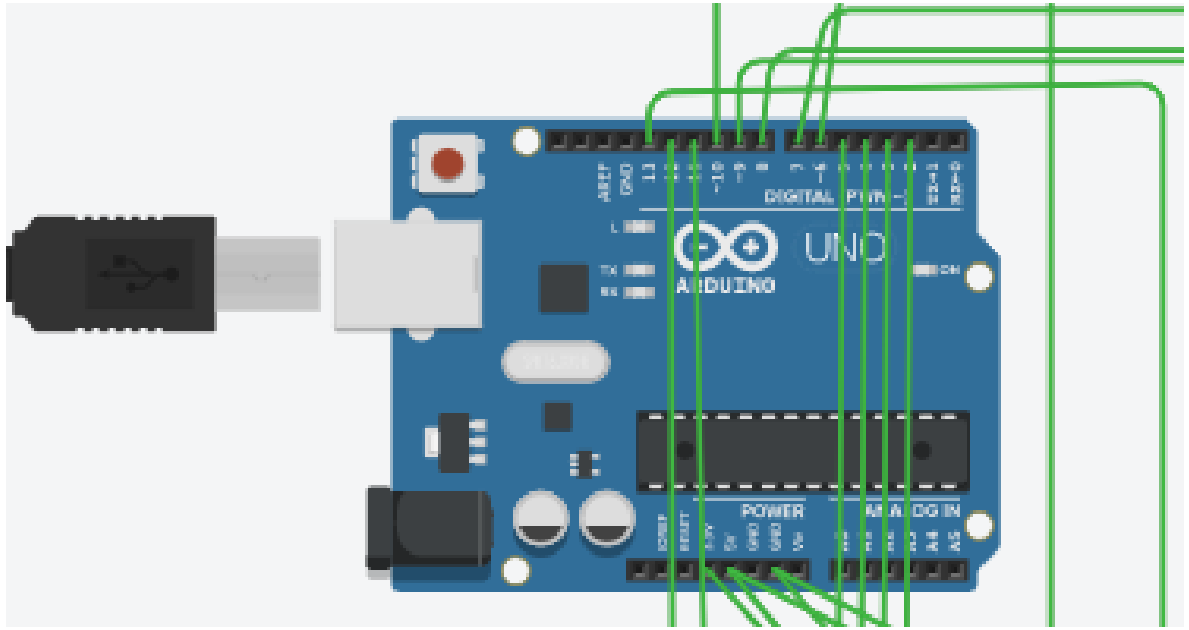
Angel Gil

4- bit Hamming code calculator



Assigning variables for connections

```
#include <LiquidCrystal.h>
int led = 10;
int switchpin1 = 9;
int switchpin2 = 8;
int switchpin3 = 7;
int switchpin4 = 6;
int switchpin5 = 13;
int a;
int b;
int c;
int d;
int e;
int f;
int g;
int switchstate;
const int rs = 12, en = 11, d4 =5, d5 = 4, d6 = 3, d7 = 2;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
void setup()
{
 Serial.begin(9600);
 lcd.begin(16,2);
 pinMode(led, OUTPUT);
 pinMode(switchpin1, INPUT);
 pinMode(switchpin2, INPUT);
 pinMode(switchpin3, INPUT);
 pinMode(switchpin4, INPUT);
 pinMode(switchpin5, INPUT);
}
```

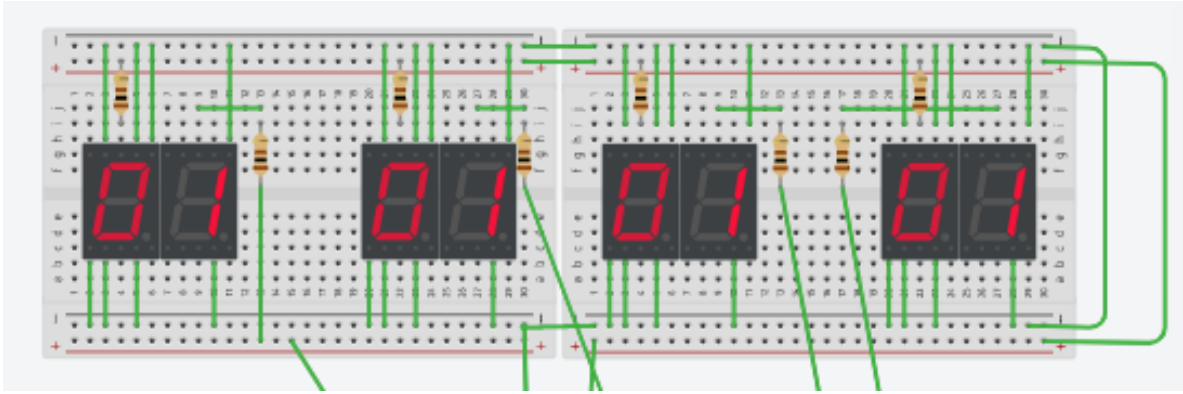


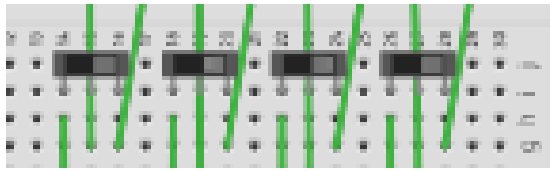
Beginning our void loop function


```
29  void loop()
30  {
31   lcd.clear();
32   lcd.setCursor(0,0);
33   lcd.print("Choose 4 bits 4");
34   lcd.setCursor(0,1);
35   lcd.print("Even hamming");
36   delay(250);
```

Reading the 7-segment displays

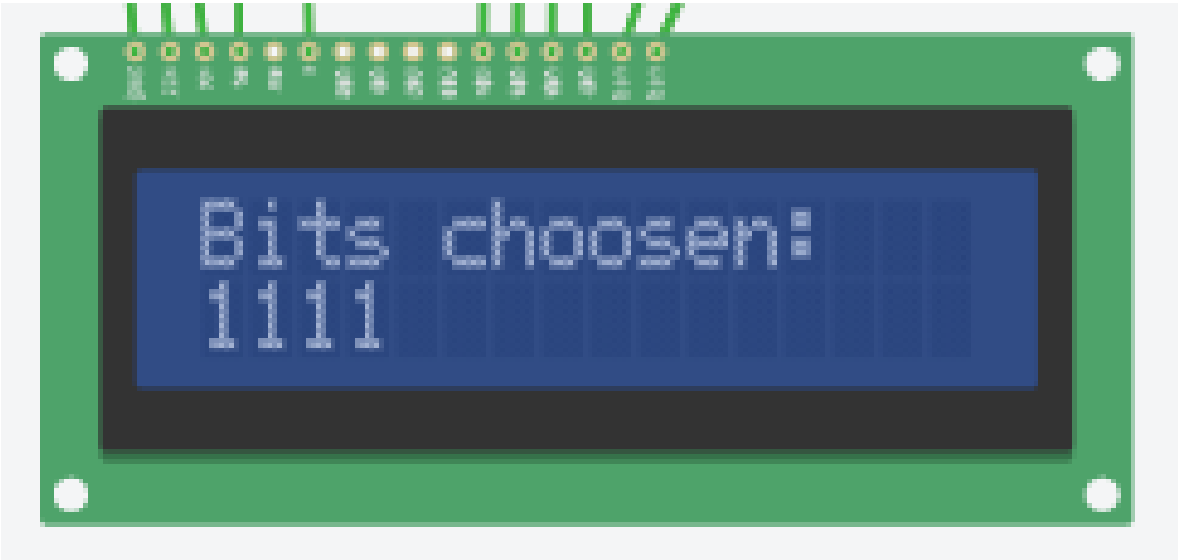
```
37      if (digitalRead(switchpin1))
38      {
39      c=1;
40      }
41      else
42      {
43      c=0;
44      }
45           if (digitalRead(switchpin2))
46           {
47           e=1;
48             delay(250);
49           }
50           else
51           {
52           e=0;
53             delay(250);
54           }
55                if (digitalRead(switchpin2))
56                {
57                f=1;
58                  delay(250);
59                }
60                else
61                {
62                f=0;
63                  delay(250);
64                }
65                     if (digitalRead(switchpin2))
66                     {
67                     g=1;
68                       delay(250);
69                     }
70                     else
71                     {
72                     g=0;
73                       delay(250);
74                     }
75
```





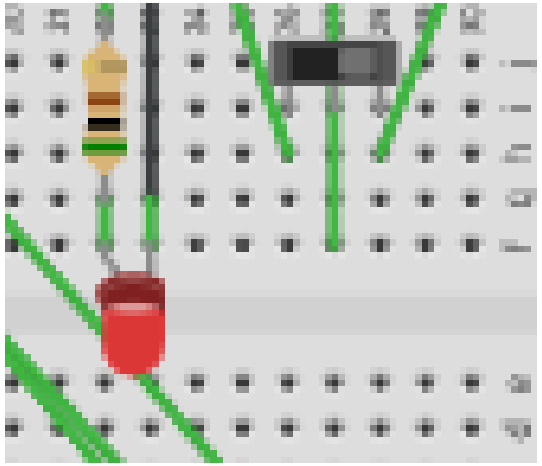
Displaying the user's 4-bit code

```
76    lcd.clear();
77    lcd.setCursor(0,0);
78    lcd.print("Bits choosen:");
79    lcd.setCursor(0,1);
80    lcd.print(c);
81    delay(250);
82    lcd.setCursor(1,1);
83    lcd.print(e);
84    delay(250);
85    lcd.setCursor(2,1);
86    lcd.print(f);
87    delay(250);
88    lcd.setCursor(3,1);
89    lcd.print(g);
90    delay(1000);
```



Calculating the even parity Hamming code

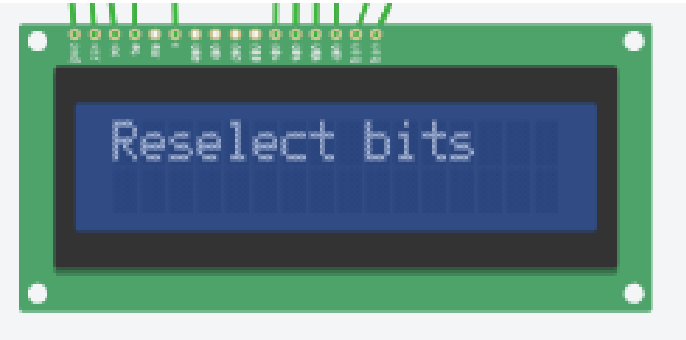
```
91    if (digitalRead(switchpin5))
92    {
93      digitalWrite(led, HIGH);    // turn the LED on (HIGH
94      delay(250);
95      lcd.clear();
96      lcd.setCursor(0,0);
97      lcd.print("Even Hamming:");
98      delay(1000);
99      if ((c+e+g)%2 ==0)// parity bit 1 checks b3,b5,b7
100     {
101     a=0;
102     }
103     else
104     {
105     a=1;
106     }
107     if ((c+f+g)%2==0) // parity bit 2 checks b3,b6,b7
108     {
109      b=0;
110     }
111     else
112     {
113      b=1;
114     }
115     if ((e+f+g)%2==0) // parity bit 3 checks b5,b6,b7
116     {
117      d=0;
118     }
119     else
120     {
121      d=1;
122     }
123     lcd.setCursor(0,1);
124     lcd.print(a);
125     lcd.setCursor(1,1);
126     lcd.print(b);
127     lcd.setCursor(2,1);
128     lcd.print(c);
129     lcd.setCursor(3,1);
130     lcd.print(d);
131     lcd.setCursor(4,1);
132     lcd.print(e);
133     lcd.setCursor(5,1);
134     lcd.print(f);
135     lcd.setCursor(6,1);
136     lcd.print(g);
```



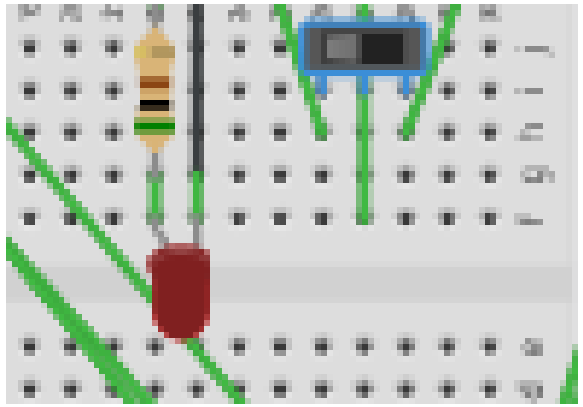
Displaying the parity bits

```
138       lcd.clear();
139       lcd.setCursor(0,0);
140       lcd.print("Parity bits");
141       lcd.setCursor(0,1);
142       lcd.print("are:");
143       lcd.setCursor(5,1);
144       lcd.print(a);
145       lcd.setCursor(6,1);
146       lcd.print(b);
147       lcd.setCursor(7,1);
148       lcd.print(d);
149       delay(1000);
150     }
```

Reselecting bits to proceed in the program




```
151   else
152   {
153     digitalWrite(led, LOW);     // turn the LED off
154     delay(250);
155     lcd.clear();
156     lcd.setCursor(0,0);
157     lcd.print("Reselect bits");
158     delay(1000);
159   }
160 }
```



- The End
- Thank you!