Week 4-5:

Arduino Uno with 7 Segment Display

Learning Outcomes:

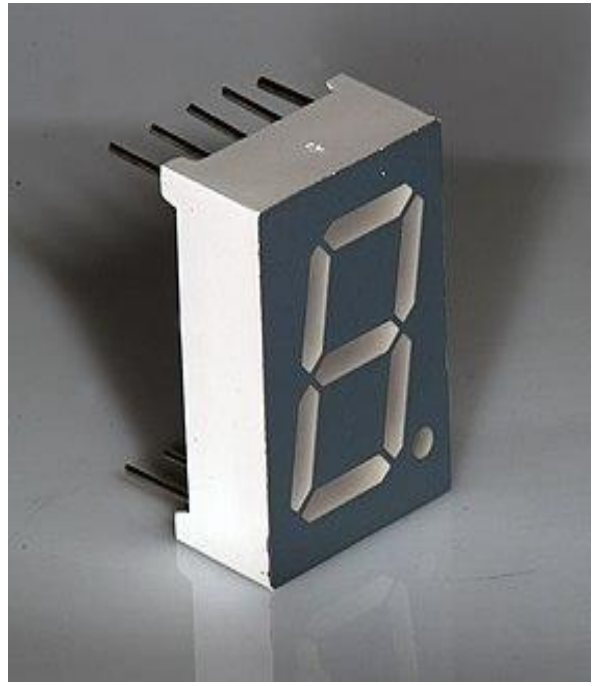
- Identify the digital and 7 segment with a subtle push button.
- Set up the 7 segment with tact push button.
- Perform the 7 segment counter 0-9 with push buttons up and down.

- A 7-segment display is an electronic display made of **seven LED bars** shaped like the number “8.”
  By turning ON different segments, we can show numbers **0–9**, and sometimes letters.
  - The earliest known inventor is **Frank W. Wood (1908).**
- In 1908, Frank W. Wood filed one of the first patents describing a segmented numerical display using incandescent lamps.
  This is considered the first conceptual design of a 7-segment display.

Structure and Components

1. Segments:

A 7-segment display consists of seven LED segments arranged in a rectangular fashion. Each segment can be lit individually to form numbers and some letters. The segments are labeled from 'a' to 'g'.



2. Common Types:

Common Cathode (CC):

In a common cathode display, all the cathodes (negative terminals) of the LED segments are connected together to a common point, usually ground.

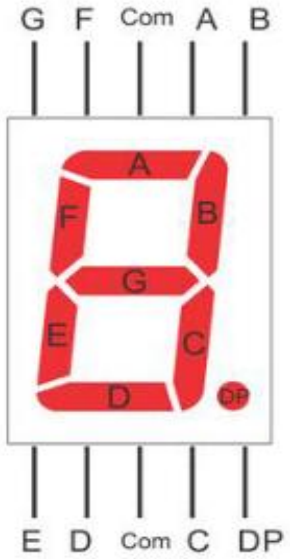
Common Anode (CA):

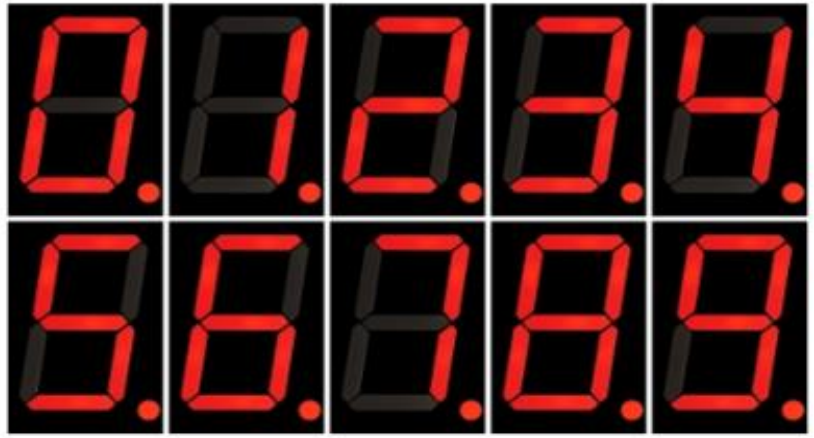
In a common anode display, all the anodes (positive terminals) are connected together to a common point, usually the power supply voltage.

Displaying Digits

- By turning on specific segments, different numbers (0-9) can be displayed:
  - **0**: a, b, c, d, e, f
  - **1**: b, c
  - **2**: a, b, d, e, g
  - **3**: a, b, c, d, g
  - **4**: b, c, f, g
  - **5**: a, c, d, f, g
  - **6**: a, c, d, e, f, g
  - **7**: a, b, c
  - **8**: a, b, c, d, e, f, g
  - **9**: a, b, c, d, f, g

Displaying Digits





PEMBEDS

Practical Applications

- **Digital Clocks:**

  7-segment displays are used to show hours, minutes, and sometimes seconds. Multiplexing is often employed to control multiple digits with fewer control lines.

- **Calculators:**

  Early calculators used 7-segment displays to show input numbers and results.

- **Counters and Meters:**

  Used in counters, voltmeters, ammeters, and other measurement devices to display numerical data.

Practical Applications

Industrial / Commercial

- Fuel pumps
- Control panels
- Elevators (floor indicators)
- CNC machines counters
- Factory display boards

Medical Equipment

- BP monitor displays
- Heart rate machines (older)
- Laboratory measurement devices

Practical Applications

Transportation

- Bus route displays
- Gas station price boards
- Train station indicators

Education

- Perfect introductory device for:
  - electronics
  - digital logic
  - microcontroller programming
  - embedded systems

Advantages and Limitations

Advantages:

Simple and inexpensive.

Easy to interface with digital systems.

Readable at a distance.

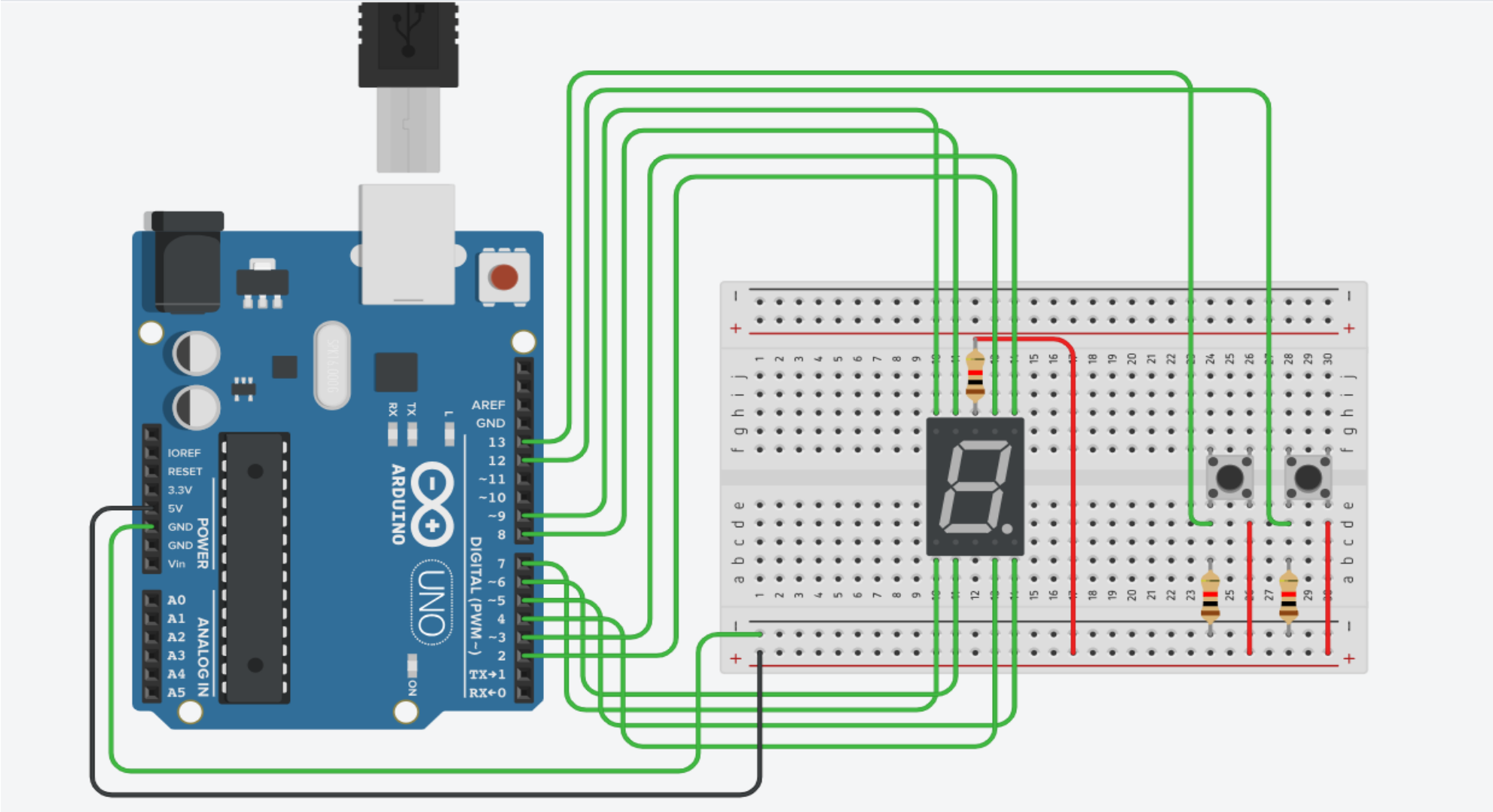
Limitations:

Limited to displaying numeric and a few alphabetic characters.

Low resolution compared to dot-matrix displays.

Not suitable for complex graphics or text.

Diagrams


IOREF
RESET
3.3V
5V
GND
GND
Vin
POWER
A0
A1
A2
A3
A4
A5
ANALOG IN
TX
RX
L
ARDUINO
UNO
ON
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM~)

Codes:

```
1
2  int A = 2;
3  int B = 3;
4  int C = 4;
5  int D = 6;
6  int DP = 5;
7  int E = 7;
8  int F = 8;
9  int G = 9;
10
11 int switchUpPin = 13;
12 int switchDownPin = 12;
13 int counter = 0;
14 int buttonUpState = 0;
15 int lastButtonUpState = 0;
16 int buttonDownState = 0;
17 int lastButtonDownState = 0;
18
19 void setup() {
20   Serial.begin(9600);
21   pinMode(A, OUTPUT);
22   pinMode(B, OUTPUT);
23   pinMode(C, OUTPUT);
24   pinMode(D, OUTPUT);
25   pinMode(E, OUTPUT);
26   pinMode(F, OUTPUT);
27   pinMode(G, OUTPUT);
28   pinMode(DP, OUTPUT);
29   digitalWrite(DP ,HIGH);
30 }
```

Codes:

```
void loop() {
  buttonUpState = digitalRead(switchUpPin);
  buttonDownState = digitalRead(switchDownPin);
  if (buttonUpState != lastButtonUpState) {
    if (buttonUpState == HIGH) {
      if(counter == 9) {
        counter = 9;
      }
      counter++;
      Serial.println(counter);
      changeNumber(counter);
      delay(100);
    } else {
      Serial.println("OFF");
    }
    delay(50);
  }
  if (buttonDownState != lastButtonDownState) {
    if (buttonDownState == HIGH) {
      if(counter == 0) {
        counter = 0;
      }
      counter--;
      Serial.println(counter);
      changeNumber(counter);
      delay(300);
    } else {
      Serial.println("OFF");
    }
    delay(50);
```

Codes:

```
62    }
63    changeNumber(counter);
64  }
65
66  void changeNumber(int buttonPress) {
67    switch (buttonPress) {
68      case 0:
69      digitalWrite(A, LOW);
70      digitalWrite(B, LOW);
71      digitalWrite(C, LOW);
72      digitalWrite(D, LOW);
73      digitalWrite(E, LOW);
74      digitalWrite(F, LOW);
75      digitalWrite(G, HIGH);
76      break;
77      case 1:
78      digitalWrite(A, HIGH);
79      digitalWrite(B, LOW);
80      digitalWrite(C, LOW);
81      digitalWrite(D, HIGH);
82      digitalWrite(E, HIGH);
83      digitalWrite(F, HIGH);
84      digitalWrite(G, HIGH);
85      break;
```

Codes:

```
86      case 2:
87      digitalWrite(A, LOW);
88      digitalWrite(B, LOW);
89      digitalWrite(C, HIGH);
90      digitalWrite(D, LOW);
91      digitalWrite(E, LOW);
92      digitalWrite(F, HIGH);
93      digitalWrite(G, LOW);
94      break;
95      case 3:
96      digitalWrite(A, LOW);
97      digitalWrite(B, LOW);
98      digitalWrite(C, LOW);
99      digitalWrite(D, LOW);
100     digitalWrite(E, HIGH);
101     digitalWrite(F, HIGH);
102     digitalWrite(G, LOW);
103     break;
104     case 4:
105     digitalWrite(A, HIGH);
106     digitalWrite(B, LOW);
107     digitalWrite(C, LOW);
108     digitalWrite(D, HIGH);
109     digitalWrite(E, HIGH);
110     digitalWrite(F, LOW);
111     digitalWrite(G, LOW);
112     break;
```

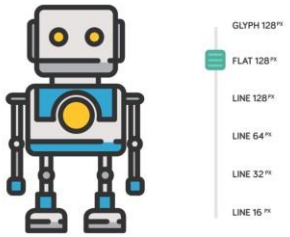
Codes:

```
113       case 5:
114       digitalWrite(A, LOW);
115       digitalWrite(B, HIGH);
116       digitalWrite(C, LOW);
117       digitalWrite(D, LOW);
118       digitalWrite(E, HIGH);
119       digitalWrite(F, LOW);
120       digitalWrite(G, LOW);
121       break;
122       case 6:
123       digitalWrite(A, LOW);
124       digitalWrite(B, HIGH);
125       digitalWrite(C, LOW);
126       digitalWrite(D, LOW);
127       digitalWrite(E, LOW);
128       digitalWrite(F, LOW);
129       digitalWrite(G, LOW);
130       break;
131       case 7:
132       digitalWrite(A, LOW);
133       digitalWrite(B, LOW);
134       digitalWrite(C, LOW);
135       digitalWrite(D, HIGH);
136       digitalWrite(E, HIGH);
137       digitalWrite(F, HIGH);
138       digitalWrite(G, HIGH);
139       break;
```

Codes:

```
140       case 8:
141       digitalWrite(A, LOW);
142       digitalWrite(B, LOW);
143       digitalWrite(C, LOW);
144       digitalWrite(D, LOW);
145       digitalWrite(E, LOW);
146       digitalWrite(F, LOW);
147       digitalWrite(G, LOW);
148       break;
149       case 9:
150       digitalWrite(A, LOW);
151       digitalWrite(B, LOW);
152       digitalWrite(C, LOW);
153       digitalWrite(D, LOW);
154       digitalWrite(E, HIGH);
155       digitalWrite(F, LOW);
156       digitalWrite(G, LOW);
157       break;
158     }
159  }
160
```


Any
Questions?


GLYPH 128PX
FLAT 128PX
LINE 128PX
LINE 64PX
LINE 32PX
LINE 16PX

PEMBEDS

END OF SLIDES

THANK YOU!