

Display Numbers 0–9 on a 7-Segment Display using ESP32

🎯 Objective:

To use an **ESP32 microcontroller** to display digits **0–9** on a **single-digit 7-segment LED display**, with each digit showing for 1 second.

🔧 Required Components:

- ESP32 board
- 1x Common Cathode 7-Segment Display
- 7x 220Ω resistors
- Breadboard & jumper wires

Pin Configuration:

Each segment of the 7-segment display is connected to specific **GPIO pins** on the ESP32.

Segment	Display Segment	ESP32 Pin
A	Top horizontal	GPIO 13
B	Upper right	GPIO 12
C	Lower right	GPIO 14
D	Bottom	GPIO 27
E	Lower left	GPIO 26
F	Upper left	GPIO 25
G	Middle horizontal	GPIO 33

Important: This assumes you're using a **common cathode** 7-segment display (where the common pin is connected to **GND**).

🧠 How a 7-Segment Display Works:

Each segment (A–G) is an LED. To display a number:

- You **light up** specific segments.
- For example, to display **0**, light up A, B, C, D, E, and F.
- Segment G (middle) stays **off**.

Code Explanation

```
#include <Arduino.h>

// Segment pins: A, B, C, D, E, F, G

const int segmentPins[7] = {13, 12, 14, 27, 26, 25, 33};

// Segment ON/OFF map for digits 0-9: A, B, C, D, E, F, G

const byte digitSegments[10][7] = {

    {1, 1, 1, 1, 1, 1, 0}, // 0

    {0, 1, 1, 0, 0, 0, 0}, // 1

    {1, 1, 0, 1, 1, 0, 1}, // 2

    {1, 1, 1, 1, 0, 0, 1}, // 3

    {0, 1, 1, 0, 0, 1, 1}, // 4

    {1, 0, 1, 1, 0, 1, 1}, // 5

    {1, 0, 1, 1, 1, 1, 1}, // 6

    {1, 1, 1, 0, 0, 0, 0}, // 7

    {1, 1, 1, 1, 1, 1, 1}, // 8

    {1, 1, 1, 1, 0, 1, 1}  // 9

};

void setup() {

    for (int i = 0; i < 7; i++) {

        pinMode(segmentPins[i], OUTPUT);

    }

}
```

```

}

void displayDigit(int digit) {

    for (int i = 0; i < 7; i++) {

        digitalWrite(segmentPins[i], digitSegments[digit][i]);

    }

}

void loop() {

    for (int i = 0; i <= 9; i++) {

        displayDigit(i);

        delay(1000);

    }

}

```

(Line by Line)

```
#include <Arduino.h>
```

- This includes all the basic functions for running code on ESP32 in the Arduino environment.

```
const int segmentPins[7] = {13, 12, 14, 27, 26, 25, 33};
```

- An **array** holding ESP32 GPIO numbers for segments A through G.
- Index **0** = Segment A, **1** = B, ..., **6** = G.

```
const byte digitSegments[10][7] = {
    {1, 1, 1, 1, 1, 1, 0}, // 0
    {0, 1, 1, 0, 0, 0, 0}, // 1
    {1, 1, 0, 1, 1, 0, 1}, // 2
    {1, 1, 1, 1, 0, 0, 1}, // 3
    {0, 1, 1, 0, 0, 1, 1}, // 4
    {1, 0, 1, 1, 0, 1, 1}, // 5
    {1, 0, 1, 1, 1, 1, 1}, // 6
    {1, 1, 1, 0, 0, 0, 0}, // 7
    {1, 1, 1, 1, 1, 1, 1}, // 8

```

```
{1, 1, 1, 1, 0, 1, 1} // 9
};
```

- This **2D array** tells us **which segments to turn ON or OFF** for each digit 0–9.
- For each digit:
 - **1** = turn ON that segment
 - **0** = turn OFF that segment

Example:

For digit **2** → {1, 1, 0, 1, 1, 0, 1}

- Segments A, B, D, E, G = ON
- Segments C, F = OFF

setup() Function:

```
void setup() {
  for (int i = 0; i < 7; i++) {
    pinMode(segmentPins[i], OUTPUT);
  }
}
```

- Set all 7 segment pins (A–G) as **output** pins.
- This prepares ESP32 to send HIGH/LOW signals to each segment.

displayDigit() Function:

```
void displayDigit(int digit) {
  for (int i = 0; i < 7; i++) {
    digitalWrite(segmentPins[i], digitSegments[digit][i]);
  }
}
```

- This function **lights up the correct segments** for a given digit:
 - Loop through all 7 segments
 - For each one, set the **digital pin HIGH or LOW** depending on the **digitSegments** array
 - Example: If **digit = 3**, it will light up segments A, B, C, D, G

loop() Function:

```
void loop() {
  for (int i = 0; i <= 9; i++) {
    displayDigit(i);
    delay(1000);
  }
}
```

- Continuously loops through digits 0 to 9:
 - Calls `displayDigit(i)` to display the number
 - Waits for **1 second** before showing the next digit

Operation Summary:

1. ESP32 initializes all the segment pins.
2. Starts displaying digits 0–9 one by one.
3. For each digit:
 - Looks up the ON/OFF segment pattern in `digitSegments`.
 - Turns on the appropriate pins using `digitalWrite`.
 - Waits 1 second.
4. Repeats forever.

📌 Important Notes:

- The display must be **common cathode**.
- Resistors (220–330Ω) should be placed **in series with each segment** to avoid damaging the LEDs.
- If the display does **not light up**, check:
 - Pin numbers
 - GND/Common Cathode is properly connected
 - Segment connections match the mapping

🖼️ Visual Example (Digit 2):

To show **2**, these segments light up:

```

--A--
|    |
F      B
|    |
--G--
|    |
E      C
|    |
--D--

```

Segment pattern = {1, 1, 0, 1, 1, 0, 1}

Segments A, B, D, E, G = ON

Segments C, F = OFF

