TM1637 4-Digit Display

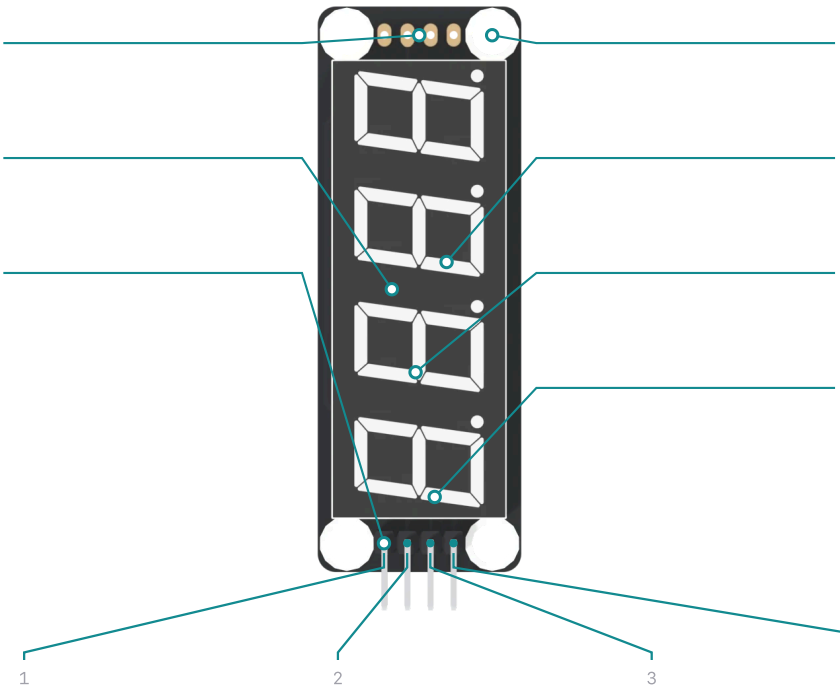
Four digits on two wires that are not I2C: no address, one DATA pin per display.

VCC goes to your board's logic supply

The 10 kΩ pull-ups on CLOCK and DATA end at VCC, so both wires rest at whatever VCC is. **5V on an Uno, 3V3 on an ESP32, ESP32-S3 or Pico.** 5V on VCC beside a 3.3 V board puts 5 V on two of its pins.

Pins, and which to wire

Drawn turned, the header along the bottom: **GND, VCC, CLOCK, DATA**. No pad is square; the back prints the names at both ends. All four are wired.



Spare pads, same four
Not a second DATA: a display wired here mirrors.

The colon
Bit 7 of digit 2: 0b01000000. The model shows points.

GND is hole 1
No pad is square. Read the names on the back.

4.6 mm holes
56 × 16 mm apart, for the base plate.

TM1637 underneath
It lights the digits: a number sent once stays.

100 nF and 10 µF
Across the supply, under the display.

Two 10 kΩ pull-ups
CLOCK and DATA to VCC: the wires rest at VCC.

Front	GND	VCC	CLOCK	DATA
Back	GND	VCC	clock	data
Wire it	GND	VCC	CLK	DIO

The colon is the second digit's bit

The dots mask counts digits from the left: 0b10000000 is the first digit, 0b01000000 the second. On a clock panel only the second is wired, to the colon; 0b10000000 lights nothing. And a 0 with leading zeros off drops the dots in library 1.2.0: pass true.

Three numbers to drive it by

Signal wires

2 pins a display

CLOCK is shared along a row; DATA is one pin per display. Six displays take seven pins.

Brightness

8 steps

1, 2, 4, 10, 11, 12, 13, 14 sixteenths. Sent only with the next write.

One update

21 ms blocked

The library waits 100 µs at each step. Its third argument shortens it.

Brightness is not even: levels 0 to 2 are the dimming, 3 to 7 differ by a few sixteenths. And the chip holds its digits by itself, so a number on the display does not prove the sketch is still running.

Numbers from Titan Micro's TM1637 datasheet V2.4 and the TM1637 library by Avishay Orpaz at 1.2.0. The update time is counted off the library's code. The datasheet specifies the chip at 4.5 to 5.5 V; blue and white panels want 5 V. Not measured on a bench.

Show 1234, then count

Arduino IDE · Board: your board · Library: "TM1637" by Avishay Orpaz · ESP32-S3: USB CDC On Boot: Enabled

```
#include <TM1637Display.h>

#define CLOCK_PIN 2    // Uno D2.  ESP32: 18.  ESP32-S3: 4.
#define DATA_PIN  3    // Uno D3.  ESP32: 19.  ESP32-S3: 5.

// CLOCK first, then DATA. The wrong way round is a dark display.
TM1637Display display(CLOCK_PIN, DATA_PIN);

void setup() {
  display.setBrightness(4);     // 0 to 7, sent with the next write
  display.showNumberDec(1234);  // stays there with no further help
  delay(2000);
}

void loop() {
  for (int i = 0; i <= 60; i++) {
    display.showNumberDec(i);   // leading zeros off: 7 is "   7"
    delay(500);
  }
}
```

`1234 → 0, 1, 2 … 60`

1234 for two seconds, then a count twice a second. Dark means CLOCK and DATA are swapped, which harms nothing.

Wiring

	FRONT	UNO	ESP32	S3	PICO
1	GND	GND	GND	GND	GND
2	VCC	5V	3V3	3V3	3V3(OUT)
3	CLOCK	D2	GPIO 18	GPIO 4	GP2
4	DATA	D3	GPIO 19	GPIO 5	GP3

Hole 1 is GND, the left-hand pin with the header along the bottom. No pad is square; the back prints GND, VCC, clock, data. VCC is the board's logic supply, never 5V beside a 3.3 V board. Any two digital pins do for CLOCK and DATA.

In MicroPython

On an ESP32, ESP32-S3 or Pico: save `tm1637.py` from the block's MicroPython folder to the board, then `from tm1637 import TM1637`, `d = TM1637(18, 19)` (CLOCK, DATA; S3: 4, 5; Pico: 2, 3) and `d.number(1234)`. `d.number(1234, 0b01000000, True)` lights the colon.

When it does not work

The display stays dark

CLOCK and DATA swapped is the usual answer, and harmless. Then check GND, VCC and the two pin numbers.

Two displays show the same number

They share a DATA line, often through the spare pads. Give each display its own DATA pin.

The blue panel is dim

Blue and white need about 3.1 V; 3.3 V leaves nothing over. Use 5 V and a level converter.

The colon will not light

A digit panel has no colon. Else the mask is 0b10000000, the first digit, or it is 0 with leading zeros off.

setBrightness does nothing

Level and on/off go out with the next write. Call it, then show the number again.

It has no I2C address

Correct: the protocol has none. Use two plain digital pins, never the I2C bus.

Specifications

Driver	TM1637, SOP-20, under the display	**Digits**	4, 0.56 inch (14.2 mm)
Separator	Colon on the TK51 and clock set	**Wires**	GND, VCC, CLOCK, DATA
Supply	5 V for all; 3.3 V usually for red to green	**Pull-ups**	10 kΩ, CLOCK and DATA to VCC
Clock	Up to 500 kHz; library about 3 kHz	**Board**	62.3 × 22.4 mm, V3