



Arduino TM1638 8-Bit LED Display with 16 Keys Module

The TM1638 is a popular integrated circuit that combines an 8-digit LED display, 8 individual LEDs, and 16 buttons (8 dual-function keys) in a single module. It communicates via a simple serial interface, making it easy to use with Arduino.

Key Features

- 8-digit 7-segment LED display
- 8 individual red/green LEDs (color depends on module version)
- 16-button keypad (8 keys with SHIFT function)
- Serial interface (STB, CLK, DIO)
- Built-in display driver and key scanning
- 5V operation compatible with Arduino

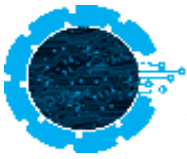
Wiring Connection

Typical connection to Arduino:

```
TM1638    ->  Arduino
VCC       ->  5V
GND       ->  GND
STB (CS)  ->  Digital pin (e.g., 4)
CLK       ->  Digital pin (e.g., 3)
DIO       ->  Digital pin (e.g., 2)
```

Basic Arduino Code Example

```
#include <TM1638.h>
```



```
// Define pins (STB, CLK, DIO)
TM1638 module(4, 3, 2);

void setup() {
  // Set display brightness (0-7)
  module.setupDisplay(true, 7);

  // Clear display
  module.clearDisplay();
}

void loop() {
  // Display a number
  module.setDisplayToDecNumber(12345678, 0, false);

  // Light up LEDs based on binary pattern (each bit represents one LED)
  module.setLEDs(0b10101010);

  // Check buttons
  byte buttons = module.getButtons();

  // If any button is pressed
  if(buttons) {
    // Display button pattern on the display
    module.setDisplayToBinNumber(buttons, 0b10000000);
  }

  delay(100);
}
```

Library Installation

You'll need the TM1638 library. You can install it:

1. Through Arduino IDE: Sketch > Include Library > Manage Libraries > Search for "TM1638"
2. Or download from: <https://github.com/rjbatista/tm1638-library>



Common Functions

- `setupDisplay(active, intensity)`: Initialize display (active=true/false, intensity=0-7)
- `clearDisplay()`: Clear all digits and LEDs
- `setDisplayToDecNumber(number, dots, leadingZeros)`: Display decimal number
- `setDisplayToHexNumber(number, dots, leadingZeros)`: Display hexadecimal number
- `setLEDs(leds)`: Control LEDs (each bit represents one LED)
- `getButtons()`: Read button states (returns byte where each bit represents a button)

Advanced Usage

For more complex applications, you can:

- Display scrolling text
- Create custom character patterns
- Implement button press/release detection
- Use the LEDs as status indicators
- Combine display and button functions for menu systems

