Erriez TM1638 library for Arduino

1.0.0

Generated by Doxygen 1.8.11

Contents

Chapter 1

Optimized TM1637 library for Arduino

This is a 2-pin serial TM1637 chip library for Arduino, optimized for size and speed. It supports a combined LED driver controller and key-scan interface to detect one key press.

Chip features

- Power CMOS process
- Display mode (8 segments × 6 digits), support common anode LED output
- Key scan (8 x 2-bit), enhanced anti-jamming button recognition circuit
- Brightness adjustment circuit (adjustable duty cycle 8)
- Two-wire serial interface (CLK, DIO)
- Oscillation mode: Built-in RC oscillator
- Built-in power-on reset circuit
- Built-in automatic blanking circuit
- Package: DIP20 / SOP20

Hardware

Connect power and 2 data pins to an Arduino board DIGITAL pins:

- VDD (Power 3.3V - 5V)
- GND (Ground)
- CLK (Clock)
- DIO (Bi-directional data input/output)

The following TM1637 pins should be connected to LED's and buttons in a matrix:

- K1∼K2 (Key-scan data input to read one key press after each other)
- SEG/GRID (Output for LED matrix)

Pins

Pin	TM1637	Arduino UNO / Nano / Micro / Pro Micro / Leonardo / Mega2560	WeMos D1 & R2 / Node MCU	WeMos LOLIN32
1	VCC	5V (or 3.3V)	3V3	3V3
2	GND	GND	GND	GND
3	CLK	2 (DIGITAL pin)	D2	0
4	DIO	3 (DIGITAL pin)	D3	4

- Check maximum regulator / diode current to prevent a burnout when using lots of LED's. Some boards can provide only 100mA, others 800mA max.

Two-wire serial interface

The TM1637 communicates with a MCU serial by using two wires:

- DIO (bi-directional input/output pin)
- SCL (Clock pin)

Note: The serial interface is not compatible with I2C or TWI, because no device address with read/write bit is used.

Example

Arduino IDE | Examples | Erriez TM1637 button and LED driver:

`TM1637`

Documentation

- Doxygen online HTML
- Doxygen PDF
- TM1637 Datasheet (Chinese)

Usage

Initialization

```
1 {c++}
2 // Include TM1637 library
3 #include "TM1637.h"
4
5 // Connect display pins to the Arduino DIGITAL pins
6 #define TM1637_CLK_PIN   2
7 #define TM1637_DIO_PIN   3
8
9 // Create tm1637 object
10 TM1637 tm1637(TM1637_CLK_PIN, TM1637_DIO_PIN);
11
12 void setup()
13 {
14     // Initialize TM1637
15     tm1637.begin();
16 }
```

Display on/off

```
1 {c++}
2 // Turn display off
3 tm1637.displayOff();
4
5 // Turn display on
6 tm1637.displayOn();
```

Turn all LED's off

```
1 {c++}
2 // Turn all LED's off
3 tm1637.clear();
```

Get keys

```
1 {c++}
2 // Get 8-bit key-scan
3 uint8_t keys = tm1637.getKeys();
```

Write Byte to display register

```
1 {c++}
2 // Write segment LED's to the first display registers 0x00..0x0F with value 0x00..0xff to
3 // display numbers and characters. Just an example which depends on the hardware:
4 tm1637.writeData(0x01, 0x01);
```

Write buffer to display registers

```
1 {c++}
2 // Creat buffer with LED's
3 uint8_t buf[] = { 0b10000110, 0b00111111, 0b00111111, 0b00111111, 0b00111111, 0b00111111};
4
5 // Write buffer to TM1637
6 tm1637.writeData(0x00, buf, sizeof(buf));
```

Optimized timing

The library uses optimized pin control for AVR targets. Other targets uses the default digitalRead() and digitalWrite() pin control functions.

Output `Benchmark` example:

Board	CLK	Read keys	Write Byte	Write 16 Bytes buffer	Clear display
Pro Mini 8MHz	84kHz	352us	344us	1080us	1072us
UNO 16MHz	170kHz	156us	152us	496us	480us
WeMos D1 & R2 80MHz	205kHz	261us	137us	396us	396us
WeMos D1 & R2 160MHz	300kHz	233us	96us	275us	271us

Arduino Pro-Mini 8MHz

Arduino UNO 16MHz

WeMos D1 & R2 80MHz

WeMos D1 & R2 160MHz

Library dependencies

- The `Benchmark` example uses `Erriez Timestamp` library.

Library installation

Please refer to the `Wiki` page.

Other Arduino Libraries and Sketches from Erriez

- `Erriez Libraries and Sketches`

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

File Index

3.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 4

Class Documentation

4.1 TM1637 Class Reference

TM1637 class.

```
#include <TM1637.h>
```

Public Member Functions

- TM1637 (uint8_t clkPin, uint8_t dioPin, bool displayOn=true, uint8_t brightness=5)

  TM1637 constructor.
- virtual void begin ()

  Initialize TM1637 controller.
- virtual void end ()

  Release TM1637 pins.
- virtual void displayOn ()

  Turn Display on.
- virtual void displayOff ()

  Turn display off.
- virtual void setBrightness (uint8_t brightness)

  Set brightness LED's.
- virtual void clear ()

  Turn all LED's off.
- virtual void writeData (uint8_t address, uint8_t data)

  Write display register.
- virtual void writeData (uint8_t address, const uint8_t *buf, uint8_t len)

  Write buffer to multiple display registers.
- virtual uint8_t getKeys ()

  Get key states.

Protected Member Functions

- virtual void writeDisplayControl ()
  Write display control register.
- virtual void writeCommand (uint8_t cmd)
  Write command to TM1637.
- virtual void start ()
  Generate start condition.
- virtual void stop ()
  Generate stop condition.
- virtual void writeByte (uint8_t data)
  Write byte to TM1637.
- virtual uint8_t readByte ()
  Read byte from TM1637.

Protected Attributes

- uint8_t _clkPin
  Clock pin.
- uint8_t _dioPin
  Data pin.
- bool _displayOn
  Display on and off status for display control register.
- uint8_t _brightness
  Display brightness for display control register.

4.1.1 Detailed Description

TM1637 class.

Definition at line 141 of file TM1637.h.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 TM1637::TM1637 (uint8_t *clkPin*, uint8_t *dioPin*, bool *displayOn* = `true`, uint8_t *brightness* = `5`)

TM1637 constructor.

Constructor with pin arguments: C-D (Clock, Data)

Parameters

clkPin	TM1637 CLK pin.
dioPin	TM1637 DIO pin.
displayOn	true: Turn display on (default) false: Turn display off
brightness	Display brightness value 0..7

Definition at line 49 of file TM1637.cpp.

4.1.3 Member Function Documentation

4.1.3.1 uint8_t TM1637::getKeys ( ) [virtual]

Get key states.

Returns

0xFF: All keys up 0x00..0x0F: Key 0..15 down

Definition at line 191 of file TM1637.cpp.

4.1.3.2 uint8_t TM1637::readByte ( ) [protected], [virtual]

Read byte from TM1637.

Returns

8-bit value.

Definition at line 316 of file TM1637.cpp.

4.1.3.3 void TM1637::setBrightness (uint8_t *brightness*) [virtual]

Set brightness LED's.

Parameters

brightness	Display brightness value 0..7

Definition at line 124 of file TM1637.cpp.

4.1.3.4 void TM1637::writeByte (uint8_t *data*) [protected], [virtual]

Write byte to TM1637.

Parameters

data	8-bit value.

Definition at line 276 of file TM1637.cpp.

4.1.3.5 void TM1637::writeData (uint8_t *address*, uint8_t *data*) [virtual]

Write display register.

Parameters

address	Display address 0x00..0x05
data	Value 0x00..0xFF

Definition at line 151 of file TM1637.cpp.

4.1.3.6 void TM1637::writeData (uint8_t *address*, const uint8_t * *buf*, uint8_t *len*) [virtual]

Write buffer to multiple display registers.

Write buffer to TM1638 with auto address increment

Parameters

address	Display address 0x00..0x05
buf	Buffer
len	Buffer length

Definition at line 172 of file TM1637.cpp.

The documentation for this class was generated from the following files:

- TM1637.h
- TM1637.cpp

Chapter 5

File Documentation

5.1 TM1637.cpp File Reference

TM1637 library for Arduino.

```
#include "TM1637.h"
```

5.1.1 Detailed Description

TM1637 library for Arduino.

Source: https://github.com/Erriez/ErriezTM1637 Documentation: https://erriez.↩
github.io/ErriezTM1637

5.2 TM1637.h File Reference

TM1637 library for Arduino.

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

Classes

- class TM1637

  TM1637 class.

Macros

- #define TM1637_CMD_DATA 0x40

  Display data command.
- #define TM1637_CMD_CTRL 0x80

  Display control command.
- #define TM1637_CMD_ADDR 0xc0

  Display address command.
- #define TM1637_DATA_WRITE 0x00

  Write data.
- #define TM1637_DATA_READ_KEYS 0x02

  Read keys.
- #define TM1637_DATA_AUTO_INC_ADDR 0x00

  Auto increment address.
- #define TM1637_DATA_FIXED_ADDR 0x04

  Fixed address.
- #define TM1637_CTRL_PULSE_1_16 0x00

  Pulse width 1/16.
- #define TM1637_CTRL_PULSE_2_16 0x01

  Pulse width 2/16.
- #define TM1637_CTRL_PULSE_4_16 0x02

  Pulse width 4/16.
- #define TM1637_CTRL_PULSE_10_16 0x03

  Pulse width 10/16.
- #define TM1637_CTRL_PULSE_11_16 0x04

  Pulse width 11/16.
- #define TM1637_CTRL_PULSE_12_16 0x05

  Pulse width 12/16.
- #define TM1637_CTRL_PULSE_13_16 0x06

  Pulse width 13/16.
- #define TM1637_CTRL_PULSE_14_16 0x07

  Pulse width 14/16.
- #define TM1637_CTRL_DISPLAY_OFF 0x00

  Display off.
- #define TM1637_CTRL_DISPLAY_ON 0x08

  Display on.
- #define TM1637_NUM_GRIDS 6

  Number of grid registers.
- #define TM1637_CLK_LOW() { digitalWrite(_clkPin, LOW); }

  CLK pin low.
- #define TM1637_CLK_HIGH() { digitalWrite(_clkPin, HIGH); }

  CLK pin high.
- #define TM1637_CLK_INPUT() { pinMode(_clkPin, INPUT); }

  CLK pin input.
- #define TM1637_CLK_OUTPUT() { pinMode(_clkPin, OUTPUT); }

  CLK pin output.
- #define TM1637_DIO_LOW() { digitalWrite(_dioPin, LOW); }

  DIO pin low.
- #define TM1637_DIO_HIGH() { digitalWrite(_dioPin, HIGH); }

  DIO pin high.
- #define TM1637_DIO_INPUT() { pinMode(_dioPin, INPUT); }

DIO pin input.

- #define TM1637_DIO_OUTPUT() { pinMode(_dioPin, OUTPUT); }

  DIO pin output.

- #define TM1637_DIO_READ() (digitalRead(_dioPin))

  DIO pin read.

- #define TM1637_PIN_DELAY()

  Delay between pin changes.

5.2.1 Detailed Description

TM1637 library for Arduino.

Source: https://github.com/Erriez/ErriezTM1637 Documentation: https://erriez.github.io/ErriezTM1637

```
Command / register definitions

MSB           LSB
 7 6 5 4 3 2 1 0
-----------------
 0 1 - - - - - -    Data command
 1 0 - - - - - -    Display control command
 1 1 - - - - - -    Address command


7.1 Data Command Set

MSB           LSB
 7 6 5 4 3 2 1 0
-----------------
 0 1 0 0 0 - 0 0    Write display data
 0 1 0 0 0 - 1 0    Read key scan data
 0 1 0 0 0 0 - -    Auto address increment
 0 1 0 0 0 1 - -    Fixed address


7.2 Address command set

MSB           LSB
 7 6 5 4 3 2 1 0
-----------------
 1 1 0 - A A A A    Address 0x00..0x0F


7.3 Display Control

MSB           LSB
 7 6 5 4 3 2 1 0
-----------------
 1 0 0 0 - 0 0 0    Set the pulse width of 1 / 16
 1 0 0 0 - 0 0 1    Set the pulse width of 2 / 16
 1 0 0 0 - 0 1 0    Set the pulse width of 4 / 16
 1 0 0 0 - 0 1 1    Set the pulse width of 10 / 16
 1 0 0 0 - 1 0 0    Set the pulse width of 11 / 16
 1 0 0 0 - 1 0 1    Set the pulse width of 12 / 16
 1 0 0 0 - 1 1 0    Set the pulse width of 13 / 16
 1 0 0 0 - 1 1 1    Set the pulse width of 14 / 16
 1 0 0 0 0 - - -    Display off
 1 0 0 0 1 - - -    Display on
```

Index