

TM1638 library for Arduino

1.1.0

Generated by Doxygen 1.8.11

Contents

1	Optimized TM1638 library for Arduino	1
2	Class Index	5
2.1	Class List	5
3	File Index	7
3.1	File List	7
4	Class Documentation	9
4.1	TM1638 Class Reference	9
4.1.1	Detailed Description	10
4.1.2	Constructor & Destructor Documentation	10
4.1.2.1	TM1638(uint8_t clkPin, uint8_t dioPin, uint8_t stbPin, bool displayOn=true, uint8_t brightness=5)	10
4.1.3	Member Function Documentation	11
4.1.3.1	getKeys()	11
4.1.3.2	readByte()	11
4.1.3.3	setBrightness(uint8_t brightness)	11
4.1.3.4	writeByte(uint8_t data)	11
4.1.3.5	writeData(uint8_t address, uint8_t data)	11
4.1.3.6	writeData(uint8_t address, const uint8_t *buf, uint8_t len)	12
4.1.3.7	writeDisplayControl()	12
5	File Documentation	13
5.1	TM1638.cpp File Reference	13
5.1.1	Detailed Description	13
5.2	TM1638.h File Reference	13
5.2.1	Detailed Description	15
5.2.2	Macro Definition Documentation	16
5.2.2.1	TM1638_CTRL_DISPLAY_ON	16
	Index	17

Chapter 1

Optimized TM1638 library for Arduino

This is a 3-pin serial [TM1638](#) chip library for Arduino, optimized for size and speed. It supports a combined LED driver controller and key-scan interface to detect multiple key presses at the same time.

Displaying numbers, characters and reading keys depends on the hardware wiring and is not part of this library. A fully operational example for a board with 8 7-segment displays, 8 dual color LED's and 8 buttons which uses this library is available here: [JY-LKM1638](#).

Hardware

Connect power and 3 data pins to an Arduino board DIGITAL pins:

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

The following [TM1638](#) pins should be connected to LED's and buttons in a matrix:

- K1~K3 (Key-scan data input to read multiple key presses at the same time)
- SEG/GRID (Output for LED matrix)

Pins

Pin	TM1638	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
5	STB0	4 (DIGITAL pin)	D4	5

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

Example

- Examples | Erriez [TM1638](#) | [Example](#)

Documentation

- [Doxygen online HTML](#)
- [Doxygen PDF](#)
- [TM1638 Datasheet](#)

Usage

Initialization

```
1 {c++}
2 // Include TM1638 library
3 #include "TM1638.h"
4
5 // Connect display pins to the Arduino DIGITAL pins
6 #define TM1638_CLK_PIN 2
7 #define TM1638_DIO_PIN 3
8 #define TM1638_STB_PIN 4
9
10 // Create tml638 object
11 TM1638 tml638(TM1638_CLK_PIN, TM1638_DIO_PIN, TM1638_STB_PIN);
12
13 void setup()
14 {
15     // Initialize TM1638
16     tml638.begin();
17 }
```

Display on/off

```
1 {c++}
2 // Turn display off
3 tml638.displayOff();
4
5 // Turn display on
6 tml638.displayOn();
```

Turn all LED's off

```
1 {c++}
2 // Turn all LED's off
3 tml638.clear();
```

Get keys

```
1 {c++}
2 // Get 32-bit key-scan
3 uint32_t keys = tml638.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 tml638.writeData(0x01, 0x01);
```

Write buffer to display registers

```
1 {c++}
2 // Create buffer with LED's
3 uint8_t buf[] = { 0b10000110, 0b00111111, 0b00111111, 0b00111111, 0b00111111, 0b00111111};
4
5 // Write buffer to TM1638
6 tml638.writeData(0x00, buf, sizeof(buf));
```

Small footprint

Measured with Arduino IDE v1.8.5 without any other peripherals, calling all library functions once:

Board	MCU flash size	MCU RAM size	TM1638 library flash	TM1638 library RAM
Arduino UNO	32kB	2048kB	1840 Bytes	40 Bytes

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	65kHz	736us	312us	2448us	2224us
UNO 16MHz	125kHz	340us	152us	1192us	1176us
WeMos D1 & R2 80MHz	200kHz	284us	116us	683us	682us
WeMos D1 & R2 160MHz	300kHz	223us	66us	474us	469us

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:

TM1638	
TM1638 class	9

Chapter 3

File Index

3.1 File List

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

TM1638.cpp	
TM1638 library for Arduino	13
TM1638.h	
TM1638 library for Arduino	13

Chapter 4

Class Documentation

4.1 TM1638 Class Reference

TM1638 class.

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

Public Member Functions

- **TM1638** (uint8_t clkPin, uint8_t dioPin, uint8_t stbPin, bool **displayOn**=true, uint8_t brightness=5)
TM1638 constructor.
- virtual void **begin** ()
Initialize TM1638 controller.
- virtual void **end** ()
Disable 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 uint32_t **getKeys** ()
Get key states.

Protected Member Functions

- virtual void [writeDisplayControl](#) ()
Write display control register.
- virtual void [writeCommand](#) (uint8_t cmd)
Write command to [TM1638](#).
- virtual void [writeByte](#) (uint8_t data)
Write byte to [TM1638](#).
- virtual uint8_t [readByte](#) ()
Read byte from [TM1638](#).

Protected Attributes

- uint8_t [_clkPin](#)
Clock pin.
- uint8_t [_dioPin](#)
Data pin.
- uint8_t [_stbPin](#)
Enable 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

[TM1638](#) class.

Definition at line 154 of file [TM1638.h](#).

4.1.2 Constructor & Destructor Documentation

4.1.2.1 [TM1638::TM1638](#) (uint8_t *clkPin*, uint8_t *dioPin*, uint8_t *stbPin*, bool *displayOn* = true, uint8_t *brightness* = 5)

[TM1638](#) constructor.

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

Parameters

<i>clkPin</i>	TM1638 CLK pin.
<i>dioPin</i>	TM1638 DIO pin.
<i>stbPin</i>	TM1638 STB pin.

Definition at line 43 of file [TM1638.cpp](#).

4.1.3 Member Function Documentation

4.1.3.1 `uint32_t TM1638::getKeys ()` [virtual]

Get key states.

Returns

One or more buttons. One bit per button.

Definition at line 190 of file TM1638.cpp.

4.1.3.2 `uint8_t TM1638::readByte ()` [protected],[virtual]

Read byte from [TM1638](#).

Returns

8-bit value.

Definition at line 251 of file TM1638.cpp.

4.1.3.3 `void TM1638::setBrightness (uint8_t brightness)` [virtual]

Set brightness LED's.

Parameters

<i>brightness</i>	Display brightness value 0..7
-------------------	-------------------------------

Definition at line 124 of file TM1638.cpp.

4.1.3.4 `void TM1638::writeByte (uint8_t data)` [protected],[virtual]

Write byte to [TM1638](#).

Parameters

<i>data</i>	8-bit value.
-------------	--------------

Definition at line 232 of file TM1638.cpp.

4.1.3.5 `void TM1638::writeData (uint8_t address, uint8_t data)` [virtual]

Write display register.

Parameters

<i>address</i>	Display address 0x00..0x0F
<i>data</i>	Value 0x00..0xFF

Definition at line 151 of file TM1638.cpp.

4.1.3.6 `void TM1638::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

<i>address</i>	Display address 0x00..0x0F
<i>buf</i>	Buffer
<i>len</i>	Buffer length

Definition at line 173 of file TM1638.cpp.

4.1.3.7 `void TM1638::writeDisplayControl ()` `[protected]`, `[virtual]`

Write display control register.

Set brightness and display on/off

Definition at line 211 of file TM1638.cpp.

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

- [TM1638.h](#)
- [TM1638.cpp](#)

Chapter 5

File Documentation

5.1 TM1638.cpp File Reference

[TM1638](#) library for Arduino.

```
#include "TM1638.h"
```

5.1.1 Detailed Description

[TM1638](#) library for Arduino.

Source: <https://github.com/Erriez/ErriezTM1638> Documentation: <https://erriez.github.io/ErriezTM1638>

5.2 TM1638.h File Reference

[TM1638](#) library for Arduino.

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

Classes

- class [TM1638](#)
[TM1638](#) class.

Macros

- `#define TM1638_CMD_DATA 0x40`
Display data command.
- `#define TM1638_CMD_CTRL 0x80`
Display control command.
- `#define TM1638_CMD_ADDR 0xc0`
Display address command.
- `#define TM1638_DATA_WRITE 0x00`
Write data.
- `#define TM1638_DATA_READ_KEYS 0x02`
Read keys.
- `#define TM1638_DATA_AUTO_INC_ADDR 0x00`
Auto increment address.
- `#define TM1638_DATA_FIXED_ADDR 0x04`
Fixed address.
- `#define TM1638_CTRL_PULSE_1_16 0x00`
Pulse width 1/16.
- `#define TM1638_CTRL_PULSE_2_16 0x01`
Pulse width 2/16.
- `#define TM1638_CTRL_PULSE_4_16 0x02`
Pulse width 4/16.
- `#define TM1638_CTRL_PULSE_10_16 0x03`
Pulse width 10/16.
- `#define TM1638_CTRL_PULSE_11_16 0x04`
Pulse width 11/16.
- `#define TM1638_CTRL_PULSE_12_16 0x05`
Pulse width 12/16.
- `#define TM1638_CTRL_PULSE_13_16 0x06`
Pulse width 13/16.
- `#define TM1638_CTRL_PULSE_14_16 0x07`
Pulse width 14/16.
- `#define TM1638_CTRL_DISPLAY_OFF 0x00`
Display off.
- `#define TM1638_CTRL_DISPLAY_ON 0x08`
Display on.
- `#define TM1638_NUM_GRIDS 16`
Number of grid registers.
- `#define TM1638_CLK_LOW() { digitalWrite(_clkPin, LOW); }`
CLK pin low.
- `#define TM1638_CLK_HIGH() { digitalWrite(_clkPin, HIGH); }`
CLK pin high.
- `#define TM1638_CLK_INPUT() { pinMode(_clkPin, INPUT); }`
CLK pin input.
- `#define TM1638_CLK_OUTPUT() { pinMode(_clkPin, OUTPUT); }`
CLK pin output.
- `#define TM1638_DIO_LOW() { digitalWrite(_dioPin, LOW); }`
DIO pin low.
- `#define TM1638_DIO_HIGH() { digitalWrite(_dioPin, HIGH); }`
DIO pin high.
- `#define TM1638_DIO_INPUT() { pinMode(_dioPin, INPUT); }`

- DIO pin input.*
- `#define TM1638_DIO_OUTPUT() { pinMode(_dioPin, OUTPUT); }`
- DIO pin output.*
- `#define TM1638_DIO_READ() (digitalRead(_dioPin))`
- DIO pin read.*
- `#define TM1638_STB_LOW() { digitalWrite(_stbPin, LOW); }`
- STB pin low.*
- `#define TM1638_STB_HIGH() { digitalWrite(_stbPin, HIGH); }`
- STB pin high.*
- `#define TM1638_STB_INPUT() { pinMode(_stbPin, INPUT); }`
- STB pin input.*
- `#define TM1638_STB_OUTPUT() { pinMode(_stbPin, OUTPUT); }`
- STB pin output.*
- `#define TM1638_PIN_DELAY()`
- Delay between pin changes.*

5.2.1 Detailed Description

TM1638 library for Arduino.

Source: <https://github.com/Erriez/ErriezTM1638> Documentation: <https://erriez.github.io/ErriezTM1638>

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

5.2.2 Macro Definition Documentation

5.2.2.1 `#define TM1638_CTRL_DISPLAY_ON 0x08`

Display on.

Pin defines

Definition at line 105 of file TM1638.h.

Index

- getKeys
 - TM1638, [11](#)
- readByte
 - TM1638, [11](#)
- setBrightness
 - TM1638, [11](#)
- TM1638, [9](#)
 - getKeys, [11](#)
 - readByte, [11](#)
 - setBrightness, [11](#)
 - TM1638, [10](#)
 - writeByte, [11](#)
 - writeData, [11](#), [12](#)
 - writeDisplayControl, [12](#)
- TM1638.cpp, [13](#)
- TM1638.h, [13](#)
 - TM1638_CTRL_DISPLAY_ON, [16](#)
- TM1638_CTRL_DISPLAY_ON
 - TM1638.h, [16](#)
- writeByte
 - TM1638, [11](#)
- writeData
 - TM1638, [11](#), [12](#)
- writeDisplayControl
 - TM1638, [12](#)