

RobotDyn 4-digit display library for Arduino
1.0.0

Generated by Doxygen 1.8.11

Contents

1	RobotDyn 4-digit LED display with TM1637 library for Arduino.	1
2	Hierarchical Index	5
2.1	Class Hierarchy	5
3	Class Index	7
3.1	Class List	7
4	File Index	9
4.1	File List	9
5	Class Documentation	11
5.1	RobotDyn4DigitDisplay Class Reference	11
5.1.1	Detailed Description	12
5.1.2	Constructor & Destructor Documentation	12
5.1.2.1	RobotDyn4DigitDisplay(uint8_t clkPin, uint8_t dioPin, bool displayOn=true, uint8_t brightness=5)	12
5.1.3	Member Function Documentation	12
5.1.3.1	dec(int value, uint8_t pad=1)	12
5.1.3.2	digit(uint8_t digit, uint8_t value)	12
5.1.3.3	doubleDots(bool on)	13
5.1.3.4	hex(unsigned int value, uint8_t pad=4)	13
5.1.3.5	rawDigit(uint8_t digit, uint8_t value)	13
5.1.3.6	time(uint8_t hour, uint8_t minute, bool doubleDotsOn=true, bool padHours=true)	13
6	File Documentation	15
6.1	RobotDyn4DigitDisplay.cpp File Reference	15
6.1.1	Detailed Description	15
6.2	RobotDyn4DigitDisplay.h File Reference	15
6.2.1	Detailed Description	16
6.2.2	Macro Definition Documentation	16
6.2.2.1	SEGMENTS_MINUS	16
Index		17

Chapter 1

RobotDyn 4-digit LED display with TM1637 library for Arduino.

This is a RobotDyn 4-digit 7-segment LED display library for Arduino. The PCB contains a two wire **TM1637 LED / button** controller.

Note: This library uses the double-dot to display a time. The LED dots per segment are not wired and cannot be controlled.

Library features

- Set brightness (0..7)
- Set digit (0..3)
- Control all individual segments per digit
- Control double dots (on/off)
- Display time (hours:minutes)
- Display decimal value (-999..9999) with optional padding
- Display hexadecimal value (0...0xFFFF) with optional padding

Hardware

Connection display with Arduino

Display	Arduino UNO / Nano / Pro Mini / Leonardo / Mega2560 / ESP8266 / Lolin32
GND	GND
VCC	5V (or 3.3V)
CLK	Any DIGITAL pin
DIO	Any DIGITAL pin

Other MCU's may work, but are not tested.

Examples

Arduino IDE | Examples | Erriez RobotDyn 4-digit display:

- [7SegmentDisplayDemo](#)

Documentation

- [Online HTML](#)
- [Download PDF](#)

Usage

Initialization

```
1 {c++}
2 #include <RobotDyn4DigitDisplay.h>
3
4 // Connect display pins to the Arduino DIGITAL pins
5 #if defined(ARDUINO_ARCH_AVR)
6 #define TM1637_CLK_PIN      2
7 #define TM1637_DIO_PIN      3
8 #elif defined(ARDUINO_ESP8266_WEMOS_D1MINI) || defined(ESP8266_WEMOS_D1MINI) ||
   defined(ARDUINO_ESP8266_NODEMCU)
9 #define TM1637_CLK_PIN      D2
10 #define TM1637_DIO_PIN      D3
11 #elif defined(ARDUINO_LOLIN32)
12 #define TM1637_CLK_PIN      0
13 #define TM1637_DIO_PIN      4
14 #else
15 #error "May work, but not tested on this target"
16 #endif
17
18 // Create display object
19 RobotDyn4DigitDisplay display(TM1637_CLK_PIN, TM1637_DIO_PIN);
20
21 void setup()
22 {
23     // Initialize TM1637
24     display.begin();
25 }
```

Clear display

```
1 {c++}
2 // Clear display
3 display.clear();    // _ _ _ _
```

Set brightness

```
1 {c++}
2 // Set brightness
3 display.setBrightness(0); // Minimum
4 display.setBrightness(7); // Maximum
```

Display time

```
1 {c++}
2 // Display time
3 display.time(11, 59);    // 11 : 59
```

Control time double dot

```
1 {c++}
2 display.doubleDots(true);    // Turn double dot on
3 display.doubleDots(false);   // Turn double dot off
```

Display decimal value

```
1 {c++}
2 // Display decimal values
3 display.dec(-999);    // - 9 9 9
4 display.dec(-1);     // _ _ - 1
5 display.dec(0);      // _ _ _ 0
6 display.dec(1);      // _ _ _ 1
7 display.dec(123);    // _ 1 2 3
8 display.dec(9999);   // 9 9 9 9
9 display.dec(10000);  // - - - -
10
11 // Display decimal values with padding
12 display.dec(1);      // _ _ _ 1 (Default no padding)
13 display.dec(1, 2);   // _ _ 0 1 (2 digits padding)
14 display.dec(1, 3);   // _ 0 0 1 (3 digits padding)
15 display.dec(1, 4);   // 0 0 0 1 (4 digits padding)
16
17 display.dec(34, 3);  // _ 0 3 4 (2 digits padding)
```

Display hexadecimal value

```
1 {c++}
2 // Display hexadecimal values
3 display.dec(0x0000);    // 0 0 0 0
4 display.dec(0x1234);    // 1 2 3 4
5 display.dec(0xABCD);    // A B C D
6 display.dec(0xBEEF);    // B E E F
7
8 // Display hexadecimal values with padding
9 display.hex(0x0001);    // _ _ _ 1 (Default no padding)
10 display.hex(0x0001, 2); // _ _ 0 1 (2 digits padding)
11 display.hex(0x0001, 3); // _ 0 0 1 (3 digits padding)
12 display.hex(0x0001, 4); // 0 0 0 1 (4 digits padding)
13
14 display.hex(0x0034, 3); // _ 0 3 4 (2 digits padding)
```

Control individual digits

```
1 {c++}
2 // Display individual digits: 1 2 3 4
3 display.digit(0, 1);
4 display.digit(1, 2);
5 display.digit(2, 3);
6 display.digit(3, 4);
```

Special characters

```
1 {c++}
2 Control individual LED-segments (bit numbers):
3   - 0 -
4   |   |
5   5   1
6   |   |
7   - 6 -
8   |   |
9   4   2
10  |   |
11  - 3 - .7
12
13 // Display error: E r _
14 display.rawDigit(0, 0b01111001);
15 display.rawDigit(1, 0b01010000);
16 display.rawDigit(2, 0b01010000);
17 display.rawDigit(3, 0b00000000);
```

```
18
19 // Display H character: _ _ _ H
20 display.rawDigit(3, 0b01110110);
21
22 // Display negative temperature: - 1 ` C
23 display.rawDigit(0, SEGMENTS_MINUS);
24 display.digit(1, 1);
25 display.rawDigit(2, SEGMENTS_DEGREE);
26 display.rawDigit(3, SEGMENTS_CELSIUS);
27
28 // Display rect
29 display.rawDigit(0, 0b00111001);
30 display.rawDigit(1, 0b00001001);
31 display.rawDigit(2, 0b00001001);
32 display.rawDigit(3, 0b00001111);
```

Library dependencies

- [Erriez TM1637](#) library

Library installation

Please refer to the [Wiki](#) page.

Other Arduino Libraries and Sketches from Erriez

- [Erriez Libraries and Sketches](#)

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

TM1637	
RobotDyn4DigitDisplay	11

Chapter 3

Class Index

3.1 Class List

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

RobotDyn4DigitDisplay	
RobotDyn4DigitDisplay class	11

Chapter 4

File Index

4.1 File List

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

RobotDyn4DigitDisplay.cpp	
RobotDyn4DigitDisplay library for Arduino	15
RobotDyn4DigitDisplay.h	
RobotDyn4DigitDisplay library for Arduino	15

Chapter 5

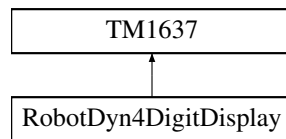
Class Documentation

5.1 RobotDyn4DigitDisplay Class Reference

[RobotDyn4DigitDisplay](#) class.

```
#include <RobotDyn4DigitDisplay.h>
```

Inheritance diagram for RobotDyn4DigitDisplay:



Public Member Functions

- [RobotDyn4DigitDisplay](#) (uint8_t clkPin, uint8_t dioPin, bool displayOn=true, uint8_t brightness=5)
Constructor RobotDyn 4-digit LED display.
- void [rawDigit](#) (uint8_t [digit](#), uint8_t value)
Display raw digit.
- void [digit](#) (uint8_t digit, uint8_t value)
Display a single digit.
- void [doubleDots](#) (bool on)
Display double time dots.
- void [time](#) (uint8_t hour, uint8_t minute, bool doubleDotsOn=true, bool padHours=true)
Display time.
- void [dec](#) (int value, uint8_t pad=1)
Display decimal value.
- void [hex](#) (unsigned int value, uint8_t pad=4)
Display hexadecimal value with optional padding.
- void [overflow](#) ()
Display overflow with four minus digits.

5.1.1 Detailed Description

[RobotDyn4DigitDisplay](#) class.

This class

Definition at line 52 of file RobotDyn4DigitDisplay.h.

5.1.2 Constructor & Destructor Documentation

5.1.2.1 RobotDyn4DigitDisplay::RobotDyn4DigitDisplay (uint8_t *clkPin*, uint8_t *dioPin*, bool *displayOn* = true, uint8_t *brightness* = 5)

Constructor RobotDyn 4-digit LED display.

Parameters

<i>clkPin</i>	Clock pins.
<i>dioPin</i>	Bi-directional data pin.
<i>displayOn</i>	Optional: Turn display on. Default: true
<i>brightness</i>	Optional: Set brightness 0..7 Default: 5.

Definition at line 84 of file RobotDyn4DigitDisplay.cpp.

5.1.3 Member Function Documentation

5.1.3.1 void RobotDyn4DigitDisplay::dec (int *value*, uint8_t *pad* = 1)

Display decimal value.

Parameters

<i>value</i>	0000..9999: Decimal value.
<i>pad</i>	0..4: Optional: Number of digits to pad with a zero. Default: 1.

Definition at line 170 of file RobotDyn4DigitDisplay.cpp.

5.1.3.2 void RobotDyn4DigitDisplay::digit (uint8_t *digit*, uint8_t *value*)

Display a single digit.

Parameters

<i>digit</i>	Digit number 0 (left digit) ... 3 (right digit)
<i>value</i>	Digit value 0..9 or 0x00..0x0F.

Definition at line 113 of file RobotDyn4DigitDisplay.cpp.

5.1.3.3 void RobotDyn4DigitDisplay::doubleDots (bool *on*)

Display double time dots.

Parameters

<i>on</i>	true: Turn double time dots on. false: Turn double time dots off.
-----------	----------------------------------------------------------------------

Definition at line 126 of file RobotDyn4DigitDisplay.cpp.

5.1.3.4 void RobotDyn4DigitDisplay::hex (unsigned int *value*, uint8_t *pad* = 4)

Display hexadecimal value with optional padding.

Parameters

<i>value</i>	0x0000..0xFFFF: Hexadecimal value
<i>pad</i>	0..4: Optional: Number of digits to pad with a zero. Default: 4.

Definition at line 224 of file RobotDyn4DigitDisplay.cpp.

5.1.3.5 void RobotDyn4DigitDisplay::rawDigit (uint8_t *digit*, uint8_t *value*)

Display raw digit.

Parameters

<i>digit</i>	Digit number 0 (left digit) .. 3 (right digit)
<i>value</i>	LED segments

Definition at line 98 of file RobotDyn4DigitDisplay.cpp.

5.1.3.6 void RobotDyn4DigitDisplay::time (uint8_t *hour*, uint8_t *minute*, bool *doubleDotsOn* = true, bool *padHours* = true)

Display time.

Parameters

<i>hour</i>	0..59: Hours
<i>minute</i>	0..59: Minutes
<i>doubleDotsOn</i>	true: Display double time dots. (Default) false: Turn double time dots off.
<i>padHours</i>	true: Display first digit as 0 when hours < 10. false: Turn first digit off when hours < 10.

Definition at line 149 of file RobotDyn4DigitDisplay.cpp.

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

- [RobotDyn4DigitDisplay.h](#)
- [RobotDyn4DigitDisplay.cpp](#)

Chapter 6

File Documentation

6.1 RobotDyn4DigitDisplay.cpp File Reference

[RobotDyn4DigitDisplay](#) library for Arduino.

```
#include <pgmspace.h>
#include "RobotDyn4DigitDisplay.h"
```

6.1.1 Detailed Description

[RobotDyn4DigitDisplay](#) library for Arduino.

Source: <https://github.com/Erriez/ErriezRobotDyn4DigitDisplay> Documentation↔
: <https://erriez.github.io/ErriezRobotDyn4DigitDisplay>

6.2 RobotDyn4DigitDisplay.h File Reference

[RobotDyn4DigitDisplay](#) library for Arduino.

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

Classes

- class [RobotDyn4DigitDisplay](#)
[RobotDyn4DigitDisplay](#) class.

Macros

- `#define ROBOT_DYN_4DIGIT_DISPLAY_NUM_DIGITS 4`
Number of display digits.
- `#define SEGMENTS_MINUS 0b01000000`
Special characters.
- `#define SEGMENTS_DEGREE 0b01100011`
Degree symbol.
- `#define SEGMENTS_CELSIUS 0b00111001`
Celsius symbol.

6.2.1 Detailed Description

[RobotDyn4DigitDisplay](#) library for Arduino.

Source: <https://github.com/Erriez/ErriezRobotDyn4DigitDisplay> Documentation↔
: <https://erriez.github.io/ErriezRobotDyn4DigitDisplay>

6.2.2 Macro Definition Documentation

6.2.2.1 `#define SEGMENTS_MINUS 0b01000000`

Special characters.

Minus sign

Definition at line 43 of file RobotDyn4DigitDisplay.h.

Index

- dec
 - RobotDyn4DigitDisplay, [12](#)
- digit
 - RobotDyn4DigitDisplay, [12](#)
- doubleDots
 - RobotDyn4DigitDisplay, [13](#)
- hex
 - RobotDyn4DigitDisplay, [13](#)
- rawDigit
 - RobotDyn4DigitDisplay, [13](#)
- RobotDyn4DigitDisplay, [11](#)
 - dec, [12](#)
 - digit, [12](#)
 - doubleDots, [13](#)
 - hex, [13](#)
 - rawDigit, [13](#)
 - RobotDyn4DigitDisplay, [12](#)
 - time, [13](#)
- RobotDyn4DigitDisplay.cpp, [15](#)
- RobotDyn4DigitDisplay.h, [15](#)
 - SEGMENTS_MINUS, [16](#)
- SEGMENTS_MINUS
 - RobotDyn4DigitDisplay.h, [16](#)
- time
 - RobotDyn4DigitDisplay, [13](#)