Grove - 4-Digit Display

Release date： 9/20/2015

Version： 1.0

Wiki: http://www.seeedstudio.com/wiki/Grove_-_4-Digit_Display

Bazaar: http://www.seeedstudio.com/depot/Grove-4Digit-Display-p-1198.html

Document Revision History

Revision	Date	Author	Description
1.0	Sep 21, 2015	Victor.He	Create file

Contents

Document Revision History 2
1. Introduction 2
2. Features 3
3. Application ideas 4
4. Specifications 5
5. Interface functions 6
6. Usage 7
   6.1 With TI LaunchPad 7
   6.2 With Arduino 8
   6.3 With Raspberry Pi 11
7. Resources 15
8. Related Projects 16
9. Support 18

Disclaimer

For physical injuries and possessions loss caused by those reasons which are not related to product quality, such as operating without following manual guide, natural disasters or force majeure, we take no responsibility for that.

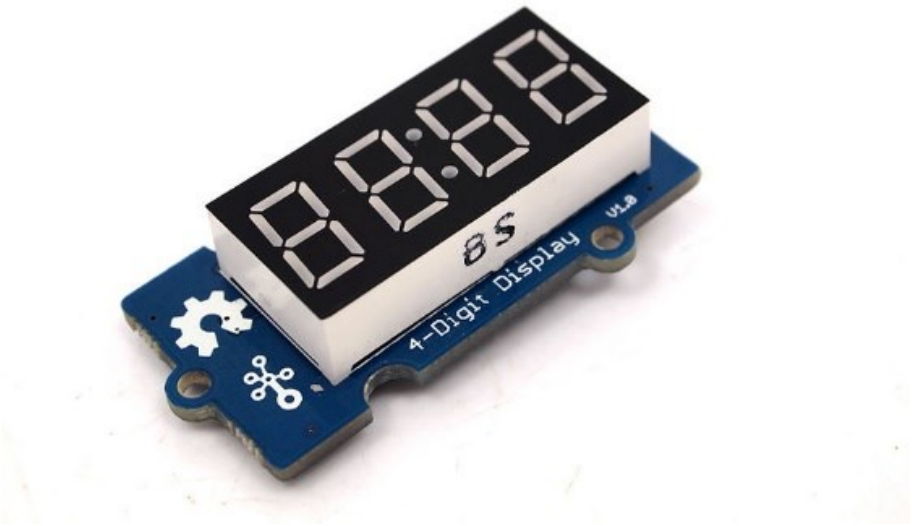
Under the supervision of Seeed Technology Inc., this manual has been compiled and published which covered the latest product description and specification. The content of this manual is subject to change without notice.

Copyright

The design of this product (including software) and its accessories is under tutelage of laws. Any action to violate relevant right of our product will be penalized through law. Please consciously observe relevant local laws in the use of this product.

1. Introduction

Grove - 4-Digit Display module is a 12-pin module. In this module, we utilise a TM1637 to scale down the number of controlling pins to 2. That is to say, it controls both the content and the luminance via only 2 digital pins of Arduino or Seeeduino. For projects that require alpha-numeric display, this can be a nice choice.



2. Features

- 4 digit red alpha-numeric display
- Grove compatible interface (3.3V/5V)
- 8 adjustable luminance levels

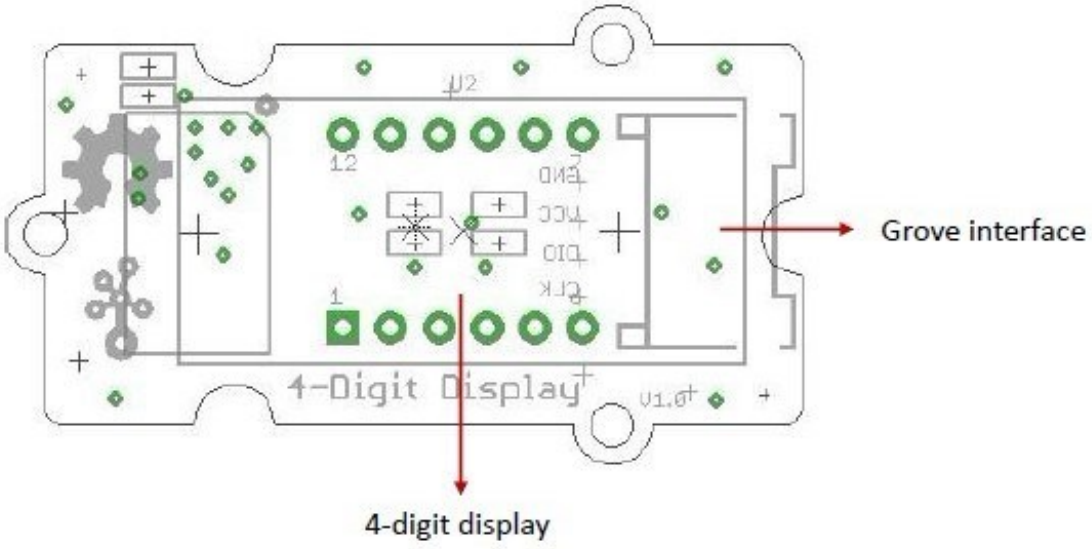
3. Application ideas

- Time display
- Stopwatch
- Sensors' input display

4. Specifications

Item	Min	Typical	Max	Unit
Voltage	3.3	5.0	5.5	VDC
Current	0.2	27	80	mA
Dimensions	42x24x14			mm
Net Weight	7±1			g

5. Interface functions



Grove interface - Can be connected to digital port on Grove - Base Shield.

4 - Digit display - Common anode digital tube.

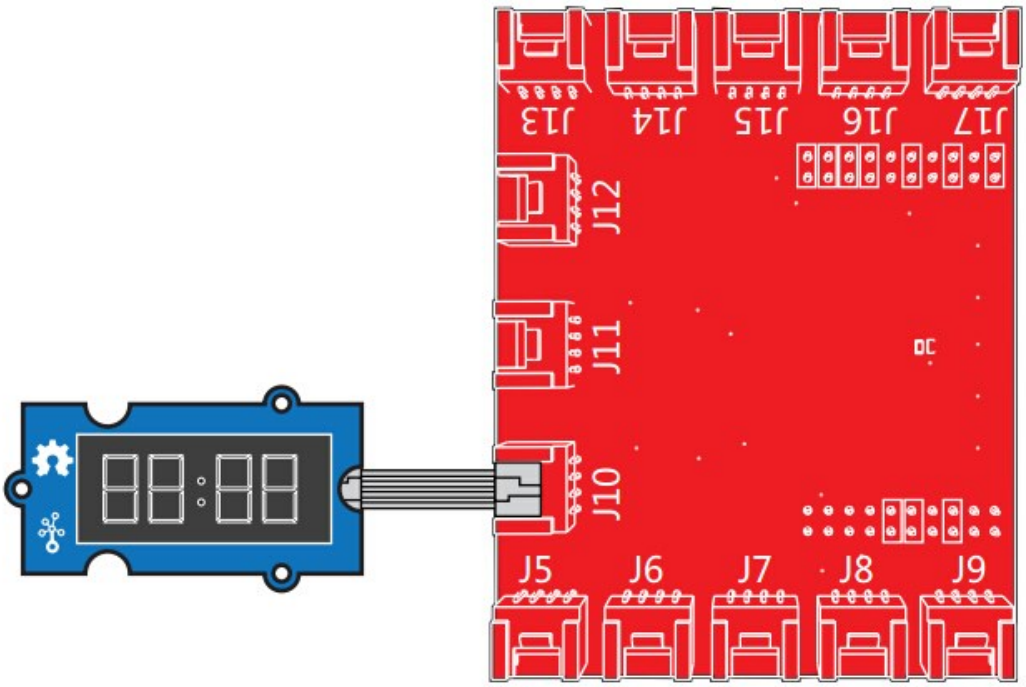
Pin definition: CLK DIO VCC GND

6. Usage

6.1 With TI LaunchPad

Displaying the Numbers (4-Digital-Display)

This example demonstrates how to display some digital numbers using a Grove-4-Digital Display.




```
/*
* TM1637.cpp
* A library for the 4 digit display
*/
#include "TM1637.h"
#define CLK 39 //pins definitions for TM1637 and can be changed to other
ports
#define DIO 38
TM1637 tm1637(CLK,DIO);
void setup()
{
    tm1637.init();
    tm1637.set(BRIGHT_TYPICAL);//BRIGHT_TYPICAL = 2,BRIGHT_DARKEST =
0,BRIGHTEST = 7;
}
void loop()
{
    int8_t NumTab[] =
{0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15};//0~9,A,b,C,d,E,F
```

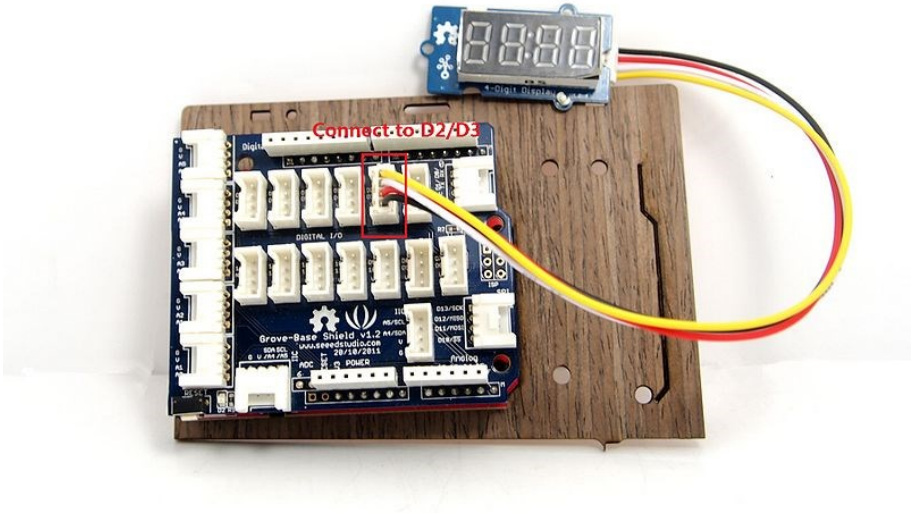
```
    int8_t ListDisp[4];
    unsigned char i = 0;
    unsigned char count = 0;
    delay(150);
    while(1)
    {
        i = count;
        count ++;
        if(count == sizeof(NumTab)) count = 0;
        for(unsigned char BitSelect = 0;BitSelect < 4;BitSelect ++)
        {
            ListDisp[BitSelect] = NumTab[i];
            i ++;
            if(i == sizeof(NumTab)) i = 0;
        }
        tm1637.display(0,ListDisp[0]);
        tm1637.display(1,ListDisp[1]);
        tm1637.display(2,ListDisp[2]);
        tm1637.display(3,ListDisp[3]);
        delay(300);
    }
}
```

6.2 With Arduino

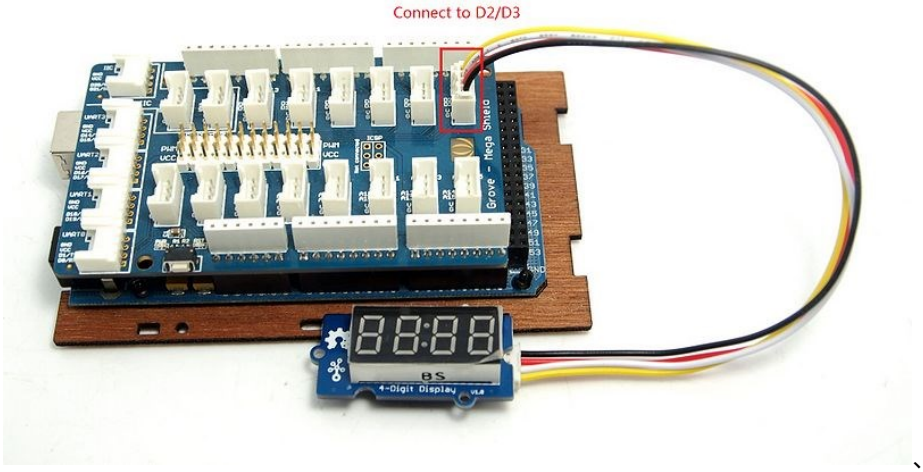
The module uses an LED drive chip - TM1637 to control the contents and change the luminance. Here we drive it to display time.

1. Connect the Grove socket marked "IN" on the LED Strip Driver and digital port 2 of the Grove - Base Shield with a Grove cable. You can change to the digital port as you like. But don't forget to change the port number in the definition of the demo code at the same time.
2. Plug onto Arduino/Seeeduino or plug Grove - Mega Shield onto Arduino Mega.

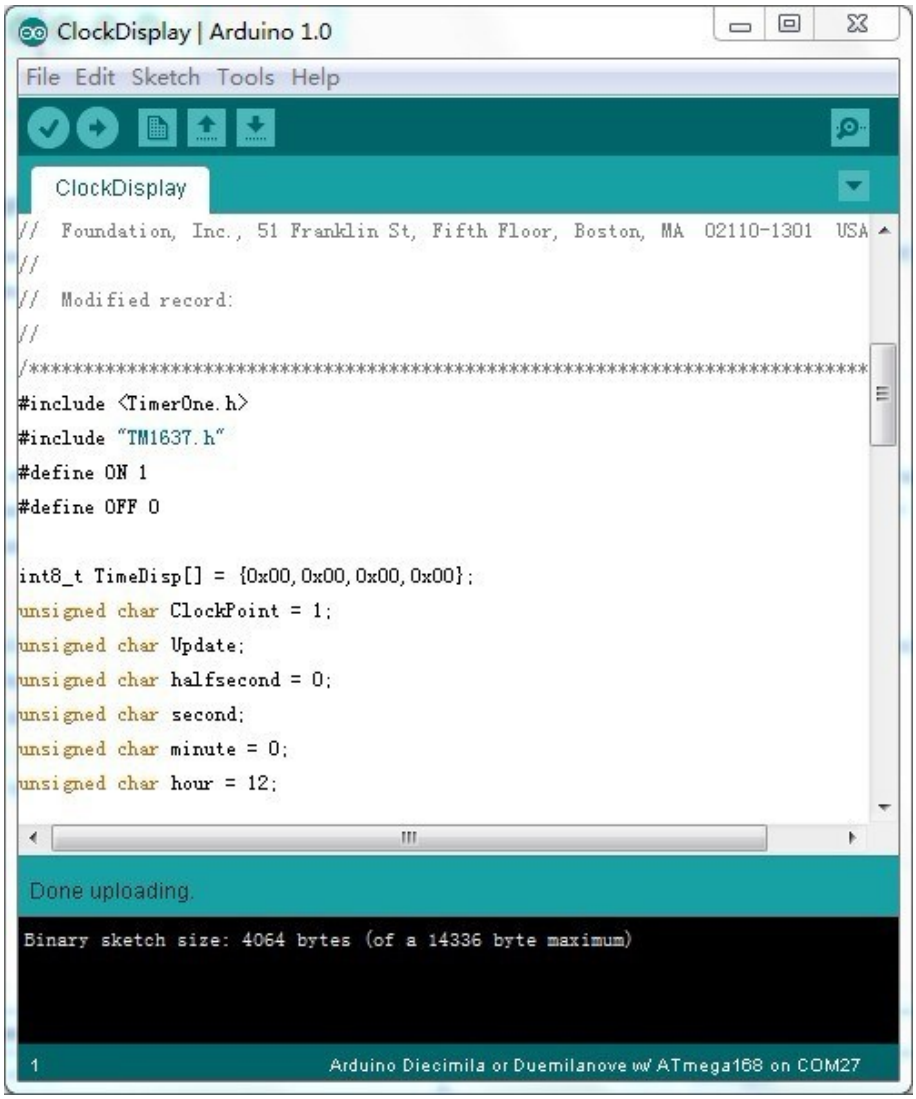
   Seeeduino and Grove - 4-digit display:



Arduino Mega and Grove - 4-digit display:

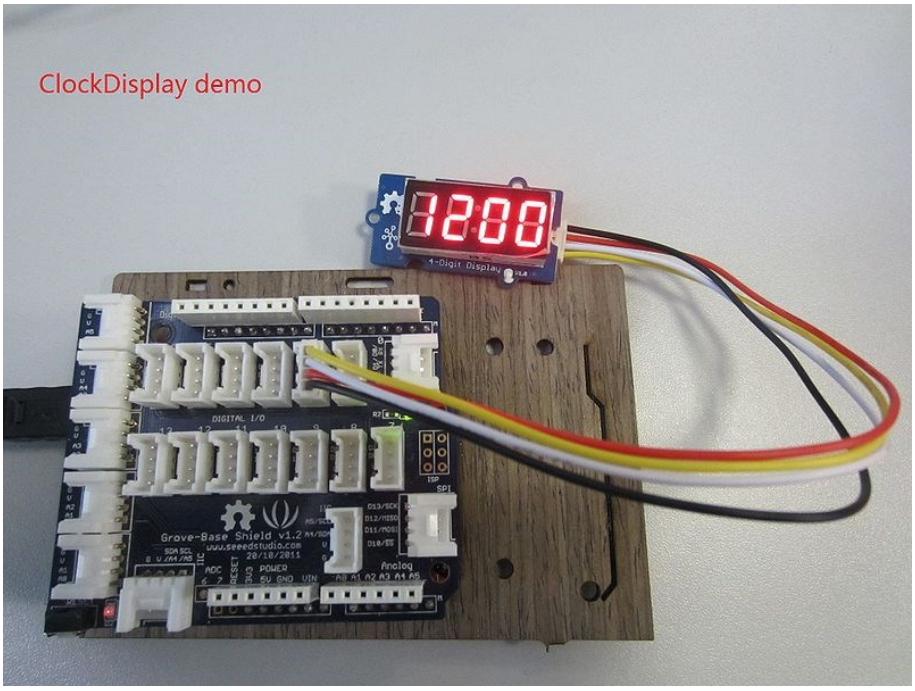


3. Connect Arduino/Seeeduino to PC via a USB cable.
4. Download the 4-Digit Display library and TimerOne library. Unzip and put them in the libraries file of Arduino IDE by the path: ..\arduino-1.0\libraries.
5. Restart the Arduino IDE, open one demo code you like, for example ClockDisplay directly by the path:File -> Example ->DigitalTube->ClockDisplay.



6. Upload the demo code and the clock will be ticking in a few seconds. Please click here if you do not know how to upload.

You can see this:



6.3 With Raspberry Pi

1. You should have got a raspberry pi and a grovepi or grovepi+.
2. You should have completed configuring the development environment, otherwise follow here.
3. Connection. Plug the sensor to grovepi socket D5 by using a grove cable.
4. Navigate to the demos' directory:

```
cd yourpath/GrovePi/Software/Python/
```

To see the code

```
nano grove_4_digit_display.py      # "Ctrl+x" to exit #
```

```
import time
import grovepi

# Connect the Grove 4 Digit Display to digital port D5
# CLK,DIO,VCC,GND
display = 5
grovepi.pinMode(display,"OUTPUT")

# If you have an analog sensor connect it to A0 so you can monitor it
below
sensor = 0
grovepi.pinMode(sensor,"INPUT")
```

```
time.sleep(.5)

# 4 Digit Display methods
# grovepi.fourDigit_init(pin)
# grovepi.fourDigit_number(pin,value,leading_zero)
# grovepi.fourDigit_brightness(pin,brightness)
# grovepi.fourDigit_digit(pin,segment,value)
# grovepi.fourDigit_segment(pin,segment,leds)
# grovepi.fourDigit_score(pin,left,right)
# grovepi.fourDigit_monitor(pin,analog,duration)
# grovepi.fourDigit_on(pin)
# grovepi.fourDigit_off(pin)

while True:
    try:
        print "Test 1) Initialise"
        grovepi.fourDigit_init(display)
        time.sleep(.5)

        print "Test 2) Set brightness"
        for i in range(0,8):
            grovepi.fourDigit_brightness(display,i)
            time.sleep(.2)
        time.sleep(.3)

        # set to lowest brightness level
        grovepi.fourDigit_brightness(display,0)
        time.sleep(.5)

        print "Test 3) Set number without leading zeros"
        leading_zero = 0
        grovepi.fourDigit_number(display,1,leading_zero)
        time.sleep(.5)
        grovepi.fourDigit_number(display,12,leading_zero)
        time.sleep(.5)
        grovepi.fourDigit_number(display,123,leading_zero)
        time.sleep(.5)
        grovepi.fourDigit_number(display,1234,leading_zero)
        time.sleep(.5)

        print "Test 4) Set number with leading zeros"
        leading_zero = 1
        grovepi.fourDigit_number(display,5,leading_zero)
        time.sleep(.5)
```

```
grovepi.fourDigit_number(display,56,leading_zero)
time.sleep(.5)
grovepi.fourDigit_number(display,567,leading_zero)
time.sleep(.5)
grovepi.fourDigit_number(display,5678,leading_zero)
time.sleep(.5)

print "Test 5) Set individual digit"
grovepi.fourDigit_digit(display,0,2)
grovepi.fourDigit_digit(display,1,6)
grovepi.fourDigit_digit(display,2,9)
grovepi.fourDigit_digit(display,3,15) # 15 = F
time.sleep(.5)

print "Test 6) Set individual segment"
grovepi.fourDigit_segment(display,0,118) # 118 = H
grovepi.fourDigit_segment(display,1,121) # 121 = E
grovepi.fourDigit_segment(display,2,118) # 118 = H
grovepi.fourDigit_segment(display,3,121) # 121 = E
time.sleep(.5)

grovepi.fourDigit_segment(display,0,57) # 57 = C
grovepi.fourDigit_segment(display,1,63) # 63 = O
grovepi.fourDigit_segment(display,2,63) # 63 = O
grovepi.fourDigit_segment(display,3,56) # 56 = L
time.sleep(.5)

print "Test 7) Set score"
grovepi.fourDigit_score(display,0,0)
time.sleep(.2)
grovepi.fourDigit_score(display,1,0)
time.sleep(.2)
grovepi.fourDigit_score(display,1,1)
time.sleep(.2)
grovepi.fourDigit_score(display,1,2)
time.sleep(.2)
grovepi.fourDigit_score(display,1,3)
time.sleep(.2)
grovepi.fourDigit_score(display,1,4)
time.sleep(.2)
grovepi.fourDigit_score(display,1,5)
time.sleep(.5)

print "Test 8) Set time"
```

```
        grovepi.fourDigit_score(display,12,59)
        time.sleep(.5)

        print "Test 9) Monitor analog pin"
        seconds = 10
        grovepi.fourDigit_monitor(display,sensor,seconds)
        time.sleep(.5)

        print "Test 10) Switch all on"
        grovepi.fourDigit_on(display)
        time.sleep(.5)

        print "Test 11) Switch all off"
        grovepi.fourDigit_off(display)
        time.sleep(.5)

    except KeyboardInterrupt:
        grovepi.fourDigit_off(display)
        break
    except IOError:
        print "Error"
```

5. Run the demo.

```
sudo python grove_4_digit_display.py
```

5. This demo may not work if your grovepi dosen't have the newest firmware, update the firmware.

```
cd yourpath/GrovePi/Firmware
sudo ./firmware_update.sh
```

7. Resources

Grove - 4-Digit Display V1.0 eagle files

Schematic in PDF

4-Digit Display library

TimerOne library

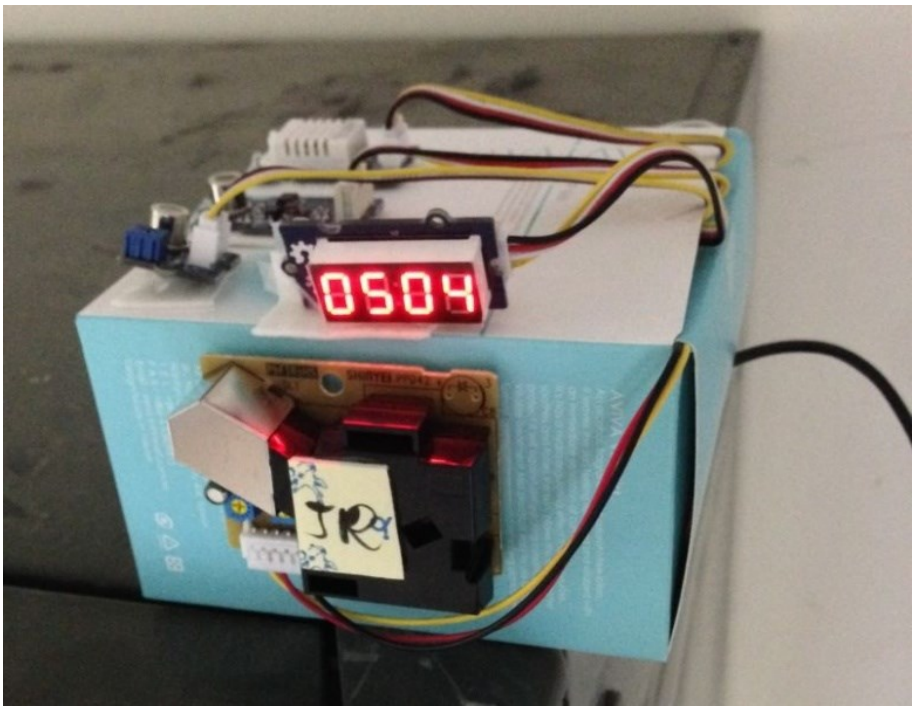
Four-Digit Display Suli Library

TM1637 datasheet

8. Related Projects

If you want to make some awesome projects by 4-Digit Display, here's one project for reference.

Air Quality Box



With this demo, we can see the air quality data on 4-Digit Display.

I want to make it.

More Awesome Projects by 4-Digit Display

Share Your Awesome Projects with Us

Born with the spirit of making and sharing, that is what we believe makes a maker.

And only because of this, the open source community can be as prosperous as it is today.

It does not matter what you are and what you have made, hacker, maker, artist or engineers.

As long as you start sharing your works with others, you are being part of the open source community and you are making your contributions.

Now share your awesome projects with us on Recipe, and win a chance to become the Core User of Seeed.

- Core Users, are those who show high interests in Seeed products and make significant contributions on Recipe.
- We cooperate with our Core Users in the development of our new product, this, in another word, the Core Users will have the chance to experience any new products of Seeed before its

official launch, and in return we expect valuable feedback from them to help us improve the product performance and user experience. And in most cases when our Core Users have some good ideas of making things, we'll offer hardware pieces, PCBA services as well as technical support. Besides, further commercial cooperation with the Core Users is highly possible.

Get more information about Core User, please email to: recipe@seeed.cc

9. Support

If you have questions or other better design ideas, you can go to our forum or wish to discuss.

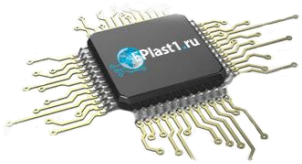
Компания «ЭлектроПласт» предлагает заключение долгосрочных отношений при поставках импортных электронных компонентов на взаимовыгодных условиях!

Наши преимущества:

- Оперативные поставки широкого спектра электронных компонентов отечественного и импортного производства напрямую от производителей и с крупнейших мировых складов;
- Поставка более 17-ти миллионов наименований электронных компонентов;
- Поставка сложных, дефицитных, либо снятых с производства позиций;
- Оперативные сроки поставки под заказ (от 5 рабочих дней);
- Экспресс доставка в любую точку России;
- Техническая поддержка проекта, помощь в подборе аналогов, поставка прототипов;
- Система менеджмента качества сертифицирована по Международному стандарту ISO 9001;
- Лицензия ФСБ на осуществление работ с использованием сведений, составляющих государственную тайну;
- Поставка специализированных компонентов (Xilinx, Altera, Analog Devices, Intersil, Interpoint, Microsemi, Aeroflex, Peregrine, Syfer, Eurofarad, Texas Instrument, Miteq, Cobham, E2V, MA-COM, Hittite, Mini-Circuits,General Dynamics и др.);

Помимо этого, одним из направлений компании «ЭлектроПласт» является направление «Источники питания». Мы предлагаем Вам помощь Конструкторского отдела:

- Подбор оптимального решения, техническое обоснование при выборе компонента;
- Подбор аналогов;
- Консультации по применению компонента;
- Поставка образцов и прототипов;
- Техническая поддержка проекта;
- Защита от снятия компонента с производства.



Как с нами связаться

Телефон: 8 (812) 309 58 32 (многоканальный)
Факс: 8 (812) 320-02-42
Электронная почта: org@eplast1.ru
Адрес: 198099, г. Санкт-Петербург, ул. Калинина, дом 2, корпус 4, литера А.