MAX7219 32x8 Dot Matrix Display Module

This is 8x8 (row by column 64x LED) dot matrix LED displays module based on MAX7219 IC. The displays are designed so that they can be mounted in a horizontal chain and can also be expanded in a vertical plane allowing versatile displays panel to be built. A convenient 3-wire serial interface connects to all common controller board like Arduino or Raspberry. Individual dot may be addressed and updated without rewriting the entire display. This module can be daisy chain to form a display panel for scrolling message board.
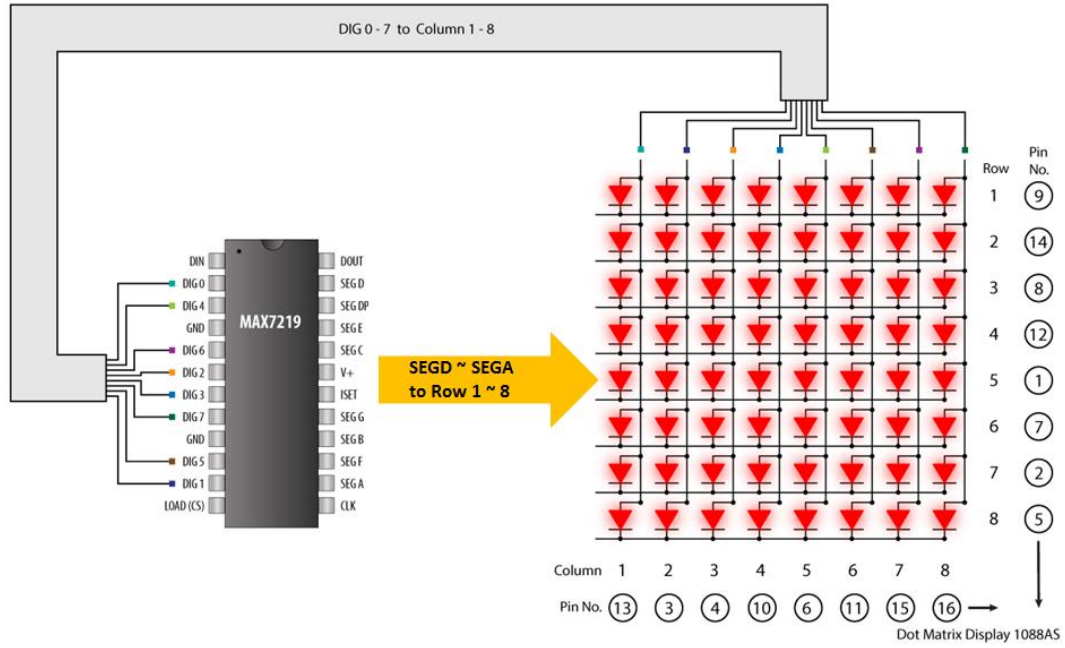
SKU: DSP-1172

Brief Data:

- Matrix Size: 32x8 (256-Dots)
- Display Size: 128x33 mm.
- Display Color: Red.
- Interface: 3-Wires Serial Interface.
- Daisy chain for multiple modules.
- Operating voltage: 4.5 ~ 5V.
- Module size: 128 x 33 x 15 mm (L x W x H).
- Mounting Hole: M3.
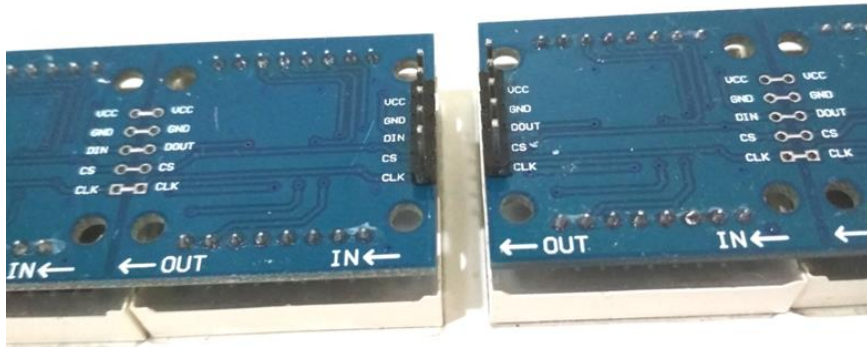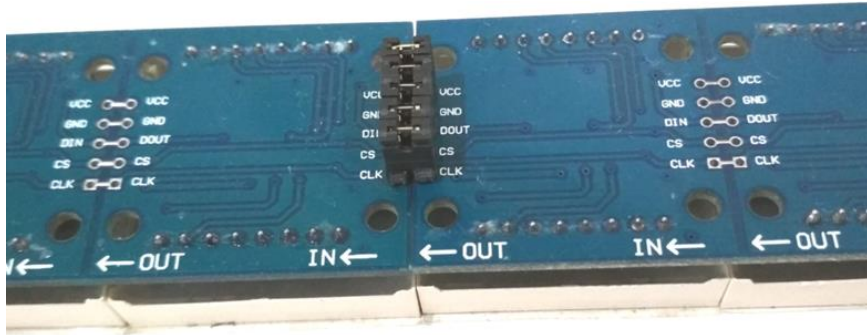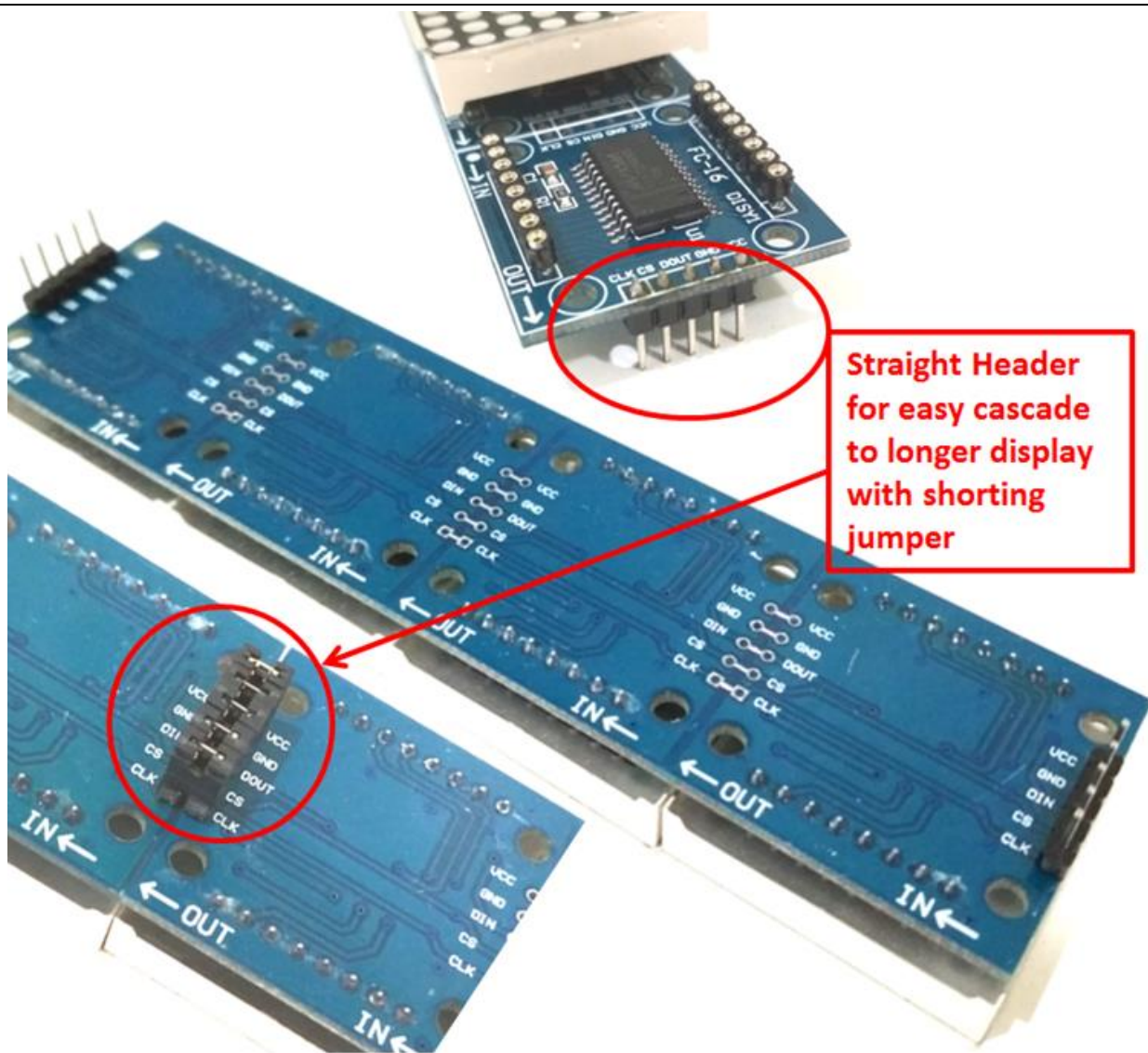
MAX7291 Module Matrix Connection:

Module Connector:

Straight connector for easy cascading for longer module with shorting jumper.
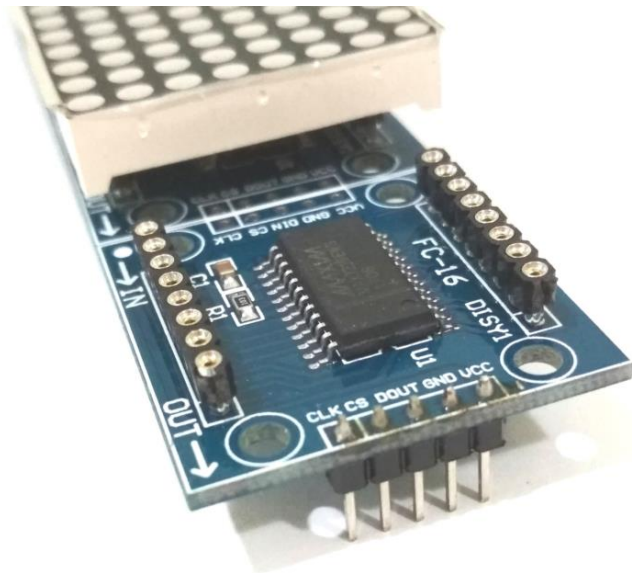
Cascading 2 Matrix Display Module with Shorting Bar:

Straight Header for easy cascade to longer display with shorting jumper
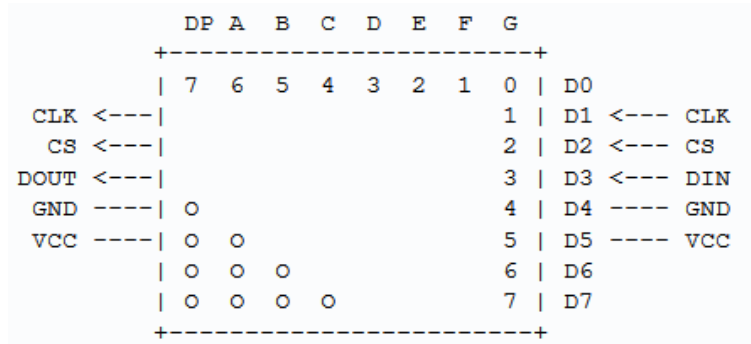
Arduino Library for MAX72xx Driver:

We will need an Arduino Matrix Display Library which supports different brands/types/makes of MAX7219-based dot matrix LED displays. This library is well documented and in the download you'll find many useful examples.

Download the libraries in the below links and copy to Arduino Libraries folder:

MAX72xx LED Matrix Display Library

Library for modular scrolling LED matrix text displays

This display module is based on FC-16 columns and rows dot matrix display orientation, we need to modify few parameters inside the library routine in order for this module to work correctly.

```
            DP  A   B   C   D   E   F   G
         +------------------------+
         | 7   6   5   4   3   2   1   0 |  D0
 CLK <---|                         1 |  D1 <--- CLK
  CS <---|                         2 |  D2 <--- CS
DOUT <---|                         3 |  D3 <--- DIN
 GND ----|  O                      4 |  D4 ---- GND
 VCC ----|  O   O                  5 |  D5 ---- VCC
         |  O   O   O              6 |  D6
         |  O   O   O   O          7 |  D7
         +------------------------+
```

FC-16 Module Dot Matrix Orientation.

Go to the Arduino Library folder located in your hard disc and locate the folder "MD_MAX72xx.h".

Libraries > MD_MAX_72xx > src > MD_MAX72xx.h
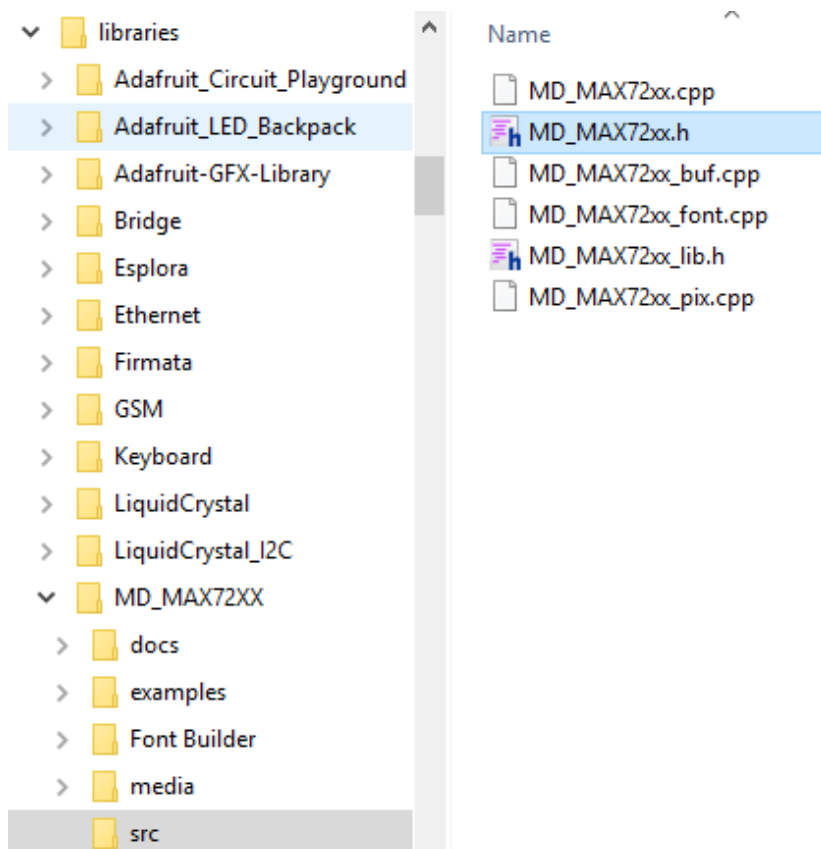
Open this file with any text editor and modify the below line and save the file:
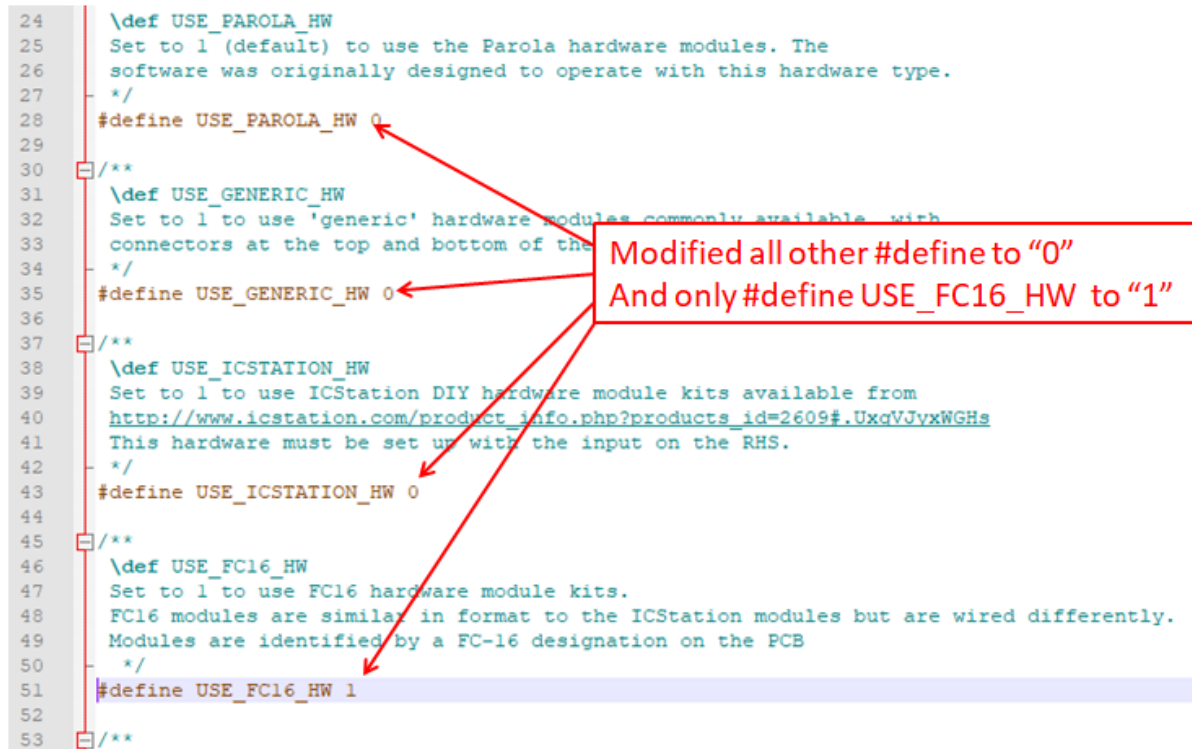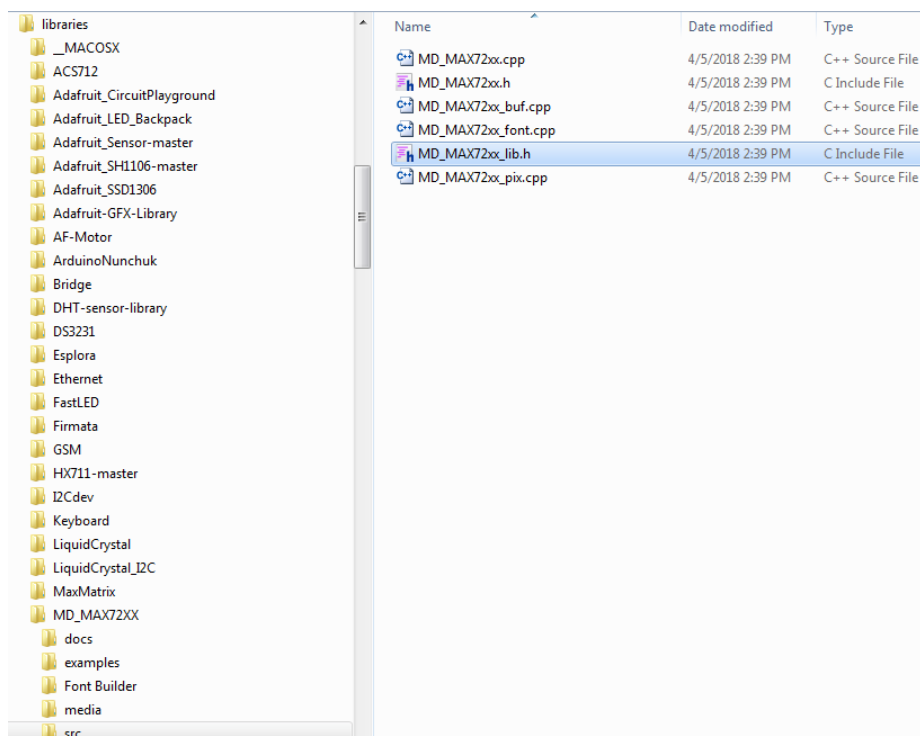
#define USE_FC16_HW 0

Modified to:

#define USE_FC16_HW 1

```
24      \def USE_PAROLA_HW
25      Set to 1 (default) to use the Parola hardware modules. The
26      software was originally designed to operate with this hardware type.
27      */
28    #define USE_PAROLA_HW 0
29
30    /**
31      \def USE_GENERIC_HW
32      Set to 1 to use 'generic' hardware modules commonly available, with
33      connectors at the top and bottom of the                        Modified all other #define to "0"
34      */                                                              And only #define USE_FC16_HW to "1"
35    #define USE_GENERIC_HW 0
36
37    /**
38      \def USE_ICSTATION_HW
39      Set to 1 to use ICStation DIY hardware module kits available from
40      http://www.icstation.com/product_info.php?products_id=2609#.UxqVJyxWGHs
41      This hardware must be set up with the input on the RHS.
42      */
43    #define USE_ICSTATION_HW 0
44
45    /**
46      \def USE_FC16_HW
47      Set to 1 to use FC16 hardware module kits.
48      FC16 modules are similar in format to the ICStation modules but are wired differently.
49      Modules are identified by a FC-16 designation on the PCB
50      */
51    #define USE_FC16_HW 1
52
53    /**
```

Next locate this file MD_MAX72xx_lib.h in the below path:

Libraries > MD_MAX_72xx > src > MD_MAX72xx_lib.h

libraries		Name	Date modified	Type
__MACOSX		MD_MAX72xx.cpp	4/5/2018 2:39 PM	C++ Source File
ACS712		MD_MAX72xx.h	4/5/2018 2:39 PM	C Include File
Adafruit_CircuitPlayground		MD_MAX72xx_buf.cpp	4/5/2018 2:39 PM	C++ Source File
Adafruit_LED_Backpack		MD_MAX72xx_font.cpp	4/5/2018 2:39 PM	C++ Source File
Adafruit_Sensor-master		MD_MAX72xx_lib.h	4/5/2018 2:39 PM	C Include File
Adafruit_SH1106-master		MD_MAX72xx_pix.cpp	4/5/2018 2:39 PM	C++ Source File
Adafruit_SSD1306				
Adafruit-GFX-Library				
AF-Motor				
ArduinoNunchuk				
Bridge				
DHT-sensor-library				
DS3231				
Esplora				
Ethernet				
FastLED				
Firmata				
GSM				
HX711-master				
I2Cdev				
Keyboard				
LiquidCrystal				
LiquidCrystal_I2C				
MaxMatrix				
MD_MAX72XX				
docs				
examples				
Font Builder				
media				
src				

Open this file with any text editor and modify the below line and verify the below setting:

```
506
507  #if USE_FC16_HW        // tested MC 23 Feb 2015
508  //#pragma message "FC16 HW selected"
509  #define HW_DIG_ROWS 1 ///< MAX72xx digits are mapped to rows in on the matrix
510  #define HW_REV_COLS 0 ///< Normal orientation is col 0 on the right. Set to 1 if reversed
511  #define HW_REV_ROWS 0 ///< Normal orientation is row 0 at the top. Set to 1 if reversed
512  #endif
```

Now the library modification is all set to run on this Matrix Display Module.

Arduino Connection Examples:

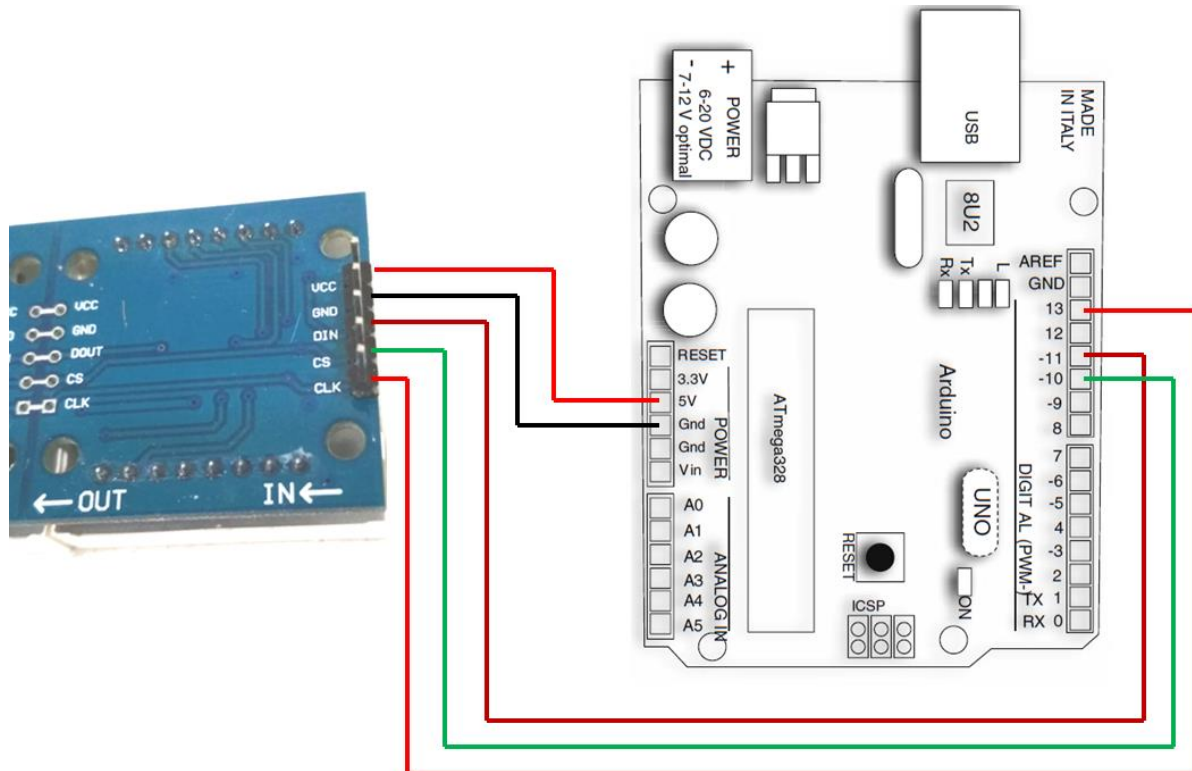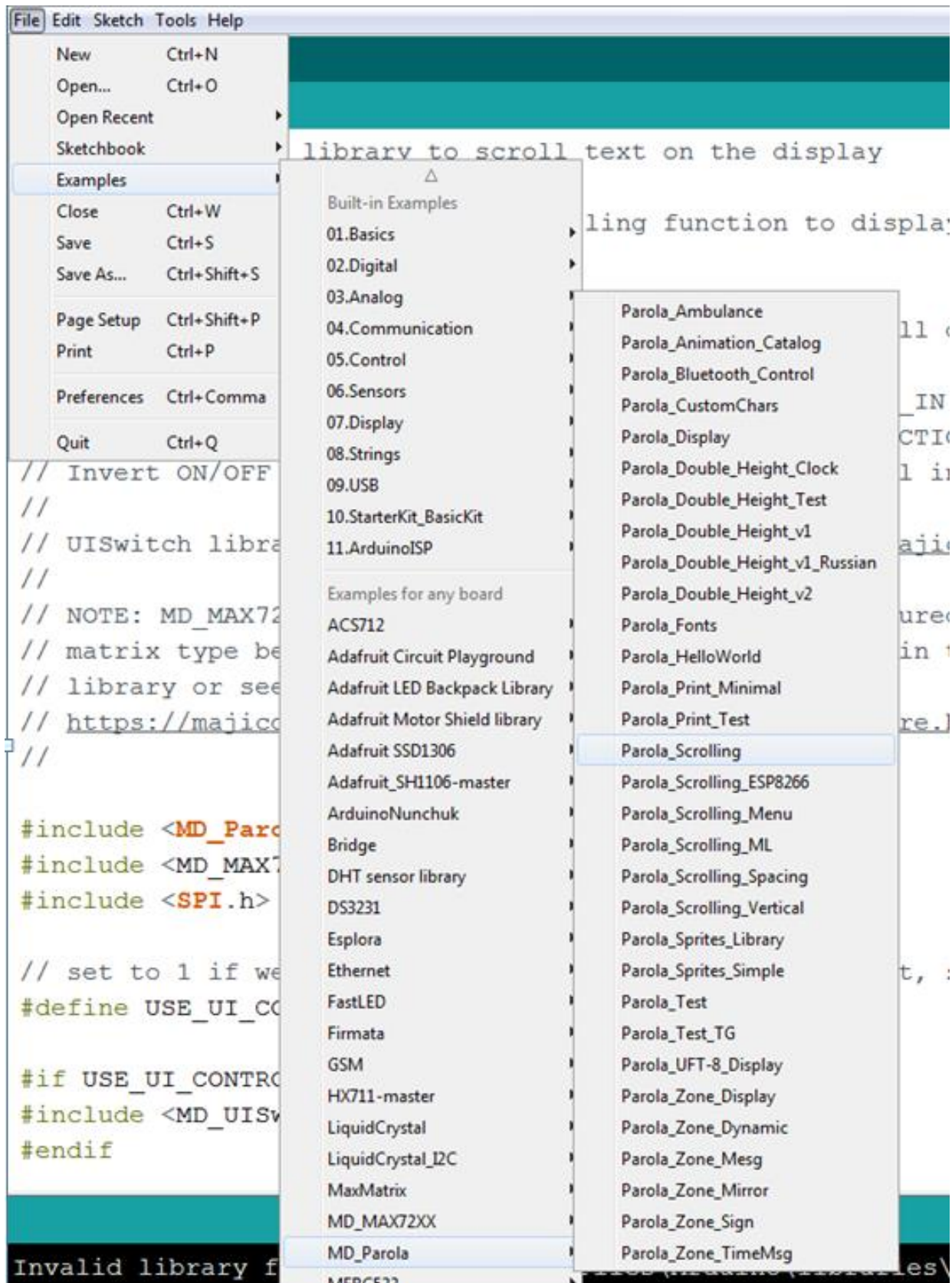Now let's connect the 8×8 LED Matrix module to the Arduino Board. Here's the circuit schematic:

Figure 1

CLK > Arduno D13

DIN > Arduino D11

CS > Arduino D10

Start Arduino IDE. Open "Parola_Scrolling" sketch from the Examples sketch folder as shown in below path:

We are going to cascade two 4-In-1 module. In this case there are total of 8 individual dot matrix displays.

Locate the line as below shown:

```
// Define the number of devices we have in
// NOTE: These pin numbers will probably no
// need to be adapted
#define MAX_DEVICES 8
#define CLK_PIN    13
#define DATA_PIN   11
#define CS_PIN     10
```

Define MAX_DEVICES to 8.

Compile and upload the sketch to Arduino Uno board as shown in Figure 1 schematic.

```
// Global message buffers shared by Serial and Scrolling functions
#define BUF_SIZE  75
char curMessage[BUF_SIZE] = { "" };
char newMessage[BUF_SIZE] = { "Handsontec > handsontec.com" };
bool newMessageAvailable = true;
```

After successfully upload the sketch, the message design in below line should run across the display:

char newMessage[BUF_SIZE] = { "Handsontec > handsontec.com" };

Change any message you like to display in this newMessage[BUF_SIZE] variable.

Web Resources:

https://www.arduinolibraries.info/libraries/md_parola

https://github.com/MajicDesigns/MD_MAX72XX

YouTube Video Demo:

- https://youtu.be/r-bAHqyaKvA
- https://youtu.be/9DuqYFv3NO0

HandsOn Technology provides a multimedia and interactive platform for everyone interested in electronics. From beginner to diehard, from student to lecturer. Information, education, inspiration and entertainment. Analog and digital, practical and theoretical; software and hardware.

Hands*On* Technology support Open Source Hardware (OSHW) Development Platform.

open source hardware

Learn : Design : Share

www.handsontec.com

The Face behind our product quality…

In a world of constant change and continuous technological development, a new or replacement product is never far away – and they all need to be tested.

Many vendors simply import and sell wihtout checks and this cannot be the ultimate interests of anyone, particularly the customer. Every part sell on Handsotec is fully tested. So when buying from Handsontec products range, you can be confident you're getting outstanding quality and value.

We keep adding the new parts so that you can get rolling on your next project.

Breakout Boards & Modules

Connectors

Electro-Mechanical Parts

Engineering Material

Mechanical Hardware

Electronics Components

P

Power Supply

Arduino Board & Shield

Tools & Accessory