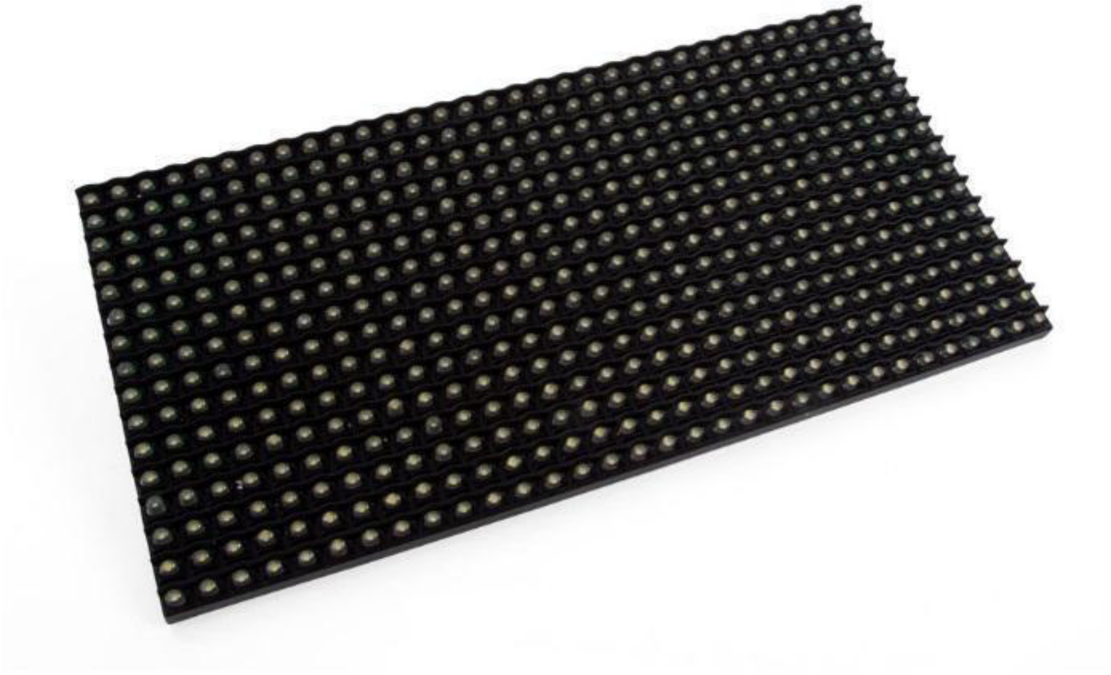
VMA419

32x16 DOT MATRIX DISPLAY



USER MANUAL

USER MANUAL

1. Introduction

To all residents of the European Union

Important environmental information about this product

This symbol on the device or the package indicates that disposal of the device after its lifecycle could harm the environment. Do not dispose of the unit (or batteries) as unsorted municipal waste; it should be taken to a specialized company for recycling. This device should be returned to your distributor or to a local recycling service. Respect the local environmental rules.

If in doubt, contact your local waste disposal authorities.

Thank you for choosing Velleman®! Please read the manual thoroughly before bringing this device into service. If the device was damaged in transit, do not install or use it and contact your dealer.

2. Safety Instructions

⚠	• This device can be used by children aged from 8 years and above, and persons with reduced physical, sensory or mental capabilities or lack of experience and knowledge if they have been given supervision or instruction concerning the use of the device in a safe way and understand the hazards involved. Children shall not play with the device. Cleaning and user maintenance shall not be made by children without supervision.

⌂	• Indoor use only. Keep away from rain, moisture, splashing and dripping liquids.

3. General Guidelines

ⓘ	• Refer to the Velleman® Service and Quality Warranty on the last pages of this manual. • Familiarise yourself with the functions of the device before actually using it. • All modifications of the device are forbidden for safety reasons. Damage caused by user modifications to the device is not covered by the warranty. • Only use the device for its intended purpose. Using the device in an unauthorised way will void the warranty. • Damage caused by disregard of certain guidelines in this manual is not covered by the warranty and the dealer will not accept responsibility for any ensuing defects or problems. • Nor Velleman nv nor its dealers can be held responsible for any damage (extraordinary, incidental or indirect) – of any nature (financial, physical…) arising from the possession, use or failure of this product. • Due to constant product improvements, the actual product appearance might differ from the shown images. • Product images are for illustrative purposes only. • Do not switch the device on immediately after it has been exposed to changes in temperature. Protect the device against damage by leaving it switched off until it has reached room temperature. • Keep this manual for future reference.

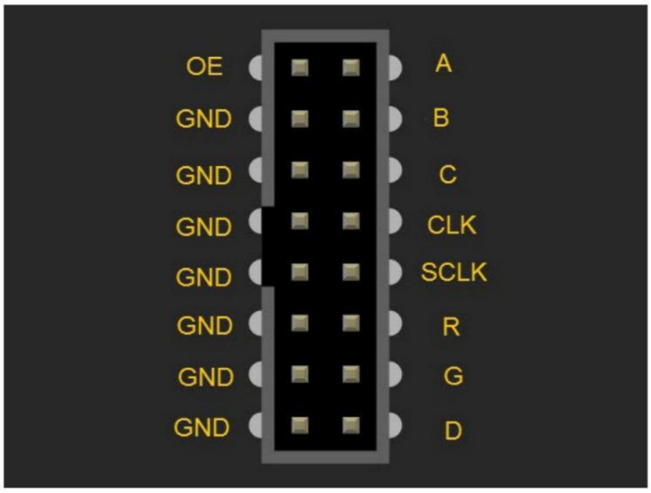
4. What is Arduino®

Arduino® is an open-source prototyping platform based in easy-to-use hardware and software. Arduino® boards are able to read inputs – light-on sensor, a finger on a button or a Twitter message – and turn it into an output – activating of a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so, you use the Arduino programming language (based on Wiring) and the Arduino® software IDE (based on Processing).

Surf to www.arduino.cc and www.arduino.org for more information.

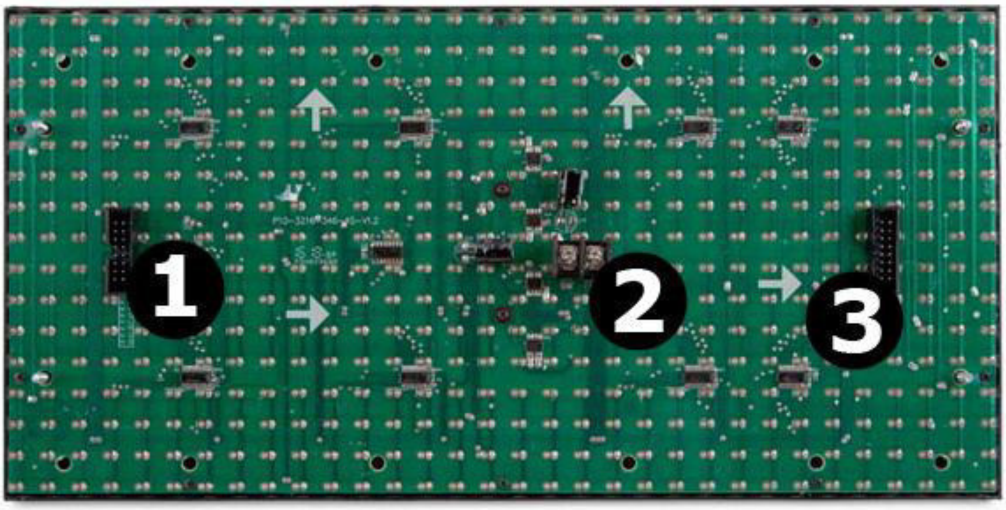
5. Pin Layout and DMD Connection

5.1 Pin Layout

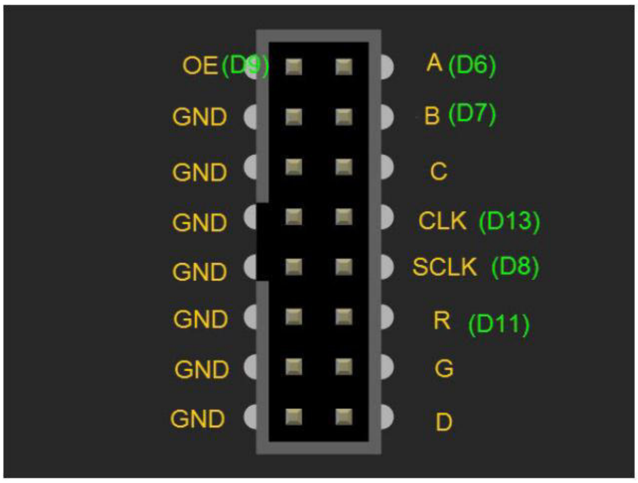


OE	enable/disable all the LEDs on the DMD
A	pins select which ¼ of the DMD is selected
B	pins select which ¼ of the DMD is selected
C	pins select which ¼ of the DMD is selected
D	pins select which ¼ of the DMD is selected
CLK	used to clock each pixel into the DMD shift register
SCLK	latches the current content of the DMD shift register
R	the raw pixel data stream (R = red, G = green on HUB12 connector (unused))

5.2 Connection



1	data input	**3**	data output
2	5 V power input		



OE ========= D9

GND ========= GND

A ========= D6

B ========= D7

C ========= no connection

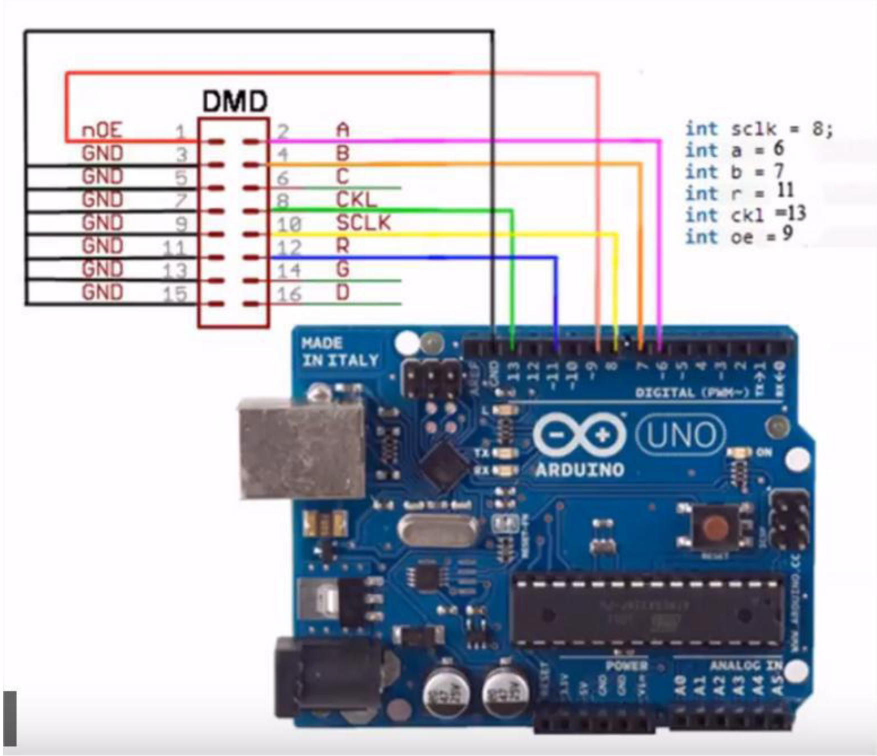
CLK ========= D13

SCLK ========= D8

R ========= D11

G ========= no connection

D ========= no connection



```
int sclk = 8;
int a = 6
int b = 7
int r = 11
int ckl =13
int oe = 9
```

IMPORTANT!

Do not use the 5 V power from your Arduino® or USB power to power the high-brightness DMD 5 V input terminals directly. Those sources are only capable of 0.5 A at best. Use the 5 VDC connection on the VMA416 instead.

If connecting a separate 5 V power to your DMD terminals while connected to your Arduino® board by USB, be very careful of your external 5 V power source polarity as you may feed damaging reverse voltage to your computer's USB port, should the wires be reversed or ground offset.

6. Installing the Libraries

The VMA419 board must work together with the TimerOne and DMD419 libraries. We are using the latest Arduino® IDE version, i.e. v1.8.5.

1. Download the TimerOne library from the VMA419 product's page on www.velleman.eu.
2. Import the library into your IDE via Sketch → Include Library → Add ZIP Library.
3. Download the DMD419 from the VMA419 product's page on www.velleman.eu and add it to your Arduino® IDE.
4. Once downloaded, import as you did for the TimerOne library.
5. Close all open Arduino® IDE windows and restart the IDE.

7. Additional Programming Information

7.1 Pre-Requisite Code to Enable your DMD

To use a DMD, the following code must be inserted into your sketch:

```
#include "SPI.h"
#include "DMD419.h"
#include "TimerOne.h"
#include "Arial_black_16.h"<arial_black_16.h>
// you can remove the fonts if unused
#include "SystemFont5x7.h"
#define DISPLAYS_ACROSS 1
#define DISPLAYS_DOWN 1
/* change these values if you have more than one DMD connected */
DMD419 dmd(DISPLAYS_ACROSS,DISPLAYS_DOWN);
void ScanDMD()
{
  dmd.scanDisplayBySPI();
}
void setup()
{
   Timer1.initialize( 5000 );
/*period in microseconds to call ScanDMD. Anything longer than 5000 (5ms)
and you can see flicker.*/

   Timer1.attachInterrupt( ScanDMD );
/*attach the Timer1 interrupt to ScanDMD which goes to
dmd.scanDisplayBySPI()*/

   dmd.clearScreen( true );
/* true is normal (all pixels off), false is negative (all pixels on) */
}
```

7.2 Displaying Text on your DMD

We will run through the functions required to display text in both static and scrolling modes.

The DMD has sixteen rows of thirty-two LEDs each. The position of an LED is noted as follows:

- the X-axis from 0 to 31
- the Y-axis from 0 to 15

Example: 0,0 is the top left LED; 31,15 is the bottom right LED.

Before displaying text, choose from the following functions as described below:

Turn all pixels on or off:

```
dmd.clearScreen(true);
```

or

```
dmd.clearScreen(false);
```

Also, select a font from the two available fonts. Select the small or large one:

```
dmd.selectFont( SystemFont5x7 );
```

or

```
dmd.selectFont( Arial_Black_16 );
```

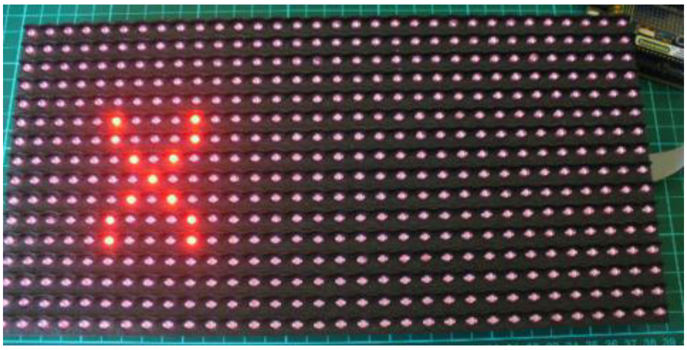
There are functions to display a single character or a line of text. To display a character:

```
dmd.drawChar(  x,  y, ‘@', GRAPHICS_NORMAL );
```

x and y are the coordinates of the top left pixel of the character to be displayed; “@” is the character to be displayed. An example displaying an x in small font

```
dmd.selectFont(SystemFont5x7);
dmd.drawChar(  5,  5, 'X', GRAPHICS_NORMAL );
```

will result in the following:

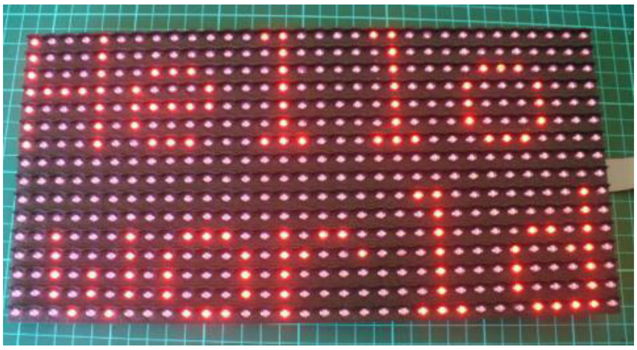


To display a line of text, encode as follows:

```
dmd.drawString( x,y, "text,", z, GRAPHICS_NORMAL );
```

x and y are the coordinates of the top left pixel of the character to be displayed; “text” is the text line to be displayed and z is the text length. An example

```
dmd.selectFont(SystemFont5x7);
dmd.drawString( 0,0, "Hello,", 5, GRAPHICS_NORMAL );
dmd.drawString( 2,9, "world,", 5, GRAPHICS_NORMAL );
```



Static text can be useful but scrolling text is even better. Longer texts can even be scrolled left to right over several interconnected DMDs.

To do so, have your sketch first assemble the text to display into a string variable. The maximum length is 255 characters.

You can also choose a fixed value, like:

```
String textToScroll = "Hello, this is a really long length of text";
```

You will also need the following function in your sketch:

```
void drawText(String dispString)
{
  dmd.clearScreen( true );
  dmd.selectFont( Arial_Black_16 );
  char newString[256];
  int sLength = dispString.length();
  dispString.toCharArray( newString, sLength+1 );
  dmd.drawMarquee(newString,sLength,( 32*DISPLAYS_ACROSS )-1 , 0 );
  long start=millis();
  long timer=start;
  long timer2=start;
  boolean ret=false;
  while(!ret){
    if ( ( timer+20 ) < millis() ) {
      ret=dmd.stepMarquee( -1 , 0 );
      timer=millis();
    }
  }
}
```

To scroll the text, call the function as required:

```
drawText(textToScroll);
```

7.3 Example

The following sketch is the total sketch for a basic text-scrolling application using the function from above:

```
#include "SPI.h"
#include "DMD419.h"
#include "TimerOne.h"
#include "Arial_black_16.h"
/* you can remove the fonts if unused */
#define DISPLAYS_ACROSS 1
#define DISPLAYS_DOWN 1
DMD419 dmd( DISPLAYS_ACROSS , DISPLAYS_DOWN );
void ScanDMD()
{
  dmd.scanDisplayBySPI();
}
void setup()
{
   Timer1.initialize( 5000 );
   Timer1.attachInterrupt( ScanDMD );
   dmd.clearScreen( true );
}
String textToScroll="Hello, this will be displayed on the DMD";
void drawText( String dispString )
{
  dmd.clearScreen( true );
  dmd.selectFont( Arial_Black_16 );
  char newString[256];
  int sLength = dispString.length();
  dispString.toCharArray( newString, sLength+1 );
  dmd.drawMarquee( newString , sLength , ( 32*DISPLAYS_ACROSS )-1 ,0);
  long start=millis();
  long timer=start;
  long timer2=start;
  boolean ret=false;
  while( !ret ){
    if ( ( timer+20 ) < millis() ) {
      ret=dmd.stepMarquee( -1 , 0 );
      timer=millis();
    }
  }
}

void loop()

{
  drawText(textToScroll);
}
```

7.4 Displaying Graphics

Using the functions described above – and with a little imagination – you can create all sorts of images. Every single pixel has to be (de-)activated separately.

To activate or de-activate a pixel:

```
dmd.writePixel( x, y, GRAPHICS_NORMAL, z );
```

x and y are the coordinates of the pixel; z turns the pixel on (with 1) or off (with 0).

To draw a circle with the centre is at x, y and where r is the radius:

```
dmd.drawCircle( x,  y,  r, GRAPHICS_NORMAL );
```

To draw a line from location x1, y1 to x2, y2:

```
dmd.drawLine(  x1, y1, x2, y2, GRAPHICS_NORMAL );
```

To draw a rectangle with the top left pixel at x1, y1 and the bottom right pixel at x2, y2:

```
dmd.drawBox(x1, y1, x2, y2, GRAPHICS_NORMAL );
```

To draw a filled rectangle with the top left pixel at x1, y1 and the bottom right pixel at x2, y2:

```
dmd.drawFilledBox(x1, y1, x2, y2, GRAPHICS_NORMAL );
```

You can see these functions demonstrated using this sketch:

Dmd_demo.pde (located under DMD419/examples)

Use this device with original accessories only. Velleman nv cannot be held responsible in the event of damage or injury resulting from (incorrect) use of this device. For more info concerning this product and the latest version of this manual, please visit our website www.velleman.eu. The information in this manual is subject to change without prior notice.

Velleman® Service and Quality Warranty

Since its foundation in 1972, Velleman® acquired extensive experience in the electronics world and currently distributes its products in over 85 countries.

All our products fulfil strict quality requirements and legal stipulations in the EU. In order to ensure the quality, our products regularly go through an extra quality check, both by an internal quality department and by specialized external organisations. If, all precautionary measures notwithstanding, problems should occur, please make appeal to our warranty (see guarantee conditions).

General Warranty Conditions Concerning Consumer Products (for EU):

• All consumer products are subject to a 24-month warranty on production flaws and defective material as from the original date of purchase.

• Velleman® can decide to replace an article with an equivalent article, or to refund the retail value totally or partially when the complaint is valid and a free repair or replacement of the article is impossible, or if the expenses are out of proportion.

You will be delivered a replacing article or a refund at the value of 100% of the purchase price in case of a flaw occurred in the first year after the date of purchase and delivery, or a replacing article at 50% of the purchase price or a refund at the value of 50% of the retail value in case of a flaw occurred in the second year after the date of purchase and delivery.

• Not covered by warranty:
- all direct or indirect damage caused after delivery to the article (e.g. by oxidation, shocks, falls, dust, dirt, humidity...), and by the article, as well as its contents (e.g. data loss), compensation for loss of profits;

- consumable goods, parts or accessories that are subject to an aging process during normal use, such as batteries (rechargeable, non-rechargeable, built-in or replaceable), lamps, rubber parts, drive belts... (unlimited list);

- flaws resulting from fire, water damage, lightning, accident, natural disaster, etc.…;

- flaws caused deliberately, negligently or resulting from improper handling, negligent maintenance, abusive use or use contrary to the manufacturer’s instructions;

- damage caused by a commercial, professional or collective use of the article (the warranty validity will be reduced to six (6) months when the article is used professionally);

- damage resulting from an inappropriate packing and shipping of the article;

- all damage caused by modification, repair or alteration performed by a third party without written permission by Velleman®.

• Articles to be repaired must be delivered to your Velleman® dealer, solidly packed (preferably in the original packaging), and be completed with the original receipt of purchase and a clear flaw description.

• Hint: In order to save on cost and time, please reread the manual and check if the flaw is caused by obvious causes prior to presenting the article for repair. Note that returning a non-defective article can also involve handling costs.

• Repairs occurring after warranty expiration are subject to shipping costs.

• The above conditions are without prejudice to all commercial warranties.

The above enumeration is subject to modification according to the article (see article’s manual).

Made in PRC
Imported by Velleman nv
Legen Heirweg 33, 9890 Gavere, Belgium
www.velleman.eu