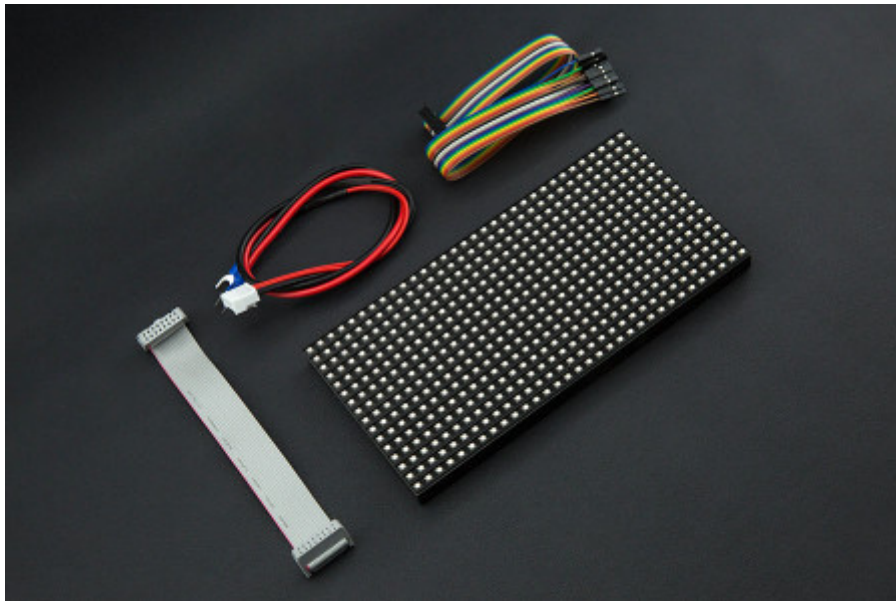# 32x16 RGB LED Matrix Panel (6mm pitch)

SKU:DFR0471

INTRODUCTION

City night is always beautiful. She is just like a charming girl, showing her beauty every time. The colorful light is jewelry, dotted on her dress.

This is a 32x16 RGB LED Matrix Panel, it has 512 full-color RGB LEDs in all. Each LED can be independently addressed and controlled. It requires at least 13 digital GPIOs to control the LED matrix. The led matrix has 2 IDC connectors (DATA_IN, DATA_OUT) on the back, you can cascade multiple panels and make a huge screen together. BUT Arduino doesn't support this function, its speed is not enough to multiple panel.

What's more, it is a high brightness, long life, no pollution, pure color LED display module. It can be used both indoor and outdoor, safety and stability, the module can not be mutually extrusion deformation, also can be used normally in harsh environment.

Note: This panel needs a 5V@2A power supply. And if you connect 4 or more screen, it needs a high performance controller, such as Mega 2560, Raspberry. .etc.

SPECIFICATION

- Operating voltage: DC 5V

- Average power consumption: <500W/㎡

- Maxim Power Consumption: <1000w/㎡

- Pixel: 32x16=512

- Level of viewing Angle: ≧160°

- Control mode: Synchronous control

- Drive mode: 1/8 scan rate

- Repetition frequency: ≧60Hz

- White Balance Brightness: ≧1200cd/㎡

- Refresh frequency : ≧300Hz

- MTTF: ≧5000 hours

- Service Life: 75000~100000 hours

- Pixel pitch: 6mm

- Dimension: 192mm*96mm/7.56*3.78 inches

- Thickness: 11mm

SHIPPING LIST

- 32x16 RGB LED Matrix Panel (6mm pitch) x1

- IDC to XH2.54 cable x1

- IDC to IDC cable x1

- Power supply cable x1

Board Overview

DATA-IN

DATA-OUT

POWER
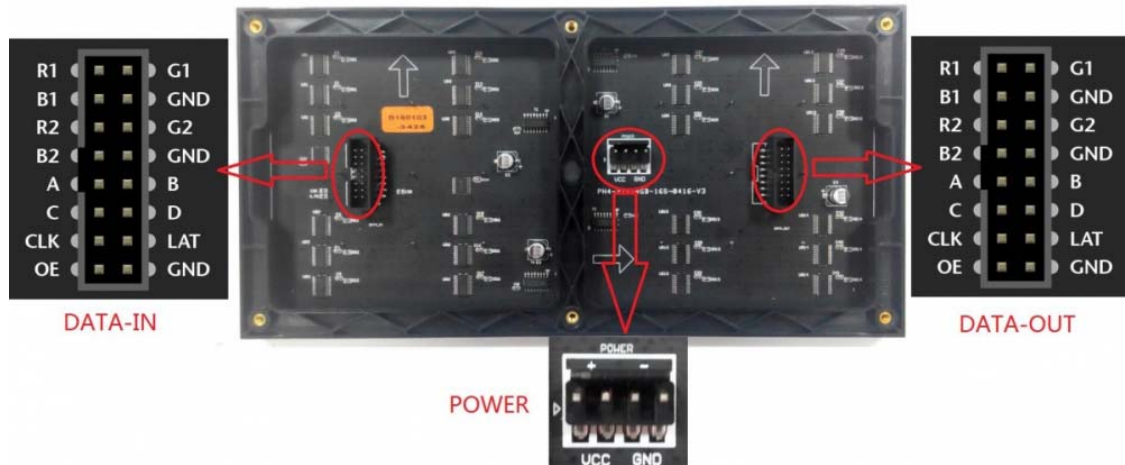
DATA-IN and DATA-OUT		
Label	**Name**	**Function**
1	DR1	High R data
2	DG1	High G data
3	DB1	High B data
4	GND	GND
5	DR2	Low R data
6	DG2	Low G data
7	DB2	Low B data
8	GND	GND
9	A	A line selection
10	B	B line selection
11	C	C line selection
12	D	D line selection
13	CLK	CLOCK

14	LAT	LATCH
15	OE	Output Enable
16	GND	GND

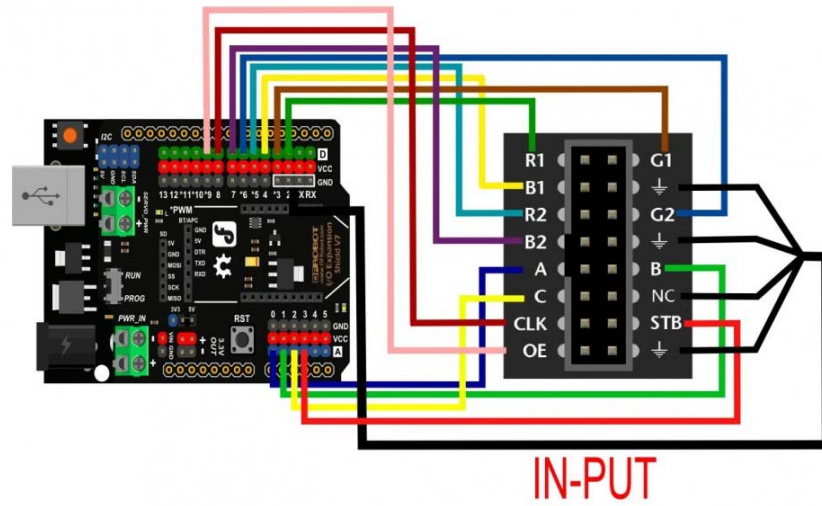| POWER		
Label	**Name**	**Function**
1	VCC	5V
2	VCC	5V
3	GND	GND
4	GND	GND

Tutorial

According to the pinout to connect, then upload the code to UNO, you will be able to see a beautiful display effect.

Requirements

- **Hardware**
  DFRduino UNO R3 x1
  DFR0471 x1
  DuPont cables

- **Software**
  Arduino IDE Click to Download Arduino IDE from Arduino®
  https://www.arduino.cc/en/Main/Software%7C
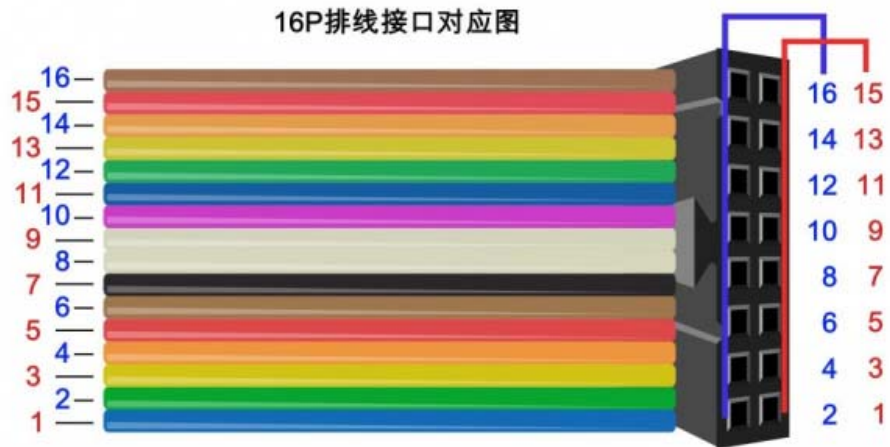
Connection Diagram

DFR0471 Connection Diagram

⚠️ Note: It needs an external power supply, the USB is only 5V@500mA, not enough power.

16P Interface Diagram

16P排线接口对应图

DFR0471 16P Pinout

Sample Code 1

Click to download the library Adafruit-GFX-Library. RGB-matrix-Panel. How to install the library?

https://github.com/adafruit/Adafruit-GFX-Library/archive/master.zip

https://github.com/adafruit/RGB-matrix-Panel

http://www.dfrobot.com.cn/community/forum.php?mod=viewthread&tid=1854&page=1&extra=#pid6955

```
/*************************************************
*
* For 32x16 RGB LED matrix.
*
* @author lg.gang
* @version  V1.0
* @date  2016-10-28
*
* GNU Lesser General Public License.
* See <http://www.gnu.org/licenses/> for details.
* All above must be included in any redistribution
* *************************************************/


#include <Adafruit_GFX.h>   // Core graphics library
#include <RGBmatrixPanel.h> // Hardware-specific library

#define CLK 8  // MUST be on PORTB! (Use pin 11 on Mega)
#define LAT A3
#define OE  9
#define A   A0
#define B   A1
#define C   A2
RGBmatrixPanel matrix(A, B, C, CLK, LAT, OE, false);

void setup() {
  matrix.begin();
```

```
  }


void loop() {


  // draw a pixel in solid white
  matrix.drawPixel(0, 0, matrix.Color333(7, 7, 7));
  delay(500);


  // fix the screen with green
  matrix.fillRect(0, 0, 32, 16, matrix.Color333(0, 7, 0));
  delay(500);


  // draw a box in yellow
  matrix.drawRect(0, 0, 32, 16, matrix.Color333(7, 7, 0));
  delay(500);


  // draw an 'X' in red
  matrix.drawLine(0, 0, 31, 15, matrix.Color333(7, 0, 0));
  matrix.drawLine(31, 0, 0, 15, matrix.Color333(7, 0, 0));
  delay(500);


  // draw a blue circle
  matrix.drawCircle(7, 7, 7, matrix.Color333(0, 0, 7));
  delay(500);


  // fill a violet circle
  matrix.fillCircle(23, 7, 7, matrix.Color333(7, 0, 7));
  delay(500);


  // fill the screen with 'black'
  matrix.fillScreen(matrix.Color333(0, 0, 0));


  // draw some text!
  matrix.setCursor(1, 0);   // start at top left, with one pixel
of spacing
```

```
    matrix.setTextSize(1);     // size 1 == 8 pixels high


    // print each letter with a rainbow color
    matrix.setTextColor(matrix.Color333(7,0,0));
    matrix.print('1');
    matrix.setTextColor(matrix.Color333(7,4,0));
    matrix.print('6');
    matrix.setTextColor(matrix.Color333(7,7,0));
    matrix.print('x');
    matrix.setTextColor(matrix.Color333(4,7,0));
    matrix.print('3');
    matrix.setTextColor(matrix.Color333(0,7,0));
    matrix.print('2');


    matrix.setCursor(1, 9);    // next line
    matrix.setTextColor(matrix.Color333(0,7,7));
    matrix.print('*');
    matrix.setTextColor(matrix.Color333(0,4,7));
    matrix.print('R');
    matrix.setTextColor(matrix.Color333(0,0,7));
    matrix.print('G');
    matrix.setTextColor(matrix.Color333(4,0,7));
    matrix.print('B');
    matrix.setTextColor(matrix.Color333(7,0,4));
    matrix.print("*");
    delay(5000);
}
```

Smaple Code 2

```
/*************************************************
*
* For 32x16 RGB LED matrix.
```

```
 *
 * @author lg.gang
 * @version  V1.0
 * @date  2016-10-28
 *
 * GNU Lesser General Public License.
 * See <http://www.gnu.org/licenses/> for details.
 * All above must be included in any redistribution
 * ************************************************/


#include <Adafruit_GFX.h>   // Core graphics library
#include <RGBmatrixPanel.h> // Hardware-specific library


// Similar to F(), but for PROGMEM string pointers rather than li
terals
#define F2(progmem_ptr) (const __FlashStringHelper *)progmem_ptr


#define CLK 8  // MUST be on PORTB! (Use pin 11 on Mega)
#define LAT A3
#define OE  9
#define A   A0
#define B   A1
#define C   A2
// Last parameter = 'true' enables double-buffering, for flicker-
free,
// buttery smooth animation.  Note that NOTHING WILL SHOW ON THE
DISPLAY
// until the first call to swapBuffers().  This is normal.
RGBmatrixPanel matrix(A, B, C, CLK, LAT, OE, true);
// Double-buffered mode consumes nearly all the RAM available on
the
// Arduino Uno -- only a handful of free bytes remain.  Even the
// following string needs to go in PROGMEM:


const char str[] PROGMEM = "Welcome to DFrobot 32x16 RGB LED Matr
ix";
```

```
int    textX   = matrix.width(),
       textMin = sizeof(str) * -12,
       hue     = 0;
int8_t ball[3][4] = {
  {  3,  0,  1,  1 }, // Initial X,Y pos & velocity for 3 bouncy
balls
  { 17, 15,  1, -1 },
  { 27,  4, -1,  1 }
};
static const uint16_t PROGMEM ballcolor[3] = {
  0x0080, // Green=1
  0x0002, // Blue=1
  0x1000  // Red=1
};


void setup() {
  matrix.begin();
  matrix.setTextWrap(false); // Allow text to run off right edge
  matrix.setTextSize(2);
}


void loop() {
  byte i;


  // Clear background
  matrix.fillScreen(0);


  // Bounce three balls around
  for(i=0; i<3; i++) {
    // Draw 'ball'
    matrix.fillCircle(ball[i][0], ball[i][1], 5, pgm_read_word(&b
allcolor[i]));
    // Update X, Y position
    ball[i][0] += ball[i][2];
    ball[i][1] += ball[i][3];
```

```
      // Bounce off edges
      if((ball[i][0] == 0) || (ball[i][0] == (matrix.width() - 1)))
        ball[i][2] *= -1;
      if((ball[i][1] == 0) || (ball[i][1] == (matrix.height() - 1))
)
        ball[i][3] *= -1;
    }


    // Draw big scrolly text on top
    matrix.setTextColor(matrix.ColorHSV(hue, 255, 255, true));
    matrix.setCursor(textX, 1);
    matrix.print(F2(str));


    // Move text left (w/wrap), increase hue
    if((--textX) < textMin) textX = matrix.width();
    hue += 7;
    if(hue >= 1536) hue -= 1536;


    // Update display
    matrix.swapBuffers(false);
  }
```

Expected Results

- Smaple Code 1: The LED module will take turns display: a white point, full screen green, full screen yellow, a yellow rectangle, a yellow rectangle, a red X and a blue circle,
- Sample Code 2: Display: "Welcome to DFrobot 32x16 RGB LED Matrix"

FAQ

For any questions, advice or cool ideas to share, please visit the **DFRobot Forum**.