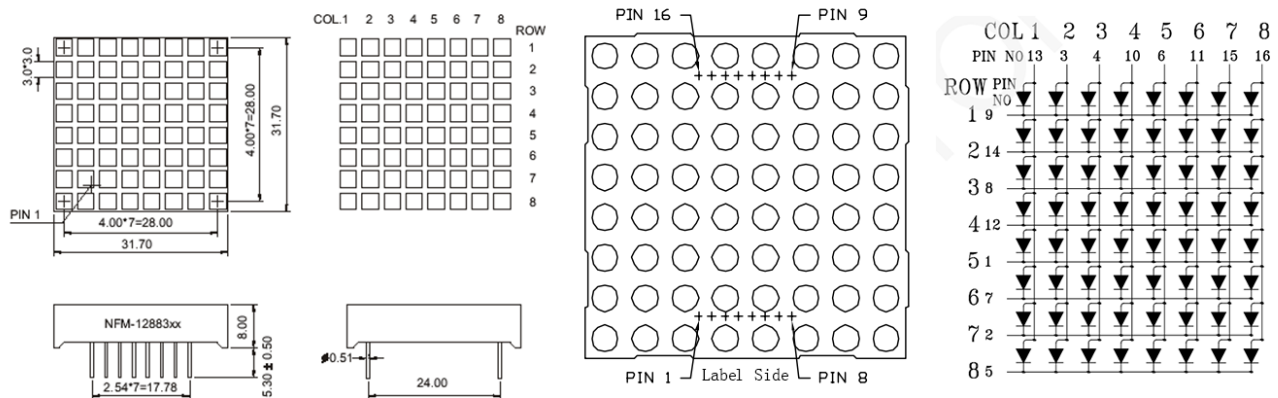

Text Scrolling with Dot Matrix 8x8

LED displays are often packaged as matrixes of LEDs arranged in rows of common anodes and columns of common cathodes, or the reverse. They can be used to display almost anything. Most modern LED sign boards uses various types of matrix boards with controllers. we are going to interface a single-color common cathode 8x8 LED matrix with Raspberry pi Pico and display a string of characters.

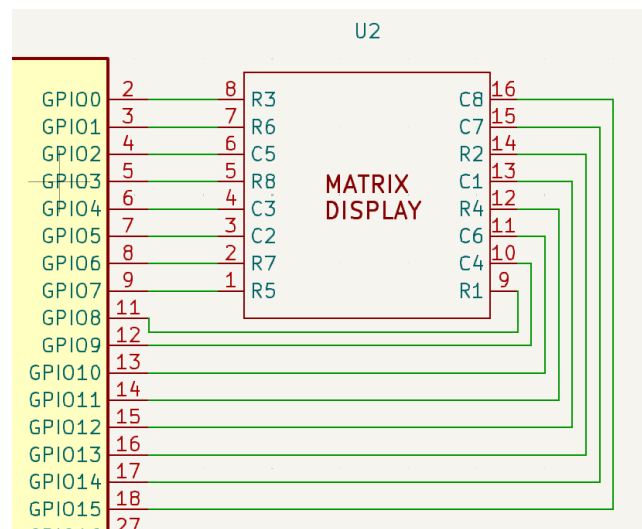
8x8 matrix consists of 64 dots or pixels. There is a LED for each pixel and these LEDs are connected to total of 16 pins.

Package dimensions & circuit and pinout diagram



Matrix of the pin connections, based on the diagram

Matrix Pin no.	Row	Column	Pi Pico's pin number
1	5	-	7
2	7	-	6
3	-	2	5
4	-	3	4
5	8	-	3
6	-	5	2
7	6	-	1
8	3	-	0
9	1	-	8
10	-	4	9
11	-	6	10
12	4	-	11
13	-	1	12
14	2	-	13
15	-	7	14
16	-	8	15



Code and explanation

Libraries and Constants

```
#define ROW_COUNT      8
#define COLUMN_COUNT   8
#define TEXT_LENGTH     56
#define SCROLL_SPEED   5
```

Here, we're defining constants for the number of rows, columns, text length, and scroll speed.

Pin Definitions

```
const int rowPins[ROW_COUNT]      = {8, 13, 0, 11, 7, 1, 6, 3};
const int columnPins[COLUMN_COUNT] = {12, 5, 4, 9, 2, 10, 14, 15};
```

We're defining the pin numbers for rows and columns of your LED matrix.

LED Matrix Pattern

[illegible]

This is the 8x56 matrix representing the text "HelloWorld"

Setup Function

```
void setup() {  
    for (int i = 0; i < ROW_COUNT; i++) {  
        pinMode(rowPins[i], OUTPUT);  
        digitalWrite(rowPins[i], HIGH);  
    }  
  
    for (int i = 0; i < COLUMN_COUNT; i++) {  
        pinMode(columnPins[i], OUTPUT);  
        digitalWrite(columnPins[i], LOW);  
    }  
}
```

In the setup() function, you set up the rows as OUTPUT and initially set them HIGH. Then you set up the columns as OUTPUT and initially set them LOW.

SelectRow Function

```
void selectRow(int row) {  
    for (int i = 0; i < ROW_COUNT; i++) {  
        digitalWrite(rowPins[i], i + 1 == row ? LOW : HIGH);  
    }  
}
```

This function selects a specific row by setting all rows to HIGH except the one passed to the function.

SetLEDInActiveRow Function

```
void setLEDInActiveRow(int column, int state) {  
    digitalWrite(columnPins[column - 1], state);  
}
```

This function sets the state (ON/OFF) of the LED in the active row.

Loop Function

```
void loop() {  
  for (int x = 0; x < 5; x++) {  
    for (int j = 0; j < ROW_COUNT; j++) {  
      selectRow(j + 1);  
      for (int i = 0; i < COLUMN_COUNT; i++) {  
        setLEDInActiveRow(i + 1, Hello_World[j][(i + position)  
- abs((i + position) / (TEXT_LENGTH)) * (TEXT_LENGTH)]);  
      }  
      delay(SCROLL_SPEED);  
    }  
  }  
  position++;  
}
```

The loop() function scrolls the text. It goes through each row, activates the LED in that row according to the pattern defined in HelloWorld, and then scrolls it to the left. It repeats this process until the end of the text. The position variable is used to shift the text to the left with each iteration.