

LED dot-matrix screen

Contents

- [6.1. Attributes](#)
- [6.2. Classes](#)
- [6.3. Example](#)

Pico:ed contains a 17x7 LED dot matrix screen.

6.1. Attributes

display

The instance of class [Display](#) represents the LED dot screen on board Pico:ed.

6.2. Classes

class Display

It is to represent a dot matrix screen driven by IS31FL3731 which is used internally in Pico:ed to generate display instances.

```
**clear()**
```

Clear the screen

```
**pixel(self, x, y, color=None, blink=None, frame=None)**
```

Set x, y pixel blink or brightness

- ****x** - ****horizontal pixel position**
- ****y** - ****vertical pixel position**
- ****color** - ****pixel brightness (0~255)**
- ****blink** - ****Set to "True", then blink the pixel**

Print to PDF ****frame** - ****sets the number of frames of the pixel**

```
**scroll(value, brightness=30)**
```

Scroll to show strings or numbers

- ****value** - ****the string or number to display**
- ****brightness** - **** brightness of the display (0~255), default value 30**

class Image

It is to describe a frame of dot matrix images, and contains a series of built-in images.

- ****value** - ****Image text description, it uses 0 to 9 to specify the brightness of the LED at the corresponding position. The format is as follows:**

'12345678908765432:'



The list of built-in images is as follows:

- NO
- SQUARE
- RECTANGLE
- RHOMBUS
- TARGET
- CHESSBOARD
- HAPPY
- SAD
- YES
- HEART
- TRIANGLE
- CHAGRIN
- SMILING_FACE
- CRY
- DOWNCAST
- LOOK_RIGHT
- LOOK_LEFT
- TONGUE
- PEEK_RIGHT
- PEEK_LEFT
- TEAR_EYES
- PROUD
- SNEER_LEFT
- SNEER_RIGHT
- SUPERCILIOUS_LOOK
- EXCITED

6.3. Example

1. Display strings and numbers

```
import time
from piced import display

# Display statically of 3 letters
display.scroll("ABC")
time.sleep(1)

# Scroll to display the numbers
display.scroll(1234567890)

# Scroll to display the strings
display.scroll("Hello, world!")
```

1. Display boxes

```
from picoed import display

# Draw a box on the screen
# Firstly, draw the top and bottom edges
for x in range(display.width):
    display.pixel(x, 0, 50)
    display.pixel(x, display.height - 1, 50)
# Now draw the left and right edges
for y in range(display.height):
    display.pixel(0, y, 50)
    display.pixel(display.width - 1, y, 50)
```

1. Display the built-in image

```
from picoed import display, Image

display.show(Image.HAPPY)
```

1. Display self-defined images

```
from picoed import display, Image

while True:
    display.show(Image(
        '12345678987654321:'
        '12345678987654321:'
        '12345678987654321:'
        '12345678987654321:'
        '12345678987654321:'
        '12345678987654321:'
        '12345678987654321:'
    ))
    display.show(Image(
        '87654321112345678:'
        '87654321112345678:'
        '87654321112345678:'
        '87654321112345678:'
        '87654321112345678:'
        '87654321112345678:'
        '87654321112345678:'
    ))
```