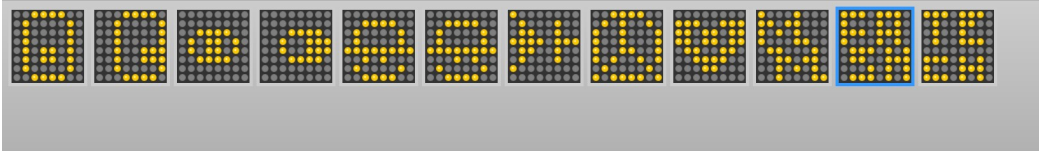


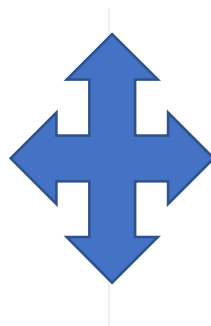
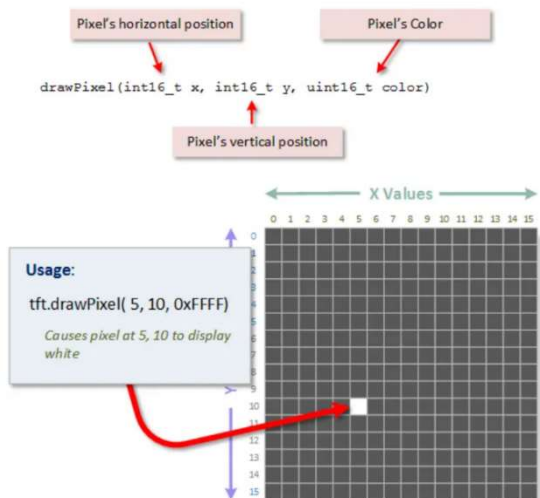
# LED Box Purse

<https://learn.adafruit.com/neopixel-matrix-snowflake-sweater/flora>

8x8 matrix patterns for Original reference--



In the end have altered some of the patterns

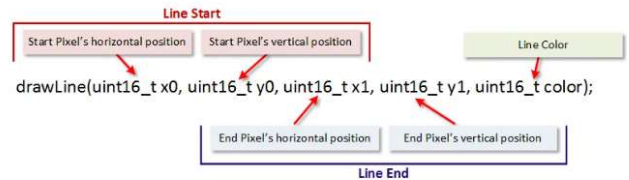


These are the best explanations of how to draw shape with the coordinate system

## How to Use It

To use this command, you provide:

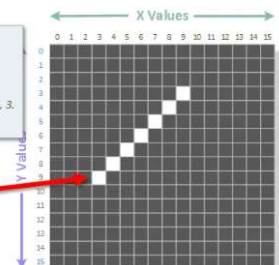
- A pair on integers that specify the x and y position of the line's starting point.
- A pair on integers that specify the x and y position of the line's ending point.
- A sixteen bit color code specifying the color of the line.



## drawPixel() Sample Sketch

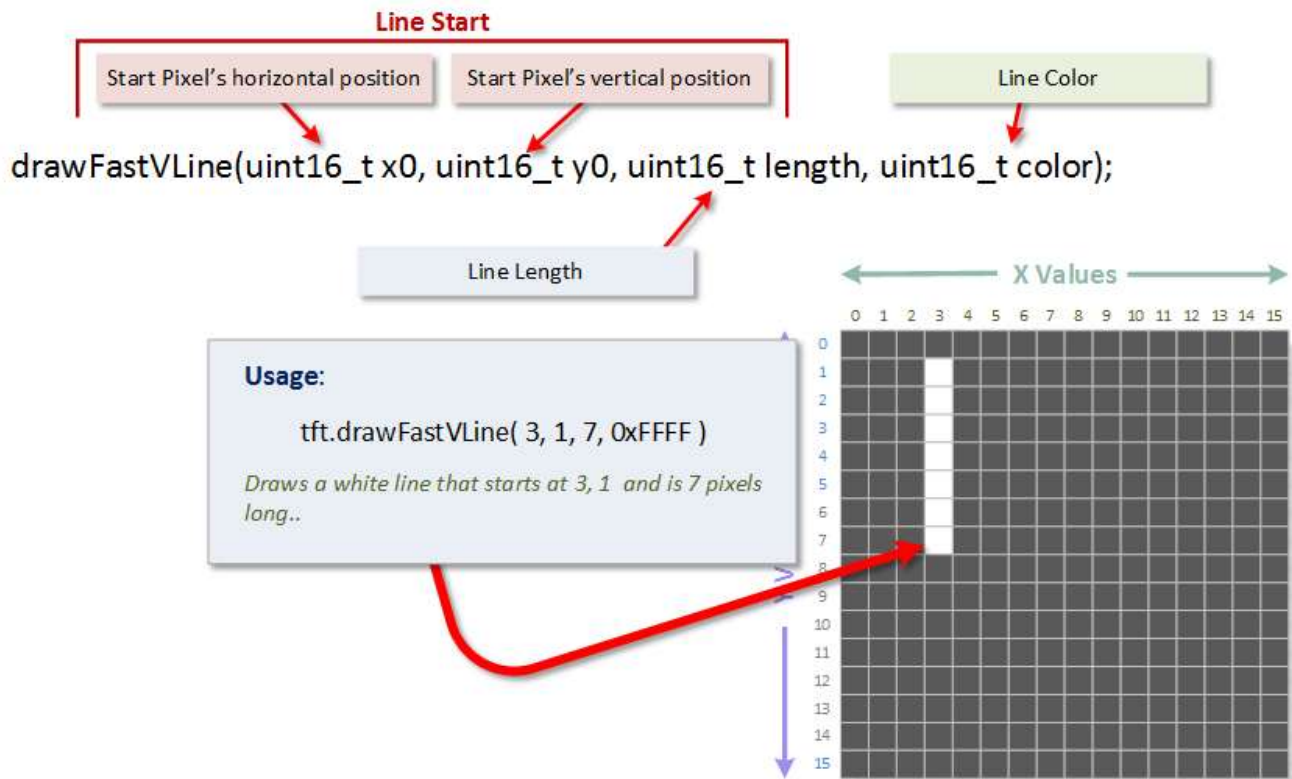
This example sketch draws a animated dot that moves across the screen horizontal. Basically we:

- 'Light' a pixel
- Delay for a little bit
- Turn the Pixel Off
- Then light the pixel to it's immediate right.



**Draws a line**

Figure 1 Draw a line



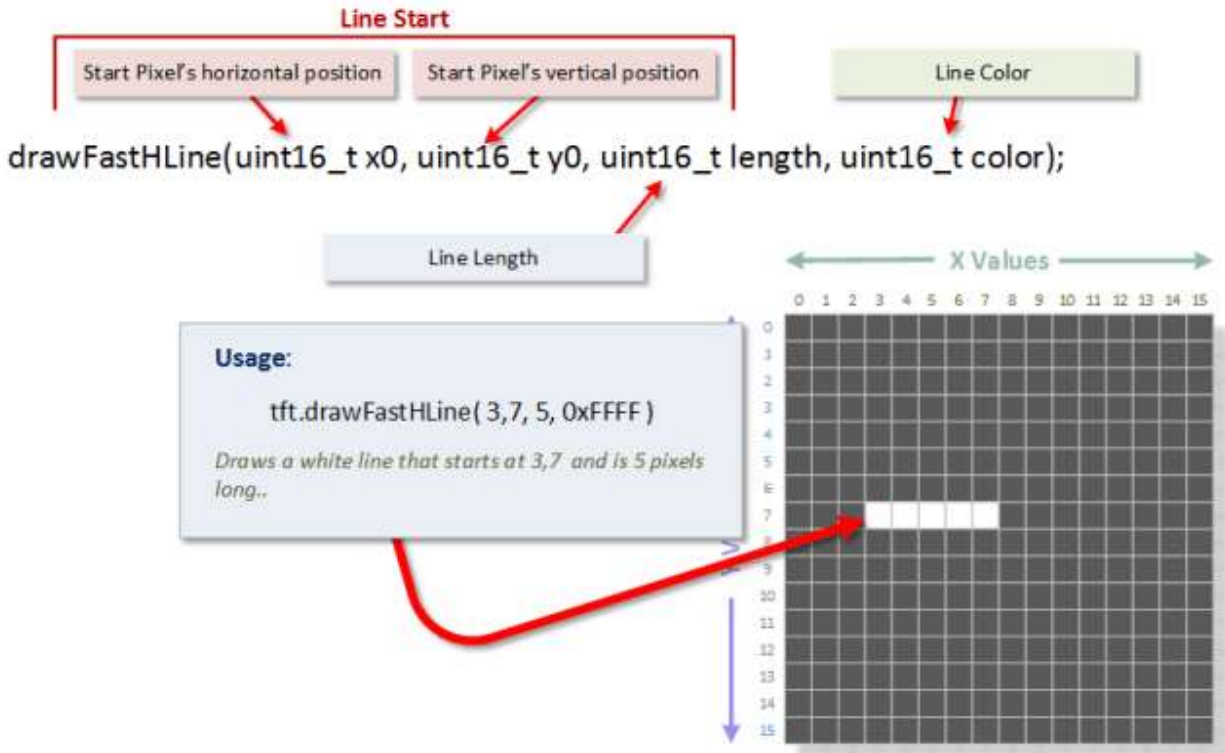
## drawFastVLine() Sample Sketch

This sketch simply draws an animated vertical line that is 30 pixels long. It moves from the top center of the display to the bottom center of the display

### How to Use It

To use this command, you provide:

- Two integers representing the horizontal and vertical position of the origin
- A line length
- A 16 bit color
- .



## How to Use It

To use this command, you insert:

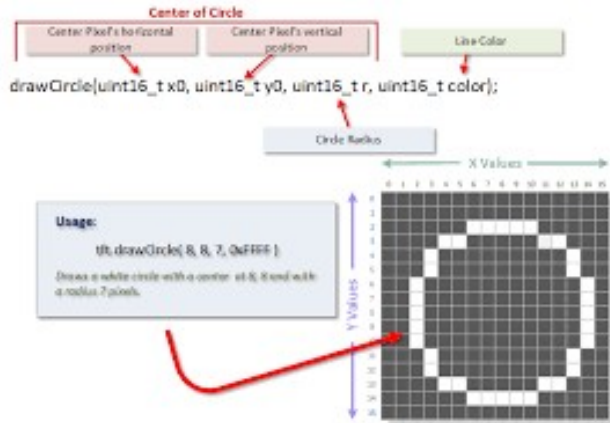
- Two integers representing the horizontal and vertical position of the origin
- A line length
- A 16 bit color

### What is Does

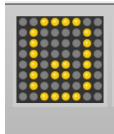
It draws a horizontal line of the length and color you specify. The key to using this routine is that this is a line that goes from left to right. Like `drawFastVLine()`, it is faster than `drawLine()`.

Matrix Draw Rectangle---?? matrix

Fill rectangle?---- matrix.fillRect



Block-9-10



```
matrix.drawFastHLine(2, 0, 4, c); // x0, y0, length, color
matrix.drawFastVLine( 1, 1, 6, c); // x0, y0, length, color
matrix.drawFastVLine( 6, 1, 6, c); // x0, y0, length, color
matrix.drawRect(4, 4, 2, 2,); // x0, y0, width, height
matrix.drawFastHLine(2, 7, 4, c); // x0, y0, length, color
```

Block-10

```
matrix.drawFastHLine(3, 0, 4, c); // x0, y0, length, color
matrix.drawFastVLine(2, 1, 6, c); // x0, y0, length, color
matrix.drawFastVLine(7, 1, 6, c); // x0, y0, length, color
matrix.drawRect(5, 5, 2, 2, c); // x0, y0, width, height
matrix.drawFastHLine(3, 7, 4, c); // x0, y0, length, color
```

### 8x8 Matrix Grid information:

Orientation of the grid-Bottom edge located under words horizontally as words run vertically

X and y axis accepts input of **0-8** ---  
**drawVLine , drawHLine**

Diagonal line (drawLine) accepts **0-7**

**Bluefruit app:** Location must be on for connection to the FLORA Bluetooth board (?? IDK)

The buttons will only control 8 of the designs. Buttons 1-4 are mapped correctly. The arrows are

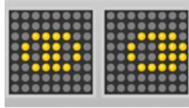
↑ =5 → =8 ↓ =6 ← =7

Color picker- remember brighter consumes mor battery life

The app often takes a minute to connect and may require multiple attempts on the main page to connect to Bluetooth.

When initially connection make sure to connect to the Adafruit Bluefruit LE

Block 3-4



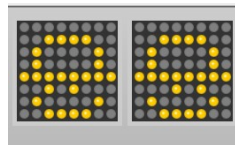
### **Block 3**

```
matrix.drawFastHLine(2, 2, 4, c); // x0, y0, length, color
matrix.drawFastHLine(2, 5, 4, c); // x0, y0, length, color
matrix.drawFastVLine(1, 3, 2, c); // x0, y0, length, color
matrix.drawFastVLine(6, 3, 2, c); // x0, y0, length, color
matrix.drawRect(3, 3, 2, 2, c); // x0, y0, width, height
```

### **Block -4**

```
matrix.drawFastHLine(3, 3, 4, c); // x0, y0, length, color
matrix.drawFastHLine(3, 5, 4, c); // x0, y0, length, color
matrix.drawFastVLine(2, 3, 2, c); // x0, y0, length, color
matrix.drawFastVLine(7, 3, 2, c); // x0, y0, length, color
matrix.drawRect(5, 3, 2, 2, c); // x0, y0, width, height
```

### **Block 5-6**



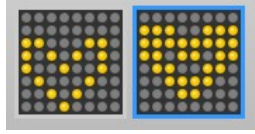
### **Block-5**

```
matrix.drawFastHLine(2, 1, 4, c); // x0, y0, length, color
matrix.drawFastHLine(0, 4, 8, c); // x0, y0, length, color
matrix.drawFastHLine(2, 5, 3, c); // x0, y0, length, color
matrix.drawPixel(3, 5, 0); // x, y, color, removes dot
matrix.drawFastHLine(2, 7, 4, c); // x0, y0, length, color
matrix.drawFastVLine(1, 2, 5, c); // x0, y0, length, color
matrix.drawPixel(1, 5, 0); // x, y, color remove dot
matrix.drawFastVLine(6, 2, 5, c); // x0, y0, length, color
matrix.drawPixel(6, 5, 0); // x, y, color remove dot
```

### **Block 6**

```
matrix.drawFastHLine(2, 0, 4, c); // x0, y0, length, color ,1st line
matrix.drawFastHLine(0, 3, 8, c); // x0, y0, length, color, brim
matrix.drawFastHLine(3, 4, 3, c); // x0, y0, length, color eyes
matrix.drawPixel(4, 4, 0); // x, y, color, removes dot
matrix.drawFastHLine(2, 6, 4, c); // x0, y0, length, color horizontal line
matrix.drawFastVLine(1, 1, 5, c); // x0, y0, length, color vertical line
matrix.drawPixel(1, 4, 0); // x, y, color remove dot
matrix.drawFastVLine(6, 1, 5, c); // x0, y0, length, color
matrix.drawPixel(6, 4, 0); // x, y, color remove dot
```

## Block 7-8



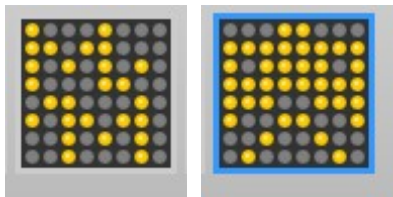
### Block-7

```
matrix.drawLine(0, 4, 3, 7, c); // x0, y0, x1, y1, color bottom left
matrix.drawLine(1, 2, 3, 4, c); // x0, y0, x1, y1, color top left
matrix.drawLine(4, 3, 5, 2, c); // x0, y0, x1, y1, color top right
matrix.drawLine(4, 6, 6, 4, c); // x0, y0, x1, y1, color borrom right
matrix.drawFastVLine(0, 2, 3, c); //left vertical line
matrix.drawFastVLine(6, 2, 3, c); //right vertical line
```

### Block-8

```
matrix.drawPixel(0, 1, c); // x, y, color left
matrix.drawPixel(7, 1, c); // x, y, color right
matrix.drawRect(1, 1, 2, 2, c); // x0, y0, width, height, left rect
matrix.drawRect(3, 2, 2, 2, c); // x0, y0, width, height, middle rect
matrix.drawRect(5, 1, 2, 2, c); // x0, y0, width, height right rect
matrix.drawLine(0, 2, 3, 5, c); // x0, y0, x1, y1, color left vertical-1
matrix.drawLine(0, 3, 3, 6, c); // x0, y0, x1, y1, color, left vert-2
matrix.drawLine(4, 5, 7, 2, c); // x0, y0, x1, y1, color, right vertical-1
matrix.drawLine(4, 6, 7, 3, c); // x0, y0, x1, y1, color right vertical-2
```

## Block -1,2,,11,12



### Block -1 Logo

```
matrix.drawFastVLine(0, 0, 4, c); //left vertical line top
matrix.drawFastVLine(4, 0, 4, c); //right vertical line top
matrix.drawFastVLine(2, 4, 4, c); //left vertical line, bottom
matrix.drawFastVLine(6, 4, 4, c); //right vertical line, bottom
```

//centers of M

```
matrix.drawPixel(1, 1, c); // x, y, color, top m
matrix.drawPixel(2, 2, c); // x, y, color, top m
matrix.drawPixel(3, 1, c); // x, y, color, top m
matrix.drawPixel(3, 5, c); // x, y, color, bottom m
```

```
matrix.drawPixel(4, 6, c); // x, y, color, bottom m
matrix.drawPixel(5, 5, c); // x, y, color, bottom m
```

```
//tails of M
```

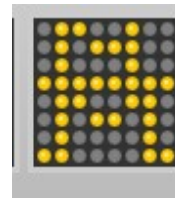
```
matrix.drawPixel(5, 3, c); // x, y, color, top m
matrix.drawPixel(6, 2, c); // x, y, color, top m
matrix.drawPixel(0, 5, c); // x, y, color, bottom m
matrix.drawPixel(2, 4, c); // x, y, color, bottom m
```

## Block-2

```
matrix.fillRect(0, 1, 8, 4, c); // x0, y0, width, height -mask
matrix.drawPixel(1, 2, 0); // x, y, color remove pixels, eye
matrix.drawPixel(6, 2, 0); // x, y, color remove pixels, eye
matrix.drawPixel(4, 4, 0); // x, y, color remove pixels
matrix.drawPixel(3, 4, 0); // x, y, color remove pixels
matrix.drawPixel(0, 5, c); // x, y, color
matrix.drawPixel(7, 5, c); // x, y, color
matrix.drawFastHLine(3, 0, 2, c); // x0, y0, length, top dot
matrix.drawLine(1, 7, 3, 5, c); //stand
matrix.drawLine(6, 7, 4, 5, c); //stand
```

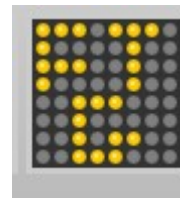
## Block 11- logo icon

```
matrix.drawFastVLine(0, 0, 8, c); //left vertical line
matrix.drawFastHLine(0, 3, 8, c); //Horizontal line brim
matrix.drawFastVLine(5, 0, 5, c); //RT vertical line, top
matrix.drawFastVLine(6, 3, 5, c); //RT vertical line, bottom
```



```
//M centers dot
```

```
matrix.drawPixel(2, 0, c); // x, y, color, top M
matrix.drawPixel(3, 1, c); // x, y, color, top m
matrix.drawPixel(4, 1, c); // x, y, color, top m
matrix.drawPixel(2, 4, c); // x, y, color, bottom m
matrix.drawPixel(3, 5, c); // x, y, color, bottom m
```



```
matrix.drawPixel(4, 5, c); // x, y, color, top m tail
matrix.drawPixel(6, 4, c); // x, y, color, top m tail
matrix.drawPixel(0, 7, c); // x, y, color, bottom m tail
matrix.drawPixel(7, 7, c); // x, y, color, bottom m tail
```



## Block 12

```
matrix.drawFastHLine(0, 0, 7, c); // x0, y0, x1, color top horiz line-- FTG
matrix.drawFastHLine(1, 2, 2, c); // x0, y0, x1, color F horiz line
matrix.drawPixel(3, 0, 0); // x, y, color remove dot
matrix.drawFastVLine(0, 0, 4, c); // top letter F
matrix.drawFastVLine(5, 1, 3, c); // top letter rt-T
matrix.drawFastVLine(2, 4, 4, c); // bottom letter rt-G
matrix.drawFastHLine(3, 4, 2, c); // bottom letter
matrix.drawFastHLine(4, 6, 2, c); // bottom letter
matrix.drawFastHLine(3, 7, 2, c); // bottom letter
matrix.show();
```

## Notes---

For some reason I will figure out later I can't seem to get 12th block to show ( I am sure it will occur to me later) But 11 shows twice and holds. The bluetooth app is a bit quirky and the location must be on for the app to work ???IDK

