Hub Reference Guide

Learn how to use the Spike Hub display and buttons!

Import the Library

```
import runloop
from hub import light_matrix, button, sound, light
```

Hub Display Basics

Show Images

```
# Show a built-in image
light_matrix.show_image(light_matrix.IMAGE_HAPPY)

# Other images you can use:
# IMAGE_HAPPY, IMAGE_SAD, IMAGE_HEART, IMAGE_SMILE
# IMAGE_YES, IMAGE_NO
# IMAGE_ARROW_N, IMAGE_ARROW_E, IMAGE_ARROW_S, IMAGE_ARROW_W
```

Show Text

```
# Show a single character
light_matrix.write('A')

# Scroll text across the display
light_matrix.write('HELLO')
```

Custom Patterns

```
# Create your own 5x5 pattern (0=off, 9=brightest)
pattern = [
    [0, 0, 9, 0, 0],
    [0, 9, 0, 9, 0],
    [9, 0, 0, 0, 9],
    [0, 9, 0, 9, 0],
    [0, 0, 9, 0, 0]
]
light_matrix.show(pattern)
```

Turn Off Display

```
light_matrix.clear()
```

Hub Buttons

Check Left Button

```
# Is left button pressed?
if button.pressed(button.LEFT) > 0:
    print("Left button pressed!")

# Wait for left button (use async in runloop)
async def main():
    await button.wait_until_pressed(button.LEFT)
    print("Left button was pressed!")

runloop.run(main())
```

Check Right Button

```
# Is right button pressed?
if button.pressed(button.RIGHT) > 0:
    print("Right button pressed!")

# Wait for right button (use async in runloop)
async def main():
    await button.wait_until_pressed(button.RIGHT)
    print("Right button was pressed!")

runloop.run(main())
```

Button Release

```
# Wait until button is released (use async in runloop)
async def main():
    await button.wait_until_released(button.LEFT)
    print("Button released!")

runloop.run(main())
```

Hub Speaker

Play Beep

```
# Simple beep
sound.beep(60, 500)  # Note 60, for 500 milliseconds

# Higher pitch
sound.beep(72, 500)
```

Change Volume

```
# Set volume (0-10)
sound.volume(8)
```

Status Light

The light around the center button!

Change Color

```
# Turn on status light
light.color(light.POWER, light.GREEN)

# Other colors: RED, BLUE, YELLOW, CYAN, MAGENTA, WHITE, ORANGE
```

Turn Off

```
light.color(light.POWER, light.BLACK)
```

Quick Examples

Example 1: Button Counter

```
import runloop
from hub import light_matrix, button

count = 0

async def main():
    global count
    while count < 5:
        await button.wait_until_pressed(button.LEFT)
```

```
            count = count + 1
            light_matrix.write(str(count))
            await button.wait_until_released(button.LEFT)

        light_matrix.show_image(light_matrix.IMAGE_HAPPY)

    runloop.run(main())
```

Example 2: Emoji Faces

```
import runloop
from hub import light_matrix, button

async def main():
    while True:
        if button.pressed(button.LEFT) > 0:
            light_matrix.show_image(light_matrix.IMAGE_HAPPY)
        elif button.pressed(button.RIGHT) > 0:
            light_matrix.show_image(light_matrix.IMAGE_SAD)
        else:
            light_matrix.show_image(light_matrix.IMAGE_SMILE)
        await runloop.sleep_ms(100)

runloop.run(main())
```

Example 3: Countdown Timer

```
import runloop
from hub import light_matrix, sound

async def main():
    for i in range(5, 0, -1):
        light_matrix.write(str(i))
        sound.beep(60, 200)
        await runloop.sleep_ms(1000)

    light_matrix.show_image(light_matrix.IMAGE_YES)
    sound.beep(72, 1000)

runloop.run(main())
```

Example 4: Animation Loop

```
import runloop
from hub import light_matrix
```

```
images = [
    light_matrix.IMAGE_ARROW_N,
    light_matrix.IMAGE_ARROW_E,
    light_matrix.IMAGE_ARROW_S,
    light_matrix.IMAGE_ARROW_W
]

async def main():
    for i in range(8):  # Loop twice
        for image in images:
            light_matrix.show_image(image)
            await runloop.sleep_ms(300)

runloop.run(main())
```

Example 5: Traffic Light

```
import runloop
from hub import light_matrix, light

async def main():
    # Red light
    light.color(light.POWER, light.RED)
    light_matrix.write('STOP')
    await runloop.sleep_ms(2000)

    # Yellow light
    light.color(light.POWER, light.YELLOW)
    light_matrix.write('WAIT')
    await runloop.sleep_ms(1000)

    # Green light
    light.color(light.POWER, light.GREEN)
    light_matrix.write('GO')
    await runloop.sleep_ms(2000)

runloop.run(main())
```

Common Display Patterns

Smiley Face

```
smiley = [
    [0, 9, 0, 9, 0],
    [0, 9, 0, 9, 0],
    [0, 0, 0, 0, 0],
    [9, 0, 0, 0, 9],
    [0, 9, 9, 9, 0]
```

```
]
light_matrix.show(smiley)
```

Heart

```
heart = [
    [0, 9, 9, 0, 0],
    [9, 9, 9, 9, 0],
    [9, 9, 9, 9, 9],
    [0, 9, 9, 9, 0],
    [0, 0, 9, 0, 0]
]
light_matrix.show(heart)
```

X Pattern

```
x_pattern = [
    [9, 0, 0, 0, 9],
    [0, 9, 0, 9, 0],
    [0, 0, 9, 0, 0],
    [0, 9, 0, 9, 0],
    [9, 0, 0, 0, 9]
]
light_matrix.show(x_pattern)
```

Happy Coding! 🎉