cedargrove_widgets.bubble_display

A CircuitPython DisplayIO.Group class for the 6E5 Triode Indicator “Magic Eye” display widget.

- Author(s): JG for Cedar Grove Studios

class cedargrove_widgets.bubble_display.BubbleDisplay(*, digits=1, mode=“Normal”, center=(0.5, 0.5), size=1.0, display_size=(None, None))

A CircuitPython class representing a multi-digit 7-segment numeric end-stackable LED display widget. The class creates a DisplayIO.Group object consisting of sub-groups for unit, digits, and digit segments. The widget is based on the HP_QDSP-6064 4-digit and the HP_5802-7433 3-digit 7-segment numeric end-stackable LED displays.

This widget class displays decimal numeric values as well as alphanumeric strings (with a limited character set). Decimal values are right-justified with the decimal point position specified by the mode parameter. Strings are also right-justified. Alpha string characters are limited to upper or lower case 'a', 'b', 'c', 'd', 'e', 'f', 'x', '.', '-', and the space character.

Defaults to an object with display center (0.5, 0.5) and radius of 0.5, specified as normalized display units (not pixels).

Parameters:

- **units** – The integer number of end-stacked widget units each containing a cluster of digits. Defaults to 1 widget.
- **digits** – The integer number of digits per unit ranging from 1 to 5 digits. Defaults to 4 digits per unit.
- **mode** —The decimal point display mode string. The default “Normal” mode places the decimal point within the ones (10^0) digit; in “HP-35” mode, the decimal point is placed in a separate digit between the ones and one-tenth (10^{-1}) digits.
- **center** – The floating point width and height tuple value representing the center of the Magic Eye specified as normalized display units. Defaults to display width and height (0.5, 0.5).
- **size** – The widget size factor relative to the display’s shorter axis. Defaults to 1.0.
- **display_size** — The host display's integer width and height tuple value in pixels. If (None, None) and the host includes a built-in display, the value becomes (board.DISPLAY.width, board.DISPLAY.height).

Properties

units

A class property that returns an integer value of the number of end-stacked widget units.

digits

A class property that returns an integer value of the number of digits per unit.

mode

A class property that returns a string representing the display’s decimal point mode.

center

A class property that returns a floating point tuple of the widget's center (width, height).

> size

A class property that returns a floating point value of the widget size.

> display_size

A class property that returns an integer tuple of the display size (width, height).

> value

A class property that sets or returns the BubbleDisplay's numeric value. Numeric values are right-justified. When a displayable value is exceeded, the BubbleDisplay will show all dashes.

> text

A class property that sets or returns the BubbleDisplay's string value. Text characters are right-justified; characters beyond the left-most digit of the display are truncated. Hexadecimal strings are displayed with an underscore replacing the "X" character.

```
# For host board with integral display (PyPortal, Clue, FunHouse, etc.)

import board
import random
import time

from cedargrove_widgets.bubble_display import BubbleDisplay

display = board.DISPLAY

# Instantiate the BubbleDisplay widget, 4 digits in a single unit
# Locate at display center (0.5, 0.5) with size = 2, "Normal" mode

bubble_display = BubbleDisplay(size=2)

display.show(bubble_display)

while True:
    # Count up and down
    for i in range(0, 100, 1):
        m = i / 100
        bubble_display.value = m
        time.sleep(0.01)
    for i in range(100, 0, -1):
        m = i / 100
        bubble_display.value = m
        time.sleep(0.01)
    time.sleep(2)

    # Display some text
    bubble_display.text = "face"
    time.sleep(2)

    # Count in hexadecimal
    for i in range(0, 256):
        bubble_display.text = hex(i)
        time.sleep(0.01)
    time.sleep(2)
```