

 kilogramham Cleanup 3rd party samples; update README.md; add some missing copyrig... Latest commit fa09f2c on Oct 28, 2021 History

Ax 2 contributors  

108 lines (93 sloc) | 2.57 KB Raw Blame ...

```
1  /**
2   * Copyright (c) 2020 Raspberry Pi (Trading) Ltd.
3   *
4   * SPDX-License-Identifier: BSD-3-Clause
5   */
6
7  #include <stdio.h>
8  #include <stdlib.h>
9
10 #include "pico/stdlib.h"
11 #include "hardware/pio.h"
12 #include "hardware/clocks.h"
13 #include "ws2812.pio.h"
14
15 #define IS_RGBW true
16 #define NUM_PIXELS 150
17
18 #ifdef PICO_DEFAULT_WS2812_PIN
19 #define WS2812_PIN PICO_DEFAULT_WS2812_PIN
20 #else
21 // default to pin 2 if the board doesn't have a default WS2812 pin defined
22 #define WS2812_PIN 2
23 #endif
24
25 static inline void put_pixel(uint32_t pixel_grb) {
26     pio_sm_put_blocking(pio0, 0, pixel_grb << 8);
27 }
28
29 static inline uint32_t urgb_u32(uint8_t r, uint8_t g, uint8_t b) {
30     return
31         ((uint32_t) (r) << 8) |
32         ((uint32_t) (g) << 16) |
33         (uint32_t) (b);
34 }
35
36 void pattern_snakes(uint len, uint t) {
37     for (uint i = 0; i < len; ++i) {
38         uint x = (i + (t >> 1)) % 64;
39         if (x < 16) {
40             put_pixel(urgb_u32(0xff, 0, 0));
41         } else if (x >= 16 && x < 32) {
42             put_pixel(urgb_u32(0, 0xff, 0));
43         } else if (x >= 32 && x < 48) {
44             put_pixel(urgb_u32(0, 0, 0xff));
45         } else {
46             put_pixel(0);
47         }
48     }
49 }
50
51 void pattern_random(uint len, uint t) {
52     if (t % 8)
53         return;
54     for (int i = 0; i < len; ++i)
55         put_pixel(rand());
56 }
57
58 void pattern_sparkle(uint len, uint t) {
59     if (t % 8)
60         return;
61     for (int i = 0; i < len; ++i)
62         put_pixel(rand() % 16 ? 0 : 0xffffffff);
63 }
64
65 void pattern_greys(uint len, uint t) {
66     int max = 100; // let's not draw too much current!
67     t %= max;
68     for (int i = 0; i < len; ++i) {
69         put_pixel(t * 0x10101);
70         if (++t >= max) t = 0;
71     }
72 }
73
74 typedef void (*pattern)(uint len, uint t);
75 const struct {
76     pattern pat;
77     const char *name;
78 } pattern_table[] = {
79     {pattern_snakes, "Snakes!"},
80     {pattern_random, "Random data"},
81     {pattern_sparkle, "Sparkles"},
82     {pattern_greys, "Greys"},
83 };
84
85 int main() {
86     //set_sys_clock_48();
87     1 stdio_init_all();
88     2 printf("WS2812 Smoke Test, using pin %d", WS2812_PIN);
89
90     // todo get free sm
91     PIO pio = pio0;
92     int sm = 0;
93     uint offset = pio_add_program(pio, &ws2812_program);
94     ws2812_program_init(pio, sm, offset, WS2812_PIN, 800000, IS_RGBW);
95
96     int t = 0;
97     while (1) {
98         int pat = rand() % count_of(pattern_table);
99         int dir = (rand() >> 30) & 1 ? 1 : -1;
100         puts(pattern_table[pat].name);
101         puts(dir == 1 ? "(forward)" : "(backward)");
102         for (int i = 0; i < 1000; ++i) {
103             pattern_table[pat].pat(NUM_PIXELS, t);
104             sleep_ms(10);
105             t += dir;
106         }
107     }
108 }
```

① init hardware

② set offset

③ init_GPIO-PIO

④ loop color pattern

↳ ④ set pixel

④.2 Send to state machine's Tx FIFO

auto

config - send to pin