


```

    }
    %}

    .program ws2812_parallel

    .define public T1 2
    .define public T2 5
    .define public T3 3

    .wrap_target
        out x, 32
        mov pins, !null [T1-1]
        mov pins, x [T2-1]
        mov pins, null [T3-2]
    .wrap

    % c-sdk {
    #include "hardware/clocks.h"

    static inline void ws2812_parallel_program_init(PIO pio, uint sm, uint offset, uint
    pin_base, uint pin_count, float freq) {
        for(uint i=pin_base; i<pin_base+pin_count; i++) {
            pio_gpio_init(pio, i);
        }
        pio_sm_set_consecutive_pindirs(pio, sm, pin_base, pin_count, true);

        pio_sm_config c = ws2812_parallel_program_get_default_config(offset);
        sm_config_set_out_shift(&c, true, true, 32);
        sm_config_set_out_pins(&c, pin_base, pin_count);
        sm_config_set_set_pins(&c, pin_base, pin_count);
        sm_config_set_fifo_join(&c, PIO_FIFO_JOIN_TX);

        int cycles_per_bit = ws2812_parallel_T1 + ws2812_parallel_T2 +
        ws2812_parallel_T3;
        float div = clock_get_hz(clk_sys) / (freq * cycles_per_bit);
        sm_config_set_clkdiv(&c, div);

        pio_sm_init(pio, sm, offset, &c);
        pio_sm_set_enabled(pio, sm, true);
    }
    %}

```