Inicializace vstupů, Zapnutí a nastavení interuptů ,Nastavení časovače

Pulse pin nastavený na OD (Open Drain), HIZ (High Impedance)

HIZ – toto nastavení umožnuje, aby pin byl „odpojen“ a tím pádem nemá téměř žádný vliv

OD – je nastaven kvůli ochraně zařízení

```
void init(void) {
    CLK_HSIPrescalerConfig(CLK_PRESCALER_HSIDIV1); // taktovani MCU na
16MHz

    // Piny
    GPIO_Init(GPIOE, GPIO_PIN_0, GPIO_MODE_IN_PU_IT); // DATA in PIN
    GPIO_Init(GPIOG, GPIO_PIN_0, GPIO_MODE_OUT_OD_HIZ_SLOW); // Trigger
    GPIO_Init(BTN_PORT, BTN_PIN, GPIO_MODE_IN_FL_NO_IT);       // Tlačítko
    GPIO_Init(GPIOD,GPIO_PIN_3,GPIO_MODE_OUT_PP_LOW_SLOW);  //Signalizace

    // Interupty
    EXTI_SetExtIntSensitivity(EXTI_PORT_GPIOE,
EXTI_SENSITIVITY_RISE_FALL);
    ITC_SetSoftwarePriority(ITC_IRQ_PORTE, ITC_PRIORITYLEVEL_1);
    enableInterrupts();

    // Časovač
    TIM4_TimeBaseInit(TIM4_PRESCALER_16, 0xffff);
    TIM4_Cmd(ENABLE);

    // UART
    init_uart1();
    init_milis();
}
```

Hlavní část kódu

```
int main(void) {
    uint32_t current_time = 0;
    uint32_t last = 0;
    int i = 0;


    initialize_system();

    while (1) {
        if (milis() - current_time > 1000 ) {
            GPIO_WriteReverse(LED_PORT, LED_PIN);
            current_time = milis();
            printf("\nTimer value: %u\n", TIM2_GetCounter());
            stav = START;
        }

        switch (stav) {
            case SLEEP:
                last = milis();
                break;

            case START:
                if (milis() - last < 19) {
                    GPIO_WriteLow(PULSE_PORT, PULSE_PIN);
                } else {
                    last = milis();
                    TIM2_SetCounter(0);
                    previous_counter = 0;
                    current_index = 0;
                    data = 0LL;
                    GPIO_WriteHigh(PULSE_PORT, PULSE_PIN);
                    stav = READ_DATA;
                }
                break;

            case READ_DATA:
                if (milis() - last > 6) {
                    last = milis();

                    uint64_t m = 1LL << 39;
```

```
                    uint8_t bit_count = 0;
                    while (m) {
                      if (data & m) {
                        putchar('1');
                    } else {
                        putchar('0');
                    }
                    if (++i % 8 == 0){
                        putchar(' ');
                    }
                    m >>=1;
                    }
                    printf("\n");
                    uint8_t RH_H = data >> 32;
                    uint8_t RH_L = data >> 24;
                    uint8_t temp_H = data >> 16;
                    uint8_t temp_L = data >> 8;
                    uint8_t kontrola = data;

                    printf("\n");
                    printf("RH: %d %%, Teplota: %d.%d °C\n", RH_H,
temp_H, temp_L);

                    if (temp_H > 27 || RH_H > 75) {
                        GPIO_WriteHigh(GPIOD, GPIO_PIN_3);
                    } else {
                        GPIO_WriteLow(GPIOD, GPIO_PIN_3);
                    }

                    stav = SLEEP;
                }
                break;

            default:
                stav = SLEEP;
                break;
        }
    }
}
```

Popis hlavní části kodu

Na začátku máme nastavení proměnných

- Co 5ms se pošle signál který zapne senzor
- Stav == SLEEP => srovnává last s milis

- Stav == START
    - Pošle startovní signál (LOW)
    - Po uplynutí potřebného času se vrátí na HIGH
- STAV READ_DATA

Funkce `while (m)`

– běží dokud proměnná **m** není 0.

Funkce prochází data a provede logický AND na **m** a **data.**

Pokud výsledek je nenulový znamená to že bit v proměnné data je 1.

Funkce `if (++i % 8 == 0)` – toto pouze pomáhá s čitelností binárního kodu pokud by jsme ho chtěli tisknout

Načtou se data(binární)

následně se rozdělí na teplotu a vlhkost(16/16 bitů + 8 bitů checksum).

dále se těch 16 bitů rozdělí na 8 a 8 (8 pro před desetinnou čárkou a 8 za)

V případě připojení UART vytiskne.

Pokud není UART tak pouze při přesáhnutí nastavené teploty a vlhkosti spustí LED připojen.