**SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE**

**FAKULTA ELEKTROTECHNIKY A INFORMATIKY**

**Vnorené riadiace systémy**

**vyhrievaná podložka s riadením teploty**

**semestrálne zadanie**

**Bc. Martin Minár**

**Bc. František Durdy**

**Bc. Juraj Fojtík**

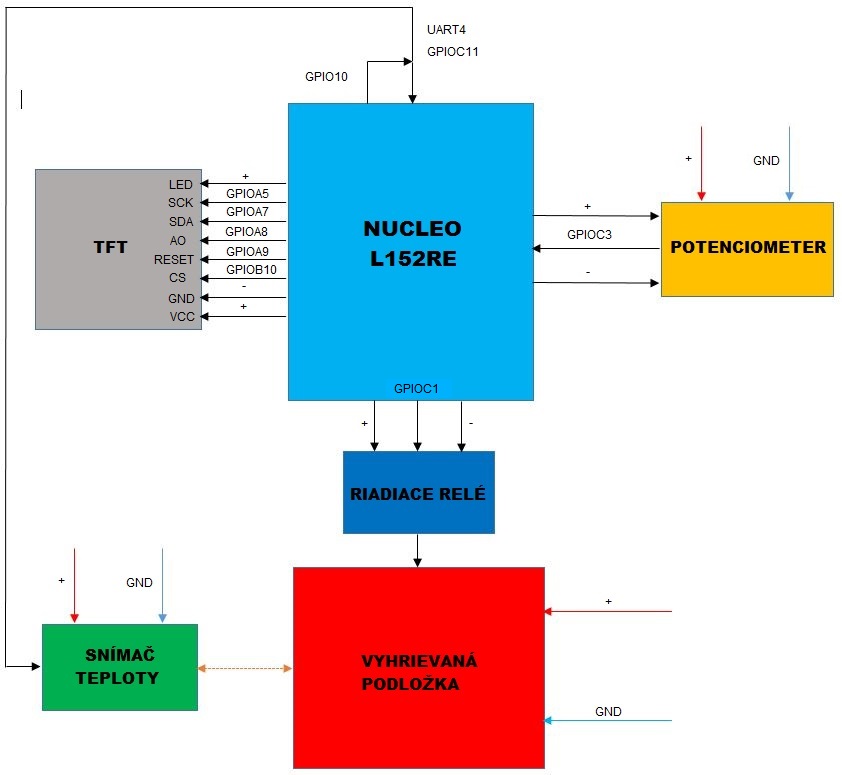
**ZS 2016/2017 Bc. Peter Čech**

1. **Hardvér zariadenia**

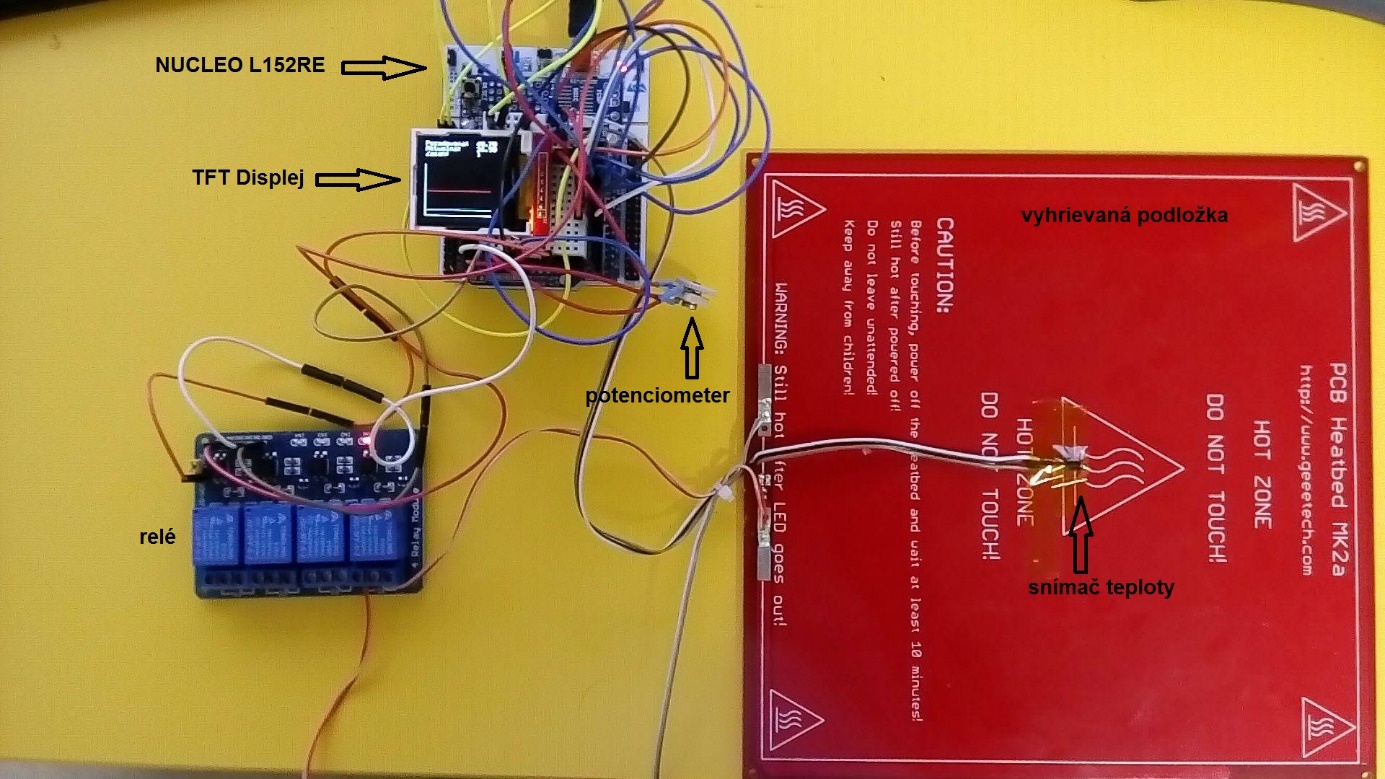
Cieľom zadania bolo zostaviť zariadenie, ktoré umožní riadiť teplotu vyhrievanej podložky, pričom sme použili nasledovné komponenty:

* Programovateľná platforma NUCLEO L152RE
* Vyhrievaná podložka
* Relé
* Potenciometer
* Snímač teploty DS18B20
* TFT displej

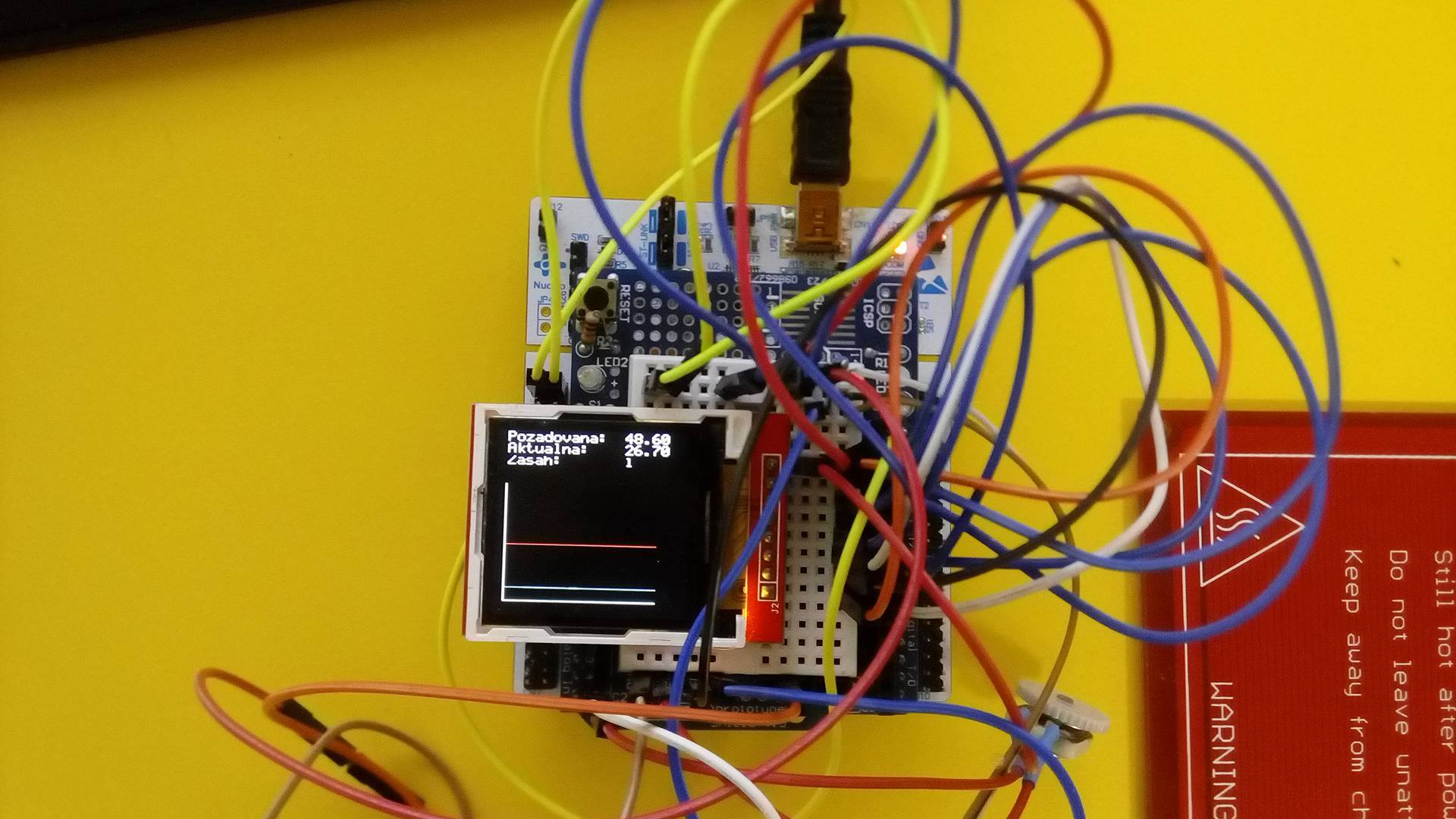
Zapojenie jednotlivých komponentov možno vidieť na obr.1



Obr.1 Bloková schéma zariadenia



Obr.2 Reálne zariadenie

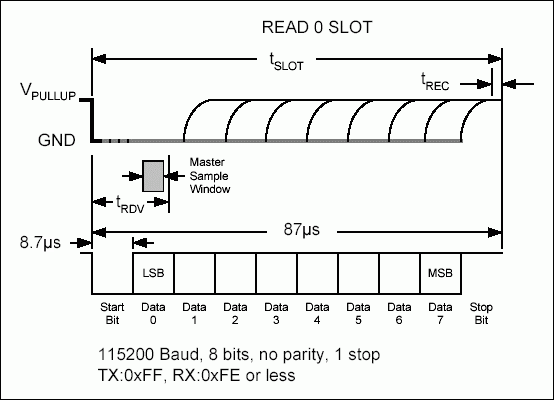


Obr.3 Displej s príslušnými hodnotami a grafickým vykresľovaním

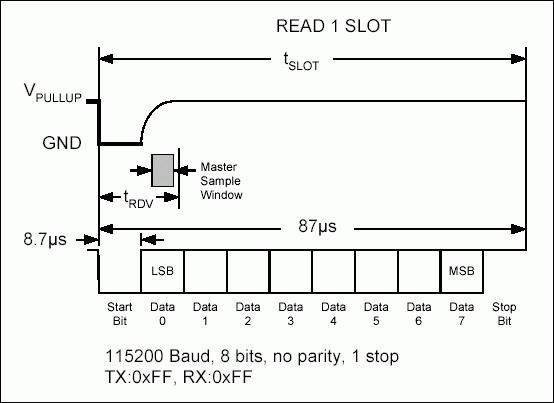
1. **Popis zbernice onewire použitej pri komunikácií z DS18B20**

One-Wire zariadenie pracuje v open-drain prostredí na napätí od 2.0 do 5.5 V. Presné logické úrovne a minimálne pullup napätie závisí od zariadenia, kde minimálne pullup napätie je 2.8V pre dobíjanie vnútornej pamäte a kondenzátor slúži na napájanie počas nízkej úrovne.

Využíva sa časovanie, teda sleduje sa aký dlhý čas trvá napäťová úroveň. Komunikácia funguje na logických úrovniach a to vysoká úroveň – nízka úroveň. Ak chcem nastaviť logickú nulu alebo logickú jednotku zopnem napäťový výstup, od dĺžky trvania napäťového signálu rozoznávam o akú logickú úroveň ide.

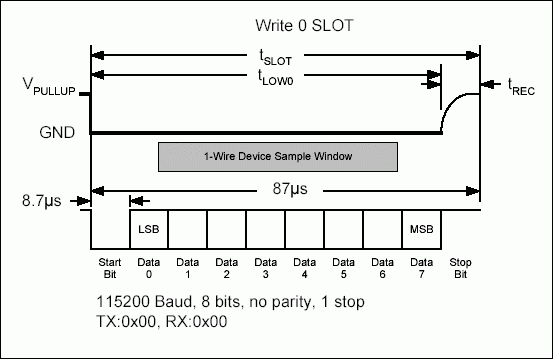
****

Obr.4 čítanie logickej nuly

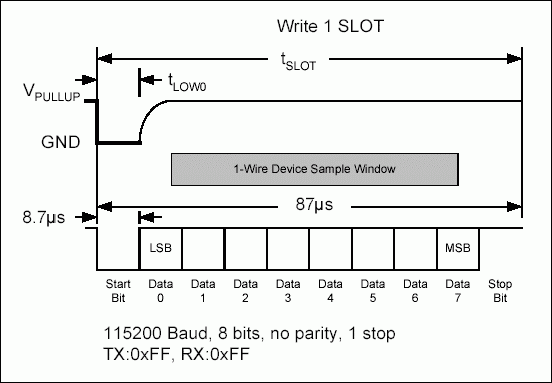


Obr.5 čítanie logickej jednotky

Ak čítam logickú nulu na RX mám 0xFE a TX 0xFF, ak čítam logickú jednotku tak Rx mám 0xFF a TX 0xFF



Obr.6 zapisovanie logickej nuly



Obr.7 zapisovanie logickej jednotky

Ak zapisujem logickú nulu budem mať RX 0x00 a TX 0x00 , pri zapisovaní logickej jednotky budem mať na RX 0xFF a TX 0xFF

1. **Dvojpólový regulátor s hysterézou**

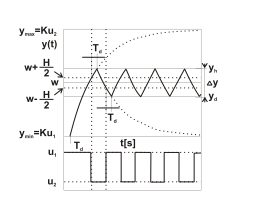
Jedná sa o zariadenie pre ovplyvnenie regulovaného systému, k dosiahnutiu a udržaniu požadovaného stavu. Porovnáva skutočnú a žiadanú hodnotu regulačnej veličiny. V závislosti od regulačnej odchýlky ovpluvňujeme akčnú veličinu. Regulátor reguluje systém s cieľom buď úplne eliminovať odchýlku, alebo udržuje odchýlku aspoň v nejakom povolenom rozsahu.

Ideálne relé: ak *e*>0, znamená to, že žiadaná hodnota je väčšia ako meraná, výstup regulátoru *u*2 → nárast regulovanej veličiny, max. prísun energie.

V opačnom prípade *u*1 → pokles regulovanej veličiny, max. odber „zápornej“ energie – nie často, napr. chladenie; prakticky – odvod energie do prostredia.

Nevhodný režim: neustále prepínanie okolo *w* – zničenie mechanických častí, citlivé na šum – prakticky nepoužiteľné

Relé s hysterézou „H“: zhoršenie kvality regulácie voči ideálnemu relé, prepínanie až na ±H/2 (vhodná veľkosť – jednotky % z rozsahu reg. veličiny).



Obr.8 Graf prepínacej priamky dvojpolohového relé s hysteréziou

1. **Programátorská príručka**

* Knižnica tft/ili9163.h
  + knižnica slúži na riadenie grafického driveru displeja(dostali sme k zadaniu)
* Knižnica tft/spi.h
  + knižnica slúži na komunikáciu z grafickým displejom prostredníctvom SPI zbernice(dostali sme k zadaniu)
* Knižnica tft/graf.h
  + **#define** GRAFOFFSETX - konštanta na nastavenie lavého vrchného rohu grafu
  + **#define** GRAFOFFSETY – konštanta na nastavenie lavého vrchného rohu grafu
  + **#define** GRAFVYSKA 80 – výška v pixeloch grafu
  + **#define** GRAFSIRKA 100 – šírka v pixeloch grafu
  + **#define** MINYGRAF 20 – minimálna hodnota na grafe
  + **#define** MAXYGRAF 80 – maximálna hodnota na grafe
  + **void** **vykresliOsiGrafu**() – vykreslí osi grafu
  + **void** **pridajPozadovanuHodnotuDoGrafu**(**float** temp) – pridáva požadovanú hodnotu do grafu a prekresluje ho
  + **void** **pridajAktualnuHodnotuDoGrafu**(**float** temp) – pridáva aktuálnu hodnotu do grafu a prekresluje ho
* Knižnica tft/texty.h
  + **void** **showPozadovanaHodnota**(**float** temp); - vypíše požadovanú hodnotu na displeji
  + **void** **showAktualnaHodnota**(**float** temp); - vypíše aktuálnu hodnotu na displeji
  + **void** **showAkcnyZasah**(**int** status); - vypíše hodnotu akčného zásahu na displeji
* Knižnica regulator/regulator.h
  + **#define** REGULATOR\_HYSTERESIS\_UP – konštanta, v ktorej je uložená hysteréza zhora
  + **#define** REGULATOR\_HYSTERESIS\_DOWN – konštanta, v ktorej je uložená hysteréza zdola
  + **#define** REGULATOR\_INVERSE - slúži na určenie, či akčný zásah je v pôvodnom tvare alebo v inverznom
  + **#define** **#define** REGULATOR\_PIN\_NUM – slúži na nastavenie pinu, na ktorom je pripojená riadiaca doska na spínanie akčného zásahu
  + **#define** REGULATOR\_PORT - slúži na nastavenie portu, na ktorom je pripojená riadiaca doska na spínanie akčného zásahu
  + **#define** REGULATOR\_CLK – slúži na nastavenie hodinového signálu pre port
  + **void** **regulatorInit**(**void**); - inicializuje výstup regulátora
  + **void** **vygenerujAkcnyZasah**(**float** pozadovana,**float** aktualna); - vygeneruje akčný zásah na základe požadovanej a aktuálnej hodnoty
  + **void** **setAkcnyZasah**(**char** status); - nastaví akčný zásah
  + **char** **getAkcnyZasah**(); - získa hodnotu akčného zásahu
* Knižnica onewire/onewire.h
  + uint8\_t **OW\_Init**(); - slúži na inicializáciu zbernice onewire, ktorá je prevádzkovaná cez UART
  + uint8\_t **OW\_Send**(uint8\_t sendReset, uint8\_t \*command, uint8\_t cLen, uint8\_t \*data, uint8\_t dLen, uint8\_t readStart); - funkcia slúži na príjem/odosielanie dát na/z zbernice onewire
* Hlavná časť programu main.c
  + **void** **adc\_init**(**void**) – inicializuje AD prevodník, pre získavanie žiadanej hodnoty
  + **void** **ADC1\_IRQHandler**(**void**) – prerušenie na získanie žiadanej hodnoty a zobrazenie na grafe, po dokončení prevodu
  + **int** **main**(**void**) {

OW\_Init(); //inicializácia onewire

lcdInitialise(LCD\_ORIENTATION1); //nastavenie rotácie a inicializácia displeja

lcdClearDisplay(decodeRgbValue(0, 0, 0)); //vymazanie displeja, nastavenie pozadia na čierno

regulatorInit(); //inicializácia regulátora

adc\_init(); //inicializácia AD prevodníka

vykresliOsiGrafu(); //vykreslenie základných osí grafu

**while** (1) {

ADC\_SoftwareStartConv(ADC1); //štart prevodu žiadanej hodnoty

OW\_Send(OW\_SEND\_RESET, "\xcc\x44", 2, NULL, NULL, OW\_NO\_READ); // odoslanie resetu a spustenie začiatku konverzie z teplotného snímača

**for** (**long** i = 0; i < 10000; i++) // počkatie požadovaného času, ovplyvňuje aj vzorkovaciu frekvenciu

;

uint8\_t buf[2];

OW\_Send(OW\_SEND\_RESET, "\xcc\xbe\xff\xff", 4, buf, 2, 2); //načítanie hodnoty zo teplotného snímača

//-----------------

**float** teplota = (**float**) ((buf[1] << 8) | buf[0]) / 16.0; //spojenie dvoch bytov do jedného čísla, plus prevod

valueAktualna = teplota;

pridajAktualnuHodnotuDoGrafu(teplota); //vysvetlené vyššie

showAktualnaHodnota(teplota); //vysvetlené vyššie

vygenerujAkcnyZasah(valuePozadovana, teplota); //vysvetlené vyššie

showAkcnyZasah(getAkcnyZasah());//vysvetlené vyššie

}

**return** 0;

}

1. **Používateľská príručka/princíp činnosti**

Pomocou potenciometra si nastavíme žiadanú teplotu podložky. Túto teplotu spolu s aktuálnou teplotou podložky vidíme na displeji, na ktorom vykresľujeme formou grafu priebehy týchto dvoch veličín, pričom žiadaná hodnota má červenú farbu a aktuálna hodnota má zeleno-modrú farbu. Aktuálnu hodnotu teploty snímame pomocou snímača teploty DS18B20. Aktuálnu a žiadanú hodnotu teploty porovnávame a na základe toho zopíname alebo vypíname relé, teda riadime vyhrievanie podložky na žiadanú teplotu. Stav relé potom sledujeme na displeji pomocou premennej zasah. Hodnota 1 znamená, že je relé zopnuté a podložka hreje a stav 0 znamené, že relé je vypnuté a podložka nehreje. Reléový regulátor má nastavenú hysteréziu 0 zhora a 0,5 zdola. V praxi to znamená to, že ak je napr. želaná hodnota 20°C a aktuálna hodnota tiež 20°C je relé vypnuté a pri poklese aktuálnej teploty na podložke na hodnotu 19,5°C sa zase zopne až pod dobu, keď nebude aktuálna teplota na podložka 20°C.