

main.c File Reference

Mereni 3D objektu. [More...](#)

```
#include <avr/io.h>
#include <avr/interrupt.h>
#include <util/delay.h>
#include <stdlib.h>
#include "lcd.h"
#include "lcd_definitions.h"
```

Include dependency graph for main.c:

Macros

#define F_CPU 16000000UL	Definice hodnoty hodinoveho signalu CLK pouziteho mikrokontroleru.
#define trig1 PC2	Pripojeni TRIGGER na PIN2 portu C.
#define echo1 PD2	Pripojeni ECHO na PIN2 portu D (INT0)
#define trig2 PC3	Pripojeni TRIGGER na PIN3 portu C.
#define echo2 PD3	Pripojeni ECHO na PIN3 portu D (INT1)
#define measure_average 10	Makro pro nastaveni poctu vzorku mereni pro vypocet prumeru merene vzdalenosti.
#define right 0x7e	Makro pro znak sipky vpravo.
#define left 0x7f	Makro pro znak sipky vlevo.
#define stupen 0xdf	Makro pro znak stupne.
#define button_enter PC0	Makro pro tlacitko ENTER.
#define button_esc PC1	Makro pro tlacitko ESC.
#define button_left PB3	Makro pro tlacitko VLEVO.
#define button_right PB4	Makro pro tlacitko VPRAVO.
#define press_enter bit_is_clear(PINC,0)	Makro pro stisk tlacitka ENTER.

```
#define press_esc bit_is_clear(PINC,1)
Makro pro stisk tlacitka ESC.

#define press_left bit_is_clear(PINB,3)
Makro pro stisk tlacitka VLEVO.

#define press_right bit_is_clear(PINB,4)
Makro pro stisk tlacitka VPRAVO.

#define temperature_ref 25
Referencni hodnota teploty pouziteho termistoru.

#define resistance_ref 10000
Referencni hodnota odporu pouziteho termistoru.

#define rezistor 10000
Hodnota odporu pouzite pri mereni teploty pomoci termistoru.

#define beta 3950
Hodnota beta faktoru udavaneho vyrobce pouziteho termistoru.
```

Enumerations

```
enum menu {
    UVOD, JAZYK_CZ, JAZYK_EN, MERENI_1D,
    MERENI_2D, MERENI_3D, MERENI_VZDALENOSTI, KONEC
}
```

Functions

```
void setup (void)
Funkce pro defaultni nastaveni programu (setup) More...

void ultrasonic1 (void)
Funkce pro vyslani impulsu na pin TRIGGER snimace 1 pro zahajeni mereni (ultrasonic1) More...

void ultrasonic2 (void)
Funkce pro vyslani impulsu na pin TRIGGER snimace 2 pro zahajeni mereni (ultrasonic2) More...

void vypocet (void)
Funkce pro vypocet namerene vzdalenosti snimacem (vypocet) More...

void speed_temperature (void)
Funkce pro urceni rychlosti zvuku v zavislosti na namerene teplote (speed_temperature) More...

void measure_delka_s1 (void)
Funkce pro mereni vzdalenosti pomoci snimace 1 (measure_delka_s1) More...

void measure_delka_s2 (void)
Funkce pro mereni vzdalenosti pomoci snimace 2 (measure_delka_s2) More...

void measure_delka (void)
Fnkce pro mereni vzdalenosti mezi 2 snimaci (measure_delka) More...

void measure_delka_objekt (void)
Funkce pro mereni 1 rozmeru mereneho objektu (measure_delka_objekt) More...

void lcd_print (char line0[17], char line1[17], uint8_t arrow)
```

Funkce pro vypis menu na LCD displej (lcd_print) [More...](#)

int **main** (void)

Funkce pro obsluhu hlavniho programu (main) [More...](#)

Variables

volatile uint16_t **number0_of_overflow** =0

Pocet pretečení citace 0.

volatile uint16_t **timer0_value** =0

Aktualni hodnota citace TC0.

char **uart_string** [5]

Definice stringu pro vypis hodnot na displej pomoci UART nebo LCD.

uint16_t **speed**

Rychlost zvuku pri dane teplote.

float **temperature**

Aktualni teplota pro urceni rychlosti zvuku ve vzduchu.

float **voltage**

Pomocna promenna pro mereni napeti na termistoru snimaneho pomoci AD prevodniku.

float **distance**

Namerena vzdalenost snimace.

uint8_t **number_of_measure** =0

Pocet vzorku mereni.

uint16_t **delka_s1** =0

Vysledna namerena vzdalenost snimace 1.

uint16_t **delka_s2** =0

Vysledna namerena vzdalenost snimace 2.

uint16_t **delka** =0

Vysledna vzdalenost mezi 2 snimaci.

uint16_t **delka_objekt** =0

Vysledny rozmer mereneho objektu.

uint16_t **delka_mereni** =0

Delka mereneho objektu.

uint16_t **sirka_mereni** =0

Sirka mereneho objektu.

uint16_t **vyska_mereni** =0

Vyska mereneho objektu.

uint8_t **EN** = 0

Pomocna promenna pro vyber anglickeho jazyka.

uint8_t **vykresli** = 0

Pomocna promenna pro vypsani textu pouze 1x.

uint8_t **vykresli_objekt** = 0

Pomocna promenna pro vypsani textu pouze 1x pri mereni vicerozmernych predmetu.

uint8_t **enter_control** = 0
Indikace stisku tlacitka ENTER.

uint8_t **esc_control** = 0
Indikace stisku tlacitka ESC.

uint8_t **left_control** = 0
Indikace stisku tlacitka VLEVO.

uint8_t **right_control** = 0
Indikace stisku tlacitka VPRAVO.

uint8_t **enter_control_OK** = 0
Pomocna promenna pro osetreni tlacitka ENTER.

uint8_t **esc_control_OK** = 0
Pomocna promenna pro osetreni tlacitka ESC.

uint8_t **left_control_OK** = 0
Pomocna promenna pro osetreni tlacitka VLEVO.

uint8_t **right_control_OK** = 0
Pomocna promenna pro osetreni tlacitka VPRAVO.

menu **pozice** = UVOD
Pomocna promenna pro spravnou orientaci v jednotlivych castech stavoveho automatu.

Detailed Description

Mereni 3D objektu.

Author

David Halas, Miroslav Hajek; Brno University of Technology; Czechia

Date

Dec 3, 2018

Note

Mereni na principu vlozeni objektu mezi 2 ultrazvukove snimace.

Function Documentation

◆ lcd_print()

```
void lcd_print ( char    line0[17],  
                 char    line1[17],  
                 uint8_t arrow  
               )
```

Funkce pro vypis menu na LCD displej (lcd_print)

Tato funkce umoznuje vypsat oba radky LCD a soucasne pridat sipky po leve, nebo prave strane, pripadne na obou stranach.

- Vstupni promenne pro tuto funkci jsou:

Parameters

- line0** - Pole o delce 16 znaku + nulovy znak na konec retezce pro vypis na prvni radek
- line1** - Pole o delce 16 znaku + nulovy znak na konec retezce pro vypis na druhy radek
- arrow** - Ciselna hodnota, která umoznuje pridani sipek na zacatek obrazovky, konec obrazovky, nebo na obe strany.

Tato funkce je tedy pouzitelna bez uprav na displej 16x2 pozic.

Ciselna hodnota zadana jako treti vstupni parametr prideluje pozici sipek dle nasledujicich hodnot:

- 0 == sipky vlevo
- 1 == sipky vpravo
- 2 == sipky vlevo i vpravo
- 3 a vice == displej bez sipek

Warning

Vstupni retezce musi mit maximalne 16 znaku

◆ main()

Funkce pro obsluhu hlavního programu (main)

Note

Na začátku programu jsou nastaveny všechny potřebné řídicí registry a používane I/O piny voláním funkce `setup()`.

V dalším kroku je volána funkce `sei()` pro povolení všech přerušení. Nasleduje časové zpoždění potřebné ke změření teploty pro správnou funkci měření.

Hlavní část programu je vykonávána v podobě stavového automatu v nekonečné smyčce.

* Jednotlivé části stavového automatu jsou popsány níže:

Uvodní sekce stavového automatu pro obsluhu menu programu (UVOD)

- V této sekci je vykreslena posloupnost několika úvodních obrazovek, kde je uživatel seznámen s obsluhou programu a následně je uživatel vyzván k výběru jazyka hlavního menu.
- Stavový automat přechází do pozice `JAZYK_CZ`

Sekce stavového automatu pro volbu českého jazyka (`JAZYK_CZ`)

- Jednou je na displej vykreslena informace o jazyku CZ a uživatel může pomocí tlačítek menu ovládat:
- * `ENTER` - Menu se nastaví na jazyk CZ a skoci na pozici `MERENI_1D`
- * `RIGHT` - Menu skoci na pozici `JAZYK_EN` a čeká na stisk dalšího tlačítka
- * `ESC` - Menu skoci na pozici `UVOD` a uživateli je opět vypsan začátek menu s instrukcemi apod.
- Vždy po stisku určitého tlačítka je příznak `..._CONTROL` nastaven na 0, aby bylo možné indikovat další stisknutí tlačítka pro ovládání další části menu.
- Stavový automat přechází do pozice `MERENI_1D`, nebo `JAZYK_EN`

Sekce stavového automatu pro volbu českého jazyka (`JAZYK_EN`)

- Jednou je na displej vykreslena informace o jazyku EN a uživatel může MENU ovládat pomocí tlačítek:
- * `ENTER` - Menu se nastaví na jazyk EN a skoci na pozici `MERENI_1D`
- * `LEFT` - Menu skoci na pozici `JAZYK_CZ` a čeká na stisk dalšího tlačítka
- * `ESC` - Menu skoci na pozici `UVOD` a uživateli je opět vypsan začátek menu s instrukcemi apod.
- Vždy po stisku určitého tlačítka je příznak `..._CONTROL` nastaven na 0, aby bylo možné indikovat další stisknutí tlačítka pro ovládání další části menu.
- Stavový automat přechází do pozice `MERENI_1D`, nebo `JAZYK_CZ`

Sekce stavového automatu pro měření 1 rozměru (`MERENI_1D`)

- Měření jednoho rozměru probíhá v 5ti krocích. Vždy po stisku tlačítka `ENTER` se provede následující krok měření.
- 1) V prvním kroku se na obrazovku vypíše možnost pro výběr měření 1D objektu a změř se vzdálenost mezi oběma snímači.
- 2) Ve druhém kroku je uživatel vyzván k vložení měřeného objektu do oblasti měření (oblast mezi 2 senzory). Následně proběhne měření vloženého objektu (viz `measure_delka_objekt()`) a výsledný rozměr je uložen do pomocné proměnné `delka_mereni`.

- 3) Ve tretim kroku je uživatel vyzvan k vyjmutí objektu z mericí oblasti, což je opět potvrzeno stiskem ENTER.
- 4) Ve čtvrtém kroku je na displej vypsána výsledná naměřená hodnota (v mm) odpovídající velikosti měřeného objektu.
- 5) V posledním kroku je potvrzena naměřená hodnota stiskem ENTER a měření je ukončeno přechodem stavového automatu na další pozici.
- Stavový automat přechází do pozice KONEC a čeká na stisk dalšího tlačítka:
- * ESC - po stisku program skoci na pozici JAZYK_CZ a vypíše uživateli vyber jazyka
- * RIGHT - po stisku program skoci na pozici MERENI_2D a vypíše se možnost vyberu 2D měření

Sekce stavového automatu pro měření 2 rozměru (MERENI_2D)

- Měření dvou rozměrů probíhá v 6ti krocích. Vždy po stisku tlačítka ENTER se provede následující krok měření.
- 1) V prvním kroku se na obrazovku vypíše možnost pro vyber měření 2D objektu a zmerí se vzdálenost mezi oběma snímači.
- 2) Ve druhém kroku je uživatel vyzvan k vložení měřeného objektu do oblasti měření (oblast mezi 2 senzory). Následně proběhne měření vloženého objektu (viz `measure_delka_objekt()`) a výsledný rozměr je uložen do pomocné proměnné `delka_mereni`.
- 3) Ve třetím kroku je uživatel vyzvan k otočení objektu a opět proběhne měření objektu. Výsledný naměřený rozměr je uložen do pomocné proměnné `sirka_mereni`.
- 4) Po potvrzení stiskem tlačítka ENTER je uživatel vyzvan k vyjmutí objektu z oblasti měření.
- 5) V patém kroku je na displej vypsána výsledná naměřená hodnota (v mm) odpovídající velikosti měřeného objektu. Výsledné hodnoty jsou zobrazeny pomocí souřadnicových identifikátorů X a Y.
- 6) V posledním kroku je potvrzena naměřená hodnota stiskem ENTER a měření je ukončeno přechodem stavového automatu na další pozici.
- Stavový automat přechází do pozice KONEC a čeká na stisk dalšího tlačítka:
- * ESC - po stisku program skoci na pozici JAZYK_CZ a vypíše uživateli vyber jazyka
- * LEFT - po stisku program skoci na pozici MERENI_1D a vypíše se možnost vyberu 1D měření
- * RIGHT - po stisku program skoci na pozici MERENI_3D a vypíše se možnost vyberu 3D měření

Sekce stavového automatu pro měření 3 rozměrů (MERENI_3D)

- Měření dvou rozměrů probíhá v 7 krocích. Vždy po stisku tlačítka ENTER se provede následující krok měření.
- 1) V prvním kroku se na obrazovku vypíše možnost pro vyber měření 2D objektu a zmerí se vzdálenost mezi oběma snímači.
- 2) Ve druhém kroku je uživatel vyzvan k vložení měřeného objektu do oblasti měření (oblast mezi 2 senzory). Následně proběhne měření vloženého objektu (viz `measure_delka_objekt()`) a výsledný rozměr je uložen do pomocné proměnné `delka_mereni`.
- 3) Ve třetím kroku je uživatel vyzvan k otočení objektu a opět proběhne měření objektu. Výsledný naměřený rozměr je uložen do pomocné proměnné `sirka_mereni`.
- 4) Uživatel je vyzvan k překlopení objektu na pravou stranu, aby bylo možné zmerit výšku objektu a poté je změřený rozměr uložen do pomocné proměnné `vyska_mereni`
- 5) Po potvrzení stiskem tlačítka ENTER je uživatel vyzvan k vyjmutí objektu z oblasti měření.
- 6) V patém kroku je na displej vypsána výsledná naměřená hodnota (v mm) odpovídající velikosti měřeného objektu. Výsledné hodnoty jsou zobrazeny pomocí souřadnicových identifikátorů X a Y a Z
- 7) V posledním kroku je potvrzena naměřená hodnota stiskem ENTER a měření je ukončeno přechodem stavového automatu na další pozici.

- Stavový automat přechází do pozice KONEC a čeká na stisk dalšího tlačítka:
- * ESC - po stisku program skoci na pozici JAZYK_CZ a vypíše uživateli vyber jazyka
- * LEFT - po stisku program skoci na pozici MERENI_2D a vypíše se možnost vyberu 2D merení
- * RIGHT - po stisku program skoci na pozici MERENI_VZDALENOSTI a vypíše se možnost vyberu merení vzdalenosti

Sekce stavového automatu pro merení vzdalenosti pomocí 1 snímače (MERENI_VZDALENOSTI)

- Při vyberu možnosti merení vzdalenosti je uživatel vyzván k umístění senzoru 2 do stabilní polohy.
- Po stisku tlačítka ENTER se provede merení vzdalenosti pomocí funkce `measure_delka_s2()` a výsledná hodnota je vypsána na obrazovku LCD.
- Merení se provádí opakovaně v určitém časovém intervalu a na displeji se vždy prepíše minulé hodnota nové naměřenou aktuální hodnotou.
- Po opětovném stisku tlačítka ENTER stavový automat přechází do pozice KONEC a čeká na stisk dalšího tlačítka:
- * ESC - po stisku program skoci na pozici JAZYK_CZ a vypíše uživateli vyber jazyka
- * LEFT - po stisku program skoci na pozici MERENI_3D a vypíše se možnost vyberu 3D merení

Sekce stavového automatu pro završení menu (KONEC)

- Nejprve je uživateli poděkováno za využití našeho měřicího přístroje a následně je vypsána možnost nového merení po stisknutí ENTER.
- Obe tyto obrazovky jsou vypsány pouze jednou a to použitím pomocné proměnné vykreslí.
- Program poté stále kontroluje, zda nebylo stisknuto tlačítko ENTER. V případě, že ano, bude pozice stavového automatu nastavena na MERENI_1D.

◆ `measure_delka()`

```
void measure_delka ( void )
```

Funkce pro merení vzdalenosti mezi 2 snímači (`measure_delka`)

Měření vzdalenosti mezi 2 snímači spočívá v postupném měření obou snímačů a následném výpočtu průměru získaných hodnot.

Nejprve je smazána hodnota posledního měření pro eliminaci chyb při výpočtu.

Poté proběhne měření vzdalenosti pomocí snímače 1 (funkce `measure_delka_s1()`) a výsledek je uložen. Poté proběhne měření pomocí snímače 2 (funkce `measure_delka_s2()`) a výsledek je opět uložen.

Následně je vypočten průměr těchto 2 hodnot a výsledek je uložen do proměnné `delka`.

V závěru jsou hodnoty získané měřeními jednotlivých snímačů smazány pro možnost opětovného měření.

◆ `measure_delka_objekt()`


```
void measure_delka_objekt ( void )
```

Funkce pro mereni 1 rozmeru mereneho objektu (measure_delka_objekt)

Mereni objektu spociva v mereni vzdalenosti snimacem 1 a nasledne snimacem 2 a vypoctu vysledne hodnoty.

V prvni kroku je smazana hodnota posledniho mereni pro eliminaci chyb pri vypoctu.

Dale probehne mereni vzdalenosti mezi objektem a snimacem 1 (pomoci funkce **measure_delka_s1()**) a mereni mezi objektem a snimacem 2 (pomoci funkce **measure_delka_s2()**).

Velikost mereneho objektu je ziskana odedcenim vzdalenosti objektu od snimace 1 a od snimace 2 od celkove vzdalenosti mezi obema snimaci.

◆ measure_delka_s1()

```
void measure_delka_s1 ( void )
```

Funkce pro mereni vzdalenosti pomoci snimace 1 (measure_delka_s1)

Mereni vzdalenosti probiha merenim casu, za který signal dojde k prekazce a zpet ke snimaci 1. Tato zmena je indikovana trvaním logické hodnoty 1 na pinu ECHO(1).

Merena hodnota je prumerovana z nekolika vzorku mereni (pocet vzorku je dan konstantou **measure_average**).

V kazdem cyklu mereni je vyslan pozadavek na pin TRIGGER1 (pomoci funkce **ultrasonic1()**), pote se ceka na zmenu pinu ECHO(1), která indikuje navrat mericiho signalu vyslaneho snimacem.

Nasledne probehne vypocet vzdalenosti (pomoci funkce **vypocet()**) a po dokončení všech mericich cyklu je proveden vypocet prumeru z namerenych vzorku.

Pomocna promenna pro pocet provedenych mereni jednotlivych vzorku je nastavena na 0 pro možnost zahajeni opetovneho mereni pomoci ultrazvukoveho snimace 1.

◆ measure_delka_s2()

```
void measure_delka_s2 ( void )
```

Funkce pro mereni vzdalenosti pomoci snimace 2 (measure_delka_s2)

Mereni vzdalenosti probiha merenim casu, za který signal dojde k prekazce a zpet ke snimaci 2. Tato zmena je indikovana trvanim logicke hodnoty 1 na pinu ECHO(2).

Merena hodnota je prumerovana z nekolika vzorku mereni (pocet vzorku je dan konstantou measure_average).

V kazdem cyklu mereni je vyslan pozadavek na pin TRIGGER2 (pomoci funkce **ultrasonic2()**), pote se ceka na zmenu pinu ECHO(2), která indikuje navrat mericiho signalu vyslaneho snimacem.

Nasledne probehne vypocet vzdalenosti (pomoci funkce **vypocet()**) a po dokonceni vseh mericich cyklu je proveden vypocet prumeru z namerenych vzorku.

Pomocna promenna pro pocet provedenych mereni jednotlivych vzorku je nastavena na 0 pro moznost zahajeni opetovneho mereni pomoci ultrazvukoveho snimace 2.

◆ setup()

```
void setup ( void )
```

Funkce pro defaultni nastaveni programu (setup)

Zde jsou nastaveny ridici registry pro nastaveni potrebných parametru citace/casovace, externiho preruseni, AD prevodniku.

Dale jsou zde nastaveny pouzite I/O piny a defaultni logicke urovne na techto pinech.

Dale je zde nastavena inicializace LCD displeje a pristup do pameti displeje.

◆ speed_temperature()

```
void speed_temperature ( void )
```

Funkce pro urceni rychlosti zvuku v zavislosti na namerene teplotě (speed_temperature)

Tato funkce slouzi k vyhodnoceni namerene teploty a nasledne urceni rychlosti sireni zvuku v prostoru, v zavislosti na okolni teplotě ziskane merenim.

Ziskana rychlost zvuku je dale pouzita pri vypoctu merene vzdalenosti.

◆ ultrasonic1()

void ultrasonic1 (void)

Funkce pro vyslani impulzu na pin TRIGGER snimace 1 pro zahajeni mereni (ultrasonic1)

Pro zajisteni spravneho mereni je pin TRIG1 nastaven na uroven 0 na kratky casovy interval. Pote je pin TRIG1 nastaven na logickou uroven 1 a po urcitem casovem intervalu (10us) je tento pin opet nastaven na logickou uroven 0.

Pri nastaveni logicke urovne 1 na pin TRIG1 se nastavi pin ECHO1 na logickou uroven 1 a z vysilace je vyslana serie pulsu. Az prijimac prijme vyslanou posloupnost zmeni pin ECHO1 na logickou uroven 0.

Doba trvani logicke urovne 1 na pinu ECHO1 odpovida casu, za který signal z vysilace dorazi k prekazce a zpet k prijimaci.

◆ ultrasonic2()

void ultrasonic2 (void)

Funkce pro vyslani impulzu na pin TRIGGER snimace 2 pro zahajeni mereni (ultrasonic2)

Pro zajisteni spravneho mereni je pin TRIG2 nastaven na uroven 0 na kratky casovy interval. Pote je pin TRIG1 nastaven na logickou uroven 1 a po urcitem casovem intervalu (10us) je tento pin opet nastaven na logickou uroven 0.

Pri nastaveni logicke urovne 1 na pin TRIG2 se nastavi pin ECHO1 na logickou uroven 1 a z vysilace je vyslana serie pulsu. Az prijimac prijme vyslanou posloupnost zmeni pin ECHO2 na logickou uroven 0.

Doba trvani logicke urovne 1 na pinu ECHO2 odpovida casu, za který signal z vysilace dorazi k prekazce a zpet k prijimaci.

◆ vypocet()

void vypocet (void)

Funkce pro vypocet namerene vzdalenosti snimacem (vypocet)

K vypoctu vzdalenosti je nutne znat rychlost sireni zvuku (ziskana pomoci funkce `speed_temperature()`) a cas, za který merici impuls urazi merenou vzdalenost (ziskan merici funkci `INT0_vect/INT1_vect`).

Zvukovy signal cestuje k prekazce a zpet, proto je nutne namereny cas vydelit 2.

Vzdalenost je ziskana vynasobenim rychlosti a casu (ziskan bitovou hodnotou poctu pretečení casovace a frekvenci hodinoveho signalu).

Hodnota vzdalenosti je ulozena do pomocne promenne a nasledne prevedena na hodnotu v milimetrech pro nasledne zpracovani.