



Dokumentácia projektu v predmete
Mikroprocesorové a vestavěné systémy
Simulace v CW: Světelné noviny

17. decembra 2015

Dávid Mikuš (xmikus15@stud.fit.vutbr.cz)

Obsah

1	Úvod	2
2	Implementácia	2
2.1	checkButtons()	3
2.2	redraw()	4
2.3	Globálne premenné	4
2.4	Pamäťová náročnosť	4
3	Záver	4

1 Úvod

Zadaním projektu je zobraziť login na displej. Na displeji bude zobrazená len časť loginu ktorá bude potom rotovať.

2 Implementácia

Projekt bol implementovaný v jazyku C v prostredí CodeWarrior. Jadrom je nekonečný *for* cyklus. Po spustení kódu sa inicializujú premenné a vojde do cyklu. Cyklus sa stará o obluhu tlačidiel. V každom priechode cyklu kontroluje či bolo stlačené tlačidlo.

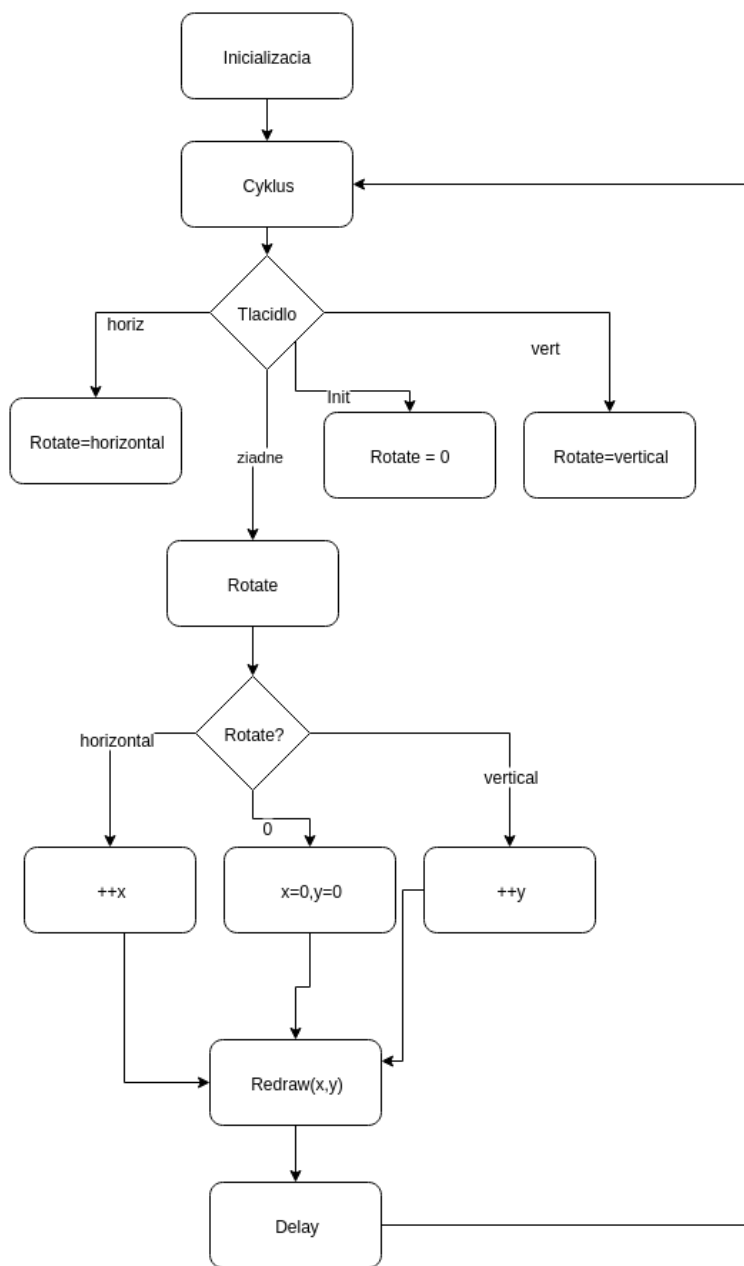
Ak bolo stlačené `init` vynulujú sa premenné `x` a `y` ktoré slúžia pre rotáciu. Taktiež sa nastaví premenná `rotate` do nuly, čo značí že sa nebude vykonávať žiadna rotácia.

Pri stlačení tlačidla `horiz` sa nastaví `rotate` na konštantu `horizontal`. Obdobne sa to deje pri tlačidle `vert`

Pri každom priechode cyklu sa kontroluje premenná `rotate` na základe ktorej sa inkrementuje premenná `x` alebo `y`.

Následne sa nastaví diody pomocou funkcie `redraw(int x, int y)`. A zavolá sa funkcia `delay()` ktorá má v sebe jeden `for` ktorý sa opakuje toľko krát aby bolo rýchlosť rotácie odpovedala nastavení slideru. V každom fore kontroluje stlačené klávesy.

```
void main(void) {
    int x = 0;
    int y = 0;
    *buttons = 0;
    *slider = 128;
    EnableInterrupts; /* enable interrupts */
    /* include your code here */
    for (;;) {
        _RESET_WATCHDOG(); /* feeds the dog */
        checkButtons();
        if(rotate == 0) {
            x = 0;
            y = 0;
        } else if(rotate == horiz)
            ++x;
        else if(rotate == vertical)
            ++y;
        redraw(x,y);
        delay();
    } /* loop forever */
    /* please make sure that you never leave main */
}
```



Obr. 1: Vývojový diagram

2.1 checkButtons()

Jednoduchá funkcia ktorá kontroluje stlačené tlačítko a na základe toho nastaví premennú `rotate`.

```

void checkButtons() {
    if(*buttons&init)
        rotate = 0;
    else if(*buttons&horiz)
        rotate = horiz;
    else if (*buttons&vertical)
        rotate = vertical;
}

```

2.2 redraw()

Funkcia `redraw()` sa stará o vybrané jednotlivých stĺpcov z globálnej premenný `login` kde je uložených 64 bajtov. Každý bajt reprezentuje jeden stĺpec a každý bit reprezentuje jednu diódu.

Najprv sa vykoná horizontálny posun pre každý stĺpec displeja.

```
index = (i-x)%size;
if(index<0)
    index += size;
ch = login[index];
```

Kde `size` je konštanta 64, `i` je stĺpec displeja a `x` je posun. Následne sa vykoná aj vertikálny posun pomocou rotáciu bitov.

```
ch = (ch >> y) | (ch << (8-y));
```

Následne sa už len výsledny stĺpec zobrazí na displej

```
display[i] = ch;
```

2.3 Globálne premenné

Globálne premenné boli použité pre mapovanie do pamäti.

```
unsigned char *buttons = (unsigned char*)0xB0;
unsigned char *display = (unsigned char*)0xB1;
unsigned char *slider = (unsigned char*)0xD1;
unsigned char rotate = 0;
```

Adresa `0xB0` je začiatok adresy pamäti RAM. Na jeden bajt boli uložené 3 tlačítka, na 1,2 a 3. bit. Pre displej bolo namapovaných 64 bajtov, ktoré obsahujú jednotlivé diody. Premenná `rotate` obsahuje akým smerom sa ma rotovať displej.

Boli použité ešte konštanty:

```
#define size 64
#define init 1
#define horiz 2
#define vertical 4
```

2.4 Pamäťová náročnosť

Segment	Data
Kódový	391
Datový	7
Počet riadkov	87

3 Záver

Projekt bol úspešne dokončený a otestovaný. Nenašla sa žiadna chyba. Jediný problém bol že záporne číslo modulo `size` dávalo ako výsledok záporne číslo, čo sa vyriešilo jedným IF-om.