



// Dílenská praxe

A4	5. Alfnumerický displej		
John Denis		1/7	Známka:
4. 1. 2023	Datum odevzdání:	25. 1. 2024	Odevzdáno:

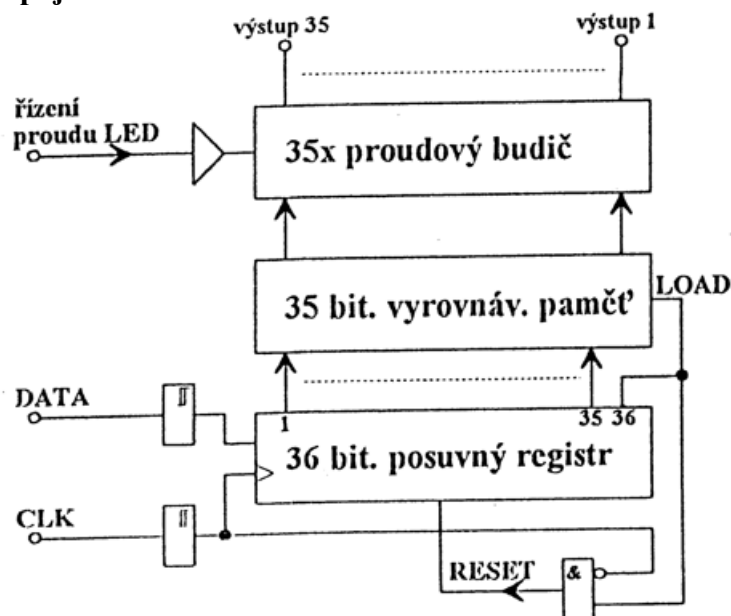


Zadání:

Zpracujte program v programovacím jazyce C ovládající alfanumerický displej tak, aby obsahoval nejméně tyto funkce:

- 1) volbu druhu displeje (7segmentový/14segmentový)
- 2) zobrazení vhodně zvolené množiny znaků pro každý typ displeje
- 3) vhodně zvolená datová a programová struktura

Blokové schema zapojení:



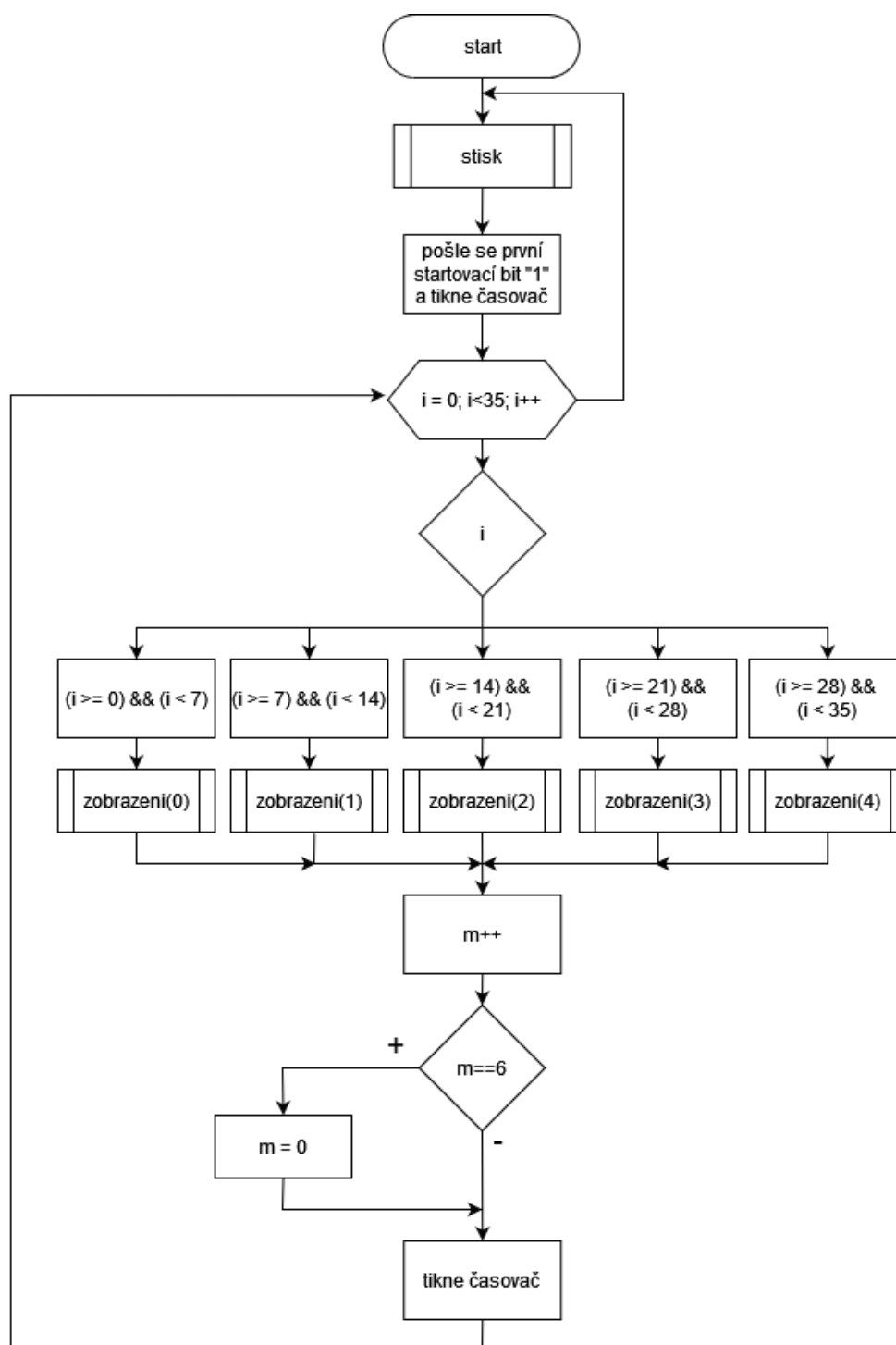
Postup (principy řešení):

Data jsem napojil na port 301 a Clk na 300. Přepínání displeje se volí na začátku kódu v proměnné „dis“ (displej). Funkcí „getch“ snímám stisknutí klávesy. Od hodnoty z ASCII tabulky odečtu 48, aby sedělo pořadí mých určených kláves v poli s ASCII tabulkou. Stisknuté klávesy ukládám do pole, kde se pozice posouvají každým stisknutím. V hlavní části programu pošlu startovací bit na výstup, a poté běží cyklus o 35 cyklech pro každý bit. Hodnotu uloženou na pozici displeje, kterou chci zobrazit, rozložím na jednotlivé bity pomocí bitové operace s nadefinovanými maskami. Postupně tak v každém cyklu zjistím jaký bit pošlu na výstup. Po každém 7. cyklu se pracuje s jinou pozicí displeje. Každá část kódu má dvě podmínky, pro numerický nebo alfanumerický displej. U alfanumerického displeje se střídá proměnná, která mění dvojice pozic.



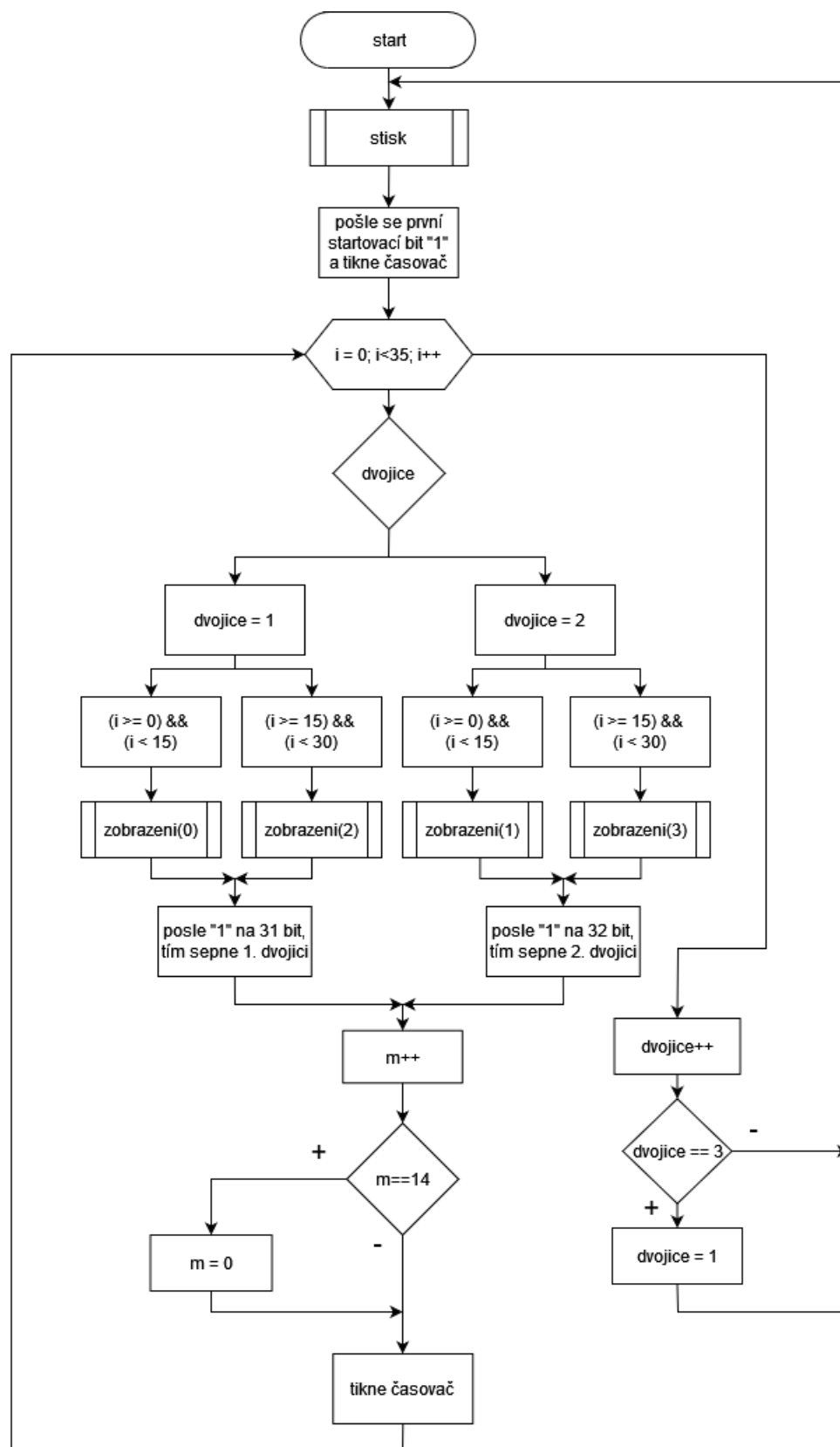
Vývojový diagram:

Funkce main pro numerický displej:





Funkce main pro alfanumerický displej:





Závěr:

Není vyzkoušena funkčnost kódu. Určil jsem si pouze zobrazení číslic na displeji a u alfanumerického pouze 5 číslic. Snažil jsem se hlavně, aby byl kód funkční a z důvodu mého pomalého pracování jsem se nezdržoval určováním zobrazení kláves na displeji.

Přílohy:

- komentovaný výpis programu

Výpis programu:

```
#include<stdio.h>
#include<conio.h>
#include<dos.h>

int dis = 1;    // volba displeje

int p = 0;    // pomocná proměnná pro posouvání pozic
int pozice[5] = {0,0,0,0,0};    // pozice numerického displeje
int cislo = 0;    // proměnná pro ukládání hodnoty z klávesnice
                // 0 1 2 3 4 5 6 7 8 9
int klavesy[10] = {0x3F,0x06,0x6D,0x4F,0x66,0x6D,0x7D,0x27,0x7F,0x6F};
int masky[7] = {0x01,0x02,0x04,0x08,0x10,0x20,0x40};    //masky pro operace
                s jednotlivými bity na num. displeji
int m = 0;    // pomocná proměnná pro masky

int poz_alf[4] = {0,0,0,0};    // pozice alfanumerického displeje
                // 0 1 2 3 4
int klav_alf[10] = {0x003F,0x0086,0x092B,0x010F,0x1560};
int mask_alf[15] = {0x01,0x02,0x04,0x08,0x10,0x20,0x40,0x80,
                0x100,0x200,0x400,0x800,0x1000,0x2000,0x4000}
                // masky pro alfan. displej
int dvojice = 1;    // pro aktivaci jednotlivých dvojic na alfan. dis.

void stisk(void)
{
    cislo = getch() - 48;    // „0“ je v ASCII tabulce 48, od stisknuté
    klávesy odečtu 48, aby se pořadí shodovalo s hodnotami určenými v poli
    if (dis == 1)    // num. dis.
    {
        for (p = 0; p < 4; p++)
        {
            // pozice na displeji se posunou
            pozice[p] = pozice[p + 1];
        }
        pozice[4] = klavesy[cislo];    // nově stisknutá hodnota se
                                        zapisuje na pozici v pravo
    }
    if (dis == 2)    // alfa. dis.
    {
        for (p = 0; p < 3; p++)
        {
            poz_alf[p] = pozice[p + 1];
        }
        poz_alf[3] = klav_alf[cislo];
    }
}
```



```
void zobrazeni(int l)
{
    if (dis == 1) // num. dis.
    {
        if ((pozice[l] & masky[m]) == 0) // pomocí masek zjistím hodnoty
            outportb(0x301, 0);          // jednotlivých bitů na pozici
        else                                a pošlu data na výstup
            outportb(0x301, 1);
    }
    if (dis == 2) // alfa. dis.
    {
        if ((poz_alf[l] & mask_alf[m]) == 0)
            outportb(0x301, 0);
        else
            outportb(0x301, 1);
    }
}

void main(void)
{
    while()
    {
        stisk();
        int i = 0;
        outportb(0x301, 1); // startovací bit log. 1 pošle na výstup
        outportb(0x300, 1); // provede se tik čítače
        outportb(0x300, 0);
        for (i = 0; i < 35; i++) // cyklus 36 bitů dat
        {
            if (dis == 1) // num. dis.
            {
                if ((i >= 0) && (i < 7)) // 1. pozice displeje
                    zobrazeni(0);

                if ((i >= 7) && (i < 14)) // 2. pozice displeje
                    zobrazeni(1);

                if ((i >= 14) && (i < 21)) // 3.pozice displeje
                    zobrazeni(2);

                if ((i >= 21) && (i < 28)) // 4.pozice displeje
                    zobrazeni(3);

                if ((i >= 28) && (i < 35)) // 5. pozice displeje
                    zobrazeni(4);
                m++;
                if (m == 6) m = 0; // proměnná pro masky se po každé
                                // pozici vynuluje
            }
        }
    }
}
```



```
if ((dis == 2) && (dvojice == 1)) // alfnum. dis. a první
                                dvojice pozic
{
    if ((i >= 0) && (i < 15))    // 1. pozice
        zobrazeni(0);
    if ((i >= 15) && (i < 30))    // 3. pozice
        zobrazeni(2);
    if (i == 30) outportb(0x301, 0); // aktivuje první dvojici
    if (i == 31) outportb(0x301, 1); // deaktivuje druhou
                                    dvojici
    m++;
    if (m == 14) m = 0;
}

if ((dis == 2) && (dvojice == 2)) // alfnum. dis. a druhá
                                dvojice pozic
{
    if ((i >= 0) && (i < 15))    // 2. pozice
        zobrazeni(1);
    if ((i >= 15) && (i < 30))    // 4. pozice
        zobrazeni(3);
    if (i == 30) outportb(0x301, 1); // aktivuje 2. dvojici
    if (i == 31) outportb(0x301, 0); // deakti. 1. pozici
    m++;
    if (m == 14) m = 0;
}
outportb(0x300, 1);    // provede se tik čítače
outportb(0x300, 0);

}

dvojice++;    // pro alfan. se vymění dvojice pozic
if (dvojice == 3) dvojice = 1;

}

}
```