



Díleňská praxe

A4	6. Model výtahu		
Tenk Jakub		1/10	Známka:
26. 1. 2022	Datum odevzdání:	16. 2. 2022	Odevzdáno:



Zadání:

Zpracujte program v programovacím jazyce C ovládající model výtahu tak, aby obsahoval nejméně tyto funkce:

- 1) ovládání pohybu kabiny výtahu pomocí tlačítek na patrech
- 2) ovládání pohybu kabiny výtahu pomocí tlačítek v kabině
- 3) ovládání pomocných funkcí výtahu
- 4) respektování funkcí tlačítek výtahu v závislosti na stavu výtahu (obsazená, případně plná kabina, . . .)
- 5) sledování provozních a chybových stavů

Propojení PC a Mikrovlnné trouby:

0x300 P1 (OUT)		0x301 P2 (OUT)	
Jméno pinu	Číslo bitu	Jméno pinu	Číslo bitu
LED UP	0	A	0
LED DOWN	1	B	1
MZ	2	C	2
MS	3	ZZ	3
SK	4		4
	5		5
	6		6
	7		7

0x300 P3 (IN)		0x301 P4 (IN)	
Jméno pinu	Číslo bitu	Jméno pinu	Číslo bitu
	0	TP 1	0
PS	1	TP 2	1
DS	2	TP 3	2
IS	3	TP 4	3
1 patro snímač	4	TK 1	4
2 patro snímač	5	TK 2	5
3 patro snímač	6	TK 3	6
4 patro snímač	7	TK 4	7

TP = Tlačítko v patře

TK = Tlačítko v kabině

Vývojový diagram:

Příloha 1 – vývojový diagram

Výpis programu:

Příloha 2 – výpis programu



Závěr:

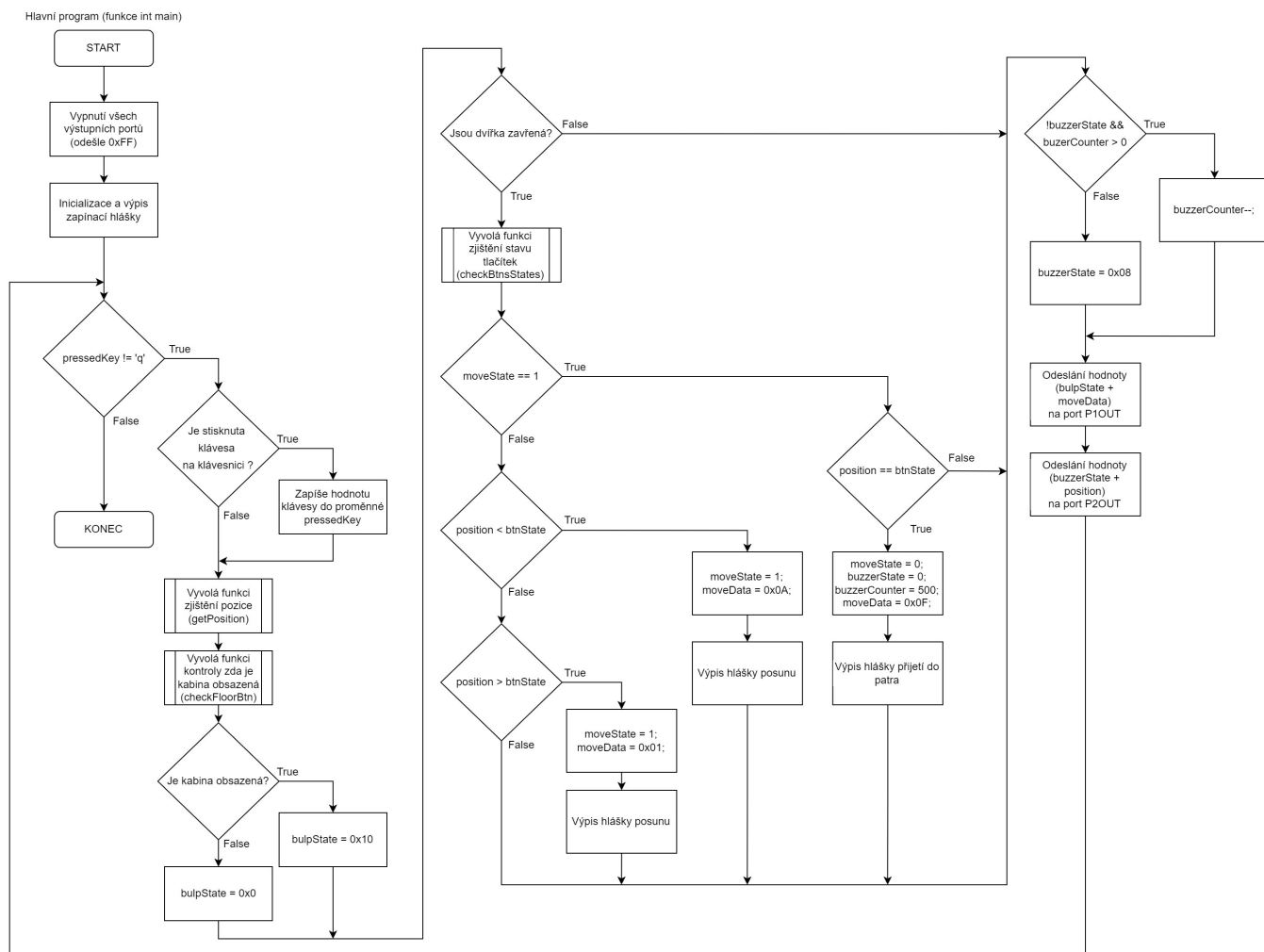
Program je sestaven dle zadání a měl by být funkční. Se zpracováním této úlohy nebyl žádný problém.

Přílohy:

- Příloha 1 – 2 strany
- Příloha 2 – 5 stran

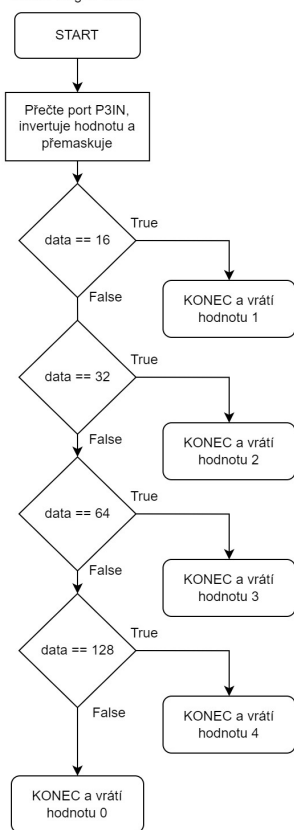


Příloha 1:

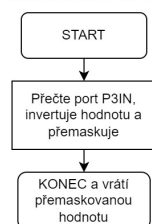




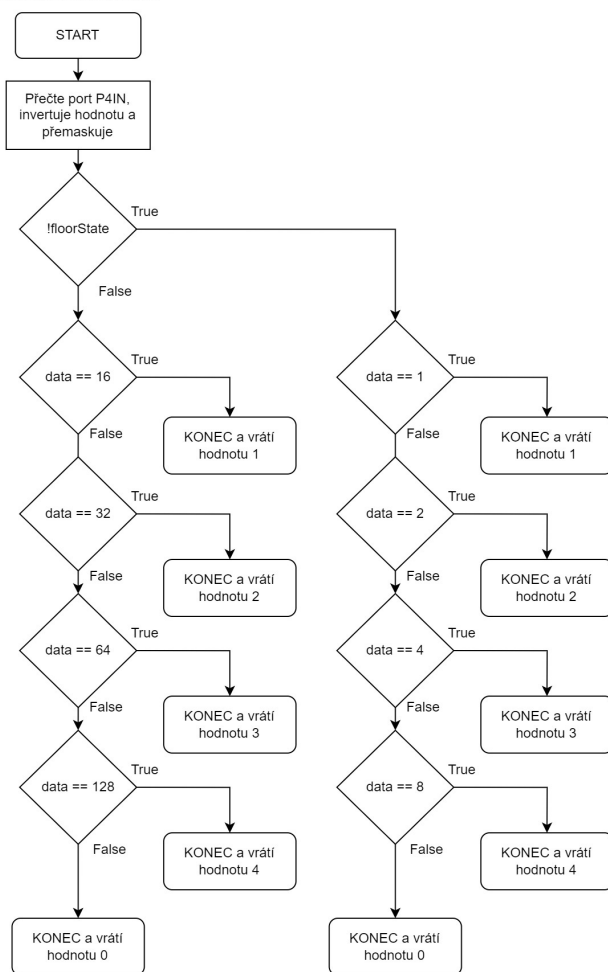
funkce int getPosition



funkce int checkFloorBtn



funkce int checkBtnsStates





Příloha 2:

/* Dokumentace

P1OUT = bity | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
 data | - | - | - | SK | MS | MZ | L2 | L1 |

P2OUT = bity | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
 data | - | - | - | - | ZZ | A | B | C |

P3IN = bity | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
 data | 4p | 3p | 2p | 1p | IS | DS | PS | - |

P4IN = bity | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
 data | Tk4 | Tk3 | Tk2 | Tk1 | Tp4 | Tp3 | Tp2 | Tp1 |

Tp = tlacitka v patrech

Tk = tlacitka v kabině

1p - 4p = snímace pater

PS = podlahový spínač

DS = dveřní spínač

MS = směr otáčení motoru

MZ = zapnutí otáčení motoru

SK = zapnutí světla kabiny

ZZ = zvukové upozornění

L1 = LED UP

L2 = LED DOWN

*/

#include <stdio.h>

#include <dos.h>

#include <time.h>

#include <conio.h>

#define P1OUT 0x300

#define P2OUT 0x301

#define P3IN 0x300

#define P4IN 0x301

int getPosition()

{
 int data = ~inportb(P3IN) & 0xF0;

 switch(data)

 {

 case 16:

 return 1;

 case 32:



```
        return 2;
    case 64:
        return 3;
    case 128:
        return 4;
    default:
        return 0;
    }
}

int checkFloorBtn()
{
    return ~inportb(P3IN) & 0x02;
}

int checkBtnsStates(int floorState)
{
    int data = ~inportb(P4IN); // Kontrola tlacitek pater

    // Kontrola 1-4 tlacitek pater a 1-4 tlacitek kabiny
    if (!floorState)
    {
        switch(data)
        {
            case 1:
                return 1;
            case 2:
                return 2;
            case 4:
                return 3;
            case 8:
                return 4;
            default:
                return 0;
        }
    }
    else
    {
        switch(data)
        {
            case 16:
                return 1;
            case 32:
                return 2;
            case 64:
                return 3;
        }
    }
}
```



```
        case 128:
            return 4;
        default:
            return 0;
    }
}
}

int checkDoorState()
{
    return ~inportb(P3IN) & 0x04;
}

int main()
{
    // Vypnutí vseho na vystupnich portech
    outportb(P1OUT, 0xFF);
    outportb(P2OUT, 0xFF);

    char pressedKey;
    int position = 0, btnState = 0, floorState = 0, moveState = 0, moveData = 0x0F;
    int buzzerState = 0x08, buzzerCounter = 0, bulpState = 0x10;

    // Vymazani obrazovky + vypis zakladnich informaci k ovladani
    printf("\x1B[2J\x1B[H");
    printf("\n\r[#####]");
    printf("\n\r Zapinani programu Osobni vytah, vytvoril Jakub Tenk");
    printf("\n\r[#####]");

    // Hlavni cyklus
    while (pressedKey != 'q')
    {
        // Cteni stavu klavesnice PC
        if (kbhit())
            pressedKey = getch();

        position = getPosition();
        floorState = checkFloorBtn();

        // Data pro sviceni kabiny, pokud je zde osoba
        bulpState = floorState ? 0x10 : 0;

        // Zjisteni stavu dveri
        if (!checkDoorState())
        {
            btnState = checkBtnsStates(floorState);
```




```
if (moveState) // Pokud kabina je v pohybu
{
    if (position == btnState) // Pokud kabina dorazi do patra
    {
        moveState = 0;
        buzzerState = 0;
        buzzerCounter = 500;
        moveData = 0x0F;
        printf("\n\r Vitejte v patre c. %i", btnState);
    }
}
else
{
    if (position < btnState) // Posun nahoru
    {
        moveState = 1;
        moveData = 0x0A;
        printf("\n\r Posun nahoru do patra c. %i", btnState);
    }
    else if (position > btnState) // Posun dolu
    {
        moveState = 1;
        moveData = 0x01;
        printf("\n\r Posun dolu do patra c. %i", btnState);
    }
}
}

// Data pro zapnutí/vypnutí bzucaku
if (!buzzerState && buzzerCounter > 0)
    buzzerCounter--;
else
    buzzerState = 0x08;

// Port pro ovladani motoru a sviceni
outportb(P1OUT, bulpState + moveData);

// Port pro ovladani bzucaku a displeje
outportb(P2OUT, buzzerState + position);

delay(1);
}

printf("\n\r\n\r[#####]");
printf("\n\r Vypinani programu Osobni vytah");
```



```
printf("\n\n[#####]");  
delay(1000);  
  
return 0;  
}
```