

```

;*****
;>>>>> Jan Bittner 4.D 2016 <<<<<<<
;*****
;>>>>>> Roboticke vozitko 03 <<<<<<<
;*****

;PA4-7: dalkove ovladani

;PB0: enable budice
;PB1: indikator moc blizko - blikacka
;PB2: indikator pomalu
;PB3: indikator rychle
;PB4: svetlo zadni prave
;PB5: svetlo zadni leve
;PB6: svetlo predni
;PB7: svetlo predni

;PC0,1,6,7: rizeni motoru

;PD2: echo zadni INT0
;PD3: echo predni INT1
;PD4: PWM prave
;PD5: PWM leve
;PD6: trig zadni
;PD7: trig predni

;-----

;r16: pracovni registr
;r17,r18: stavy tlacitek
;r19: stavove slovo
;r22,r13,r12: podprogramy Delay
;r21: podprogram Delay1 v preruseni
;r23: pracovni registr v preruseni pretececi T0
;r20: pracovni registr v preruseni start_pulsu
;r0: vzdalenost zadni
;r1: vzdalenost predni
;r2: stavovy registr na mereni rychlosti - suda dopredu, licha dozadu

;-----

.include "M32def.inc"

.equ DelayTime = 20 ;zpozdeni 20 x 0.768ms = 15.36ms ->65Hz

.equ PocatecniStav = 0b11110011 ;pocatecni nastaveni svetel a ledek
.equ MaskaRychlost = 0b00001000 ;rychlost
.equ MaskaSvetla = 0b11110000 ;svetla

;rychlosti motoru
.equ rychlost0=0 ;stop
.equ rychlost1=160 ;pomalu
.equ rychlost2=255 ;rychle

;tlacitka dalkove ovladani
.equ TlacDopredu = 0b01110000
.equ TlacDozadu = 0b10110000
.equ TlacVpravo = 0b11010000
.equ TlacVlevo = 0b11100000
.equ TlacDopreduR = 0b01010000
.equ TlacDopreduL = 0b01100000
.equ TlacDozaduR = 0b10010000
.equ TlacDozaduL = 0b10100000
.equ Tlac1 = 0b00000111
.equ Tlac2 = 0b00001011
.equ Tlac3 = 0b00001101
.equ Tlac4 = 0b00001110
.equ TlacStop = 0b11110000

;-----

.macro DOPREDU
    ldi r16, 0b10000001

```

```
        out PORTC, r16
        call rychlost
.endmacro

.macro DOZADU
        ldi r16, 0b01000010
        out PORTC, r16
        call rychlost
.endmacro

.macro DOPREDU_R
        ldi r16, 0b000000001
        out PORTC, r16
        call rychlost
.endmacro

.macro DOZADU_R
        ldi r16, 0b000000010
        out PORTC, r16
        call rychlost
.endmacro

.macro DOPREDU_L
        ldi r16, 0b100000000
        out PORTC, r16
        call rychlost
.endmacro

.macro DOZADU_L
        ldi r16, 0b010000000
        out PORTC, r16
        call rychlost
.endmacro

.macro OTOC_R
        ldi r16, 0b010000001
        out PORTC, r16
        POMALU
.endmacro

.macro OTOC_L
        ldi r16, 0b100000010
        out PORTC, r16
        POMALU
.endmacro

.macro STUJ
        ldi r16, 0b000000000
        out PORTC, r16
        ldi r16, rychlost0
        out OCR1AL, r16      ;dolni byte
        out OCR1BL, r16      ;dolni byte
.endmacro

.macro POMALU
        ldi r16, rychlost1
        out OCR1AL, r16      ;dolni byte
        out OCR1BL, r16      ;dolni byte
.endmacro

.macro RYCHLE
        ldi r16, rychlost2
        out OCR1AL, r16      ;dolni byte
        out OCR1BL, r16      ;dolni byte
.endmacro

.dseg
.org 0x100
data:.byte 1024
;-----

.cseg
.org 0
```

```

reset:  jmp start          ;reset vektor
;-----

.org 0x02
        jmp cidlo_zadni

.org 0x04
        jmp cidlo_predni

.org 0x08
        jmp start_pulsu

.org 0x16
        jmp pretezeniT0

;-----

.org 0x100
start:  ldi r16,0x5F        ;nastaveni ukazatele
        out SPL,r16        ;zasobniku na 0x085F
        ldi r16,0x08        ;konec SRAM
        out SPH,r16

        ldi r16,0x00        ;vstupni
        out DDRA,r16        ;Port A vstupni
        ldi r16,0xff        ;zapnuti pull-up rezistoru
        out PORTA,r16

        ldi r16,0xff        ;vystupni
        out DDRB,r16        ;Port B vystupni

        ldi r16,0xff        ;vystupni
        out DDRC,r16        ;Port C vystupni

        ldi r16, 0xf0        ;vystupni 7,6,5,4
        out DDRD, r16        ;Port D vystupni
        ldi r16,0x0f        ;zapnuti pull-up rezistoru
        out PORTD,r16

        ldi r16, 0x05
        out MCUCR, r16        ;preruseni od nabezne a sestupne hrany INT0, INT1
        sei                  ;povoleni preruseni
;-----

        ldi r16, 0x00        ;nulovani citace 1
        out TCNT1H,r16
        out TCNT1L,r16

        ldi r16, 0x00        ;nastaveni komparacniho registru 1A
        out OCR1AH,r16        ;horni byte
        ldi r16,rychlost0
        out OCR1AL,r16        ;dolni byte

        ldi r16, 0x00        ;nastaveni komparacniho registru 1B
        out OCR1BH, r16        ;horni byte
        ldi r16, rychlost0
        out OCR1BL, r16        ;dolni byte

        ldi r16, 0b10101101
        out TCCR1A, r16

        ldi r16, 0b00001001
        out TCCR1B, r16

;-----

        ldi r16, 0x00        ;nulovani citace 2
        out TCNT2, r16

        ldi r16, 244          ;nastaveni komparacniho registru 2
        out OCR2, r16        ;odpovida 16 Hz

        ldi r16, 0x07        ;deleno 1024

```

```
    out TCCR2, r16

    ldi r16, 0x00

    out ASSR, r16

    ldi r16, 0x81      ;povoleni preruseni
    out TIMSK, r16     ;komparacni jednotky 2
                      ;pretecení T0

;=====

    STUJ               ;pocatecni stav
    ldi r19, PocatecniStav

    clr r2

;-----

cyklus: out PORTB, r19      ;Aktualizace LED
        call Delay

        in r17, PINA       ;Nacteni stavu tlacitek
        mov r18, r17
        andi r17, 0xf0     ;R17 - rizeni jizdy
        andi r18, 0x0f     ;R18 - pridavne funkce

        cpi r17, TlacDopredu
        brne c01
        DOPREDU

c01:    cpi r17, TlacDopreduR
        brne c02
        DOPREDU_R

c02:    cpi r17, TlacDopreduL
        brne c03
        DOPREDU_L

c03:    cpi r17, TlacDozadu
        brne c04
        DOZADU

c04:    cpi r17, TlacDozaduR
        brne c05
        DOZADU_R

c05:    cpi r17, TlacDozaduL
        brne c06
        DOZADU_L

c06:    cpi r17, TlacVpravo      ;pri tlacitku doprava
        brne c07                ;-> otacej vpravo na miste
        OTOC_R

c07:    cpi r17, TlacVlevo      ;pri tlacitku vlevo
        brne c08                ;-> otacej doleva na miste
        OTOC_L

c08:    cpi r17, TlacStop       ;test na uvolneni tlacitek
        breq c10                ;neni stisknute tlacitko jizdy
        rjmp cyklus            ;opakovani cyklu pri jizde

;-----

c10:    STUJ

        cpi r18, Tlac1          ;pri tlacitku 1
        brne c11                ;-> prepinej rychlost
        call zmena_rychlosti

c11:    cpi r18, Tlac2          ;pri tlacitku 2
```

```
        brne c12                ;-> prepinej svetla
        call zmena_svetel

c12:
        rjmp cyklus

;-----
konec:  rjmp konec

;=====

zmena_rychlosti:

        sbrs r19, 3
        rjmp c31

        ldi r16, MaskaRychlost
        com r16
        and r19, r16            ;kdyz je 1 -> zmena na 0
        rjmp c32

c31:    ori r19, MaskaRychlost   ;kdyz je 0 -> zmena na 1

c32:    call Delay2
        ret

;-----

rychlost:

        sbrs r19, 3            ;pokud je nastavena rychla rychlost
        rjmp c41

        POMALU                 ;jede se pomalu
        rjmp c42

c41:    RYCHLE                 ;jede se rychle

c42:    ret

;-----

zmena_svetel:

        sbrs r19, 6
        rjmp c51

        ldi r16, MaskaSvetla
        com r16
        and r19, r16            ;kdyz je 1 -> zmena na 0
        rjmp c52

c51:    ori r19, MaskaSvetla    ;kdyz je 0 -> zmena na 1

c52:    call Delay2
        ret

;=====

cidlo_predni:                    ;INT1
        sbis PIND, 3
        rjmp cp0

        ldi r20, 0x00           ;nulovani citace 0
        out TCNT0, r20
        ldi r20, 0x01           ;nulovani priznaku pretecení T0
        out TIFR, r20
        ldi r20, 0x04           ;/4MHz/256 -> 64us
        out TCCR0, r20          ;64/58= 1.1cm x hodnota

        rjmp cpl
```

```

cp0:    ldi r20, 0x00        ;zastaveni citace 0
        out TCCR0, r20
        in r1, TCNT0

        ldi r20, 0x00
        out GICR, r20        ;zakazani preruseni

cp1:    ldi r20, 0xFF
        out GIFR, r20        ;nulovani priznaku preruseni
        reti

;-----

cidlo_zadni:    ;INT0
                sbis PIND, 2
                rjmp cz0

                ldi r20, 0x00        ;nulovani citace 0
                out TCNT0, r20
                ldi r20, 0x01        ;nulovani priznaku pretecení T0
                out TIFR, r20
                ldi r20, 0x04        ;/4MHz/256 -> 64us
                out TCCR0, r20      ;64/58= 1.1cm x hodnota

                rjmp cz1

cz0:    ldi r20, 0x00        ;zastaveni citace 0
        out TCCR0, r20
        in r0, TCNT0

        ldi r20, 0x00
        out GICR, r20        ;zakazani preruseni

cz1:    ldi r20, 0xFF
        out GIFR, r20        ;nulovani priznaku preruseni
        reti

;-----

start_pulsu:
        ldi r20, 0x00        ;nulovani citace 2
        out TCNT2, r20

        sbrs r2, 0          ;stridani predni/zadni cidlo
        rjmp s0

        sbi PORTD, 7        ;predni
        call delay1
        cbi PORTD, 7

        ldi r20, 0xFF
        out GIFR, r20        ;nulovani priznaku preruseni
        ldi r20, 0x80
        out GICR, r20        ;povoleni preruseni INT1

        rjmp s2

s0:    sbi PORTD, 6          ;zadni
        call delay1
        cbi PORTD, 6

        ldi r20, 0xFF
        out GIFR, r20        ;nulovani priznaku preruseni
        ldi r20, 0x40
        out GICR, r20        ;povoleni preruseni INT0

s2:    inc r2                ;zmena predni/zadni cidlo
        reti

;-----
preteceniT0:

```

```
    ldi r23, 0x00      ;zastaveni citace 0
    out TCCR0, r23

    ldi r23, 0xff      ;nastaveni citace na max
    out TCNT0, r23
    reti
```

;-----

```
Delay:    ldi r22, DelayTime ;nastaveni zpozdeni (x0.768ms)
          clr r12            ;pro 4 MHz
Delay0:    nop
          nop
          nop
          nop
          nop
          nop
          nop
          nop
          nop
          dec r12            ;snizeni hodnoty
          brne Delay0       ;opakovani dokud neni 0 (256x)
          dec r22            ;snizeni hodnoty
          brne Delay0       ;opakovani dokud neni 0 (R22x)
          ret
```

;-----

```
Delay1:    ldi r21, 12      ;nastaveni zpozdeni (x10us)
Delay10:    dec r21          ;snizeni hodnoty
          brne Delay10      ;opakovani dokud neni 0
          ret
```

;-----

```
Delay2:    ldi r22, 10      ;nastaveni zpozdeni (x50ms)
          clr r13            ;pro 4 MHz
          clr r12
Delay20:    dec r12          ;snizeni hodnoty
          brne Delay20      ;opakovani dokud neni 0 (256x)
          dec r13            ;snizeni hodnoty
          brne Delay20      ;opakovani dokud neni 0 (256x)
          dec r22            ;snizeni hodnoty
          brne Delay20      ;opakovani dokud neni 0 (R22x)
          ret
```