.include "8535def.inc"

.def temp=r16
.def level=r18
.def curr=r19
.def Delay=r22
.def Delay2=r23
.def Delay3=r24

;**Program look-up table to flash memory
.org 200
.include "iodefs.asm"
.db 0
.org 0

;**Set PortA as output, set all low
ldi temp,$ff
out PORTA,temp
out DDRA,temp
ldi temp,$00
out PORTA,temp

;**Set PortA p1 high to start ADC clock
ldi temp,$02
out PORTA,temp

;**Set up PortB as output
ldi temp,$ff
out PORTB,temp
out DDRB,temp

;**Set up PortC as input
ldi temp,$00
out PORTC,temp
out DDRC,temp

;**Set up PortD as input
ldi temp,$00
out PORTD,temp
out DDRD,temp

;**Two second delay for DC-DC converter transient to reach steady state
ldi Delay,14
ldi Delay2,255
ldi Delay3,255

DLYA:
  ldi Delay2,255
  dec Delay
DLYB:
  ldi Delay3,255
  dec Delay2
DLYC:
dec    Delay3
brne   DLYC
tst    Delay2
brne   DLYB
tst    Delay
brne   DLYA

;**03/27/03 Set Port B, output to PWM to 25% duty cycle initially
ldi    temp,170
out    PORTB,temp

start:
;**Delay for clock: 10usec for 50 kHz sampling
ldi    Delay,$03
DLY:
    dec    Delay
    brne   DLY

;**Set PortA p1 low and PortA p0 low for reading output current at falling edge
cbi    PORTA,1

;**Find correct current level for look-up table
in     curr,PINC

current:
;**test for table1
;**57 is cutoff for table1 when a gain of ten is placed on the sensed current
mov    r20,curr
subi   r20,255
tst    r20
breq   table1
brcs   table1

;**test for table2
;**81 is cutoff for table2
mov    r20,curr
subi   r20,81
tst    r20
breq   table2
brcs   table2

;**test for table3
;**96 is cutoff for table3
mov    r20,curr
subi   r20,96
tst    r20
breq   table3
brcs   table3

;**test for table4
;**112 is cutoff for table4
mov    r20,curr
subi   r20,112
tst    r20
breq   table4
table1:
;**Change stack pointer(SPL and SPH),ZH,ZL for table 1 in the flash memory
;**This sets stack pointer to last internal RAM location before zeros
ldi temp,low(RAMEND)
out SPL,temp
ldi temp,high(RAMEND)
out SPH,temp
ldi ZH,high(2*tableone)
ldi ZL,low(2*tableone)
rjmp voltage

table2:
;**Change SPL and SPH, ZH, ZL for table 2
ldi temp,$7F
out SPL,temp
ldi temp,$00
out SPH,temp
ldi ZH,high(2*tabletwo)
ldi ZL,low(2*tabletwo)
rjmp voltage

table3:
;**Change SPL and SPH, ZH, ZL for table 3
ldi temp,$BF
out SPL,temp
ldi temp,$00
out SPH,temp
ldi ZH,high(2*tablethree)
ldi ZL,low(2*tablethree)
rjmp voltage

table4:
;**change SPL and SPH, ZH, ZL for table 4
ldi temp,$FF
out SPL,temp
ldi temp,$00
out SPH,temp
ldi ZH,high(2*tablefour)
ldi ZL,low(2*tablefour)
rjmp voltage

;**Loop for matching input voltage level.
voltage:
;**Set PortA p1 low and PortA p0 high allowing a read from ch2 of ADC
ldi temp,$03
out PORTA,temp
;**Delay for clock before it goes low to read input voltage on ch2 of ADC
ldi     Delay,$05
DLY3:
    dec     Delay
    brne    DLY3

;**Set PortA p1 high to read ch2 of ADC for input voltage level
cbi     PORTA,1
;**Read sensed input voltage level from PORTC
in      level,PINC

;**Search lookup table and output duty cycle voltage level for the input voltage
findmatch:
lpm
mov     r17,level
sub     r17,r0
tst     r17
breq    output
brcs    output
adiw    ZL,2
rjmp    findmatch

output:
adiw    ZL,1
lpm
out     PORTB,r0

;**Set PortA p1 high and set p0 low to read ch1 of ADC for output current level
cbi     PORTA,0
sbi     PORTA,1

rjmp    start