```
post114 code

// swordfish basic three-channel software pwm using tmr2 and adc inputs
device = 18f4520
clock = 32

include "intosc.bas"
// do not include setdigitalio
// to use portb for the adc all pins must be in analog mode due to the way
// these old pics map adc inputs
dim test as porte.0
dim z as byte
include "adc.bas"
dim x as byte
// adc pot inputs
// used to set the pwm duty cycle for pwm1-pwm3
// connect pot upper lug = vdd, lower lug = gnd, wiper = port io
dim
    pot1 as portb.0,    // rb0/an12
    pot2 as portb.1,    // rb1/an10
    pot3 as portb.2     // rb2/an8

// adc channels
const
    ch_pot1 = 12,       // an12
    ch_pot2 = 10,       // an10
    ch_pot3 = 8         // an8

// pwm outputs
dim
    pwm1 as porta.0,
    pwm2 as porta.1,
    pwm3 as porta.2

// pwm duty cycles (from adc) 0=min, 255=max
dim
    pwm1_duty as byte,
    pwm2_duty as byte,
    pwm3_duty as byte

dim pwm_period as byte

// pwm timer tmr2
dim
    tmr2if as pir1.bits(1),
    tmr2ie as pie1.bits(1),
    tmr2on as t2con.bits(2)

// set io pin directions and initial settings
sub initio()
    // set inputs
    input(pot1)
    input(pot2)
    input(pot3)

    // set outputs (low to start)
    low(pwm1)
    low(pwm2)
    low(pwm3)
end sub

// pwm tmr2 interrupt
interrupt tmr2_isr()
    tmr2if = 0
    pwm_period = pwm_period + 1
    if (pwm_period >= pwm1_duty) then
        pwm1 = 0
    else
        pwm1 = 1
    endif
    if (pwm_period >= pwm2_duty) then
        pwm2 = 1
    else
        pwm2 = 0
    endif
    if (pwm_period >= pwm3_duty) then
        pwm3 = 1
    else
        pwm3 = 0
    endif
end interrupt


main:
for z = 0 to 5
test = 1
delayms(500)
test = 0
delayms(500)
next
initio()

// adc setup
adcon1 = $00        // all pins set to analog mode, vref = vdd/gnd
adcon2 = adc.frc    // adc clock = frc
adc.adfm = 0        // left justify (we only use the 8 msb's)
adc.setacqtime(100) // 100us delay

pwm_period = 0
pwm1_duty = 0
pwm2_duty = 0
pwm3_duty = 0

// setup pwm timer tmr2
// 25khz = 40us/bit -> 40us x 256 = 10240us period, ~10ms period (100hz)
t2con = %00000001   // t2outps<3:0>=%000 (1:1), tmr2on=0, t2ckps<1:0>=%01 (1:4)
pr2 = 176
tmr2 = 0
tmr2if = 0
tmr2ie = 1
tmr2on = 1

enable(tmr2_isr)

while (true)


    pwm1_duty = adc.read(ch_pot1) >> 8
    pwm2_duty = adc.read(ch_pot2) >> 8
    pwm3_duty = adc.read(ch_pot3) >> 8
end while

```