

```

-----
*****
" CNC MPG Pendant's rotary switches encoder interface by Yioryos Dimitriadis June/July 2023 "
pic18f1330 @ 32MHz internal oscillator with 4x PLL
*****
-----
*          DEVICE INITIALISATION          *
-----

list      p=pic18f1330, r=dec
#include   p18f1330.inc

-----
Device Fuses
-----
Only PIC18FXXJ devices do not support this directive, (__config), not PIC18FXXXX!
(Assembler/Linker/Librarian User's Guide DS33014J-page 58)
NOTE: For each config byte, you must write all flags for it to work
e.g. if you write __config __CONFIG2L, __BOR_OFF_2L only, it will NOT work!
__config __CONFIG1H, __OSC_INTIO2_1H & __FCMEN_OFF_1H & __IESO_OFF_1H
__config __CONFIG2L, __PWRT_ON_2L & __BOR_OFF_2L & __BORV_3_2L
__config __CONFIG2H, __WDT_OFF_2H & __WDTPS_32768_2H
__config __CONFIG3L, __HPOL_HIGH_3L & __LPOL_HIGH_3L & __PWMPIN_OFF_3L
__config __CONFIG3H, __FLTAMX_RA5_3H & __T10SCMX_LOW_3H & __MCLRE_OFF_3H
__config __CONFIG4L, __STVREN_ON_4L & __BBSIZ_BB256_4L & __XINST_OFF_4L & __DEBUG_OFF_4L
; config OSC = INTIO2, FCMEN = OFF, IESO = OFF
; config PWRT = ON, BOR = OFF, BORV = 3 ;*(config directive for bor=off does not work, only the old __config)*
; config WDT = OFF, WDTPS = 32768
; config FLTAMX = RA5, T10SCMX = LOW, MCLRE = OFF
; config STVREN = ON, BBSIZ = BB256, XINST = OFF, DEBUG = OFF

-----
Equivalence Declarations
-----
a      equ      0      ; use access RAM bank for reading/writing data
b      equ      1      ; use BSR-specified RAM bank for reading/writing data
f      equ      1      ; destination=register
w      equ      0      ; destination=wreg
p1_1   equ      10011000b ; 1/1 prescaler constant
p1_2   equ      10010000b ; 1/2 prescaler constant
p1_4   equ      10010001b ; 1/4 prescaler constant
p1_8   equ      10010010b ; 1/8 prescaler constant
p1_16  equ      10010011b ; 1/16 prescaler constant
p1_32  equ      10010100b ; 1/32 prescaler constant
p1_64  equ      10010101b ; 1/64 prescaler constant
p1_128 equ      10010110b ; 1/128 prescaler constant
t_dt_f equ      8       ; default dtmf duration multiplication factor
tone_dt equ      250     ; default dtmf duration (value*t_dt_f=(50us resolution))
; (e.g.: 50ms @ a value of 250 and t_dt_f=4)

-----
#define off_pos porta, 2, a ; "off" rotary switch1 position input pin assignment
#define x_pos   porta, 0, a ; "x" rotary switch1 position input pin assignment

```

```

#define y_pos    porta, 1, a    ; "y" rotary switch1 position input pin assignment
#define z_pos    porta, 4, a    ; "z" rotary switch1 position input pin assignment
#define a_pos    porta, 5, a    ; "a" rotary switch1 position input pin assignment
#define b_pos    porta, 3, a    ; "b" rotary switch1 position input pin assignment
#define c_pos    portb, 0, a    ; "c" rotary switch1 position input pin assignment
#define x1_pos   portb, 3, a    ; "x1" rotary switch2 position input pin assignment
#define x10_pos  portb, 2, a    ; "x10" rotary switch2 position input pin assignment
#define x100_pos porta, 7, a    ; "x100" rotary switch2 position input pin assignment
#define A0_out   latb, 6, a     ; "A0" output pin assignment
#define A1_out   latb, 5, a     ; "A1" output pin assignment
#define A2_out   latb, 4, a     ; "A2" output pin assignment
#define B0_out   lata, 6, a     ; "B0" output pin assignment
#define B1_out   latb, 7, a     ; "B1" output pin assignment
;
;-----
;               Program Variables
;-----
;
;               org      0      ; beginning of general purpose register ram, (000h)-bank0
;                               ; (000h -07fh access ram low)
;
presc    res      1            ; timer0 prescaler variable
dtmfkey  res      1            ; dtmf tone 2b played
dtmfdr   res      1            ; dtmf duration register
dtmfdrb  res      1            ; dtmf duration buffer register
dtmfdrf  res      1            ; dtmf duration multiplication factor register
dtmfdrfb res      1            ; dtmf duration multiplication factor bufferregister
dp_col   res      1            ; column datapoints
dp_row   res      1            ; row datapoints
sin_r    res      1            ; 8 bit sin value of row frequency
sin_c    res      1            ; 8 bit sin value of column frequency
r_sw_1   res      1            ; rotary switch 1 register
r_sw_2   res      1            ; rotary switch 2 register
;
;-----
;               **      PROGRAM INITIALIZATION      **
;-----
;
;               org      0      ; go to beginning of program memory, (flash)
;
;               goto     init    ; "goto" in case pclath is another page
;
;               org      8      ; high priority interrupt vector
;
;               goto     $      ; never here
;
;               org      24     ; low priority interrupt vector
;
;               goto     $      ; never here
;
init
;-----
;               Initialize status, bsr & pclath
;

```

```

;-----
;      clr f    status, a
;      clr f    bsr, a
;      clr f    pclath, a
;      clr f    pclatu, a
;      clr f    intcon, a      ; disable interrupts
;-----
;      Internal oscillator setup
;-----
;      bsf      osccon, ircf0, a      ; \
;      bsf      osccon, ircf1, a      ; -|->8MHz internal oscillator
;      bsf      osccon, ircf2, a      ; /
;      bsf      osctune, pllen, a     ; 4x PLL enabled (32MHz Fosc)
;-----
;      Ports setup
;-----
;      clr f    lata, a      ; initialize porta by clearing output data latches
;      movl w    0x07, a      ; configure a/d
;      movwf    adcon1, a     ; for digital inputs
;      movl w    0, a         ; configure comparators
;      movwf    cmcon, a      ; for digital input
;      movl w    10111111b, a
;      movwf    trisa, a      ; porta pins inputs, except pin 6 which is an output
;-----
;      clr f    latb, a      ; initialize portb by clearing output data latches
;      movl w    0x0f, a      ; set rb<4:0> as
;      movwf    adcon1, a     ; digital i/o pins
;      movl w    00001101b, a
;      movwf    trisb, a      ; portb pins 1,4,5,6 & 7 outputs, pins 0,2 & 3 inputs
;-----
;      Timer0 setup
;-----
;      movl w    00010100b
;      movwf    t0con, a      ; timer0 off, 16bit counter, int.instr.clk, prescaler assigned=1/32
;-----
;      PWM setup
;-----
;      clr f    ptperh, a     ; \
;      movl w    255, a       ; -|->period for Fpwm=31.25kHz (8bit resolution at Fosc=32MHz)
;      movwf    ptperl, a     ; /
;      bsf      pwmcon0, pwmen0, a ; pwm on pin rb1 only
;-----
;      Variables Initialisation
;-----
;      clr f    r_sw_1, a     ; initialise rotary switch 1 register
;      clr f    r_sw_2, a     ; initialise rotary switch 2 register
;      call     dur_default    ; set default dtmf duration
;-----
;      *** MAIN PROGRAM ***
;-----

```

```

chk_rsw1
    btfss    off_pos      ; rotary switch 1 at "off" position?
    bra     chk_off_pos   ; yes, it seems like this, check if it was a bounce
    btfss    x_pos       ; no, rotary switch 1 at "x" position?
    bra     chk_x_pos     ; yes, it seems like this, check if it was a bounce
    btfss    y_pos       ; no, rotary switch 1 at "y" position?
    bra     chk_y_pos     ; yes, it seems like this, check if it was a bounce
    btfss    z_pos       ; no, rotary switch 1 at "z" position?
    bra     chk_z_pos     ; yes, it seems like this, check if it was a bounce
    btfss    a_pos       ; no, rotary switch 1 at "a" position?
    bra     chk_a_pos     ; yes, it seems like this, check if it was a bounce
    btfss    b_pos       ; no, rotary switch 1 at "b" position?
    bra     chk_b_pos     ; yes, it seems like this, check if it was a bounce
    btfss    c_pos       ; no, rotary switch 1 at "c" position?
    bra     chk_c_pos     ; yes, it seems like this, check if it was a bounce

chk_rsw2
    btfss    x1_pos      ; no, rotary switch 2 at "x1" position?
    bra     chk_x1_pos   ; yes, it seems like this, check if it was a bounce
    btfss    x10_pos     ; no, rotary switch 2 at "x10" position?
    bra     chk_x10_pos  ; yes, it seems like this, check if it was a bounce
    btfss    x100_pos    ; no, rotary switch 2 at "x100" position?
    bra     chk_x100_pos ; yes, it seems like this, check if it was a bounce
    rcall   del_10ms     ; no, delay 10 milliseconds
    bra     chk_rsw1     ; and go back to rotary switch 1 checking

chk_off_pos
    rcall   del_30ms     ; delay 30 milliseconds
    btfsc   off_pos      ; rotary switch 1 STILL at "off" position?
    bra     chk_rsw2     ; no, it was a bounce, resume to check rotary switch 2
    btfsc   r_sw_1, 0, a ; yes, rsw1 is at "off" position. Was it at "off" previously?
    bra     chk_rsw2     ; yes, resume to check rotary switch 2
    clrf    r_sw_1, a    ; \ no, update
    bsf     r_sw_1, 0, a ; / rsw1 register
    bcf     A0_out       ; \
    bcf     A1_out       ; | -> update output logic
    bcf     A2_out       ; /
    call    snd_off      ; play "off" sound sequence
;    call    cl_enc       ; ** try cl_enc instead of snd_off here **
    bra     chk_rsw2     ; resume to check rotary switch 2

chk_x_pos
    rcall   del_30ms     ; delay 30 milliseconds
    btfsc   x_pos        ; rotary switch 1 STILL at "x" position?
    bra     chk_rsw2     ; no, it was a bounce, resume to check rotary switch 2
    btfsc   r_sw_1, 1, a ; yes, rsw1 is at "x" position. Was it at "x" previously?
    bra     chk_rsw2     ; yes, resume to check rotary switch 2
    clrf    r_sw_1, a    ; \ no, update
    bsf     r_sw_1, 1, a ; / rsw1 register
    bcf     A0_out       ; \
    bcf     A1_out       ; | -> update output logic
    bcf     A2_out       ; /
    call    snd_x        ; play "x" sound sequence

```

```

bra      chk_rsw2      ; resume to check rotary switch 2
chk_y_pos
    rcal  del_30ms      ; delay 30 milliseconds
    btfsc y_pos         ; rotary switch 1 STILL at "y" position?
    bra   chk_rsw2      ; no, it was a bounce, resume to check rotary switch 2
    btfsc r_sw_1, 2, a   ; yes, rsw1 is at "y" position. Was it at "y" previously?
    bra   chk_rsw2      ; yes, resume to check rotary switch 2
    clrf  r_sw_1, a      ; \ no, update
    bsf   r_sw_1, 2, a   ; / rsw1 register
    bcf   A0_out         ; \
    bsf   A1_out         ; |-> update output logic
    bcf   A2_out         ; /
    call  snd_y          ; play "y" sound sequence
    bra   chk_rsw2      ; resume to check rotary switch 2
chk_z_pos
    rcal  del_30ms      ; delay 30 milliseconds
    btfsc z_pos         ; rotary switch 1 STILL at "z" position?
    bra   chk_rsw2      ; no, it was a bounce, resume to check rotary switch 2
    btfsc r_sw_1, 3, a   ; yes, rsw1 is at "z" position. Was it at "z" previously?
    bra   chk_rsw2      ; yes, resume to check rotary switch 2
    clrf  r_sw_1, a      ; \ no, update
    bsf   r_sw_1, 3, a   ; / rsw1 register
    bcf   A0_out         ; \
    bsf   A1_out         ; |-> update output logic
    bcf   A2_out         ; /
    call  snd_z          ; play "z" sound sequence
    bra   chk_rsw2      ; resume to check rotary switch 2
chk_a_pos
    rcal  del_30ms      ; delay 30 milliseconds
    btfsc a_pos         ; rotary switch 1 STILL at "a" position?
    bra   chk_rsw2      ; no, it was a bounce, resume to check rotary switch 2
    btfsc r_sw_1, 4, a   ; yes, rsw1 is at "a" position. Was it at "a" previously?
    bra   chk_rsw2      ; yes, resume to check rotary switch 2
    clrf  r_sw_1, a      ; \ no, update
    bsf   r_sw_1, 4, a   ; / rsw1 register
    bcf   A0_out         ; \
    bcf   A1_out         ; |-> update output logic
    bsf   A2_out         ; /
    call  snd_a          ; play "a" sound sequence
    bra   chk_rsw2      ; resume to check rotary switch 2
chk_b_pos
    rcal  del_30ms      ; delay 30 milliseconds
    btfsc b_pos         ; rotary switch 1 STILL at "b" position?
    bra   chk_rsw2      ; no, it was a bounce, resume to check rotary switch 2
    btfsc r_sw_1, 5, a   ; yes, rsw1 is at "b" position. Was it at "b" previously?
    bra   chk_rsw2      ; yes, resume to check rotary switch 2
    clrf  r_sw_1, a      ; \ no, update
    bsf   r_sw_1, 5, a   ; / rsw1 register
    bsf   A0_out         ; \
    bcf   A1_out         ; |-> update output logic

```

```

        bsf      A2_out      ; /
        call     snd_b       ; play "b" sound sequence
        bra      chk_rsw2    ; resume to check rotary switch 2
chk_c_pos
        rcall    del_30ms    ; delay 30 milliseconds
        btfsc    c_pos      ; rotary switch 1 STILL at "c" position?
        bra      chk_rsw2    ; no, it was a bounce, resume to check rotary switch 2
        btfsc    r_sw_1, 6, a ; yes, rsw1 is at "c" position. Was it at "c" previously?
        bra      chk_rsw2    ; yes, resume to check rotary switch 2
        clrf     r_sw_1, a   ; \ no, update
        bsf      r_sw_1, 6, a ; / rsw1 register
        bcf      A0_out      ; \
        bsf      A1_out      ; | -> update output logic
        bsf      A2_out      ; /
        call     snd_c       ; play "c" sound sequence
        bra      chk_rsw2    ; resume to check rotary switch 2
chk_x1_pos
        rcall    del_30ms    ; delay 30 milliseconds
        btfsc    x1_pos     ; rotary switch 2 STILL at "x1" position?
        bra      chk_rsw1    ; no, it was a bounce, resume to check rotary switch 1
        btfsc    r_sw_2, 0, a ; yes, rsw2 is at "x1" position. Was it at "x1" previously?
        bra      chk_rsw1    ; yes, resume to check rotary switch 1
        clrf     r_sw_2, a   ; \ no, update
        bsf      r_sw_2, 0, a ; / rsw2 register
        bsf      B0_out      ; \ update output
        bcf      B1_out      ; / logic
        btfss    r_sw_1, 0, a ; rsw1 at "off" position?
        call     snd_x1      ; no, play "x1" sound sequence
        bra      chk_rsw1    ; yes, skip audio out & resume to check rotary switch 1
chk_x10_pos
        rcall    del_30ms    ; delay 30 milliseconds
        btfsc    x10_pos    ; rotary switch 2 STILL at "x10" position?
        bra      chk_rsw1    ; no, it was a bounce, resume to check rotary switch 1
        btfsc    r_sw_2, 1, a ; yes, rsw2 is at "x10" position. Was it at "x10" previously?
        bra      chk_rsw1    ; yes, resume to check rotary switch 1
        clrf     r_sw_2, a   ; \ no, update
        bsf      r_sw_2, 1, a ; / rsw2 register
        bcf      B0_out      ; \ update output
        bsf      B1_out      ; / logic
        btfss    r_sw_1, 0, a ; rsw1 at "off" position?
        call     snd_x1      ; no, play "x10" sound sequence
        bra      chk_rsw1    ; yes, skip audio out & resume to check rotary switch 1
chk_x100_pos
        rcall    del_30ms    ; delay 30 milliseconds
        btfsc    x100_pos   ; rotary switch 2 STILL at "x100" position?
        bra      chk_rsw1    ; no, it was a bounce, resume to check rotary switch 1
        btfsc    r_sw_2, 2, a ; yes, rsw2 is at "x100" position. Was it at "x100" previously?
        bra      chk_rsw1    ; yes, resume to check rotary switch 1
        clrf     r_sw_2, a   ; \ no, update
        bsf      r_sw_2, 2, a ; / rsw2 register

```

```

    bsf      B0_out      ; \ update output
    bsf      B1_out      ; / logic
    btfss    r_sw_1, 0, a ; rsw1 at "off" position?
    call     snd_x100     ; no, play "x100" sound sequence
    bra      chk_rsw1     ; yes, skip audio out & resume to check rotary switch 1

;-----
;*** PROGRAM SUBROUTINES ***
;-----

;*** 50us EVENT TIMER ***
;-----
event_tmr
    clrf     tmr0h, a
    movlw    98
    movwf    tmr0l, a
    movlw    p1_1
    movwf    presc, a
    bcf      t0con, tmr0on, a ; turn off timer0
    bcf      intcon, tmr0if, a ; precautionary clear tmr0 interrupt flag
    comf      tmr0h, 1, a
    comf      tmr0l, 1, a ; FIRST WRITE IN tmr0h so that it will be updated after writing to tmr0l!!
    movff    presc, t0con ; timer0 on, 16bit counter, int.instr.clk, prescaler assigned=1/x
    return

event_tmr_off
    bcf      intcon, tmr0if, a ; clear tmr0 interrupt flag
    bcf      t0con, tmr0on, a ; turn off timer0
    return

;-----
;*** VARIABLE DELAY ***
;-----
vard
    bcf      t0con, tmr0on, a ; turn off timer0
    bcf      intcon, tmr0if, a ; precautionary clear tmr0 interrupt flag
    comf      tmr0h, 1, a
    comf      tmr0l, 1, a ; FIRST WRITE IN tmr0h so that it will be updated after writing to tmr0l!!
    movff    presc, t0con ; timer0 on, 16bit counter, int.instr.clk, prescaler assigned=1/x
    btfss    intcon, tmr0if, a ; overflow?
    bra      $-1*2 ; no
    bcf      intcon, tmr0if, a ; yes, clear tmr0 interrupt flag
    bcf      t0con, tmr0on, a ; turn off timer0
    return

; NOTE: x*prescaler value*0.5=desired delay in useconds
; where x is the 16 bit value loaded in tmr0 before calling vard!
; (prescaler value for above is eg 256 for a 1/256 prescaler setting)
;-----
;*** FIXED DELAYS ***
;-----
del_5us

```

```

        clrf      tmr0h, a
        movl w    25                ; instead of 40 for compensating for tcy losses
        movwf    tmr0l, a
        movl w    p1_1
        movwf    presc, a
        rcal l    vard
        return

del_50us
        clrf      tmr0h, a
        movl w    35                ; instead of 50 for compensating for tcy losses
        movwf    tmr0l, a
        movl w    p1_8
        movwf    presc, a
        rcal l    vard
        return

del_2_5ms
        movl w    0x02
        movwf    tmr0h, a
        movl w    0x71
        movwf    tmr0l, a
        movl w    p1_32
        movwf    presc, a
        rcal l    vard
        return

del_5ms
        movl w    0x04
        movwf    tmr0h, a
        movl w    0xE2
        movwf    tmr0l, a
        movl w    p1_32
        movwf    presc, a
        rcal l    vard
        return

del_10ms
        movl w    0x09
        movwf    tmr0h, a
        movl w    0xC4
        movwf    tmr0l, a
        movl w    p1_32
        movwf    presc, a
        rcal l    vard
        return

del_20ms
        movl w    0x13
        movwf    tmr0h, a
        movl w    0x88
        movwf    tmr0l, a
        movl w    p1_32
        movwf    presc, a
        rcal l    vard

```



```
    return
del_30ms
    movl w    0x1D
    movwf    tmr0h, a
    movl w    0x4C
    movwf    tmr0l, a
    movl w    p1_32
    movwf    presc, a
    rcal l    vard
    return
del_50ms
    movl w    0x30
    movwf    tmr0h, a
    movl w    0xD4
    movwf    tmr0l, a
    movl w    p1_32
    movwf    presc, a
    rcal l    vard
    return
del_100ms
    movl w    0x61
    movwf    tmr0h, a
    movl w    0xA8
    movwf    tmr0l, a
    movl w    p1_32
    movwf    presc, a
    rcal l    vard
    return
del_200ms
    movl w    0xC3
    movwf    tmr0h, a
    movl w    0x50
    movwf    tmr0l, a
    movl w    p1_32
    movwf    presc, a
    rcal l    vard
    return
del_250ms
    movl w    0xF4
    movwf    tmr0h, a
    movl w    0x24
    movwf    tmr0l, a
    movl w    p1_32
    movwf    presc, a
    rcal l    vard
    return
del_0_5s
    movl w    0xF4
    movwf    tmr0h, a
    movl w    0x24
```

```

        movwf    tmr0l, a
        movl w    p1_64
        movwf    presc, a
        rcall    vard
        return

del_1s
        movl w    0xF4
        movwf    tmr0h, a
        movl w    0x24
        movwf    tmr0l, a
        movl w    p1_128
        movwf    presc, a
        rcall    vard
        return

```

```

-----
***      DTMF  ENCODING  SUBROUTINES      ***
-----

```

```

; Send DTMF tone via a PWM on pin RB1
-----

```

```

send_DTMF
        rcall    event_tmr      ; 50us timer on
        movf     dtmfkey, w, a
        subl w    "0"           ; w-literal, result in w
        btfsc    status, z, a    ; is dtmfkey="0"?
        rcall    dtmf_0         ; yes
        movf     dtmfkey, w, a
        subl w    "1"           ; w-literal, result in w
        btfsc    status, z, a    ; is dtmfkey="1"?
        rcall    dtmf_1         ; yes
        movf     dtmfkey, w, a
        subl w    "2"           ; w-literal, result in w
        btfsc    status, z, a    ; is dtmfkey="2"?
        rcall    dtmf_2         ; yes
        movf     dtmfkey, w, a
        subl w    "3"           ; w-literal, result in w
        btfsc    status, z, a    ; is dtmfkey="3"?
        rcall    dtmf_3         ; yes
        movf     dtmfkey, w, a
        subl w    "4"           ; w-literal, result in w
        btfsc    status, z, a    ; is dtmfkey="4"?
        rcall    dtmf_4         ; yes
        movf     dtmfkey, w, a
        subl w    "5"           ; w-literal, result in w
        btfsc    status, z, a    ; is dtmfkey="5"?
        rcall    dtmf_5         ; yes
        movf     dtmfkey, w, a
        subl w    "6"           ; w-literal, result in w
        btfsc    status, z, a    ; is dtmfkey="6"?
        rcall    dtmf_6         ; yes

```

```

    movf    dtmfkey, w, a    ; no
    subl w  "7"              ; w-literal, result in w
    btfsc   status, z, a     ; is dtmfkey="7"?
    rcall   dtmf_7           ; yes
    movf    dtmfkey, w, a    ; no
    subl w  "8"              ; w-literal, result in w
    btfsc   status, z, a     ; is dtmfkey="8"?
    rcall   dtmf_8           ; yes
    movf    dtmfkey, w, a    ; no
    subl w  "9"              ; w-literal, result in w
    btfsc   status, z, a     ; is dtmfkey="9"?
    rcall   dtmf_9           ; yes
    movf    dtmfkey, w, a    ; no
    subl w  "A"              ; w-literal, result in w
    btfsc   status, z, a     ; is dtmfkey="A"?
    rcall   dtmf_A           ; yes
    movf    dtmfkey, w, a    ; no
    subl w  "B"              ; w-literal, result in w
    btfsc   status, z, a     ; is dtmfkey="B"?
    rcall   dtmf_B           ; yes
    movf    dtmfkey, w, a    ; no
    subl w  "C"              ; w-literal, result in w
    btfsc   status, z, a     ; is dtmfkey="C"?
    rcall   dtmf_C           ; yes
    movf    dtmfkey, w, a    ; no
    subl w  "D"              ; w-literal, result in w
    btfsc   status, z, a     ; is dtmfkey="D"?
    rcall   dtmf_D           ; yes
    movf    dtmfkey, w, a    ; no
    subl w  "*"              ; w-literal, result in w
    btfsc   status, z, a     ; is dtmfkey="*"?
    rcall   dtmf_s           ; yes
    movf    dtmfkey, w, a    ; no
    subl w  "#"              ; w-literal, result in w
    btfsc   status, z, a     ; is dtmfkey="#"?
    rcall   dtmf_di          ; yes
    return

;-----
; DTMF "1"      (1209Hz+697Hz)
;-----
dtmf_1
    movl w  33                ; 33 data points with 2 sine periods
    movwf   dp_col, a
    movl w  86                ; 86 data points with 3 sine periods
    movwf   dp_row, a
dtmf_1_loop
    rcall   f_1209Hz
    rcall   f_697Hz
    rcall   add_freq
    movl w  0

```

```

        cpfseq    dtmfdrf, a
        bra       dtmf_1_loop
        bra       dtmf_end
;-----
; DTMF "2"      (1336Hz+697Hz)
;-----
dtmf_2
        movl w    15                ; 15 data points with 1 sine period
        movwf     dp_col, a
        movl w    86                ; 86 data points with 3 sine periods
        movwf     dp_row, a
dtmf_2_loop
        rcal l    f_1336Hz
        rcal l    f_697Hz
        rcal l    add_freq
        movl w    0
        cpfseq    dtmfdrf, a
        bra       dtmf_2_loop
        bra       dtmf_end
;-----
; DTMF "3"      (1477Hz+697Hz)
;-----
dtmf_3
        movl w    27                ; 27 data points with 2 sine periods
        movwf     dp_col, a
        movl w    86                ; 86 data points with 3 sine periods
        movwf     dp_row, a
dtmf_3_loop
        rcal l    f_1477Hz
        rcal l    f_697Hz
        rcal l    add_freq
        movl w    0
        cpfseq    dtmfdrf, a
        bra       dtmf_3_loop
        bra       dtmf_end
;-----
; DTMF "4"      (1209Hz+770Hz)
;-----
dtmf_4
        movl w    33                ; 33 data points with 2 sine periods
        movwf     dp_col, a
        movl w    26                ; 26 data points with 1 sine period
        movwf     dp_row, a
dtmf_4_loop
        rcal l    f_1209Hz
        rcal l    f_770Hz
        rcal l    add_freq
        movl w    0
        cpfseq    dtmfdrf, a
        bra       dtmf_4_loop

```

```

        bra        dtmf_end
;-----
; DTMF "5"        (1336Hz+770Hz)
;-----
dtmf_5
        movl w      15                ; 15 data points with 1 sine period
        movwf       dp_col, a
        movl w      26                ; 26 data points with 1 sine period
        movwf       dp_row, a
dtmf_5_loop
        rcal l      f_1336Hz
        rcal l      f_770Hz
        rcal l      add_freq
        movl w      0
        cpfseq      dtmfdrf, a
        bra         dtmf_5_loop
        bra         dtmf_end
;-----
; DTMF "6"        (1477Hz+770Hz)
;-----
dtmf_6
        movl w      27                ; 27 data points with 2 sine periods
        movwf       dp_col, a
        movl w      26                ; 26 data points with 1 sine period
        movwf       dp_row, a
dtmf_6_loop
        rcal l      f_1477Hz
        rcal l      f_770Hz
        rcal l      add_freq
        movl w      0
        cpfseq      dtmfdrf, a
        bra         dtmf_6_loop
        bra         dtmf_end
;-----
; DTMF "7"        (1209Hz+852Hz)
;-----
dtmf_7
        movl w      33                ; 33 data points with 2 sine periods
        movwf       dp_col, a
        movl w      47                ; 47 data points with 2 sine periods
        movwf       dp_row, a
dtmf_7_loop
        rcal l      f_1209Hz
        rcal l      f_852Hz
        rcal l      add_freq
        movl w      0
        cpfseq      dtmfdrf, a
        bra         dtmf_7_loop
        bra         dtmf_end
;-----

```

```

; DTMF "8"      (1336Hz+852Hz)
; -----
dtmf_8
    movl w    15          ; 15 data points with 1 sine period
    movwf    dp_col, a
    movl w    47          ; 47 data points with 2 sine periods
    movwf    dp_row, a
dtmf_8_loop
    rcal l    f_1336Hz
    rcal l    f_852Hz
    rcal l    add_freq
    movl w    0
    cpfseq   dtmfdrf, a
    bra      dtmf_8_loop
    bra      dtmf_end
; -----
; DTMF "9"      (1477Hz+852Hz)
; -----
dtmf_9
    movl w    27          ; 27 data points with 2 sine periods
    movwf    dp_col, a
    movl w    47          ; 47 data points with 2 sine periods
    movwf    dp_row, a
dtmf_9_loop
    rcal l    f_1477Hz
    rcal l    f_852Hz
    rcal l    add_freq
    movl w    0
    cpfseq   dtmfdrf, a
    bra      dtmf_9_loop
    bra      dtmf_end
; -----
; DTMF "*"      (1209Hz+941Hz)
; -----
dtmf_s
    movl w    33          ; 33 data points with 2 sine periods
    movwf    dp_col, a
    movl w    85          ; 85 data points with 4 sine periods
    movwf    dp_row, a
dtmf_s_loop
    rcal l    f_1209Hz
    rcal l    f_941Hz
    rcal l    add_freq
    movl w    0
    cpfseq   dtmfdrf, a
    bra      dtmf_s_loop
    bra      dtmf_end
; -----
; DTMF "0"      (1336Hz+941Hz)
; -----

```

```

dtmf_0
    movl w    15                ; 15 data points with 1 sine period
    movwf    dp_col, a
    movl w    85                ; 85 data points with 4 sine periods
    movwf    dp_row, a
dtmf_0_loop
    rcal l    f_1336Hz
    rcal l    f_941Hz
    rcal l    add_freq
    movl w    0
    cpfseq   dtmfdrf, a
    bra      dtmf_0_loop
    bra      dtmf_end

; -----
; DTMF "#"      (1477Hz+941Hz)
; -----
dtmf_di
    movl w    27                ; 27 data points with 2 sine periods
    movwf    dp_col, a
    movl w    85                ; 85 data points with 4 sine periods
    movwf    dp_row, a
dtmf_di_loop
    rcal l    f_1477Hz
    rcal l    f_941Hz
    rcal l    add_freq
    movl w    0
    cpfseq   dtmfdrf, a
    bra      dtmf_di_loop
    bra      dtmf_end

; -----
; DTMF "A"      (1633Hz+697Hz)
; -----
dtmf_A
    movl w    49                ; 49 data points with 4 sine periods
    movwf    dp_col, a
    movl w    86                ; 86 data points with 3 sine periods
    movwf    dp_row, a
dtmf_A_loop
    rcal l    f_1633Hz
    rcal l    f_697Hz
    rcal l    add_freq
    movl w    0
    cpfseq   dtmfdrf, a
    bra      dtmf_A_loop
    bra      dtmf_end

; -----
; DTMF "B"      (1633Hz+770Hz)
; -----
dtmf_B
    movl w    49                ; 49 data points with 4 sine periods

```

```

        movwf    dp_col, a
        movl w    26                ; 26 data points with 1 sine period
        movwf    dp_row, a
dtmf_B_loop
        rcal l    f_1633Hz
        rcal l    f_770Hz
        rcal l    add_freq
        movl w    0
        cpfseq    dtmfdrf, a
        bra      dtmf_B_loop
        bra      dtmf_end
;-----
; DTMF "C"      (1633Hz+852Hz)
;-----
dtmf_C
        movl w    49                ; 49 data points with 4 sine periods
        movwf    dp_col, a
        movl w    47                ; 47 data points with 2 sine periods
        movwf    dp_row, a
dtmf_C_loop
        rcal l    f_1633Hz
        rcal l    f_852Hz
        rcal l    add_freq
        movl w    0
        cpfseq    dtmfdrf, a
        bra      dtmf_C_loop
        bra      dtmf_end
;-----
; DTMF "D"      (1633Hz+941Hz)
;-----
dtmf_D
        movl w    49                ; 49 data points with 4 sine periods
        movwf    dp_col, a
        movl w    85                ; 85 data points with 4 sine periods
        movwf    dp_row, a
dtmf_D_loop
        rcal l    f_1633Hz
        rcal l    f_941Hz
        rcal l    add_freq
        movl w    0
        cpfseq    dtmfdrf, a
        bra      dtmf_D_loop
        bra      dtmf_end
;-----
dtmf_end
        bcf      intcon, tmr0if, a    ; clear tmr0 interrupt flag
        bcf      t0con, tmr0on, a    ; turn off timer0
        bcf      ptcon1, pten, a     ; pwm time base off
        clrf     ptmrh, a
        clrf     ptmrl, a            ; clear time base register

```



```

        movf    dtmfdrfb, w, a    ; \ restore dtmf tone duration
        movwf   dtmfdrf, a        ; / multiplication factor register
        return

;-----
; Add frequencies and transmit
;-----
add_freq
        movf    sin_r, w, a
        addwf   sin_c, f, a        ; add column and row frequencies to create dtmf waveform, result in sin_c
        btfss   intcon, tmr0if, a  ; tmr0 overflow?(i.e.: 50us sampling interval elapsed?)
        bra     $-1*2              ; no
        rrcf    sin_c, w, a        ; yes, divide by 2 to maintain 8 bits addition result
        mullw   4                  ; \
        movff   prodl, pdc0l       ; | -> update pwm duty cycle, (x4 needed by duty cycle register
        movff   prodh, pdc0h       ; | as stated at DS39758C-page 125)
        movlw   0xFE               ; \
        movwf   tmr0h, a           ; | -> update timer0 for next 50us
        movlw   0x89               ; | sampling period counting (50us-9Tcy 4 pwm update - 14tcy 4 tmr0 update)
        movwf   tmr0l, a           ; /
        bcf     intcon, tmr0if, a  ; clear tmr0 interrupt flag
        btfss   ptcon1, pten, a    ; time base on?
        bsf     ptcon1, pten, a    ; no, turn time base on
        decfsz  dtmfdr, f, a       ; decrement dtmf duration register
        return
        movf    dtmfdrb, w, a      ; \ restore dtmf
        movwf   dtmfdr, a          ; / duration register
        decf    dtmfdrf, f, a      ; decrement dtmf duration multiplication factor register
        return

;-----
; Frequency encoding subroutines
;-----
f_1633Hz
        movlw   upper(table_1633Hz); \
        movwf   tblptru, a         ; |
        movlw   high(table_1633Hz); | -> start address
        movwf   tblptrh, a         ; | of the table
        movlw   low(table_1633Hz); |
        movwf   tblptrl, a         ; /
        movf    dp_col, w, a       ; w=column data pointer
        sublw   49                 ; w-literal, result in w
        btfsc   status, z, a       ; table at start address?
        bra     read_col_f         ; yes
        movf    dp_col, w, a       ; \
        rlncf   wreg, f, a         ; |
        addwf   tblptrl, f, a       ; | -> table not at start address,
        clrf    wreg, a            ; | adjust table pointer
        addwfc  tblptrh, f, a       ; |
        addwfc  tblptru, f, a       ; /
        bra     read_col_f
;-----

```

f_1477Hz

```

movl w    upper(table_1477Hz); \
movwf     tbl_ptru, a          |
movl w    high(table_1477Hz);  | ->start address
movwf     tbl_ptrh, a          |   of the table
movl w    low(table_1477Hz);   |
movwf     tbl_ptrl, a          | /
movf      dp_col, w, a         ; w=column data pointer
subl w    27, w                ; w-literal, result in w
btfsc     status, z, a         ; table at start address?
bra       read_col_f          ; yes
movf      dp_col, w, a         \
rlncf     wreg, f, a           |
addwf     tbl_ptrl, f, a       | ->table not at start address,
clr f     wreg, a              |   adjust table pointer
addwfc    tbl_ptrh, f, a       |
addwfc    tbl_ptru, f, a       | /
bra       read_col_f

```

f_1336Hz

```

movl w    upper(table_1336Hz); \
movwf     tbl_ptru, a          |
movl w    high(table_1336Hz);  | ->start address
movwf     tbl_ptrh, a          |   of the table
movl w    low(table_1336Hz);   |
movwf     tbl_ptrl, a          | /
movf      dp_col, w, a         ; w=column data pointer
subl w    15, w                ; w-literal, result in w
btfsc     status, z, a         ; table at start address?
bra       read_col_f          ; yes
movf      dp_col, w, a         \
rlncf     wreg, f, a           |
addwf     tbl_ptrl, f, a       | ->table not at start address,
clr f     wreg, a              |   adjust table pointer
addwfc    tbl_ptrh, f, a       |
addwfc    tbl_ptru, f, a       | /
bra       read_col_f

```

f_1209Hz

```

movl w    upper(table_1209Hz); \
movwf     tbl_ptru, a          |
movl w    high(table_1209Hz);  | ->start address
movwf     tbl_ptrh, a          |   of the table
movl w    low(table_1209Hz);   |
movwf     tbl_ptrl, a          | /
movf      dp_col, w, a         ; w=column data pointer
subl w    33, w                ; w-literal, result in w
btfsc     status, z, a         ; table at start address?
bra       read_col_f          ; yes
movf      dp_col, w, a         \

```

```

    rlnsf    wreg, f, a      ;
    addwf    tblptrl, f, a   ; ->table not at start address,
    clrf     wreg, a         ;     adjust table pointer
    addwfc    tblptrh, f, a   ;
    addwfc    tblptru, f, a   ; /
    bra      read_col_f      ;
;-----
f_697Hz
    movl w    upper(table_697Hz); \
    movwf     tblptru, a       ;
    movl w    high(table_697Hz); ->start address
    movwf     tblptrh, a       ;     of the table
    movl w    low(table_697Hz); \
    movwf     tblptrl, a       ;
    movf      dp_row, w, a      ; w=row data pointer
    subl w    86               ; w-literal, result in w
    btfsc     status, z, a     ; table at start address?
    bra       read_row_f       ; yes
    movf      dp_row, w, a      ; \
    rlnsf     wreg, f, a       ;
    addwf     tblptrl, f, a     ; ->table not at start address,
    clrf      wreg, a          ;     adjust table pointer
    addwfc    tblptrh, f, a     ;
    addwfc    tblptru, f, a     ; /
    bra       read_row_f       ;
;-----
f_770Hz
    movl w    upper(table_770Hz); \
    movwf     tblptru, a       ;
    movl w    high(table_770Hz); ->start address
    movwf     tblptrh, a       ;     of the table
    movl w    low(table_770Hz); \
    movwf     tblptrl, a       ;
    movf      dp_row, w, a      ; w=row data pointer
    subl w    26               ; w-literal, result in w
    btfsc     status, z, a     ; table at start address?
    bra       read_row_f       ; yes
    movf      dp_row, w, a      ; \
    rlnsf     wreg, f, a       ;
    addwf     tblptrl, f, a     ; ->table not at start address,
    clrf      wreg, a          ;     adjust table pointer
    addwfc    tblptrh, f, a     ;
    addwfc    tblptru, f, a     ; /
    bra       read_row_f       ;
;-----
f_852Hz
    movl w    upper(table_852Hz); \
    movwf     tblptru, a       ;
    movl w    high(table_852Hz); ->start address
    movwf     tblptrh, a       ;     of the table

```

```

    movl w    low(table_852Hz) ; |
    movwf    tblptrl, a        ; /
    movf     dp_row, w, a      ; w=row data pointer
    subl w    47               ; w-literal, result in w
    btfsc    status, z, a      ; table at start address?
    bra      read_row_f        ; yes
    movf     dp_row, w, a      ; \
    rlnsf    wreg, f, a        ; |
    addwf    tblptrl, f, a      ; | ->table not at start address,
    clrf     wreg, a           ; |   adjust table pointer
    addwfc   tblptrh, f, a      ; /
    addwfc   tblptru, f, a
    bra      read_row_f

;-----
f_941Hz
    movl w    upper(table_941Hz) ; \
    movwf    tblptru, a          ; |
    movl w    high(table_941Hz)  ; | ->start address
    movwf    tblptrh, a          ; |   of the table
    movl w    low(table_941Hz)   ; /
    movwf    tblptrl, a
    movf     dp_row, w, a        ; w=row data pointer
    subl w    85               ; w-literal, result in w
    btfsc    status, z, a        ; table at start address?
    bra      read_row_f          ; yes
    movf     dp_row, w, a        ; \
    rlnsf    wreg, f, a          ; |
    addwf    tblptrl, f, a        ; | ->table not at start address,
    clrf     wreg, a             ; |   adjust table pointer
    addwfc   tblptrh, f, a        ; /
    addwfc   tblptru, f, a
    bra      read_row_f

;-----
read_row_f
    tblrd*+   ; read into tblat and increment
    movff    tblat, dp_row        ; get row data point position
    tblrd*    ; read into tblat
    movff    tblat, sin_r         ; get sin row data
    return

;-----
read_col_f
    tblrd*+   ; read into tblat and increment
    movff    tblat, dp_col        ; get column data point position
    tblrd*    ; read into tblat
    movff    tblat, sin_c         ; get sin column data
    return

;-----
*** DTMF DURATIONS ***
;-----
dur_default ; 100ms

```

```

    movl w    tone_dt
    movwf    dtmfdr, a      ; initialise dtmf duration register
    movwf    dtmfdrb, a    ; and its respective buffer register
    movl w    t_dt_f
    movwf    dtmfdrf, a    ; initialise dtmf duration multiplication factor register
    movwf    dtmfdrfb, a   ; and its respective buffer register
    return

; -----
dur_50ms
    movl w    tone_dt
    movwf    dtmfdr, a      ; initialise dtmf duration register
    movwf    dtmfdrb, a    ; and its respective buffer register
    movl w    4
    movwf    dtmfdrf, a    ; initialise dtmf duration multiplication factor register
    movwf    dtmfdrfb, a   ; and its respective buffer register
    return

; -----
dur_25ms
    movl w    tone_dt
    movwf    dtmfdr, a      ; initialise dtmf duration register
    movwf    dtmfdrb, a    ; and its respective buffer register
    movl w    2
    movwf    dtmfdrf, a    ; initialise dtmf duration multiplication factor register
    movwf    dtmfdrfb, a   ; and its respective buffer register
    return

; -----
dur_150ms
    movl w    tone_dt
    movwf    dtmfdr, a      ; initialise dtmf duration register
    movwf    dtmfdrb, a    ; and its respective buffer register
    movl w    12
    movwf    dtmfdrf, a    ; initialise dtmf duration multiplication factor register
    movwf    dtmfdrfb, a   ; and its respective buffer register
    return

; -----
dur_250ms
    movl w    tone_dt
    movwf    dtmfdr, a      ; initialise dtmf duration register
    movwf    dtmfdrb, a    ; and its respective buffer register
    movl w    20
    movwf    dtmfdrf, a    ; initialise dtmf duration multiplication factor register
    movwf    dtmfdrfb, a   ; and its respective buffer register
    return

; -----
dur_300ms
    movl w    tone_dt
    movwf    dtmfdr, a      ; initialise dtmf duration register
    movwf    dtmfdrb, a    ; and its respective buffer register
    movl w    24
    movwf    dtmfdrf, a    ; initialise dtmf duration multiplication factor register

```

```

        movwf    dtmfdrfb, a        ; and its respective buffer register
        return
; -----
dur_500ms
        movl w    tone_dt
        movwf     dtmfdr, a        ; initialise dtmf duration register
        movwf     dtmfdrb, a      ; and its respective buffer register
        movl w    40
        movwf     dtmfdrf, a      ; initialise dtmf duration multiplication factor register
        movwf     dtmfdrfb, a     ; and its respective buffer register
        return
; -----
dur_1s
        movl w    tone_dt
        movwf     dtmfdr, a        ; initialise dtmf duration register
        movwf     dtmfdrb, a      ; and its respective buffer register
        movl w    80
        movwf     dtmfdrf, a      ; initialise dtmf duration multiplication factor register
        movwf     dtmfdrfb, a     ; and its respective buffer register
        return
; -----
; *** DTMF SEQUENCES ***
; -----
snd_off                                ; off sound
        movl w    "1"
        movwf     dtmfkey, a
        call      dur_50ms
        call      send_DTMF
        rcall     del_50ms
        movl w    "2"
        movwf     dtmfkey, a
        call      dur_50ms
        call      send_DTMF
        rcall     del_50ms
        movl w    "3"
        movwf     dtmfkey, a
        call      dur_50ms
        call      send_DTMF
        rcall     del_50ms
        movl w    "A"
        movwf     dtmfkey, a
        call      dur_50ms
        call      send_DTMF
        rcall     del_50ms
        return
; -----
cl_enc        movl w    "#"        ; try close encounters in place of snd_off
                                ; to hear a poor imitation when rotary switch 1=off

```

```

    movwf    dtmfkey, a
    call    dur_default
    call    send_DTMF
    rcall   del_20ms
    movlw    "D"
    movwf    dtmfkey, a
    call    dur_150ms
    call    send_DTMF
    rcall   del_20ms
    movlw    "2"
    movwf    dtmfkey, a
    call    dur_default
    call    send_DTMF
    rcall   del_50ms
    movlw    "1"
    movwf    dtmfkey, a
    call    dur_150ms
    call    send_DTMF
    rcall   del_20ms
    movlw    "0"
    movwf    dtmfkey, a
    call    dur_150ms
    call    send_DTMF
    rcall   del_10ms
    return

; -----
snd_x
    movlw    "1"
    movwf    dtmfkey, a
    call    dur_50ms
    call    send_DTMF
    return

; -----
snd_y
    movlw    "2"
    movwf    dtmfkey, a
    call    dur_50ms
    call    send_DTMF
    return

; -----
snd_z
    movlw    "3"
    movwf    dtmfkey, a
    call    dur_50ms
    call    send_DTMF
    return

; -----
snd_a
    movlw    "A"
    movwf    dtmfkey, a

```

```

        call    dur_50ms
        call    send_DTMF
        return
; -----
snd_b   movl    w    "4"
        movwf   dtmfkey, a
        call    dur_50ms
        call    send_DTMF
        return
; -----
snd_c   movl    w    "5"
        movwf   dtmfkey, a
        call    dur_50ms
        call    send_DTMF
        return
; -----
snd_x1  movl    w    "6"
        movwf   dtmfkey, a
        call    dur_50ms
        call    send_DTMF
        return
; -----
snd_x10 movl    w    "B"
        movwf   dtmfkey, a
        call    dur_50ms
        call    send_DTMF
        return
; -----
snd_x100 movl    w    "7"
        movwf   dtmfkey, a
        call    dur_50ms
        call    send_DTMF
        return
; -----
; -----
; *** DTMF TABLES (WORD= 8-bit MSW=sin, 8-bit LSW=index) ***
; -----
table_1633Hz; 49 data points with 4 sine periods
        dw      0xBD01
        dw      0xEC02
        dw      0xFE03
        dw      0xF004
        dw      0xC405
        dw      0x8706

```


dw	0x4807
dw	0x1708
dw	0x0109
dw	0x0B0A
dw	0x330B
dw	0x6F0C
dw	0xAF0D
dw	0xE20E
dw	0xFC0F
dw	0xF610
dw	0xD111
dw	0x9712
dw	0x5713
dw	0x2114
dw	0x0315
dw	0x0516
dw	0x2717
dw	0x5F18
dw	0x9F19
dw	0xD71A
dw	0xF91B
dw	0xFB1C
dw	0xDD1D
dw	0xA71E
dw	0x671F
dw	0x2D20
dw	0x0821
dw	0x0222
dw	0x1C23
dw	0x4F24
dw	0x8F25
dw	0xCB26
dw	0xF327
dw	0xFD28
dw	0xE729
dw	0xB62A
dw	0x772B
dw	0x3A2C
dw	0x0E2D
dw	0x002E
dw	0x122F
dw	0x4130
dw	0x7F31

table_1477Hz; 27 data points with 2 sine periods

dw	0xB801
dw	0xE502
dw	0xFC03
dw	0xF904
dw	0xDB05

```
dw 0xAA06
dw 0x7007
dw 0x3908
dw 0x1109
dw 0x000A
dw 0x0A0B
dw 0x2D0C
dw 0x620D
dw 0x9C0E
dw 0xD10F
dw 0xF410
dw 0xFE11
dw 0xED12
dw 0xC513
dw 0x8E14
dw 0x5415
dw 0x2316
dw 0x0517
dw 0x0218
dw 0x1919
dw 0x461A
dw 0x7F1B
```

table_1336Hz; 15 data points with 1 sine period

```
dw 0xB301
dw 0xDD02
dw 0xF803
dw 0xFD04
dw 0xED05
dw 0xCA06
dw 0x9907
dw 0x6508
dw 0x3409
dw 0x110A
dw 0x010B
dw 0x060C
dw 0x210D
dw 0x4B0E
dw 0x7F0F
```

table_1209Hz; 33 data points with 2 sine periods

```
dw 0xAE01
dw 0xD702
dw 0xF303
dw 0xFE04
dw 0xF705
dw 0xDF06
dw 0xB907
dw 0x8B08
dw 0x5B09
```

dw	0x300A
dw	0x110B
dw	0x010C
dw	0x040D
dw	0x180E
dw	0x3A0F
dw	0x6710
dw	0x9711
dw	0xC412
dw	0xE613
dw	0xFA14
dw	0xFD15
dw	0xED16
dw	0xCE17
dw	0xA318
dw	0x7319
dw	0x451A
dw	0x1F1B
dw	0x071C
dw	0x001D
dw	0x0B1E
dw	0x271F
dw	0x5020
dw	0x7F21

table_697Hz; 86 data points with 3 sine periods

dw	0x9B01
dw	0xB502
dw	0xCD03
dw	0xE104
dw	0xF005
dw	0xFA06
dw	0xFE07
dw	0xFC08
dw	0xF409
dw	0xE60A
dw	0xD40B
dw	0xBD0C
dw	0xA40D
dw	0x880E
dw	0x6D0F
dw	0x5210
dw	0x3911
dw	0x2412
dw	0x1313
dw	0x0714
dw	0x0115
dw	0x0116
dw	0x0717
dw	0x1318

```
dw 0x2419
dw 0x391A
dw 0x521B
dw 0x6D1C
dw 0x881D
dw 0xA41E
dw 0xBD1F
dw 0xD420
dw 0xE621
dw 0xF422
dw 0xFC23
dw 0xFE24
dw 0xFA25
dw 0xF026
dw 0xE127
dw 0xCD28
dw 0xB529
dw 0x9B2A
dw 0x7F2B
dw 0x632C
dw 0x492D
dw 0x312E
dw 0x1D2F
dw 0x0E30
dw 0x0431
dw 0x0032
dw 0x0233
dw 0x0A34
dw 0x1835
dw 0x2A36
dw 0x4137
dw 0x5A38
dw 0x7639
dw 0x913A
dw 0xAC3B
dw 0xC53C
dw 0xDA3D
dw 0xEB3E
dw 0xF73F
dw 0xFD40
dw 0xFD41
dw 0xF742
dw 0xEB43
dw 0xDA44
dw 0xC545
dw 0xAC46
dw 0x9147
dw 0x7648
dw 0x5A49
dw 0x414A
```

dw	0x2A4B
dw	0x184C
dw	0x0A4D
dw	0x024E
dw	0x004F
dw	0x0450
dw	0x0E51
dw	0x1D52
dw	0x3153
dw	0x4954
dw	0x6355
dw	0x7F56

table_770Hz; 26 data points with 1 sine period

dw	0x9D01
dw	0xBA02
dw	0xD303
dw	0xE804
dw	0xF605
dw	0xFD06
dw	0xFD07
dw	0xF608
dw	0xE809
dw	0xD30A
dw	0xBA0B
dw	0x9D0C
dw	0x7F0D
dw	0x610E
dw	0x440F
dw	0x2B10
dw	0x1611
dw	0x0812
dw	0x0113
dw	0x0114
dw	0x0815
dw	0x1616
dw	0x2B17
dw	0x4418
dw	0x6119
dw	0x7F1A

table_852Hz; 47 data points with 2 sine periods

dw	0xA101
dw	0xC002
dw	0xDA03
dw	0xEE04
dw	0xFB05
dw	0xFE06
dw	0xF807

dw	0xEA08
dw	0xD409
dw	0xB80A
dw	0x980B
dw	0x770C
dw	0x550D
dw	0x370E
dw	0x1E0F
dw	0x0C10
dw	0x0211
dw	0x0112
dw	0x0813
dw	0x1914
dw	0x3015
dw	0x4D16
dw	0x6E17
dw	0x9018
dw	0xB119
dw	0xCE1A
dw	0xE51B
dw	0xF61C
dw	0xFD1D
dw	0xFC1E
dw	0xF21F
dw	0xE020
dw	0xC721
dw	0xA922
dw	0x8723
dw	0x6624
dw	0x4625
dw	0x2A26
dw	0x1427
dw	0x0628
dw	0x0029
dw	0x032A
dw	0x102B
dw	0x242C
dw	0x3E2D
dw	0x5D2E
dw	0x7F2F

table_941Hz; 85 data points with 4 sine periods

dw	0xA401
dw	0xC602
dw	0xE103
dw	0xF504
dw	0xFD05
dw	0xFB06
dw	0xEF07
dw	0xD808

```
dw 0xBA09
dw 0x960A
dw 0x710B
dw 0x4D0C
dw 0x2D0D
dw 0x140E
dw 0x050F
dw 0x0010
dw 0x0611
dw 0x1712
dw 0x3113
dw 0x5114
dw 0x7615
dw 0x9B16
dw 0xBE17
dw 0xDB18
dw 0xF119
dw 0xFC1A
dw 0xFD1B
dw 0xF31C
dw 0xDE1D
dw 0xC21E
dw 0x9F1F
dw 0x7A20
dw 0x5621
dw 0x3422
dw 0x1A23
dw 0x0824
dw 0x0025
dw 0x0426
dw 0x1227
dw 0x2928
dw 0x4929
dw 0x6C2A
dw 0x922B
dw 0xB52C
dw 0xD52D
dw 0xEC2E
dw 0xFA2F
dw 0xFE30
dw 0xF631
dw 0xE432
dw 0xCA33
dw 0xA834
dw 0x8435
dw 0x5F36
dw 0x3C37
dw 0x2038
dw 0x0B39
dw 0x013A
```

```
dw 0x023B
dw 0x0D3C
dw 0x233D
dw 0x403E
dw 0x633F
dw 0x8840
dw 0xAD41
dw 0xCD42
dw 0xE743
dw 0xF844
dw 0xFE45
dw 0xF946
dw 0xEA47
dw 0xD148
dw 0xB149
dw 0x8D4A
dw 0x684B
dw 0x444C
dw 0x264D
dw 0x0F4E
dw 0x034F
dw 0x0150
dw 0x0951
dw 0x1D52
dw 0x3853
dw 0x5A54
dw 0x7F55
```

```
;-----
;-----
;-----
end
```