AppKit:
Using the CM8880 DTMF Transceiver

This AppKit shows how to use the California Microdevices CM8880 DTMF transceiver chip with PIC microcontrollers and Parallax BASIC Stamp® single-board computers.

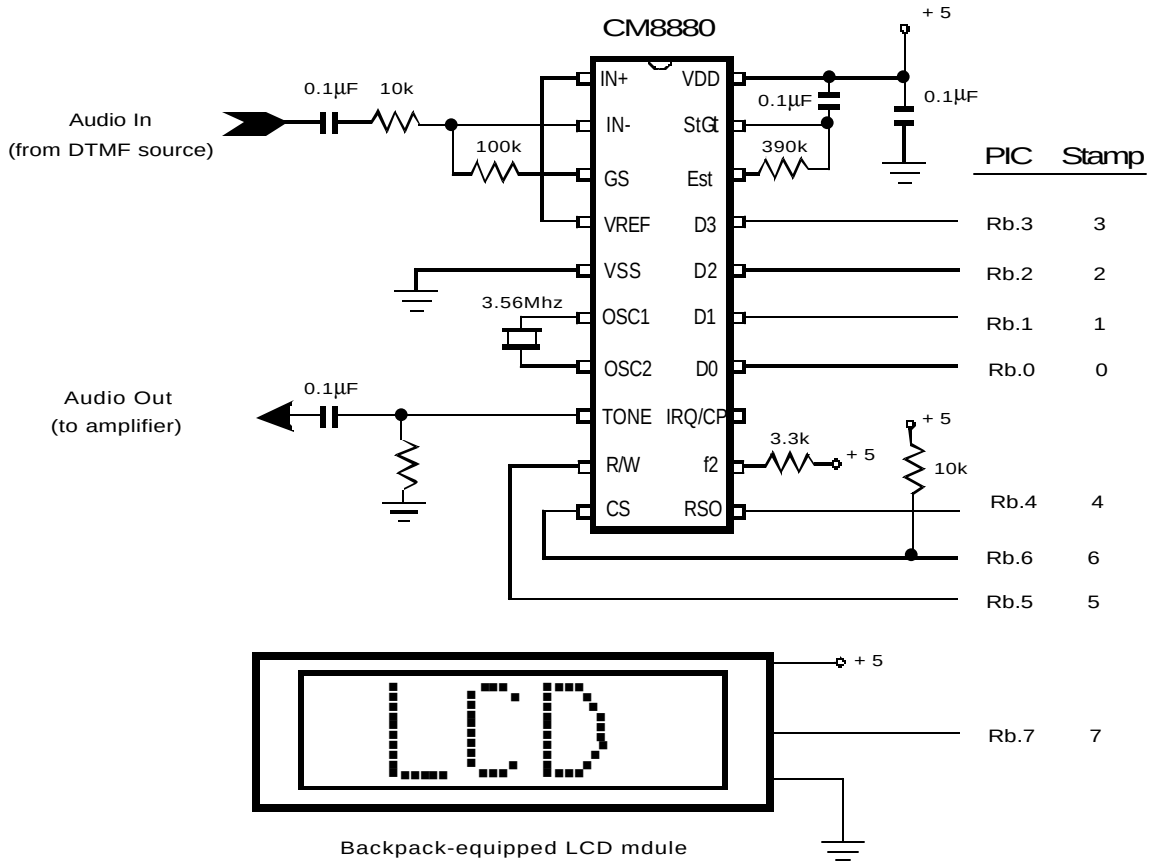
Description

The CM8880 is a dual-tone, multifrequency (DTMF) transceiver on a chip. It can be configured to send or receive "touch" tones used in many phone and radio communication systems.

Hardware interface

Communication with the CM8880 takes place over a 4-bit bus, consisting of D0 through D3, with three additional bits selecting modes of operation. Those bits are chip select (CS), read/write (RW) and register select (RS0).

The figure shows how to connect the CM8880 to the PIC or Stamp for the demo programs. Do not omit the bypass capacitor—not even if you feel that your power supply is solid and well-filtered. Noise from the digital parts of your circuit, particularly strobing of the CS line, can fall into the audio range and interfere with the CM8880's ability to hear and distinguish DTMF tones.



Software interface

The software needed to interface with the CM8880 is relatively simple, thanks to a four-bit data bus. A write cycle consists of the following steps (starting with the CM8880's CS pin high to deselect it):

(1) Put the data pins into output mode
(2) Write the data to the bus
(3) Set up RS0: 0 = write data; 1 = write instructions
(4) Clear the RW bit to request a write
(5) Clear CS to activate the CM8880
(6) Set CS to terminate the write operation and deactivate the CM8880

Reading the CM8880 is similar. Starting with CS high, the steps are:

(1) Put the data pins into input mode
(2) Set the RW bit to request a read
(3) Set up RS0: 0 = read data; 1 = read instructions
(4) Clear CS to activate the CM8880
(5) Read the data from the bus
(6) Set CS to terminate the read operation and deactivate the CM8880

The table below summarizes the interaction of the CM8880's control pins. To sum up the table, the 8880 is active only when CS is 0. The RW bit determines the data direction; 1 = read (data from 8880 to Stamp) and 0 = write (data from Stamp to 8880). The RS bit determines whether the transaction involves data (DTMF tones) or internal CM8880 functions (instructions or status); 1 = instructions/status and 0 = data.

CS	RW	RS0	Description
0	0	0	Active: write data (i.e., send DTMF)
0	0	1	Active: write instructions to 8880
0	1	0	Active: read data (i.e., receive DTMF)
0	1	1	Active: read status from 8880
1	0	0	Inactive
1	0	1	Inactive
1	1	0	Inactive
1	1	1	Inactive

The CM8880 has quite a few operational modes and options, and not enough control pins to select them. Instead, a program must store these settings in a pair of internal registers known as control register A and control register B (CRA and CRB). The tables that follow summarize the operation of these registers. Note that some of the control settings interact, or have different meanings depending on other settings.

Functions of Control Register A

Bit	Name	Function
0	Tone Out	0 = tone generator disabled 1 = tone generator enabled
1	Mode Control	0 = Send and receive DTMF 1 = Send DTMF, receive call-progress tones (DTMF bursts lengthened to 104 ms)
2	Interrupt Enable	0 = Make controller check for DTMF rec'd 1 = Interrupt controller via pin 13 when DTMF rec'd
3	Register Select	0 = Next instruction write goes to CRA. 1 = Next instruction write goes to CRB.

Functions of Control Register B

Bit	Name	Function
0	Burst	0 = Output DTMF bursts of 52 or 104 ms 1 = Output DTMF as long as enabled
1	Test	0 = Normal operating mode 1 = Present test timing bit on pin 13
2	Single/Dual	0 = Output dual (real DTMF) tones. 1 = Output separate row or column tones
3	Column/Row	0 = If above = 1, select row tone. 1 = If above = 1, select column tone.

Tips for using the CM8880

- A phone can serve as a suitable source of DTMF tones. Connect a 9V battery across the wires that normally connect to the phone line and press the keys. You will hear DTMF tones in the handset. To connect this setup to the circuit shown in the schematic, ground the battery's negative terminal and connect the positive terminal to the point marked "Audio In."

- To hear DTMF tones generated by the circuit, connect an audio amplifier to the point marked "Audio Out." Radio Shack's 277-1008C is suitable. Keep the volume low! You may dial a phone by holding the amplified speaker close to the telephone mouthpiece.

- Connecting hardware directly to the phone lines requires FCC approval. A shortcut is to use interface hardware that is already approved by the FCC. One source for this hardware, known as a Data Access Arrangement (DAA) is Cermetek, Sunnyvale, CA, 408-752-5000.

PIC Program I Listing: DTMF Dialing

```
; Program DIAL.SRC
; PIC controls a CM8880 to generate DTMF tones.
; Note that there are no delays between CS clear and set actions when
; the program strobes data into or out of the CM8880. The reason is that
; all of the 8880's timing parameters are in the 10s of nanoseconds.
; Also, this program polls the CM8880's transmit-ready flag between
; DTMF bursts, so it sends consistent 100-ms tones regardless of PIC
; clock speed.

CS          =       rb.6            ; Chip select, active low.
RS          =       rb.4            ; Register select bit. High = registers selected.
RW          =       rb.5            ; Read/Write bit. Low = write.

dials       =       1011b           ; Control register setting for normal dialing (100ms).
talk        =       0               ; TRIS value for talking to CM8880.
listen      =       15              ; TRIS value for listening to CM8880.

            org     8               ; Variables above special-function registers.
temp        ds      1               ; Temporary variable for delays, etc.
digit       ds      1               ; Index for dialing digits from table.
cntrl       ds      1               ; Variable copy of data for control register.
stats       ds      1               ; Variable copy of status register

xm_rdy      =       stats.1         ; Flag set when 8880 is ready to send DTMF.

            device pic16c54,xt_osc,wdt_off,protect_off
            reset start
            org     0               ; Program origin.

; ============================
;           SUBROUTINES
; ============================

; A table of numbers to dial.
dial_these
            jmp     pc+w
            retw    4,5,9,0,6,2,3

; Generates a brief delay to allow CM8880 to get settled after power-on.
delay       clr     temp
:loop       djnz    temp,:loop      ; Loop 256 times.
            ret                     ; Return.
```

PIC Program I Listing: DTMF Dialing (cont.)

```
; Accepts data to be written to the pair of four-bit control registers in
; the variable "cntrl." Data for register 1 must be in the lower nybble;
; register 2 in the upper nybble.
set_controls
        mov     w,cntrl         ; Get the control value.
        AND     w,#00001111b    ; Strip upper four bits
        OR      w,#00011000b    ; Set register-select, clear CS, set RS2.
        mov     rb,w            ; Write to output, (CS low).
        setb    CS              ; Deselect to complete the 1st write.
        mov     w,<>cntrl       ; Get nybble-swapped control value.
        AND     w,#00001111b    ; Strip upper four bits
        OR      w,#00010000b    ; Set register-select, clear CS.
        mov     rb,w            ; Write to output, (CS low).
        setb    CS              ; Deselect to complete the 2nd write.
        ret

; Gets the contents of the 8880's status register and stores 'em
; to the variable "stats." The port TRIS (data direction) must
; be set to input before calling this routine.
get_status
        setb    RW              ; Set to read.
        setb    RS              ; Select the status register.
        clrb    CS              ; Activate chip.
        mov     stats,#15       ; Bit mask for 4-bit data.
        AND     stats,rb        ; Get lower 4 lsbs of rb (data).
        setb    CS              ; Deselect chip.
        ret

; Dials a DTMF digit represented by the lower four bits of the
; w register. Converts a value of "0" to "10" in accordance with
; the DTMF encoding scheme.
dial_digit
        AND     w,#15           ; Strip MSB, update z flag in the process.
        snz                     ; If w=0, change it to 10 for proper..
        mov     w,#10           ; ..DTMF encoding.
        mov     rb,w            ; Write to output.
        setb    CS              ; Terminate the write
        ret
```

PIC Program I Listing: DTMF Dialing (cont.)

```
; ============================
;                MAIN PROGRAM
; ============================

start
                mov     rb,#255         ; All bits high to deselect CM8880.
                mov     !rb,#talk       ; Set port to output.
                call    delay           ; Wait for things to settle down.
                mov     cntrl,#dials    ; Set control register for dialing.
                call    set_controls    ; Write settings to control registers.
                clr     digit           ; Clear the digit counter to 0.
:loop
                mov     w,digit         ; Now, put the digit number into w..
                call    dial_these      ; ..and get the next digit from the table.
                call    dial_digit      ; Dial the digit stored in w.
:hold           mov     !rb,#listen     ; Set port to input.
                call    get_status      ; Get the status register in "stats."
                jnb     xm_rdy,:hold    ; If transmitter's not ready, wait here.
                mov     !rb,#talk       ; Restore port to output.
                inc     digit           ; Point to the next digit in the table.
                cjb     digit,#7,:loop  ; Stop when all digits have been dialed.
                jmp     $               ; Freeze the program here--reset to rerun.
```

PIC Program II Listing: DTMF Decoding

```
; Program DTMF_RCV.SRC
; PIC uses the CM8880 to receive and decode DTMF tones for display
; on an LCD Serial Backpack-equipped LCD module (9600 bps) or terminal.

CS          =       rb.6            ; Chip select, active low.
RS          =       rb.4            ; Register select bit. High = registers selected.
RW          =       rb.5            ; Read/Write bit. Low = write.

decode      =       1000b           ; Control register settings for DTMF receive.
talk        =       0               ; TRIS value for talking to CM8880.
listen      =       15              ; TRIS value for listening to CM8880.
baud        =       18              ; Delay constant for 9600 baud at 4MHz.
out_pin     =       rb.7            ; Serial output pin for Backpack.
LCDw        =       16              ; Width of the LCD screen
LCDcls      =       1               ; LCD clear-screen command.
I           =       254             ; LCD instruction toggle.

            org     8               ; Variables above special-function registers.
temp        ds      1               ; Temporary variable for delays, etc.
temp2       ds      1               ; Temporary variable for delays, etc.
cntrl       ds      1               ; Variable copy of data for control register.
statdat     ds      1               ; Variable copy of status register or data.
LCDcol      ds      1               ; Current column of LCD screen for wrap.
xbyte       ds      1               ; Byte for Serout to transmit.
return      ds      1               ; Return address for clearAll routine.

dt_flag     =       statdat.2       ; Flag set when 8880 has decoded a tone.

            device pic16c54,xt_osc,wdt_off,protect_off
            reset start
            org     0               ; Program origin.

; ============================
;           SUBROUTINES
; ============================

; Generates a brief delay to allow CM8880 to get settled after power-on.
delay       clr     temp
:loop       djnz    temp,:loop      ; Loop 256 times.
            ret                     ; Return.

; Table to correlate DTMF values 0-15 to printable symbols
symbols jmp pc+w
            retw    'D','1','2','3','4','5','6','7','8','9','0'
            retw    '*','#','A','B','C'
```

PIC Program II Listing: DTMF Decoding (cont.)

```
; Accepts data to be written to the pair of four-bit control registers in
; the variable "cntrl." Data for register 1 must be in the lower nybble;
; register 2 in the upper nybble.
set_controls
                mov     w,cntrl             ; Get the control value.
                AND     w,#00001111b        ; Strip upper four bits
                OR      w,#00011000b        ; Set register-select, clear CS, set RS2.
                mov     rb,w                ; Write to output, (CS low).
                setb    CS                  ; Deselect to complete the 1st write.
                mov     w,<>cntrl           ; Get nybble-swapped control value.
                AND     w,#00001111b        ; Strip upper four bits
                OR      w,#00010000b        ; Set register-select, clear CS.
                mov     rb,w                ; Write to output, (CS low).
                setb    CS                  ; Deselect to complete the 2nd write.
                ret

; Piggybacks on "get_status" below to get data from the CM8880.
get_data
                clrb    RS                  ; Select data
                skip                        ; Skip first instruction (setb RS)
                                            ; of get_status below.

; Gets the contents of the 8880's status register and stores 'em
; to the variable "statdat." The port TRIS (data direction) must
; be set to input before calling this routine.
get_status
                setb    RS                  ; Select the status register.
:data           setb    RW                  ; Set to read.
                clrb    CS                  ; Activate chip.
                mov     statdat,#15         ; Bit mask for 4-bit data.
                AND     statdat,rb          ; Get lower 4 lsbs of rb (data).
                setb    CS                  ; Deselect chip.
                ret

; Sends a byte of data serially through out_pin at 9600 bps (assuming a
; 4-MHz clock). Drives the LCD Serial Backpack, or terminal software
; running on a PC.
Serout
:xmit           clc                         ; Set up start bit.
                call    emitBit             ; Send start bit.
                mov     temp2,#8            ; Send 8 data bits.
```

PIC Program II Listing: DTMF Decoding (cont.)

```
:bits       rr      xbyte       ; Rotate bit into carry.
            call    emitBit     ; Send the data bit.
            djnz    temp2,:bits ; All 8 bits sent? No: send more bits.
            rr      xbyte       ; Rotate data back into original position.
            stc                 ; Set up stop bit.
            call    emitBit     ; Send the stop bit.
            ret                 ; Yes: return.

; Send bit and generate time delays for Serout.
emitBit
            movb    out_pin,/c  ; Inverted output.
            mov     temp,#baud  ; Baud-rate delay.
:loop       jmp     $+1         ; Two-cycle nops for time delay.
            djnz    temp,:loop  ; Number of trips through loop
            nop                 ; Pad to exact timing.
            ret                 ; Set by baud rate in temp3.

; Resets the LCD position counter and clears the LCD screen.
; Before jumping to this subroutine, the program must store
; the return address in the variable "return." This procedure
; is necessary because Serout uses both levels of the built-in
; call stack--this allows us to simulate one more level. Be
; careful to ensure that the return address isn't greater than 255.
clearAll
            clr     LCDcol      ; Reset LCD column counter.
            mov     xbyte,#I    ; Toggle LCD to instruction mode.
            call    Serout      ; Send it serially.
            mov     xbyte,#LCDcls ; Clear the LCD screen.
            call    Serout      ; Send.
            mov     xbyte,#I    ; Toggle LCD back to data mode.
            call    Serout      ; Send.
            mov     w,return    ; Point to return address.
            jmp     w           ; Go there.

; ============================
;           MAIN PROGRAM
; ============================

start
            mov     rb,#127     ; Bits 0-6 high to deselect CM8880.
            mov     !rb,#talk   ; Set port to output.
            call    delay       ; Wait for things to settle down.
            mov     cntrl,#decode ; Set control register for decoding.
            call    set_controls ; Write settings to control registers.
            mov     return,#:loop ; Save return address.
            jmp     clearAll    ; Jump to clear routine.
```

PIC Program II Listing: DTMF Decoding (cont.)

```
:loop   mov     !rb,#listen                 ; Set port to input.
        call    delay
        call    get_status                  ; Check the status register.
        jnb     dt_flag,:loop               ; No tone detected, loop.
        inc     LCDcol                      ; Point to next LCD position.
        cjb     LCDcol,#LCDw,:nocls  ; If its > LCD width, clear..
        mov     return,#:nocls              ; Save return address.
        jmp     clearAll                    ; Jump to clear routine.
:nocls  call    get_data                    ; Get the data.
        mov     w,statdat
        call    symbols              ; Look up dtmf digit in table.
        mov     xbyte,w               ; Write the digit to the display.
        call    Serout
        goto    :loop                ; Do it again.
```

BASIC Stamp I (BS1-IC) and BASIC Stamp Ver. D Program 1 Listing: DTMF Dialing

```
' Program: DIAL.BAS (Sends a string of DTMF tones via the 8880)
' This program demonstrates how to use the CM8880 as a DTMF tone
' generator. All that's required is to initialize the 8880 properly,
' then write the number of the desired DTMF tone to the 8880's
' 4-bit bus.

' The symbols below are the pin numbers to which the 8880's
' control inputs are connected, and one variable used to read
' digits out of a lookup table.

SYMBOL      RS_p = 4    ' Register-select pin (0=data).
SYMBOL      RW_p = 5    ' Read/Write pin (0=write).
SYMBOL      CS_p = 6    ' Chip-select pin (0=active).
SYMBOL      digit = b2  ' Index of digits to dial.

' This code initializes the 8880 for dialing by writing to its
' internal control registers CRA and CRB. The write occurs when
' CS (pin 6) is taken low, then returned high. See the accompanying
' article for an explanation of the 8880's registers.

let pins = 255              ' All pins high to deselect 8880.
let dirs = 255              ' Set up to write to 8880 (all outputs).
let pins = %00011011        ' Set up CRA, next write to CRB.
high CS_p
let pins = %00010000        ' Clear register B; ready to send DTMF.
high CS_p

' This for/next loop dials the seven digits of my fax number. For
' simplicity, it writes the digit to be dialed directly to the output
' pins. Since valid digits are between 0 and 15, this also takes RS,
' RW, and CS low--perfect for writing data to the 8880. To complete
' the write, the CS line is returned high. The initialization above
' sets the 8880 for tone bursts of 200 ms duration, so we pause
' 250 ms between digits. Note: in the DTMF code as used by the phone
' system, zero is represented by ten (1010 binary) not 0. That's why
' the phone number 459-0623 is coded 4,5,9,10,6,2,3.

for digit = 0 to 6
  lookup digit,(4,5,9,10,6,2,3),pins        ' Get digit from table.
  high CS_p                                 ' Done with write.
  pause 250                                 ' Wait to dial next digit.
next digit

end
```

BASIC Stamp I (BS1-IC) and BASIC Stamp Ver. D Program 2 Listing: DTMFDecoding

```
' Program: DTMF_RCV.BAS (Receives and display DTMF tones using the 8880)
' This program demonstrates how to use the 8880 as a DTMF decoder. As
' each new DTMF digit is received, it is displayed on an LCD Serial
' Backpack screen. If no tones are received within a period of time
' set by sp_time, the program prints a dash (or other selected character)
' to the LCD to record the delay. When the display reaches the righthand
' edge of the screen, it clears the LCD and starts over at the left edge.

SYMBOL      RS_p = 4      ' Register-select pin (0=data).
SYMBOL      RW_p = 5      ' Read/Write pin (0=write).
SYMBOL      CS_p = 6      ' Chip-select pin (0=active).
SYMBOL      dtmf = b2     ' Received DTMF digit.
SYMBOL      dt_Flag = bit0 ' DTMF-received flag.
SYMBOL      home_Flag = bit1            ' Flag: 0 = cursor at left edge of LCD.
SYMBOL      polls = w2    ' Number of unsuccessful polls of DTMF.
SYMBOL      LCDw = 16     ' Width of LCD screen.
SYMBOL      LCDcol = b3   ' Current column of LCD screen for wrap.
SYMBOL      LCDcls = 1    ' LCD clear-screen command.
SYMBOL      I = 254                     ' LCD instruction toggle.
SYMBOL      sp_time = 1000              ' Print space this # of polls w/o DTMF.

' This code initializes the 8880 for receiving by writing to its
' internal control registers CRA and CRB. The write occurs when
' CS (pin 6) is taken low, then returned high.

let pins = %01111111        ' Pin 7 (LCD) low, pins 0 through 6 high.
let dirs = %11111111        ' Set up to write to 8880 (all outputs).
let pins = %00011000        ' Set up  register A, next write to register B.
high CS_p
let pins = %00010000        ' Clear register B; ready to send DTMF.
high CS_p
let dirs = %11110000        ' Now make set the 4-bit bus to input.
high RW_p                   ' And set RW to "read."
serout 7,n2400,(I,LCDcls,I)  ' Clear the LCD screen.

' In the loop below, the program checks the 8880's status register
' to determine whether a DTMF tone has been received (indicated by
' a '1' in bit 2). If no tone, the program loops back and checks
' again. If a tone is present, the program switches from status to
' data (RS low) and gets the value (0-15) of the tone. This
' automatically resets the 8880's status flag.
again:
  high RS_p                         ' Read status register.
  low CS_p                          ' Activate the 8880.
  let dt_flag = pin2                ' Store status bit 2 into flag.
  high CS_p                         ' End the read.
```

BASIC Stamp I (BS1-IC) and BASIC Stamp Ver. D Program 2 Listing: DTMFDecoding (cont.)

```
if dt_Flag = 1 then skip1                    ' If tone detected, continue.
let polls = polls+1                          ' Another poll without DTMF tone.
if polls < sp_time then again    ' If not time to print a space, poll again.
if LCDcol = LCDw then skip2 ' Don't erase the screen to print spaces.
let dtmf = 16                                ' Tell display routine to print a space.
gosub Display                                ' Print space to LCD.
skip2:
let polls = 0                                ' Clear the counter.
goto again                                   ' Poll some more.
skip1:                                       ' Tone detected:
let polls = 0                                ' Clear the poll counter.
low RS_p                                          ' Get the DTMF data.
low CS_p                                          ' Activate 8880.
let dtmf = pins & %00001111 ' Strip off upper 4 bits using AND.
high CS_p                                    ' Deactivate 8880.
gosub display                                ' Display the data.
goto again                                   ' Do it all again.

Display:
if LCDcol < LCDw then skip3 ' If not at end of LCD, don't clear screen.
serout 7,n2400,(I,LCDcls,I)   ' Clear the LCD screen.
let LCDcol = 0                               ' And reset the column counter.
skip3:                                            ' Look up the symbol for the digit.
if LCDcol=0 AND dtmf=16 then ret             ' No spaces at first column.
lookup dtmf,("D1234567890*#ABC-"),dtmf
serout 7,n2400,(dtmf)                        ' Write it to the Backpack display.
let LCDcol = LCDcol + 1                      ' Increment the column counter.
ret:
return
```

BASIC Stamp II (BS2-IC) Program 2 Listing: DTMF Dialing

```
' Program: DIAL.BS2 (BS2 sends DTMF tones via the 8880)
' This program demonstrates how to use the CM8880 as a DTMF tone
' generator. All that's required is to initialize the 8880 properly,
' then write the number of the desired DTMF tone to the 8880's
' 4-bit bus.

' The symbols below are the pin numbers to which the 8880's
' control inputs are connected, and one variable used to read
' digits out of a lookup table.

RS    con     4             ' Register-select pin (0=data).
RW    con     5             ' Read/Write pin (0=write).
CS    con     6             ' Chip-select pin (0=active).
digit var     nib           ' Index of digits to dial, 1-15.

' This code initializes the 8880 for dialing by writing to its
' internal control registers CRA and CRB. The write occurs when
' CS (pin 6) is taken low, then returned high. See the accompanying
' article for an explanation of the 8880's registers.

OUTL = 127                  ' Pins 0-6 high to deselect 8880.
DIRL = 127                  ' Set up to write to 8880 (pins 0-6 outputs).
OUTL = %00011011            ' Set up  register A, next write to register B.
high CS
OUTL = %00010000            ' Clear register B; ready to send DTMF.
high CS

' This for/next loop dials the seven digits of my fax number. For
' simplicity, it writes the digit to be dialed directly to the output
' pins. Since valid digits are between 0 and 15, this also takes RS,
' RW, and CS low--perfect for writing data to the 8880. To complete
' the write, the CS line is returned high. The initialization above
' sets the 8880 for tone bursts of 200 ms duration, so we pause
' 250 ms between digits. Note: in the DTMF code as used by the phone
' system, zero is represented by ten (1010 binary) not 0. That's why
' the phone number 459-0623 is coded 4,5,9,10,6,2,3.

for digit = 0 to 6
  lookup digit,[4,5,9,10,6,2,3],OUTL        ' Get digit from table.
  high CS                                   ' Done with write.
  pause 250                                 ' Wait to dial next digit.
next
end
```

BASIC Stamp II (BS2-IC) Program Listing: DTMF Decoding

```
' Program: DTMF_RCV.BS2 (Receives/displays DTMF using 8880 with BS2)
' This program demonstrates how to use the 8880 as a DTMF decoder. As
' each new DTMF digit is received, it is displayed on an LCD Serial
' Backpack screen. If no tones are received within a period of time
' set by sp_time, the program prints a space (or other selected character)
' to the LCD to record the delay. When the display reaches the righthand
' edge of the screen, it clears the LCD and starts over at the left edge.

RS      con     4                       ' Register-select pin (0=data).
RW      con     5                       ' Read/Write pin (0=write).
CS      con     6                       ' Chip-select pin (0=active).

dtmf    var     byte                    ' Received DTMF digit.
dt_Flag         var     bit                     ' DTMF-received flag.
dt_det          var     INL.bit2        ' DTMF detected status bit.
home_Flag       var     bit                     ' Flag: 0 = cursor at left edge of LCD.
polls   var     word                    ' Number of unsuccessful polls of DTMF.
LCDw            con     16                      ' Width of LCD screen.
LCDcol          var     byte                    ' Current column of LCD screen for wrap.
LCDcls          con     1                       ' LCD clear-screen command.
I       con     254                     ' LCD instruction toggle.
sp_time         con     1500                    ' Print space this # of polls w/o DTMF.
n24n    con     $418D                   ' Serout constant: 2400 baud inverted.

' This code initializes the 8880 for receiving by writing to its
' internal control registers CRA and CRB. The write occurs when
' CS (pin 6) is taken low, then returned high.

OUTL = %01111111                        ' Pin 7 (LCD) low, pins 0 through 6 high.
DIRL = %11111111                        ' Set up to write to 8880 (all outputs).
OUTL = %00011000                        ' Set up  register A, next write to register B.
high CS
OUTL = %00010000                        ' Clear register B; ready to send DTMF.
high CS
DIRL = %11110000                        ' Now set the 4-bit bus to input.
high RW                                 ' And set RW to "read."
serout 7,n24n,[I,LCDcls,I]      ' Clear the LCD screen.

' In the loop below, the program checks the 8880's status register
' to determine whether a DTMF tone has been received (indicated by
' a '1' in bit 2). If no tone, the program loops back and checks
' again. If a tone is present, the program switches from status to
' data (RS low) and gets the value (0-15) of the tone. This
' automatically resets the 8880's status flag.
again:
  high RS               ' Read status register.
  low CS                                ' Activate the 8880.
  dt_flag = dt_det      ' Store DTMF-detected bit into flag.
  high CS               ' End the read.
```

BASIC Stamp II (BS2-IC) Program Listing: DTMF Decoding (cont.)

```
if dt_Flag = 1 then skip1       ' If tone detected, continue.
polls = polls+1                 ' Another poll without DTMF tone.
if polls < sp_time then again   ' If not time to print a space, poll.
if LCDcol = LCDw then skip2 ' Don't erase the screen to print spaces.
dtmf = 16                                   ' Tell display routine to print a space.
gosub Display                               ' Print space to LCD.
skip2:
polls = 0                       ' Clear the counter.
goto again                      ' Poll some more.
skip1:                          ' Tone detected:
polls = 0                       ' Clear the poll counter.
low RS                                      ' Get the DTMF data.
low CS                                      ' Activate 8880.
dtmf = INL & %00001111      ' Strip off upper 4 bits using AND.
high CS                                     ' Deactivate 8880.
gosub display                   ' Display the data.
goto again                      ' Do it all again.

Display:
if LCDcol < LCDw then skip3 ' If not at end of LCD, don't clear screen.
serout 7,N24N,[I,LCDcls,I]    ' Clear the LCD screen.
LCDcol = 0                                  ' And reset the column counter.
skip3:                                          ' Look up the symbol for the digit.
if LCDcol=0 AND dtmf=16 then ret            ' No spaces at first column.
lookup dtmf,["D1234567890*#ABC-"] ,dtmf
serout 7,N24N,[dtmf]                        ' Write it to the Backpack display.
LCDcol = LCDcol + 1                         ' Increment the column counter.
ret:
return
```

ADVANCED PRODUCT INFORMATION

CALIFORNIA MICRO DEVICES **CM8880/CM8880-1/CM8880-2**

CMOS Integrated DTMF Transceiver

Features

- Advanced CMOS technology for low power consumption and increased noise immunity
- Complete DTMF Transmitter/Receiver
- Standard 6500/6800 microprocessor port
- Central office quality and performance
- Adjustable Guard Time
- Automatic Tone Burst mode
- Call Progress mode
- Single +5 volt power supply
- 20-pin DIP, 20-pin SOIC, 28-pin PLCC packages
- 2MHz microprocessor port operation
- No continuous Φ2 clock required, only strobe

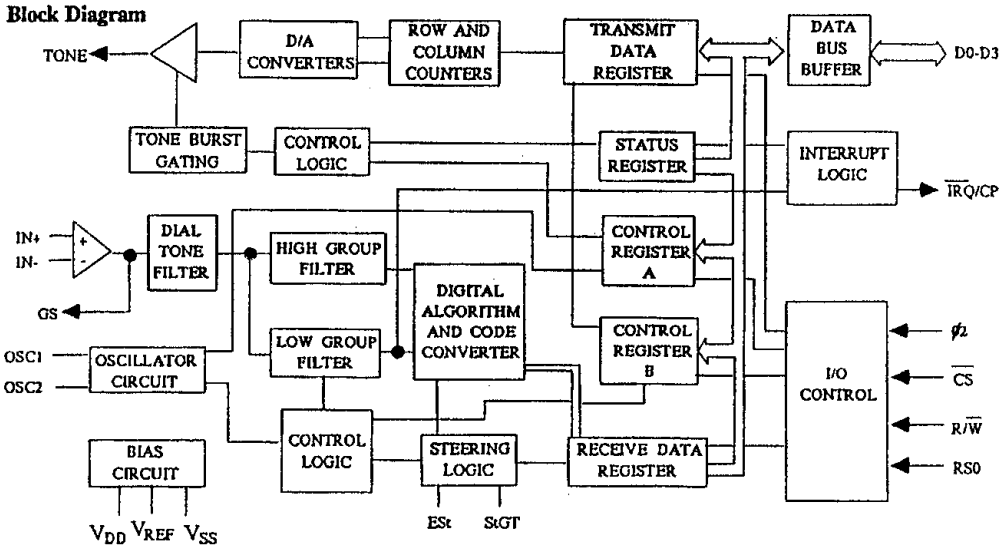
Applications

- Paging systems
- Repeater systems/mobile radio
- Interconnect dialers
- PABX systems
- Computer systems
- Fax machines
- Pay telephones
- Credit card verification

General Description

The CMD CM8880 is a fully integrated DTMF Transceiver, featuring adjustable guard time, automatic tone burst mode, call progress mode and a fully compatible 6500/6800 microprocessor interface. The CM8880 is manufactured using state-of-the-art advanced CMOS technology for low power consumption and precise data handling. The CM8880 is based on the industry standard CM8870 DTMF Receiver, while the transmitter utilizes a switched-capacitor D/A converter for low distortion, highly accurate DTMF signaling. Internal counters provide an automatic tone burst mode which allows tone bursts to be transmitted with precise timing. A call progress filter can be selected by an external microprocessor for analyzing call progress tones. The CM8880-1 is functionally equivalent to the CM8880 but has input sensitivity to meet European specifications. The CM8880-2 is electrically equivalent to the CM8880 but does not include the call progress function.

Block Diagram



This is advance information and specifications are subject to change without notice.

2000 W. 14TH ST., TEMPE, AZ 85281 • (602)921-6000 • FAX (602)921-6298

Ratings	Symbol	Value
Supply Voltage (V_{DD} - V_{SS})	V_{DD}	+6.0V Max
Voltage on any Pin	V_{dc}	-0.3V to V_{DD} +0.3V
Current on any Pin	I_{DD}	10 mA Max
Operating Temperature	T_A	-40°C to +85°C
Storage Temperature	T_S	-65°C to 150°C

This device contains input protection against damage due to static voltages or electric fields; however, precautions should be taken to avoid application of voltages higher than the maximum rating.

Note: Exceeding these ratings may cause permanent damage. Functional operation under these conditions is not implied.

DC Characteristics: All voltages referenced to V_{SS}, V_{DD} = 5.0V ± 5%, fc = 3.579545 MHz, T_A = -40°C to +85°C unless otherwise noted.

Parameter	Symbol	Min	Typ*	Max	Units
Operating Supply Voltage	V_{DD}	4.75	5.0	5.25	V
Operating Supply Current	I_{DD}			10	mA
Power Consumption	P_O			52.5	mW
Inputs					
High Level Input Voltage OSC1	V_{IHO}	3.5			V
Low Level Input Voltage OSC1	V_{ILO}			1.5	V
Input Impedance (@1KHz) IN+, IN-	R_{IN}		10.0		MΩ
Steering Threshold Voltage (V_{DD}=5.0V)	V_{TSt}	2.2	2.3	2.5	V
Outputs					
High Level Output Voltage (No Load) OSC2	V_{OHO}	4.9			V
Low Level Output Voltage (No Load) OSC2	V_{OLO}			0.1	V
Output Leakage Current (V_{OH}=2.4V) IRQ	I_{OZ}		1.0	10.0	μA
V_{REF} Output Voltage (No Load)	V_{REF}	2.4		2.7	V
V_{REF} Output Resistance	R_{OR}			1	KΩ
Data Bus (DO-D3, Φ2, R/$\overline{W}$, RSO, $\overline{CS}$)					
Low Level Input Voltage	V_{IL}			0.8	V
High Level Input Voltage	V_{IH}	2.0			V
Low Level Output Voltage (I_{OL}=1.6mA)	V_{OL}			0.4	V
High Level Output Voltage (I_{OH}=400μA)	V_{OH}	2.4			V
Input Leakage Current (V_{IN} = 0.4 to 2.4V)	I_{IZ}			10.0	μA

2000 W. 14TH STREET, TEMPE, AZ 85281 • (602)921-6000 • FAX (602)921-6000

Electrical Characteristics - Gain Setting Amplifier: All voltages referenced to V_{SS} unless otherwise noted. V_{DD} = 5.0V, V_{SS} = 0V, T_A = 25°C

Parameter	Symbol	Min	Typ*	Max	Units
Input Leakage Current ($V_{SS} \le V_{IN} \le V_{DD}$)	I_{IN}		100		nA
Input Resistance	R_{IN}		10		MΩ
Input Offset Voltage	V_{OS}		25		mV
Power Supply Rejection (1 KHz)	PSRR		60		dB
Common Mode Rejection (-3.0V $\le V_{IN} \le$ 3.0V)	CMRR		60		dB
DC Open-Loop Voltage Gain	A_{VOL}		65		dB
Unity Gain Bandwidth	BW		1.5		MHz
Output Voltage Swing ($R_L \ge$ 100KΩ to V_{SS})	V_O		4.5		V_{PP}
Maximum Capacitive Load GS	C_L		100		pF
Maximum Resistive Load GS	R_L		50		KΩ
Common Mode Range (No Load)	V_{CM}		3.0		V_{PP}

AC Characteristics: All Voltages referenced to V_{SS} unless otherwise noted. V_{DD} = 5.0V, V_{SS} = OV, fc = 3.579545 MHz

Receive Signal Conditions

Valid Input Signal Levels CM8880, CM8880-2 (Each Tone of Composite Signal; Notes 1, 2, 3, 5, 6, 9)		-29 27.5		+1 869	dBm mV_{RMS}
Valid Input Signal Levels CM8880-1 (Each Tone of Composite Signal; Notes 1, 2, 3, 5, 6, 9)		-31 21.8		+1 869	dBm mV_{RMS}
Input Signal Level Reject CM8880-1 (Each Tone of Composite Signal; Notes 1, 2, 3, 5, 6, 9)		-37 10.9			dBm mV_{RMS}
Positive Twist Accept (Notes 2, 3, 6, 9)				6	dB
Negative Twist Accept (Notes 2, 3, 6, 9)				6	dB
Freq. Deviation Accept (Notes 2, 3, 5, 9)		±1.5% ±2 Hz			Nom.
Freq. Deviation Reject (Notes 2, 3, 5)		±3.5%			Nom.
Third Tone Tolerance (Notes 2, 3, 4, 5, 9, 10)			-16		dB
Noise Tolerance (Notes 2, 3, 4, 5, 7, 9, 10)			-12		dB
Dial Tone Tolerance (Notes 2, 3, 4, 5, 8, 9, 11)			+22		dB

Call Progress (CM8880, CM8880-1)

Lower Frequency (@ -25 dBm) ACCEPT	f_{LA}		320		Hz
Upper Frequency (@ -25 dBm) ACCEPT	f_{HA}		510		Hz
Lower Frequency (@ -25 dBm) REJECT	f_{LR}		290		Hz
Upper Frequency (@ -25 dBm) REJECT	f_{HR}		540		Hz

2000 W. 14TH STREET, TEMPE, AZ 85281 • (602)921-6000 • FAX (602)921-6298

Receive Timing

Parameter	Symbol	Min	Typ*	Max	Units
Tone Present Detect Time	t_{DP}	5	11	14	mS
Tone Absent Detect Time	t_{DA}	0.5	4	8.5	mS
Tone Duration Accept (Ref. Fig. 9)	t_{REC}			40	mS
Tone Duration Reject (Ref. Fig. 9)	$t_{\overline{REC}}$	20			mS
Interdigit Pause Accept (Ref. Fig. 9)	t_{ID}			40	mS
Interdigit Pause Reject (Ref. Fig. 9)	t_{DO}	20			mS
Delay St to b3	t_{PSb3}		13		μS
Delay St to RX_0-RX_3	t_{PSRX}		8		μS

Transmit Timing

Tone Burst Duration (DTMF Mode)	t_{BST}	50		52	mS
Tone Pause Duration (DTMF Mode)	t_{PS}	50		52	mS
Tone Burst Duration (Extended, Call Process Mode)	t_{BSTE}	100		104	mS
Tone Pause Duration (Extended, Call Process Mode)	t_{PSE}	100		104	mS

Tone Output

High Group Output Level (R_L = 10 KΩ)	V_{HOUT}	-6.1		-2.1	dBm
Low Group Output Level (R_L = 10 KΩ)	V_{LOUT}	-8.1		-4.1	dBm
Pre-emphasis (R_L - 10KΩ)	dB_P	0	2	3	dB
Output Distortion (R_L = 10KΩ, 3.4 KHz Bandwidth)	THD		-25		dB
Frequency Deviation (f = 3.5795 MHz)	f_D		±0.7	±1.5	%
Output Load Resistance	R_{LT}	10		50	KΩ

2000 W. 14TH STREET, TEMPE, AZ 85281 • (602)921-6000 • FAX (602)921-6298

AC Characteristics: V_{DD} = 5.0V ± 5%, V_{SS} = 0V, T_A = -40°C to +85°C

Microprocessor Interface

Parameter	Symbol	Min	Typ*	Max	Units
Φ2 Cycle Period	t_{CYC}	0.5			μs
Φ2 High Pulse Width	t_{CH}	200			ns
Φ2 Low Pulse Width	t_{CL}	180			ns
Φ2 Rise and Fall Time	t_R, t_F			25	ns
Address, R/$\overline{W}$ Hold Time	t_{AH}, t_{RWH}	10			ns
Address, R/$\overline{W}$ Setup Time (Prior to Φ2)	t_{AS}, t_{RWS}	23			ns
Data Hold Time (Read)	t_{DHR}	22			ns
Φ2 to Valid Data Delay (Read) (200pF load)	t_{DDR}			150	ns
Data Setup Time (Write)	t_{DSW}	45			ns
Data Hold Time (Write)	t_{DHW}	10			ns
Input Capacitance DO-D3	C_{IN}		5		pF
Output Capacitance $\overline{IRQ}$ CP	C_{OUT}		5		pF

DTMF Clock

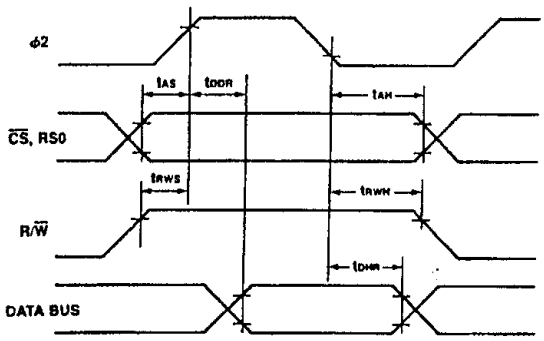
Crystal Clock Frequency	fc	3.5759	3.5795	3.5831	MHz
Clock Input Rise Time (External Clk)	t_{LHCL}			110	ns
Clock Input Fall Time (External Clk)	t_{HLCL}			110	ns
Clock Input Duty Cycle (External Clk)	DC_{CL}	40	50	60	%
Capacitive Load OSC2	C_{LO}			30	pF

* Typical values are for use as design aids only, and are not guaranteed or subject to production testing.

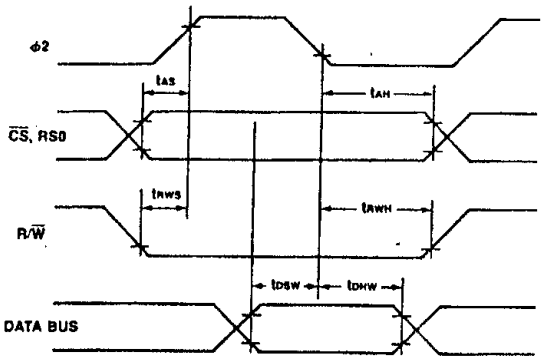
Notes:
1. dBm = decibels above or below a reference power of 1 mW into a 600 ohm load.
2. Digit sequence consists of all 16 DTMF tones.
3. Tone duration = 40 mS. Tone pause = 40 mS.
4. Nominal DTMF frequencies are used.
5. Both tones in the composite signal have an equal amplitude.
6. The tone pair is deviated by ± 1.5% ± 2Hz.
7. Bandwidth limited (3 KHz) Gaussian noise.
8. The precise dial tone frequencies are 350 and 440 Hz (±2%)
9. For an error date of less than 1 in 10,000.
10. Referenced to the lowest amplitude tone in the DTMF signal.
11. Referenced to the minimum valid accept level.

2000 W. 14TH STREET, TEMPE, AZ 85281 • (602)921-6000 • FAX (602)921-6298

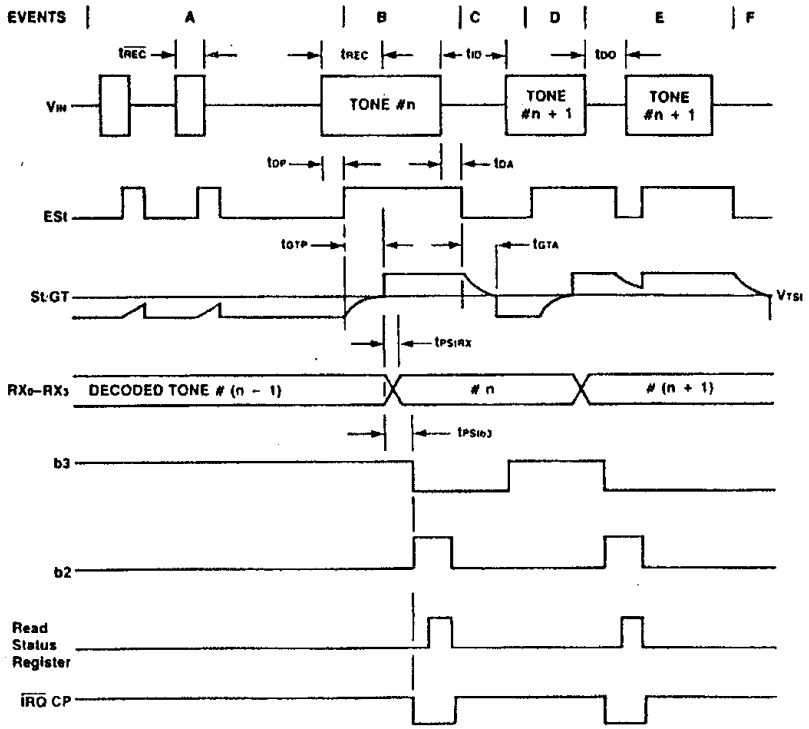
Microprocessor Read Cycle



Microprocessor Write Cycle



General Transceiver Timing

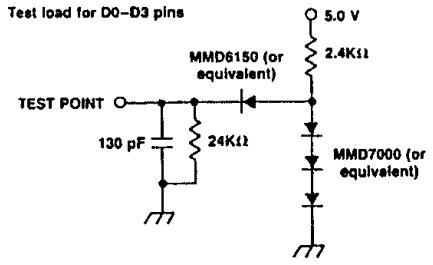


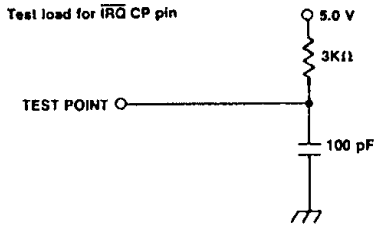
Explanation of Events

A	Tone bursts detected, tone duration invalid, RX data register not updated.
B	Tone #n detected, tone duration valid, tone decoded and latched in RX data register.
C	End of tone #n detected, tone absent duration valid, information in RX data register retained until next valid tone pair.
D	Tone #n + 1 detected, tone duration valid, tone decoded and latched in RX data register.
E	Acceptable dropout of tone #n + 1, tone absent duration invalid, data remains unchanged.
F	End of tone #n + 1 detected, tone absent duration valid, information in RX data register retained until next valid tone pair.

Explanation of Symbols

V_{IN}	DTMF composite input signal.
ESt	Early steering output. Indicates detection of valid tone frequencies.
St/GT	Steering input/guard time output. Drives external RC timing circuit.
RX_0-RX_3	4-bit decoded data in receive data register.
b3	Delayed steering. Indicates that valid frequencies have been present/absent for the required guard time thus consituting a valid signal. Active low for the duration of a valid DTMF signal.
b2	Indicates that valid data is in the receive data register. The bit is cleared after the status register is read.
IRQ/CP	Interrupt is active indicating that new data is in the R X data register. The interrupt is cleared after the status register is read.
$\overline{tREC}$	Maximum DTMF signal duration not detected as valid.
tREC	Minimum DTMF signal duration required for valid recognition.
t_{ID}	Minimum time between valid sequential DTMF signals.
t_{DO}	Maximum allowable dropout during valid DTMF signal.
t_{DP}	Time to detect valid frequencies present.
t_{DA}	Time to detect valid frequencies absent.
t_{GTP}	Guard time, tone present.
t_{GTA}	Guard time, tone absent.



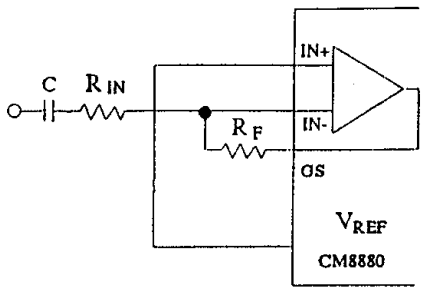


Functional Description

The CM8880 Integrated DTMF Transceiver provides the design engineer with not only low power consumption, but central office quality performance. The CM8880's internal architecture consists of a high performance DTMF receiver with an internal Gain Setting Amplifier and DTMF Generator. The DTMF Generator contains a Tone Burst Counter for generating precise tone bursts and pauses. The Call Progress mode, when selected, allows the detection of call progress tones. A standard 8051, 8086/8 series microprocessor interface allows access to an internal status register, two control registers and two data registers within the CM8880.

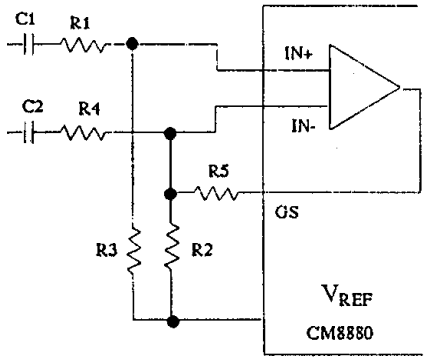
Input Configuration

The CM8880 input arrangement consists of a differential input operational amplifier and bias sources (V_{REF}) for biasing the amplifier inputs at $V_{DD}/2$. Provisions are made for the connection of a feedback resistor to the op amp output (GS) for gain adjustment. In the single-ended configuration, the input pins should be connected as shown in Figure 1, while Figure 2 shows the necessary connections for a differential input configuration.



VOLTAGE GAIN
$(A_V) = R_F / R_{IN}$

Figure 1. Single-Ended Input Configuration



DIFFERENTIAL INPUT AMPLIFIER
C1 = C2 = 10nF
R1 = R4 = R5 = 100KΩ
R2 = 60KΩ, R3 = 37.5KΩ
R3 = (R2R5)/(R2 + R5)
VOLTAGE GAIN
(Av diff) = R5/R1
INPUT IMPEDANCE
$(Z_{IN}\ \text{diff}) = \sqrt{2\ R1^2 + (1/wC)^2}$

Figure 2. Differential Input Configuration

Receiver Section

Separation of the low and high-group tones is achieved by applying the DTMF signal to the inputs of two sixth-order switched capacitor bandpass filters, the bandwidths of which correspond to the low and high-group frequencies as shown in Figure 5. The low-group filter incorporates notches at 350 Hz and 440 Hz for excellent dial-tone rejection. Each filter output is followed by a single-order switched capacitor filter section which smoothes the signals prior to limiting. Limiting is performed by high-gain comparators with hysteresis to prevent detection of unwanted low-level signals. The outputs of the comparators provide full-rail logic swings at the incoming DTMF signals frequencies.

Following the filter section is a decoder which employs digital counting techniques to determine the frequencies of the incoming tones, and to verify that the incoming tones correspond to standard DTMF frequencies. A complex averaging algorithm protects against tone simulation by extraneous signals (e.g. voice), while still providing tolerance to small deviations in frequency. The averaging algorithm was developed to ensure an optimum combination of immunity to talk-off, as well as a tolerance to the presence of interfering frequencies (3rd tones) and noise. When the detector recognizes the presence of two valid tones (sometimes referred to as "signal condition" in industry publications), the "Early Steering" (ESt) output will go to an active state. Any subsequent loss of signal condition will cause ESt to assume an inactive state.

Steering Circuit

Before registration of a decoded tone pair, the receiver checks for a valid signal duration (referred to as Character Recognition Condition). This check is performed by an external RC time constant driven by ESt. A logic high on ESt causes V_C (See Figure 3) to rise as the capacitor discharges. Provided that the signal condition is maintained (ESt remains high) for the validation period (t_{GTP}), Vc reaches the threshold (V_{TSt}) of the steering logic to register the tone pair, latching its corresponding 4-bit code (See Figure 5) into the Receive Data Register. At this point the GT output is activated and drives V_C to V_{DD}. GT continues to drive high as long as ESt remains high. Finally, after a short delay to allow the output latch to settle, the Delayed Steering output flag goes high, signalling that a received tone pair has been registered. It is possible to monitor the status of the Delayed Steering flag by checking the appropriate bit in the Status Register. If Interrupt Mode has been selected, the IRQ/CP pin will pull low when the Delayed Steering flag is active.

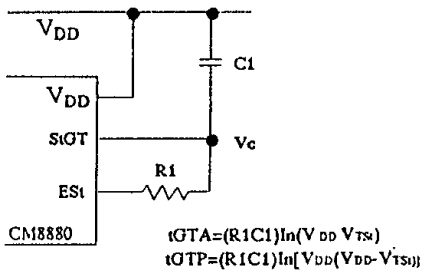


Figure 3. Basic Steering Circuit

The contents of the output latch are updated on an active Delayed Steering transition. This data is presented to the 4-bit bi-directional data bus when the Receive Data Register is read. The steering circuit works in reverse to validate the interdigit pause between signals. Thus, as well as rejecting signals too short to be considered valid, the receiver will tolerate signal interruptions (drop out) too short to be considered a valid pause. This facility, together with the capability of selecting the steering time constants externally, allows the designer to tailor performance to meet a wide variety of system requirements.

Guard Time Adjustment

The simple steering circuit shown in Figure 3 is adequate for most applications. Component values are chosen according to the formula:

$$tREC = tDP + tGTP$$
$$tID = tDA + tGTA$$

The value of tDP is a device parameter and tREC is the minimum signal duration to be recognized by the receiver. A value for C of 0.1 uF is recommended for most applications, leaving R to be selected by the designer. Different steering arrangements may be used to select independently the guard times for tone present (tGPT) and tone absent (tGTA). This may be necessary to meet system specifications which place both accept and reject limits on both tone duration and interdigital pause. Guard Time adjustment also allows the designer to tailor system parameters such as talk-off and noise immunity. Increasing tREC improves talk-off performance since it reduces the probability that tones simulated by speech will maintain signal condition long enough to be registered. Alternatively, a relatively short tREC with a long tDO would be appropriate for extremely noisy environments where fast acquisition time and immunity to tone drop-outs are required. Design information for Guard Time adjustment is shown in Figure 4.

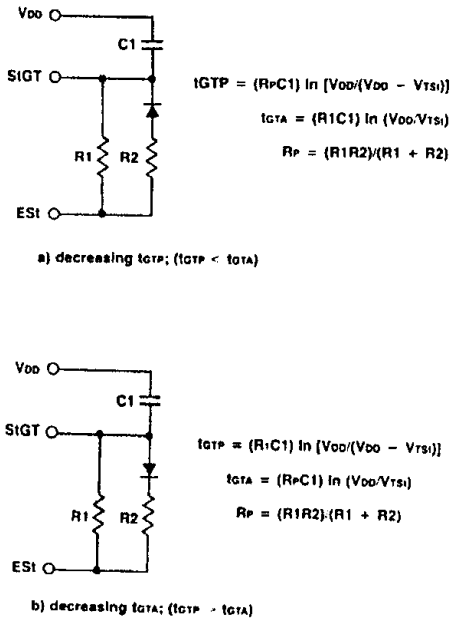


Figure 4. Guard Time Adjustment

FLOW	FHIGH	DIGIT	D3	D2	D3	DO
697	1209	1	0	0	0	1
697	1336	2	0	0	1	0
697	1477	3	0	0	1	1
770	1209	4	0	1	0	0
770	1336	5	0	1	0	1
770	1477	6	0	1	1	0
852	1209	7	0	1	1	1
852	1336	8	1	0	0	0
852	1477	9	1	0	0	1
941	1336	0	1	0	1	0
941	1209	*	1	0	1	1
941	1477	#	1	1	0	0
697	1633	A	1	1	0	1
770	1633	B	1	1	1	0
852	1633	C	1	1	1	1
941	1633	D	0	0	0	0

0 = LOGIC LOW, 1 = LOGIC HIGH

Figure 5. Functional Encode/Decode

Call Progress Filter (CM8880, CM8880-1)
A Call Progress (CP) Mode can be selected allowing the detection of various tones which identify the progress of a telephone call on the network. The Call Progress tone input and DTMF input are common, however, call progress tones can only be detected when the CP Mode has been selected. DTMF signals cannot be detected if the CP Mode has been selected (see Table 5). Figure 6 indicates the useful detect bandwidth of the Call Progress filter. Frequencies presented to the input (IN+ and IN-) which are within the 'accept' bandwidth limits of the filter are hard-limited by a high-gain comparator with the $\overline{IRQ}$/CP pin serving as the output. The square wave output obtained from the schmitt trigger can be analyzed by a microprocessor or counter arrangement to determine the nature of the Call Progress tone being detected. Frequencies which are in the 'reject' area will not be detected, and consequently there will be no activity on $\overline{IRQ}$/CP as a result of these frequencies.

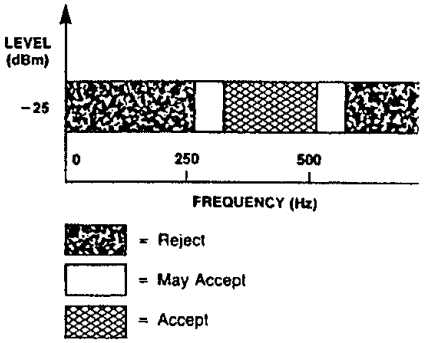


Figure 6. Call Progress Response

DTMF Generator
The DTMF transmitter employed in the CM8880 is capable of generating all sixteen standard DTMF tone pairs with low distortion and high accuracy. All frequencies are derived from an external 3.58 MHz crystal. The sinusoidal waveforms for the individual tones are digitally synthesized using row and column programmable dividers and switched capacitor D/A converters. The row and column tones are mixed and filtered providing a DTMF signal with low total harmonic distortion and high accuracy. To specify a DTMF signal, data conforming to the encoding format shown in Figure 5 must be written to the Transmit Data Register. Note that this is the same as the receiver output code. The individual tones which are generated (f_{LOW} and f_{HIGH}) are referred to as low-group and high-group tones. As seen from Table 1, the Low-Group frequencies are

697, 770, 852, and 941 Hz; the High-Group frequencies are 1209, 1336, 1477 and 1633 Hz. Typically the High-Group to Low-Group amplitude ratio (twist) is 2dB to compensate for High-Group attenuation on long loops.

DTMF Generator Operation

The period of each tone consists of 32 equal time segments. The period of a tone is controlled by varying the length of these time segments. During write operations to the transmit data register, 4-bit data on the bus is latched and converted to 2 of 8 coding for use by the programmable divider circuitry. This code is used to specify a time segment length which will ultimately determine the frequency of the tone. When the divider reaches the appropriate count as determined by the input code, a reset pulse is issued and the counter starts again. The number of time segments is fixed at 32; however, by varying the segment length as described above, the frequency can also be varied. The divider output clocks another counter which addresses the sinewave lookup ROM. The lookup table contains codes which are used by the switched capacitor D/A converter to obtain discrete and highly accurate DC voltage levels. Two identical circuits are employed to produce row and column tones which are then mixed using a low noise summing amplifier. The oscillator described needs no "start-up" time as in other DTMF generators since the crystal oscillator is running continuously, thus providing a high degree of tone burst accuracy. Under conditions when there is no tone output signal, the TONE pin assumes a DC level of 2.5 volts (typ). A bandwidth limiting filter is incorporated and serves to attenuate distortion products above 4 KHz. It can be seen from Figure 7 that the distortion products are very low in amplitude.

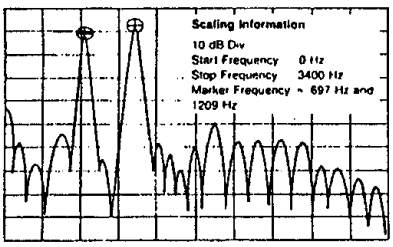


Figure 7. Spectrum Plot

ACTIVE CELL	OUTPUT FREQUENCY (Hz)		% ERROR
	SPECIFIED	ACTUAL	
L1	697	699.1	+0.30
L2	770	766.2	-0.49
L3	852	847.4	-0.54
L4	941	948.0	+0.74
H1	1209	1215.9	+0.57
H2	1336	1331.7	-0.32
H3	1447	1471.9	-0.35
H4	1633	1645.0	+0.73

Table 1. Actual Frequencies Versus Standard Requirements

Burst Mode

In certain telephony applications it is required that DTMF signals being generated be of a specific duration determined either by the particular application or by any one of the exchange transmitter specifications currently existing. Standard DTMF signal timing can be accomplished by making use of the Burst Mode. The transmitter is capable of issuing symmetric bursts/pauses of predetermined duration. This burst/pause duration is 51 mS ± 1 mS which is a standard interval for autodialer and central office applications. After the burst pause has been issued, the appropriate bit is set in the Status Register, indicating that the transmitter is ready for more data. The timing described above is available when the DTMF Mode has been selected. However, when CP Mode (Call Progress Mode) is selected, a secondary burst/pause time is available such that this interval is extended to 102 mS ± 2 mS. The extended interval is useful when precise tone bursts of longer than 51 mS duration and 51 mS pause are desired. Note that when CP mode and burst mode have been selected, DTMF tones may be transmitted only and not received. In certain applications where a non-standard burst/pause time is desirable, a software timing loop or external timer can be used to provide the timing pulses when the burst mode is disabled by enabling and disabling the transmitter.

The CM8880 is initialized on power-up sequence such that DTMF mode and burst mode are selected.

CP mode and burst mode have been selected, DTMF tones may be transmitted only and not received. In certain applications where a non-standard burst/pause time is desirable, a software timing loop or external timer can be used to provide the timing pulses when the burst mode is disabled by enabling and disabling the transmitter.

The CM8880 is initialized on power-up sequence such that DTMF mode and burst mode are selected.

Single Tone Generation

A Single Tone Mode is available whereby individual tones from the low-group or high-group can be generated. This mode can be used for DTMF test equipment applications, acknowledgment tone generation and distortion measurements. Refer to Control Register B (Table 6) description for details.

Distortion Calculations

The CM8880 is capable of producing precise tone bursts with minimal error in frequency (See Table 1). The internal summing amplifier is followed by a first-order low-pass switched-capacitor filter to minimize harmonic components and intermodulation products.

$$THD(\%) = 100\,\frac{\sqrt{V^2_{2f} + V^2_{3f} + V^2_{4f} + \ldots V^2_{nf}}}{V_{fundamental}}$$

Equation 1. THD(%) For a Single Tone

$$THD(\%) = 100\,\frac{\sqrt{V^2_{2L} + V^2_{3L} + \ldots V^2_{nL} + V^2_{2H} + V^2_{3H} + \ldots V^2_{nH} + V^2_{IMD}}}{\sqrt{V^2_L + V^2_H}}$$

Equation 2. THD(%) For a Dual Tone

The total harmonic distortion for a single tone can be calculated using Equation 1 which is the ratio of the total power of all the extraneous frequencies to the power of the fundamental frequency expressed as a percentage. The Fourier components of the tone output correspond to V2f....Vnf as measured on the output waveform. The total harmonic distortion for a dual tone can be calculated using Equation 2. V_L and V_H correspond to the low-group amplitude and high-group amplitude, respectively, and V^2_{IMD} is the sum of all the intermodulation components. The internal switched-capacitor filter following the D/A converter keeps distortion products down to a very low level as shown in Figure 7.

DTMF Clock Circuit

The internal clock circuit is completed with the addition of a standard television color burst crystal having a resonant frequency of 3.579545 MHz. A number of CM8880 devices can be connected as shown in Figure 8 such that only one crystal is required.

Figure 8. Common Crystal Connection

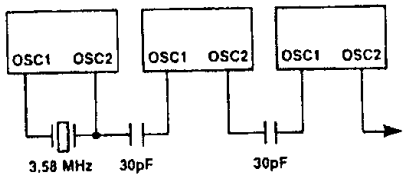


Table 2. Internal Register Functions

RSO	RD	WR	Function
0	1	0	Write to Transmitter
0	0	1	Read from Receiver
1	1	0	Write to Control Register
1	0	1	Read from Status Register

Table 3. CRA Bit Positions

b3	b2	b1	b0
RSEL	IRQ	CP $\overline{\text{DTMF}}$	TOUT

Table 4. CRB Bit Positions

b3	b2	b1	b0
C/R	S/D	TEST	BURST

Microprocessor Interface

The CM8880 employs a microprocessor interface which allows precise control of transmitter and receiver functions. There are five internal registers associated with the microprocessor interface which can be subdivided into three categories, ie; data transfer, transceiver control and transceiver status.

There are two registers associated with data transfer operations. The Receive Data Register contains the output code of the last valid DTMF tone pair to be decoded and is a read-only register. The data entered in the Transmit Data Register will determine which tone pair is to be generated (See Figure 5 for coding details). Data can only be written to the Transmit Data Register. Transceiver control is accomplished with two Control Registers (CRA and CRB) which occupy the same address space. A write operation to CRB can be executed by setting the appropriate bit in CRA. The following write operation to the same address will then be directed to CRB and subsequent write cycles will then be directed back to CRA. Internal reset circuitry will clear the control registers on power-up; however, as a precautionary measure the initialization software should include a routine to clear the registers. Refer to Table 5 and 6 for details concerning the Control Registers. The $\overline{IRQ}$/CP pin can be programmed such that it will provide an interrupt request signal upon validation of DTMF signals, or when the transmitter is ready for more data (Burst mode only). The $\overline{IRQ}$/CP pin is configured as an open-drain output device and as such requires a pull-up resistor (See Figure 9).

Table 5. Control Register A Description

Bit	Name	Function	Description
b0	TOUT	Tone Output	A logic '1' enables the Tone Output. This function can be implemented in either the Burst Mode or Non-Burst Mode.
b1	CP/$\overline{DTMF}$	Mode Control	In DTMF Mode (logic '0'), the device is capable of generating and receiving Dual Tone Multi-Frequency signals. When the CP (Call Progress) mode is selected (logic '1'), a 6th order bandpass filter is enabled to allow Call Progress tones to be detected. Call Progress tones which are within the specified bandwidth will be presented at the $\overline{IRQ}$/CP pin in rectangular wave format if the IRQ bit has been enabled (b2=1). Also when the CP mode and Burst Mode have both been selected, the transmitter will issue DTMF signals with a burst and pause of 102 mS (typ) duration. This signal duration is twice that obtained from the DTMF transmitter, if DTMF mode had been selected. Note that signals cannot be decoded when the CP mode of operation has been selected.
b2	IRQ	Interrupt Enable	A logic '1' enables the Interrupt Mode. When this mode is active and the DTMF Mode has been selected (b1=0), the $\overline{IRQ}$/CP pin will pull to a logic '0' condition when either 1) a validDTMF signal has been received and has been present for the guard time duration or 2) the transmitter is ready for more data (Burst Mode only).
b3	RSEL	Register Select	A logic '1' selects Control Register B on the next write cycle to the Control Register address. Subsequent write cycles to the Control Register are directed back to Control Register A.

Table 6. Control Register B Description

Bit	Name	Function	Description
b0	BURST	Burst Mode	A logic '0' enables the Burst Mode. When this mode is selected, data corresponding to the desired DTMF tone pair can be written to the Transmit Data Register, resulting in a tone burst of a specific duration (See AC Characteristics). Subsequently, a pause of the same duration is induced. Immediately following the pause, the Status Register is updated indicating that the Transmit Data Register is ready for further instructions, and an interrupt will be generated if the Interrupt Mode has been enabled. Additionally, if Call Progress (CP) Mode has been enabled, the burst and pause duration is increased by a factor of two. When the Burst Mode is not selected (logic '1') tone bursts of any desired duration may be generated.
b1	TEST	Test Mode	By enabling the Test Mode (logic '1') the $\overline{IRQ}$/CP pin will present the delayed steering (inverted) signal from the DTMF receiver. Refer to General Transceiver Timing (b3 waverform) for details concerning the output waveform. DTMF Mode must be selected (CRA b1 = 0) before Test Mode can be implemented.
b2	S/$\overline{D}$	Single/Dual Tone Generation	A logic '0' will allow Dual Tone Multi-Frequency signals to be produced. If single-tone generation is enabled (logic '1'). either row or column tones (low-group or high-group) can be genrated depending on the state of b3 in Control Register B.
b3	C/$\overline{R}$	Column/Row Tones	When used in conjunction with b2 (above) the transmitter can be made to generate single-row or single-column frequencies. A logic '0' will select row frequencies and a logic '1' will select column frequencies.

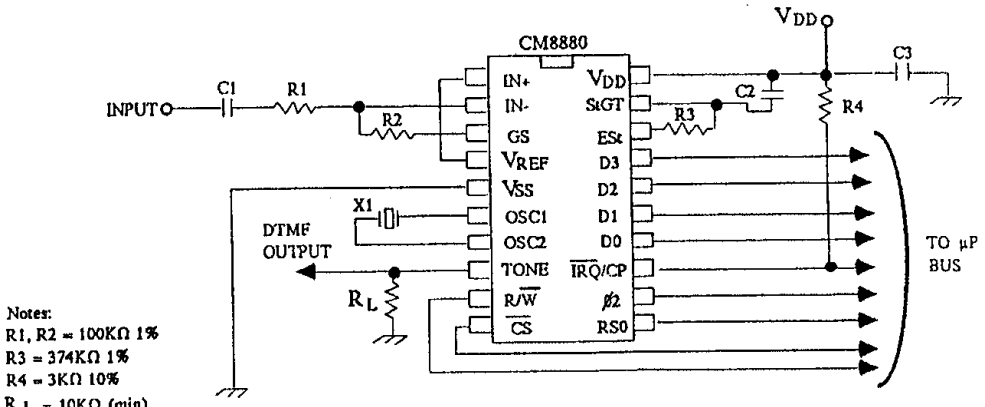


Figure 9. Application Circuit (Single Ended Input)

Table 7. Status Register Description

Bit	Name	Status Flag Set	Status Flag Cleared
b0	IRQ	Interrupt has occurred. Bit one (b1) and/or bit two (b2) is set.	Interrupt is inactive. Cleared after Status Register is read.
b1	Transmit Data Register Empty (Burst Mode Only)	Pause duration has terminated and transmitter is ready for new data.	Cleared after Status Register is read or when in Non-Burst Mode.
b2	Receive Data Register Full	Valid data is in the Receive Data Register.	Cleared after Status Register is read.
b3	$\overline{\text{Delayed Steering}}$	Set upon the valid detection of the absence of a DTMF signal.	Cleared upon the detection of a valid DTMF signal.

Pin Function Table

Name	Description
IN+	Non-inverting op-amp input.
IN-	Inverting op-amp input.
GS	Gain Select. Gives access to output of front end differential amplifier for connection of feedback resistor.
V_{REF}	Reference voltage output. Nominally $V_{DD}/2$ is used to bias inputs at mid-rail (see application circuit).
V_{SS}	Negative power supply input.
OSC1	DTMF clock/oscillator input.
OSC2	Clock output. A 3.5795 MHz crystal connected between OSC1 and OSC2 completes the internal oscillator circuit.
TONE	Dual Tone Multi-Frequency (DTMF) output.
R/$\overline{W}$	Read/write input. Controls the direction of data transfer to and from the microprocessor and the CM8880. TTL compatible.
CS	Chip Select. TTL input (CS = 0 to select the chip).
RSO	Register select input. See register decode table. TTL compatible.
Φ2	System clock input. May be continuous or strobed only during read or write. TTL compatible.

Name	Description
$\overline{\text{IRQ}}$/CP	Interrupt request to microprocessor (open-drain output). Also, when Call Progress (CP) Mode has been selected and Interrupt enabled the IRQ/CP pin will output a rectangular wave signal representative of the input signal applied at the input op-amp. The input signal must be within the bandwidth limits of the Call Progress filter. See Filter 6.
D0-D3	Microprocessor data bus. TTL compatible.
ESt	Early Steering output. Presents a logic high once the digital algorithm has detected a valid tone pair (signal condition). Any momentary loss of signal condition will cause ESt to return to a logic low.
StGT	Steering input/Guard Time output (bidirectional). A voltage greater than V_{TSt} detected at St causes the device to register the detected tone pair and update the output latch. A voltage less than V_{TSt} frees the device to accept a new tone pair. The GT output acts to reset the extenal steering time-constant; its state is a function of ESt and the voltage on St.
V_{DD}	Positive power supply input.