

DTMF Transceiver by Zarlink Semiconductor (#604-00003)

Introduction

This documentation shows how to use the Zarlink Semiconductor DTMF transceiver chip with Parallax BASIC Stamp[®] microcontrollers. The MT8880 is a dual-tone, multi-frequency (DTMF) transceiver on a chip. It can be configured to send or receive “touch” tones used in many phone and radio communication systems.

Parts Listing (sold separately)

The parts below are available from the online Components Shop at www.parallax.com. These parts are required to perform the experiment in this documentation.

- MT8880 DTMF Transceiver Chip (#604-00003) Qty. 1
- .1 uf mono radial capacitor (#200-01040) Qty. 4
- 3.3K ohm resistor ¼ watt (#150-03320) Qty. 1
- 10K ohm resistor ¼ watt (#150-01030) Qty. 3
- 100K ohm resistor ¼ watt (#150-01011) Qty. 1
- 390K ohm resistor ¼ watt (#150-03940) Qty. 1
- 3.58 MHz resonator (#250-03550) Qty. 1

Applications

- Credit Card Systems
- Paging Systems
- Repeater Systems/Mobile Radio
- Interconnect Dialers
- Personal Computers

Technical Specifications

Characteristics	Sym	Min	Type	Max	Units
Operating supply voltage	Vdd	4.75	5.0	5.5	V
Operating supply current	Idd		7.0	11	mA

Power consumption	Pc		57.8	mW	
Operating temperature	To	-40	+85	Celcius	
Crystal clock frequency	fCLK	3.575965	3.579545	3.583124	MHz

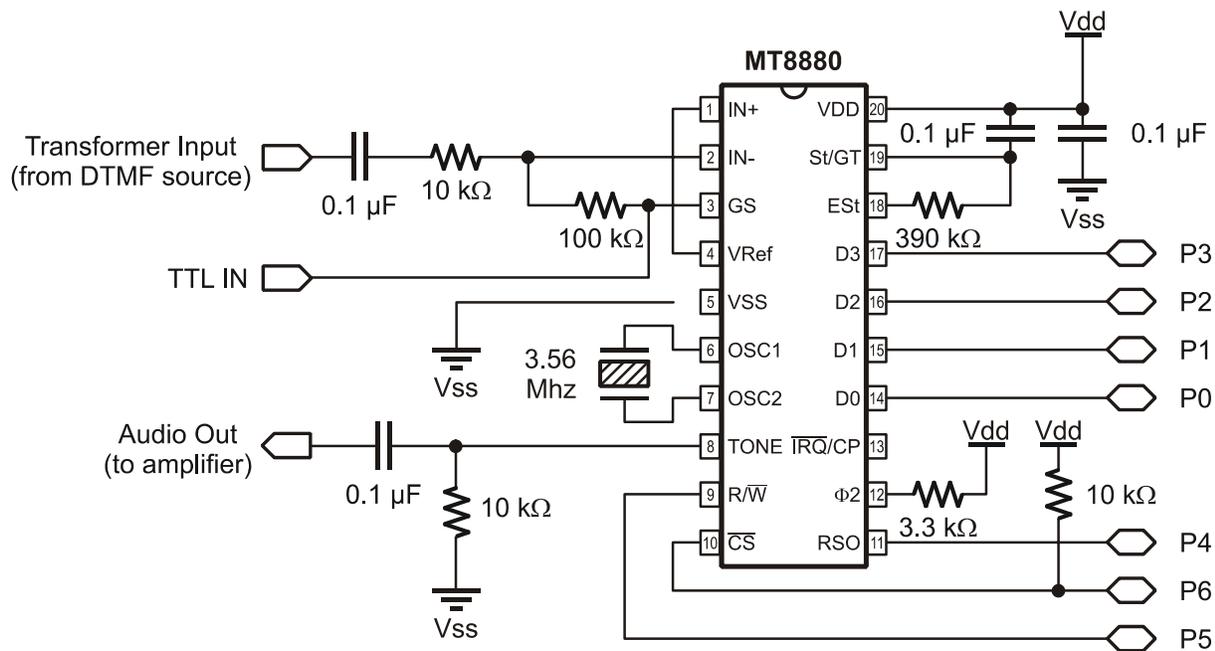
(Full data sheet attached)

Hardware interface

Communication with the MT8880 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 MT8880 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 MT8880's ability to hear and distinguish DTMF tones.

Connections



Software interface

The software needed to interface with the MT8880 is relatively simple, thanks to a four-bit data bus. A write cycle consists of the following steps (starting with the MT8880'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 MT8880
- (6) Set CS to terminate the write operation and deactivate the MT8880

Reading the MT8880 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 MT8880
- (5) Read the data from the bus
- (6) Set CS to terminate the read operation and deactivate the MT8880

The table below summarizes the interaction of the MT8880's control pins. To sum up the table, the MT8880 is active only when CS is 0. The RW bit determines the data direction; 1 = read (data from MT8880 to Stamp) and 0 = write (data from Stamp to MT8880). The RS bit determines whether the transaction involves data (DTMF tones) or internal MT8880 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 MT8880
0	1	0	Active: read data (i.e., receive DTMF)
0	1	1	Active: read status from MT8880
1	0	0	Inactive
1	0	1	Inactive
1	1	0	Inactive
1	1	1	Inactive

The MT8880 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 MT8880

- 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.

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

```
' Program: DIAL.BAS (Sends a string of DTMF tones via the MT8880)
' This program demonstrates how to use the MT8880 as a DTMF tone
' generator. All that's required is to initialize the MT8880 properly,
' then write the number of the desired DTMF tone to the MT8880's
' 4-bit bus.

' The symbols below are the pin numbers to which the MT8880'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 MT8880 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 MT8880's registers.

let pins = 255           ' All pins high to deselect MT8880.
let dirs = 255           ' Set up to write to MT8880 (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 MT8880. To complete
' the write, the CS line is returned high. The initialization above
' sets the MT8880 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 1 (BS1-IC) and BASIC Stamp Ver. D Program 2 Listing: DTMFDecoding

```

' Program: DTMF_RCV.BAS (Receives and display DTMF tones using the MT8880)
' This program demonstrates how to use the MT8880 as a DTMF decoder. As
' each new DTMF digit is received, it is displayed on Debug terminal.
' 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 Debug terminal.

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 polls = w2      ' Number of unsuccessful polls of DTMF.
SYMBOL sp_time = 1000 ' Print space this # of polls w/o DTMF.

' This code initializes the MT8880 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 MT8880 (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."

' In the loop below, the program checks the MT8880'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 MT8880's status flag.

again:
  high RS_p      ' Read status register.
  low CS_p      ' Activate the MT8880.
  let dt_flag = pin2      ' Store status bit 2 into flag.
  high CS_p      ' End the read.

```

BASIC Stamp 1 (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 Debug terminal.

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 MT8880.
    let dtmf = pins & %00001111 ' Strip off upper 4 bits using AND.
    high CS_p              ' Deactivate MT8880.
    gosub display          ' Display the data.
    goto again             ' Do it all again.

Display:
    Debug @ dtmf
ret:
return
```

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

```
' Program: DIAL.BS2 (BS2 sends DTMF tones via the MT8880)
' This program demonstrates how to use the MT8880 as a DTMF tone
' generator. All that's required is to initialize the MT8880 properly,
' then write the number of the desired DTMF tone to the MT8880's
' 4-bit bus.

' The symbols below are the pin numbers to which the MT8880'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 MT8880 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 MT8880's registers.

OUTL = 127      ' Pins 0-6 high to deselect MT8880.
DIRL = 127      ' Set up to write to MT8880 (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 MT8880. To complete
' the write, the CS line is returned high. The initialization above
' sets the MT8880 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 2(BS2-IC) Program Listing: DTMF Decoding

```

' Program: DTMF_RCV.BS2 (Receives/displays DTMF using MT8880 with BS2)
' This program demonstrates how to use the MT8880 as a DTMF decoder. As
' each new DTMF digit is received, it is displayed on Debug terminal. 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 Debug terminal to record the delay. When the display reaches the right-hand
' 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.
polls  var    word      ' Number of unsuccessful polls of DTMF.
sp_time con    1500     ' Print space this # of polls w/o DTMF.
n24n   con    $418D     ' Serout constant: 2400 baud inverted.

' This code initializes the MT8880 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 MT8880 (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."

' In the loop below, the program checks the MT8880'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 MT8880's status flag.

again:
    high RS              ' Read status register.
    low CS               ' Activate the MT8880.
    dt_flag = dt_det     ' Store DTMF-detected bit into flag.
    high CS              ' End the read.

```

BASIC Stamp 2 (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            ' Debug

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 MT8880.
  dtmf = INL & %00001111  ' Strip off upper 4 bits using AND.
  high CS                  ' Deactivate MT8880.
  gosub display            ' Display the data.
  goto again              ' Do it all again.

Display:
  DEBUG DEC dtmf
ret:
  return
```