MX·COM, INC.
## COMMUNICATION SEMICONDUCTORS

EV8380

USER MANUAL

Evaluation Kit for
CMX838 FRS/PMR446/GMRS
Family Radio Processor

- **Evaluation Platform for the CMX838**
  - **Reduces development time and risk**
  - **Supports rapid prototyping**
- **Convenient access to all CMX838 functions**
  - **Mic input jack**
  - **Speaker driver**
  - **Speaker output jack**
  - **Off-chip signal processing paths**
  - **SMB connectors for synthesizer RF input, charge pump output and reference oscillator input**

- **Windows™ software and PCMCIA card**
  - **PC acts as CMX838 host controller**
  - **Easily read/write CMX838 registers**
  - **Perform simple CMX838 control experiments with tab-delimited script files**
  - **Example scripts provided**
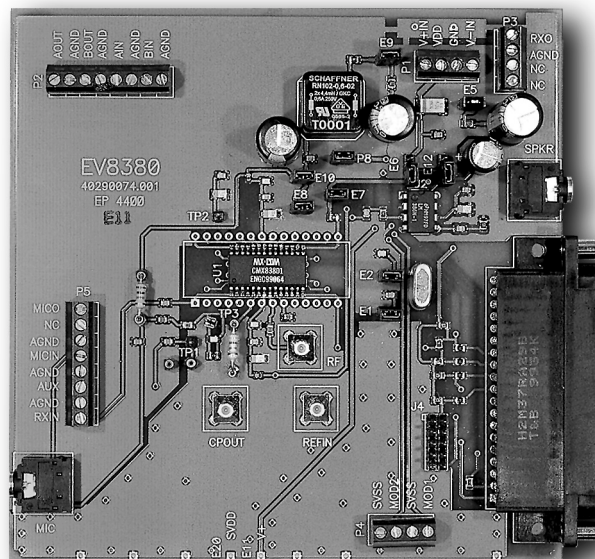
The EV8380 is an evaluation platform for the CMX838 Family Radio Processor. It speeds development time by allowing CMX838 experiments to be performed without developing embedded host firmware.

The EV8380 includes a main PC board containing a CMX838 FRS/PMR446/GMRS Family Radio Processor, a PCMCIA card and Windows™ software. The main board includes interfaces to all CMX838 sections including: RF (synthesizer RF input, charge pump output with loop filter, and synthesizer reference oscillator input), audio (mic jack, mic bias source, and speaker amplifier, speaker output jack) and host (PC via PCMCIA card and software).

Windows™ software presents a 'type and click' interface to CMX838 registers for read and write operations. It also executes tab-delimited text script files to automate command sequences.

The EV8380 supports all CMX838 baseband clock source modes including: connected baseband clock crystal (included), derived from synthesizer reference oscillator and external baseband clock source.

The EV8380 main board can be used in a standalone manner or connected to external circuits for more complete system testing. The main board is powered from an external 2.7-5.5VDC power supply.

The CMX838 data bulletin should be consulted during use of the EV8380.

CONTENTS

MX-COM, Inc. reserves the right to change specifications at any time and without notice.

1   Preliminary Information

1.1   Laboratory Equipment

The following laboratory equipment is needed to use this evaluation kit:

1) 2.7-5.5VDC Power Supply. Note that there is no on-board regulator on the EV8380, so the power supply should be set to the desired operating voltage. Please refer to the CMX838 data bulletin for information on maximum allowed voltage settings.

2) An IBM compatible PC; 66MHz '486 or better, running Microsoft Windows™ 95/98/2000. The PCMCIA card used by the EV8380 does not provide support for Windows™NT.

3) An available PCMCIA slot for the PCMCIA interface card.

1.2   Handling Precautions

Like most evaluation kits, this product is designed for use in office and laboratory environments. The following practices will help ensure its proper operation.

1.2.1   Static Protection

This product uses low power CMOS circuits which can be damaged by electrostatic discharge. Partially damaged circuits can function erroneously and lead to deceptive results. Observe ESD precautions at all times when handling this product.

1.2.2   Contents - Unpacking

Please ensure that you have received all of the items on the separate information sheet (EK8380) and notify MX-COM within 7 working days if the delivery is incomplete.

2   General Description

The EV8380 is the evaluation platform for the CMX838, MX-COM's advanced processor IC for FRS, GMRS, and PMR446 radio services. This innovative device combines much of the audio processing, subaudio processing, and RF synthesizer components that are required for a complete FRS radio. With the addition of a few external components, realistic testing of FRS systems utilizing the CMX838 can be performed.

The EV8380 is shipped from the factory pre-configured for operation:

1) With a 4.0MHz crystal as the CMX838's clock source.

2) With the CMX838 power supplied from the VDD IN pin (P1:2) and AGND (P1:3).

3   Operation

This section provides instructions for users who wish to experiment immediately with the evaluation kit. A more complete description of the kit and its operation appears later in this document.

The EV8380 offers considerable configuration flexibility, such as the ability to select different power supplies and timing sources. The EV8380 is shipped from the factory for operation at 5.0V with an external 4.0MHz crystal installed. While other configurations are possible, the following information will allow the user to operate the EV8380 with a minimum of adjustments.

3.1   Setting-Up

1) Make sure power is not applied to the EV8380.

2) Install PCMCIA software from supplied CD-ROM:

- Install "EV8380 Software VerX.X" CD-ROM. (X.X will be a version number e.g. "1.2".)

- Go into "Windows Explorer", open the CD-ROM drive selection, and select "Disk 1".

- Double click the "Setup.exe" file.

   o For Windows 95/98 systems, when prompted, select "yes" to allow modifications to the "AUTOEXEC.BAT" and "CONFIG.SYS" files.

- Once the installation of the EV8380 software is complete, reboot the computer.

- After the PC has restarted, install the PCMCIA card, without any cable attached, into an available slot in the computer.
- The operating system should automatically detect and configure the PCMCIA card.
- After the PCMCIA card installation is complete, go to "Start > Programs > MX-Com > InsCal32".
- When prompted, click "OK" to allow the detected board to be added to the Instacal configuration list. This enables the PCMCIA card for access by the provided software.
- Exit "Instacal".
- Go to "Start > Programs > MX-Com > EV8380".
- Once the "EV8380" dialog box appears, the user should select "Control > CBUS".
- At this point, the EV8380 graphical user interface (GUI) should appear on the screen.

3) Ensure the following jumpers are configured on the EV8380 as follows:
   a) E1 and E2 installed (allow the use of external crystal oscillator circuit with CMX838)
   b) E6 and E12 removed (prevents PCMCIA interface from supplying power to board)
   c) E9 installed (allows speaker driver amplifier to receive power from "VDD IN")

4) Connect a 2.7-5.5V power supply to "VDD IN" (P1:1) and "GND" (P1:3).
   a) Note: There is no on-board regulator on the EV8380, so the power supply should be set to the desired operating voltage. Please refer to the CMX838 data bulletin for information on maximum allowed voltage settings.

5) Apply power to EV8380 from its 2.7-5.5V power supply.

6) Connect one end of the supplied PCMCIA card cable to the PCMCIA card, and connect the other end to J1 of the EV8380.
   a) The PCMCIA card and ribbon cable both have white dots on them. When connecting the ribbon cable to the PCMCIA card, ensure that the cable connector side with the "white dot" is on the matching side of the PCMCIA card.

7) Connect a microphone, if desired, to the "MIC" input jack of the EV8380.

8) Connect a speaker, if desired, to the "SPKR" output jack of the EV8380.

In this configuration, the EV8380 user has the ability to have the CMX838:

1) Generate a standard or non-standard CTCSS tone, invert the signal (if desired), amplify it (if desired), and have it presented at either (or both) of the TX MOD 1 or 2 outputs.

2) Generate a 'ringing/alerting' tone and have it presented at the AUX I/O or RXOUT pin.

The setup beyond this point depends on the desired testing to be performed. The addition of appropriate signal sources/equipment will allow the user to exercise the CMX838 as follows (this is not an all-inclusive list):

1) Detect a single standard CTCSS tone.

2) Detect a user-defined subaudio tone.

3) Detect a tone from a user-defined set of tones i.e. a tone 'watch list.'

4) Perform tone cloning by rapidly scanning the received signal for the presence of any one of the 50 TIA-603 standard CTCSS tones, and providing indication of the detected tone.

5) Perform various filtering functions (low pass, high pass, band pass, pre-emphasis, de-emphasis) to allow the user to observe their respective frequency responses.

6) Apply a square wave 'melody tone' signal to the AUX I/O pin (such as might come from a host μC) and allow the CMX838 filters to create the desired melody tone output.

7) Apply an audio signal to the CMX838 (i.e. from a microphone), have the CMX838 generate a CTCSS tone, and use the digitally adjustable gain stages to individually trim the audio and sub-audio signals prior to their delivery to the TX MOD 1 & 2 modulation drivers.

8) Adjust the TX modulation output level by manipulating the digitally adjustable TX MOD 1 & 2 gain stages.

9) Configure the TX MOD 1 & 2 drivers to allow for either single-point modulation or two-point modulation.

10) Configure the TX MOD 1 or 2 drivers (or both) to invert the phase of the transmitted signal by 180° (as desired to compensate for inverting response in either an attached VCO or VC/TCXO).

11) Activate the external signal processing paths to allow for off-chip companding and/or scrambling.

12) Switch between RF channels for either of the FRS, GMRS, or PMR446 radio services.

13) Setup the CMX838 to transmit on one RF channel and receive on another.

14) Program the CMX838 to transmit on a RF channel other than that used in the FRS/GMRS/PMR446 radio services (i.e. generic synthesizer programming).

15) Connect the EV8380 to external circuits to allow complete radio system testing.

Tab-delimited text files are available in the "Examples" folder of the EV8380 software installation directory to allow quick operation of the CMX838. Operations such as melody tone generation and tone cloning can be quickly performed by the use of these tab-delimited text files. Please review Section 3.2, "CMX838 Operation", for more information on how these text files should be used.

3.2 EV8380 Graphical User Interface (GUI)

3.2.1 General Comments

The CMX838 is controlled by writing to/reading from its internal registers over its C-BUS serial host µC interface. Using the EV8380, CMX838 C-BUS transactions are entered in a simple Windows™ graphical user interface, the EV8380 GUI. The EV8380 GUI assembles C-BUS transactions from user input and causes them to be executed between the PC and the CMX838 by way of the PCMCIA card. The C-BUS transactions required to accomplish various CMX838 functionality can be entered in one of two ways:

1) Direct C-BUS register read and/or write operations.

2) Use of tab-delimited text files that contain a sequence of script commands with desired register addresses and contents.

3.2.2 Starting the EV8380 GUI

Both of the above approaches involve the use of the EV8380 graphical user interface (GUI). In order to access the GUI after the EV8380 software has been installed, select "Start > Programs > MX-Com > EV8380". A dialog box named "EV8380" should now appear on the screen. To access the GUI, the user should click on the "Control" menu of the "EV8380" dialog box, and then select the "CBUS" option. The GUI (named "C-BUS Control") should now be on the screen.

3.2.3 EV8380 GUI Buttons and Text Boxes

The buttons and text boxes in the GUI are as follows:

1) SEND FILE: This button will allow selection of a "*.txt" script file in the PC directory where the EV8380.exe file is located. When this file is found, the tab-delimited contents of that script file are translated into C-BUS register commands and sent out to the CMX838 via the PCMCIA card interface.

2) C-BUS ADDRESS: This text box contains the CMX838 register address (in hex) to which a read or write operation is about to be performed. (All of the following buttons and descriptions are referenced to the register address stored in this text box.)

3) FIRST DATA: This text box stores the first byte of data, in hex, to be written to a CMX838 register.

4) SECOND DATA: This text box stores the second byte of data, in hex, to be written to a CMX838 register (if required).

5) FIRST READ: This text box stores the first byte of data, in hex, that was read out of a CMX838 register.

6) SECOND READ: This text box stores the second byte of data, in hex, that was read out of a CMX838 register (if required).

7) WRITE 1: This button will cause one byte (stored in C-BUS ADDRESS) to be written to a CMX838 register (i.e. 'General Reset' command).

8) WRITE 2: This button will cause two bytes (stored in C-BUS ADDRESS and FIRST DATA) to be written to a CMX838 register (i.e. one address byte and one data byte).

9) WRITE 3: This button will cause three data bytes (stored in C-BUS ADDRESS, FIRST DATA, and SECOND DATA) to be written to a CMX838 register (i.e. one address byte and two data bytes).

10) READ 1: This button will cause one byte of data to be read out of a CMX838 register.

11) READ 2: This button will cause two bytes of data to be read out of a CMX838 register.

3.2.4 EV8380 GUI Examples: Direct Register Operations

As described earlier, the EV8380 GUI can be used either with tab-delimited text files or with direct register manipulations**.** Here are a few examples of how direct register operations can be performed with the EV8380 GUI:

1) To issue a 'General Reset' to the CMX838:
   a. Type in 01 (hex address corresponding to General Reset command) into C-BUS ADDRESS box.
   b. Click "WRITE 1" box to send the General Reset 'address' to the CMX838.

2) To write a byte to the SETUP register:
   a. Type in 80 (SETUP register hex address) in the C-BUS ADDRESS box.
   b. Type in desired register value in FIRST DATA box.
   c. Click "WRITE 2" box to send the address and data byte to the CMX838.

3) To write two bytes to the Synthesizer 1st IF Offset register:
   a. Type in 8D (Synthesizer 1st Offset register hex address) in the C-BUS ADDRESS box.
   b. Type first data byte into FIRST DATA box.
   c. Type second data byte into SECOND DATA box.
   d. Click "WRITE 3" box to send the address and two data bytes to the CMX838.

4) To read a byte from the SUB-AUDIO STATUS register:
   a. Type in 94 (SUB-AUDIO STATUS register hex address) in the C-BUS ADDRESS box.
   b. Click on "READ 1" box to read the contents of the desired register.

3.2.5 EV8380 GUI Examples: Using Text Script Files with the "SEND FILE" Utility

Several tab-delimited text files are provided on the EV8380 CD-ROM to serve as examples and facilitate testing. These text files can be imported with the "SEND FILE" command to rapidly exercise the CMX838's feature set without requiring the user to manually initialize and change the necessary register settings. Once the user is comfortable with the GUI and general CMX838 operation these text files can be modified or new ones may be created as desired.

To use the files provided on the EV8380 CD-ROM, the user should do the following:

- From within the EV8380 GUI, click the "SEND FILE" button.
- When the "Open" dialog box appears, click on the tab-delimited text file you wish to have loaded to the CMX838.
- Immediately after this selection, a "Save As" dialog box will appear.
  o After a file has been loaded into the CMX838, the "echoed response" of the CMX838 will be sent back to the PC. The echoed response is saved into a file for later viewing, if desired.
  o Type in the desired name of the "echoed contents" file, and click "OK".
- At this point, the CMX838 will perform the tasks that were loaded in the tab-delimited text file.

The user can create their own tab-delimited text files and use them with the "SEND FILE" utility. Any files created for this purpose should be tab-delimited and should be saved as *.txt files.

3.2.6 Creating EV8380 Text Script Files

EV8380 script files are tab-delimited text files that contain a sequence of specific C-BUS read and write operations to automate control of the CMX838 installed in the EV8380. These files may also include comments to make them easier to understand.

EV8380 GUI scripting is provided as a simple tool to support rapid development. It is not a complete programming language with conditional branches, syntax checking, bounds checking, etc. so the users must ensure their scripts are accurate.

3.2.6.1   Script Command Types and Syntax

Script commands are of two types: C-BUS operations and script directive. Their syntax is shown in the below.

Script Command Type	Syntax	Notes
C-BUS	command<tab>register<tab>[argument]	• The selected C-BUS register address determines whether an argument is required and its size. • Both C-BUS register address and argument (if any) must be in hexadecimal form without any preceding characters such as "$" or "0x".
Pause	command<tab>argument	• Argument must be in decimal form without any preceding characters such as "0d".

<div align="center">**Table 1, Script Command Types**</div>

Comments are also supported by using "//" to indicate the start of a comment. Text following "//" in any line is treated as a comment and ignored.

3.2.6.2   C-BUS Type Commands

Valid values for the C-BUS type script command fields are described in the following tables.

Script command	Description
W1	single byte C-BUS write; single byte is a C-BUS address
W2	two byte C-BUS write; first byte is a C-BUS address, second byte is an 8 bit C-BUS data argument
W3	three byte C-BUS write; first byte is a C-BUS address, second and third bytes are a 16 bit C-BUS data argument
R2	single byte C-BUS read: single byte is the address of the 8 bit C-BUS register being read
R3	double byte C-BUS read: single byte is the address of the 16 bit C-BUS register being read

<div align="center">**Table 2, C-BUS Type Commands**</div>

NOTES:

1. Both C-BUS register address and argument (if any) must be in hexadecimal form without any preceding characters such as "$" or "0x".

2. Each C-BUS register supports a single argument size and operation type (read or write). It is an error to use a script command that does not match a register's size e.g. using a W1 script command to write to a 2 byte C-BUS register.

CMX838 Register Name	Register Address	CMX838 function area	Appropriate C-BUS Script Command
GENERAL RESET	01	system	W1
AUDIO POWER AND BANDWIDTH CONTROL	83	system	W2
SYNTHESIZER BASEBAND CLK CONTROL	89	system	W2
SETUP REGISTER	80	audio	W2
AUDIO CONTROL	81	audio	W2
SYNTHESIZER GENERAL CONTROL	8A	synthesizer	W2
SYNTHESIZER CHANNEL SELECT	8B	synthesizer	W2
SYNTHESIZER 1ST IF OFFSET	8D	synthesizer	W3
SYNTHESIZER STATUS	8C	synthesizer	R2
SUBAUDIO GENERAL CONTROL	93	subaudio	W2
SUBAUDIO STATUS	94	subaudio	R2
8 BIT SUBAUDIO TASK DATA	95	subaudio	W2
16 BIT SUBAUDIO TASK DATA	8E	subaudio	W3
SUBAUDIO ANALOG LEVEL CONTROL	97	subaudio	W3
TX MOD 1&2 CONTROL	88	output	W3
RX VOLUME CONTROL	82	output	W2

Table 3, CMX838 Registers and Valid C-BUS Script Commands

3.2.6.3   Pause Type Commands

Pauses are supported to allow the installed CMX838 to 'dwell' in a configured state for an extended period. For example, the CMX838 may be configured to emit an audio tone and script pause commands may be used to determine the length of the tone before it is turned off.

One pause command is of a fixed duration. The other has a maximum duration but is terminated when the CMX838 asserts its hardware interrupt.

Script command	Description	NOTE
T	Pause for the specified number of 10uS 'ticks' written in decimal form e.g. an argument value of 6500 yields a pause duration of 65mS.	The maximum valid argument value is 6500.
I	Pause for the specified number of 10uS 'ticks' in decimal form but terminate the pause immediately upon receiving a CMX838 interrupt.	The maximum valid argument value is 6500.

Table 4, Pause Type Commands

NOTES:

1.   Argument must be in decimal form without any preceding characters such as "0d".

2.   The "I" script command is the letter "i", not the letter "L".

3.2.6.4   Script Example

A short segment of sample script follows. Note there are tabs between each field that are not visible in this document.

```
// general reset
W1     01
W2     89     40       // clock control base band from XTAL 4MHz
T      6500            // pause 65mS
T      6500            // pause 65mS
T      6500            // pause 65mS
W2     80     DE       // TX mode, Aux I/O Selected, HPF Out to LPF/Lim In
W2     81     61       // LPF + -7.5 dB Gain
W2     82     24       // Bypass de-emphasis, Rx out Gain = -28.5 dB
W2     83     56       // audio lowest current mode with Narrow Band LPF
W3     88     503F     // Audio & Sub-audio on  Mod2 0dB, Audio on Mod1 7.5dB
W3     97     1010     // Sub Audio Analog Levels
```

3.3 EV8380 External Connections

Primary CMX838 C-BUS interface signals are exposed at a connector.

$\overline{\text{CS}}$	J1:12	J4:10	$\overline{\text{CS 2}}$
$\overline{\text{IRQ}}$	J1: 2	J4: 8	$\overline{\text{IRQ 2}}$
Reply Data	J1: 1	J4: 6	Reply Data 2
Command Data	J1:11	J4: 4	Command Data 2
Serial Clock	J1:10	J4: 2	Serial Clock 2

Second set of C-BUS interface signals is exposed at a connector to support user implemented daisy-chained board configurations controlled by a single PCMCIA card.
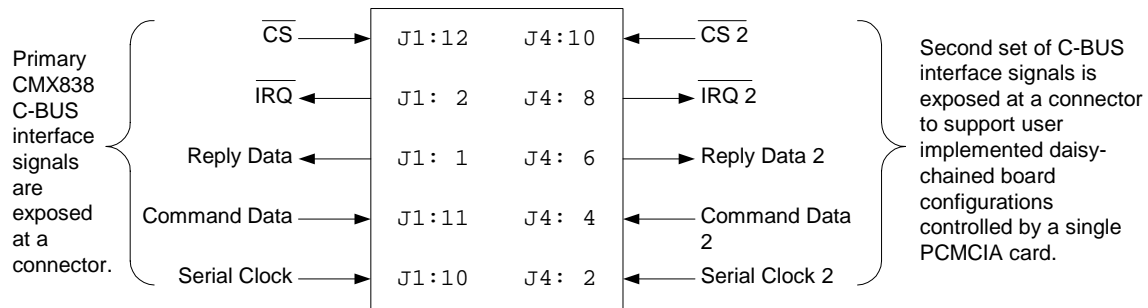
Figure 1: EV8380 PC and C-BUS Connections

Rx Input	P5:1,2	P2:1,2	A_{OUT}
Rx Output	P3:1,2	P2:3,4	B_{OUT}
TX MOD1 Output	P4:1,2	P2:5,6	A_{IN}
TX MOD2 Output	P4:3,4	P2:7,8	B_{IN}
Mic input	J2	J3	Speaker Output
AUX I/O	P5:3,4		

To support user-added external companding and scrambling processing.
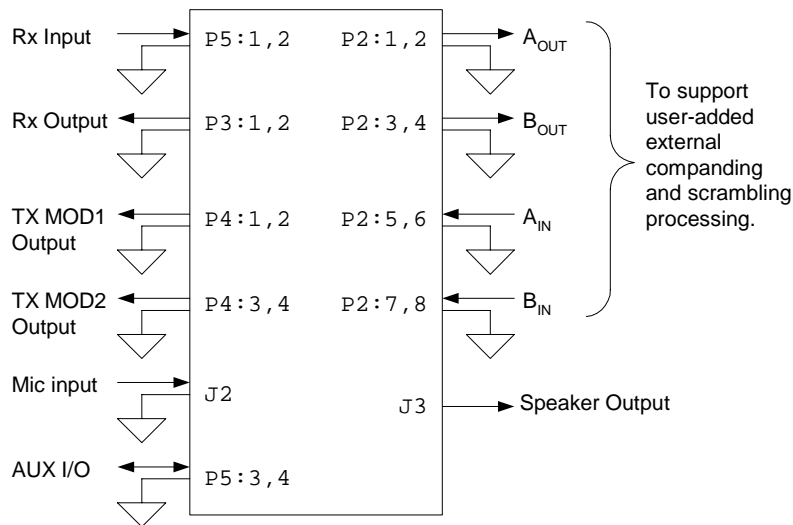
Figure 2: EV8380 Baseband Signal Connections

RF input from external VCO	SM2
Input from external synthesizer reference oscillator e.g. TCXO.	SM1
SM3	Charge pump output current to external VCO loop filter

Figure 3: EV8380 RF Connections

4   General Notes

4.1   Baseband Clock Generation

The CMX838 offers considerable flexibility in the choice of clock sources. Baseband circuits on the CMX838 can receive their clock from any of the system RF reference oscillator (i.e. VC/TCXO), a dedicated crystal oscillator or an externally applied clock signal of suitable frequency and amplitude.

4.1.1   Using the RF Reference Oscillator as the CMX838 Clock Source

The desired mode of CMX838 operation involves the derivation of the baseband clocks from an external RF reference oscillator. When the device is operated in this manner, all CMX838 circuits (RF and baseband) are synchronized to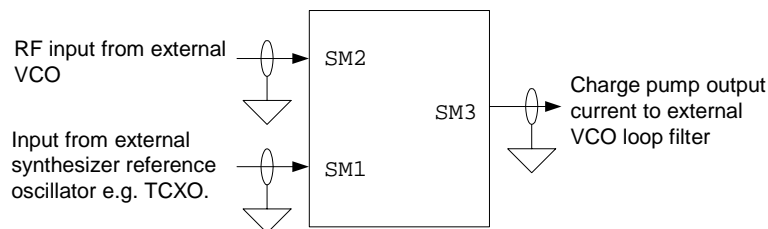 the same clock signal**.** Eight different RF reference oscillator frequencies, from 4.0MHz to 21.25MHz, are available for CMX838 baseband clock generation (please refer to CMX838 data bulletin for details on available frequencies).

To operate the CMX838 with an RF reference oscillator as its baseband clock source:

    1)   Connect an external RF reference oscillator to connector SM3 on the EV8380 board**.**

Please note that the reference oscillator frequencies compatible with the CMX838 are listed in the CMX838 data bulletin.

    2)   Write a data byte to the SYNTHESIZER BASEBAND CLOCK CONTROL ($89) register to configure the CMX838 clock generation circuits.

For example, if the external RF reference oscillator operates at 12.8MHz, a byte of $90 will enable baseband clock generation from the external 12.8MHz reference oscillator.

When the CMX838 is operated with an RF reference oscillator as its baseband clock source, the normal mode of operation for the CMX838 on-chip crystal amplifier is 'disabled'.

4.1.2   Using External Sources as the CMX838 Clock Source

If it is not desirable to use the RF reference oscillator as the device clock source, the CMX838 can develop its baseband clock from an external clock source; a crystal oscillator or a system clock signal can be used in this situation**.** Eight different frequencies, from 4.0MHz to 24.0MHz, are available for CMX838 baseband clock generation when operated in this manner (please refer to CMX838 data bulletin for details on available frequencies).

4.1.2.1   Operating the CMX838 from a Crystal Oscillator

To operate the CMX838 with an external crystal oscillator:

    1)   Connect crystal of desired frequency to the 'X1' socket on the EV8380.

        a.   The EV8380 board is shipped from the factory with a 4.0MHz crystal.

        b.   If a different crystal frequency is desired, please consult the CMX838 data bulletin for other supported crystal frequency choices.

        c.   Please consult the CMX838 data bulletin for recommendations concerning crystal type, drive levels, etc.

    2)   Ensure jumpers E1 and E2 are installed.

    3)   Write a byte to the SYNTHESIZER BASEBAND CLOCK CONTROL ($89) register to configure the CMX838 clock generation circuits.

        a.   For example, if the crystal operates at 4.0MHz, a byte of $40 will enable baseband clock generation from the 4.0MHz crystal oscillator.

4.1.2.2   Operating the CMX838 from a System Clock

To operate the CMX838 from a system clock signal e.g. a host microcontroller's clock:

1) Please consult the CMX838 data bulletin for appropriate system clock frequency choices.

2) Ensure jumpers E1 and E2 are removed.

3) Connect the system clock signal to the XTALIN pin of the CMX838 by way the jumper E1 pins.

   a. Please consult the CMX838 data bulletin for recommendations on drive level.

4) Write a byte to the SYNTHESIZER BASEBAND CLOCK CONTROL ($89) register to configure the CMX838 clock generation circuits.

   a. For example, if the system clock operates at 4.0MHz, a byte of $40 will enable baseband clock generation from the 4.0MHz system clock signal.

4.2   Subaudio Processing

The CMX838 offers tremendous flexibility in the ability to process subaudio tones:

1) Single tone generation (standard and user-defined CTCSS tones)

2) Single tone detection (standard and user-defined CTCSS tones)

3) For tone detection, the formation of a tone 'watch list' is possible, with the list comprised of as many as 50 different standard and non-standard CTCSS tones that are effectively detected in parallel.

4) Multiple tone detection; the CMX838 can be configured to search for all of the standard CTCSS tones and/or a user programmed tone.

5) Custom tone detection: one of the CMX838 parallel CTCSS tone decoders is programmable and allows configuration of an arbitrary tone

Flexibility is also offered by way of adjustable filter settings and gain stages in the CMX838 subaudio signal path.

The CMX838 supports individual enabling/disabling of tones to allow users to configure their own tone sets with each enabled tone effectively detected in parallel with all other enabled tones. The CMX838 also offers two pre-configured sets of CTCSS tones so a full set may be activated in one step. One set includes all 39 TIA-603-A tones. The second set includes 50 commonly encountered CTCSS tones (including all 39 tones specified in the TIA-603-A standard).

Once activated, a pre-configured tone set may be edited. For example, the set may be configured to enable all 39 TIA-603-A tones in one step and subsequently edited to remove or add more tones.

In general, to generate or detect one of these tones, the hex value for the tone of interest should be loaded into the SUBAUDIO TASK 8 BIT DATA ($95) register. In typical applications, once the tone signaling processor is initialized, its primary behavior (CTCSS encode and decode) is steered by the $\overline{TX/RX}$ bit of the SETUP Register ($80) so tone sets do not have to be repeatedly configured.

CTCSS Tone	Tone Index (hex)	CTCSS Tone	Tone Index (hex)
67.0	01	**156.7**	1A
69.3	02	**159.8**	1B
71.9	03	**162.2**	1C
74.4	04	**165.5**	1D
77.0	05	**167.9**	1E
79.7	06	**171.3**	1F
82.5	07	**173.8**	20
85.4	08	**177.3**	21
88.5	09	**179.9**	22
91.5	0A	**183.5**	23
94.8	0B	**186.2**	24
97.4	0C	**189.9**	25
100.0	0D	**192.8**	26
103.5	0E	**196.6**	27
107.2	0F	**199.5**	28
110.9	10	**203.5**	29
114.8	11	**206.5**	2A
118.8	12	**210.7**	2B
123.0	13	**218.1**	2C
127.3	14	**225.7**	2D
131.8	15	**229.1**	2E
136.5	16	**233.6**	2F
141.3	17	**241.8**	30
146.2	18	**250.3**	31
151.4	19	**254.1**	32
$33 = user programmable mode			
$3E = all TIA-603 tones selected			
$3F = all tones selected (user programming and TIA-603)			

Table 2: CMX838 Standard CTCSS Tone Table

The following information describes important timing aspects of the various operating modes of the tone signaling processor:

1) CMX838 clocks must be properly set up and running before any tasks can be issued to the tone signaling processor.

2) During normal operation, tasks can be issued to the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register at a maximum rate of one task every 250μs.

   a. If the tone signaling processor is in its "Fast Initialization" mode, tasks can be issued to the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register at a maximum rate of one task every 50μs.

3) For tasks that involve "argument data" (such as information corresponding to a particular tone), the argument data must be loaded into the argument register first. After the argument data has been loaded, the appropriate task should then be loaded into the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.

   a) Once "argument data" has been passed to the argument register, no delay is required before writing the corresponding task to the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.

4) After a task has been loaded into the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register, a minimum wait time of 250μs must be observed before modifying the contents of any argument data register.

5) If the CMX838 is placed in its "Fast Initialization" mode, either directly or indirectly (through the issuance of a "Soft Reset" task):

   a. An initial delay of 250μs must be observed before the next task can be written to the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.

   b. Once this initial 250μs delay has been observed, tasks can be issued to the device at a maximum rate of one task per 50μs. Any "argument data" required for the next task may be loaded into the argument register during this initial 250μs delay.

   c. While in the "Fast Initialization" mode, for tasks other than "Soft Reset" or "Fast Initialization", ensure that the required argument registers are not updated for at least 50μs after the corresponding task has been issued.

6) To determine the data byte that should be loaded into the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register, the power control settings and the IRQ settings should be logically "OR-ed" with the desired task field. The resulting byte should then be loaded into the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.

4.2.1   Power-Up Initialization Sequence for CMX838

The tone signaling processor of the CMX838 should be fully initialized after the chip is powered up. The first time the tone signaling processor is enabled after power is applied to the CMX838, a byte should be written to the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register with the 'task field' set to $F. This setting will clear the configuration memory and place the tone signaling processor in a "Fast Initialization" mode. When the CMX838 tone signaling processor is in this mode, the tone encoders and decoders do not run.

To resume normal operation, after all initialization is completed, a byte with the 'task field' set to $0 should be written to the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.

The following sequence should be utilized when applying power to the CMX838 (i.e. after turning on the EV8380). **This sequence only needs to be executed upon "cold" power-up**. For example, issuance of a 'General Reset' command after power has been applied does not require the execution of this initialization routine:

1) Apply power to the EV8380.
2) Issue 'General Reset' (single C-BUS write of $01).
3) Write $4F to the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.

   b) This applies a 'soft reset' to the subaudio section of the CMX838, clearing its configuration memory.

4) Wait 250μs for the 'Soft Reset' task to complete.
5) At the end of the 250μs delay, the CTCSS processor will automatically be in the "Fast Initialization" mode. The tone encoders and decoders do not run while in this special mode. Please review section 4.2.1 and the CMX838 data bulletin for more information on the "Fast Initialization" mode.
6) Configuration of the CMX838 tone signaling processor can now be performed as desired.
7) **Once the configuration is complete, a task of $0 should be written to the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register in order to resume normal operation.**

The initialization procedure is now complete.

After these steps have been performed, other registers should be set as necessary to achieve desired operation (i.e. audio path switching, gain levels, baseband clock generation, etc).

5   Programming Examples

5.1   Device Setup

The following sections provide specific instructions on how to program the CMX838 to perform some of its functions but, in a typical application, the CMX838 will also require some "setup" instructions (i.e. Tx or Rx, amplifier gains, powersave settings, etc.). Example transmit and receive setup sequences follow. (More examples can be found in the example tab-delimited files provided on the EV8380 CD-ROM.) The actual initialization may vary depending on the specific application.

TX SETUP SEQUENCE

1) Write $40 to SYNTHESIZER BASEBAND CLOCK CONTROL ($89) register:
   a. Baseband clock source supplied from external crystal.
   b. Bits 5-2 are "don't care", as CMX838 RF synthesizer clock is not used for baseband clock generation.
   c. 4.0MHz crystal oscillator used for baseband clock generation.

2) Write $FF to SETUP ($80) register:
   a. Tx enabled.
   b. Microphone amplifier output signal passed through "Audio Input 1 Select" switch to pre-emphasis/LPF network.
   c. Audio signal passed from HPF network output to LPF/limiter input.
   d. Audio signal is not passed to de-emphasis network

3) Write $1F to AUDIO CONTROL ($81) register:
   a. First stage of audio filtering configured for pre-emphasis response.
   b. Audio signal is passed through HPF.
   c. Audio level (gain stage between pre-emphasis/LPF and HPF) set to +7.5dB.

4) Write $17 to RX AUDIO LEVEL CONTROL ($82) register:
   a. Audio signal is passed through deviation limiter and post-deviation limiter LPF.
   b. De-emphasis network is not bypassed.
   c. Volume control set to 0.0dB.

5) Write $54 to AUDIO POWER & BANDWIDTH CONTROL ($83) register:
   a. Low power operation for modulation digitally controlled amplifiers (DCA) and microphone amplifier.
   b. Low power operation for audio filters, deviation limiter, and audio level DCA.
   c. Low power operation for de-emphasis network and Rx Audio Out DCA.
   d. Post-deviation limiter LPF configured for wideband output.

6) Write $5035 to TX MOD 1 & 2 CONTROL ($88) register:
   a. Subaudio signal presented on MOD2 output.
   b. MOD2 output level set to 0.0dB.
   c. Audio tone presented on MOD1 output.
   d. MOD1 output level set to +2.5dB.

7) Write $44 to SYNTHESIZER GENERAL CONTROL ($8A) register:
   a. Synthesizer is enabled.
   b. Lock detect IRQ is masked
   c. Magnitude of charge pump current is 40* I_{SET}.
   d. Positive slope for VCO gain
   e. FRS radio service selected

8) Write $01 to SYNTHESIZER CHANNEL SELECT ($8B) register:
   a. Bits 7 and 6 set to zero.
   b. Lock detect comparison window set to +/- 20ns.
   c. FRS channel 1 selected.

<u>RX SETUP SEQUENCE:</u>

1) Write $40 to SYNTHESIZER BASEBAND CLOCK CONTROL ($89) register:
   a. Baseband clock source supplied from external crystal.
   b. Bits 5-3 are "don't care", as CMX838 RF synthesizer clock is not used for baseband clock generation.
   c. 4.0MHz crystal oscillator used for baseband clock generation.

9) Write $3F to SETUP ($80) register:
   a. Rx Enabled
   b. RXIN selected as input signal source
   c. HPF output passed to deviation limiter input
   d. Rx O/P enabled (this bit should be set to 1 for CMX838 engineering samples)

10) Write $1F to AUDIO CONTROL ($81) register:
   a. First stage of audio filtering configured for pre-emphasis response
   b. Audio signal routed through HPF
   c. Audio level (gain stage between pre-emphasis/LPF and HPF) set to +7.5dB

11) Write $99 to RX AUDIO LEVEL CONTROL ($82) register:
   a. Deviation limiter is bypassed
   b. Audio signal is passed through post-deviation limiter LPF & de-emphasis network
   c. Rx volume control set to +3.0dB

12) Write $54 to AUDIO POWER AND BANDWIDTH CONTROL ($83) register:
   a. Modulation digitally controlled amplifiers (DCA) and microphone amplifier configured for low power consumption
   b. Audio filters, deviation limiter, and audio level DCA configured for low power consumption
   c. De-emphasis network and Rx Audio Out DCA configured for low power consumption
   d. Post-deviation limiter LPF set to wide setting
   e. Bit 0 is unused

13) Write $0000 to TX MOD 1 & 2 CONTROL ($88) register:
   a. Tx Mod 2 output set to Vbias
   b. Tx Mod 2 output gain set to off
   c. Tx Mod 1 output set to Vbias
   d. Tx Mod 1 output gain set to off

14) Write $44 to SYNTHESIZER GENERAL CONTROL ($8A) register:
   a. Synthesizer is enabled
   b. Lock detect IRQ is masked
   c. Magnitude of charge pump current is 40*Iset
   d. Positive VCO gain slope
   e. FRS channels selected

15) Write $01 to SYNTHESIZER CHANNEL SELECT ($8B) register:
   a. Bits 7 and 6 should always be set to zero
   b. Lock detect comparison window set to +/- 20ns
   c. FRS channel 1 selected

5.2   Tone Signaling Processor Examples

NOTE that before any tasks can be issued to the tone signaling processor, the baseband clocks must first be setup properly and running.

5.2.1   How to Generate a Standard CTCSS Tone

To select a pre-programmed subaudio tone, the user must first load the tone index (i.e. argument data) into the argument data register and then issue task # 8 (select subaudio tone from pre-programmed list) to the tone signaling processor.

NOTE: To determine the value to load in the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register, the power control and IRQ control bits should be logically "OR-ed" with the desired task field.

For example, to set up the CMX838 to transmit an 114.8Hz tone (standard CTCSS tone), the required C-BUS sequence would be:

1. Write $11 to SUBAUDIO TASK DATA ($95) register.
2. Write $48 to SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.

5.2.2   How to Generate A User-Defined TX Subaudio Tone

To program the CMX838 to generate a user-defined subaudio tone, the user must first calculate the argument data according to the following formula:

$$Argument = INT\left(0.5 + \frac{36 \cdot 65536 \cdot f}{100000}\right)$$

where f is the tone to be transmitted. Load the binary equivalent of the argument data into 16 BIT SUBAUDIO TASK DATA register ($8E), then issue task 9 (program user-defined subaudio tone) to the tone signaling processor.

NOTE: To determine the value to load in the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register, the power control and IRQ control bits should be logically "OR-ed" with the desired task field.

For example, to set up the processor to generate a 65 Hz tone, the required C-BUS sequence would be

1. Write $05FE to 16 BIT SUBAUDIO TASK DATA register ($8E).
2. Write $49 to SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.

5.2.3   How to Generate an Audio Tone at the AUX I/O Pin

The CMX838 audio tone generator produces ringing/alert square wave tones at user defined frequencies. It can also filter those audio signals to make them more sinusoidal.

To use a generated audio tone signal the AUX I/O pin must be configured to internally connect to the audio tone generator output. (When the AUX I/O pin is configured in this manner, external connections to the AUX I/O pin may load and affect the audio tone generator output.)

If desired, the AUX I/O pin may also be internally switched to drive the CMX838 audio processing path to filter and pass an audio tone to both loudspeaker and TXMOD1 and TXMOD2 output pins. This allows the CMX838 to internally generate an audio tone and internally connect it to the audio processing path, using the AUX I/O pin to do so. If desired, the generated audio tone may be driven off-chip via the AUX I/O pin without also internally connecting it to the CMX838 audio processing path.

5.2.3.1 Set the Audio Tone Frequency

To program the CMX838 to generate a user-defined audio tone the user must first compute the "argument data" according to the following formula:

$$Argument = INT\left(0.5 + \frac{6 \cdot 65536 \cdot f}{100000}\right)$$

where f is the tone to be transmitted. Load the binary equivalent of the argument data into 16 BIT SUBAUDIO TASK DATA register ($8E), then issue task 10 (program audio frequency ringing tone).

NOTE: To determine the value to load in the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register, the power control and IRQ control bits should be logically "OR-ed" with the desired task field.

For example, to transmit a 620Hz tone, the required C-BUS sequence would be

1. Write $0986 to 16 BIT SUBAUDIO TASK DATA register ($8E).
2. Write $4A to SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.

5.2.3.2 Internally Connect the AUX I/O Pin to the Audio Tone Generator

To internally connect the audio tone generator output to the AUX I/O pin, argument data for the AUX I/O pin configuration must be loaded into the argument register, and task 14 (configure aux pin as output) should then be sent to the tone signaling processor.

The argument data for AUX I/O pin configuration has the following format in the SUBAUDIO TASK DATA ($95) register:

Bit 7	Bit 6-3	Bit 2-0 (These bits are Don't Care if Bit 7 is a logic 0)			
1=enable aux pin as output 0=enable aux pin as input	Don't care	**Bit 2**	**Bit 1**	**Bit 0**	**AUX I/O pin output signal**
		1	0	0	RX Decode Status bit
		1	0	1	Audio Tone
		1	1	0	Output logic 0
		1	1	1	Output logic 1

NOTE: To determine the value to load in the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register, the power control and IRQ control bits should be logically "OR-ed" with the desired task field.

For example, to have the CMX838 produce a 620Hz ringing tone and present the tone on the AUX I/O pin, set up the audio tone with task 10 and then enable the AUX I/O pin output with task 14.

1. Write $0986 to 16 BIT SUBAUDIO TASK DATA ($8E) register.
2. Write $4A to SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.
3. Wait 250µs.
4. Write $85 to SUBAUDIO TASK DATA ($95) register.
5. Write $4E to SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.

If the CMX838 is already in the Tx mode, the newly programmed tone will immediately begin to be transmitted.

Note that once the audio tone generator is enabled, the frequency can be changed by reissuing task 10.

5.2.4 How to Detect a Standard CTCSS Tone

To program the CMX838 to detect a standard (TIA-603) CTCSS tone, the user must first load the argument data corresponding to the tone of interest into the argument data register, then issue task 4 (enable/disable tone detector) to the tone signaling processor. (NOTE: These steps can be repeated as needed to configure a "tone watch list" but a special mode can be used to quickly enable/disable multiple detectors; see section 5.2.5 for more information.)

The argument data has the following format in the SUBAUDIO TASK DATA ($95) register:

Bit 7	Bit 6	Bits 5-0
1=enable 0=disable	Don't care	Tone detector index number (1-51) Additionally using index 63 can enable or disable all detectors while issuing just one task. Enabling index 62 enables detection of all TIA-603 Tones. There is no single command to disable just the TIA-603 Tone Detectors – instead use index 63 to disable all detectors.

NOTE: To determine the value to load in the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register, the power control and IRQ control bits should be logically "OR-ed" with the desired task field.

For example, to configure the CMX838 to detect a 67Hz tone:

1. Write $81 to SUBAUDIO TASK DATA ($95) register.
2. Write $64 to SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.

5.2.5   How to Quickly Enable/Disable Multiple Detectors

Task 13 places the tone signaling processor in a mode that allows multiple detectors to be quickly configured. Just as for task 12, neither the tone encoders nor the tone decoders operate in this mode. The argument data for this task is defined as for task 4 (enable/disable tone detector).

When task 13 is executed, the tone signaling processor reverts to "Fast Initialization Mode" when any other task is issued. To return to normal operation, task 0 (normal operation) must be issued to the tone signaling processor.

NOTE: When enabling or disabling tone detectors with task 13, the argument data in the SUBAUDIO TASK DATA ($95) register has the following format:

Bit 7	Bit 6	Bits 5-0
1=enable 0=disable	Don't care	Tone detector index number (1-51) Additionally using index 63 can enable or disable all detectors while issuing just one task. Enabling index 62 enables detection of all TIA-603 Tones. There is no single command to disable just the TIA-603 Tone Detectors – instead use index 63 to disable all detectors.

NOTE: To determine the value to load in the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register, the power control and IRQ control bits should be logically "OR-ed" with the desired task field.

The following example shows how to enable tone detectors 1, 7, 10, 12, 18, and 20 only. (NOTE: Multiple calls to Task 4 can also be used to enable or disable multiple detectors but would, however, require more C-BUS transactions and a waiting period of 250μs after **each** task 4 call. The use of multiple calls to task 4 would allow the tone decoders to continue to run, however.)

To disable all tone detectors and enter "quickly enable multiple detectors mode":

1. Write $3F to SUBAUDIO TASK DATA ($95) register.
2. Write $4D to SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.
3. Wait 250μs to ensure the CMX838 performs Task 13

4. Write the following values to the SUBAUDIO TASK DATA ($95) register.

     a. $81   (to enable tone detector 1 (67.0 Hz))   value = 0x80 | 0x01, then wait 50µs minimum

     b. $87   (to enable tone detector 7 (82.5 Hz)), then wait 50µs minimum

     c. $8A   (to enable tone detector 10 (91.5 Hz)), then wait 50µs minimum

     d. $8C   (to enable tone detector 12 (97.4 Hz), then wait 50µs minimum

     e. $92   (to enable tone detector 18 (118.8 Hz)), then wait 50µs minimum

     f. $94   (to enable tone detector 20 (127.3 Hz)), then wait 50µs minimum

5. Write $40 to SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.

     a.   This step includes power control, IRQ control, and the issuance of task 0 to place the device back in normal running mode

5.2.6   How to Detect a User-Defined Subaudio Tone

Task 5 (program user-defined subaudio tone) effectively programs the center frequency of user programmable detector 51.

To detect the presence of a user-defined subaudio tone, the user must first calculate the argument data corresponding to the desired tone according to the following formulas:

$$N = INT\left(\frac{96 \cdot 511 \cdot f}{100000}\right)$$

$$R = INT\left(0.5 + 511 - \frac{100000 \cdot N}{96 \cdot f}\right)$$

$$Argument = N \cdot 64 + R$$

The argument data for 65 Hz would be 31*64+14 = 1998 = $07CE.

The argument data is loaded into the argument data register, and task 5 is then issued to the tone signaling processor.

The programmed center frequency can be back calculated by:

$$f = \frac{100000 \cdot N}{96(511 - R)}$$

In the above example, the actual center frequency would be 64.97 Hz.

NOTE: To determine the value to load in the SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register, the power control and IRQ control bits should be logically "OR-ed" with the desired task field.

For example, a C-BUS sequence to enable detection of a 65Hz tone (user-defined) would be:

1. Write $07CE to 16 BIT SUBAUDIO TASK DATA ($8E) register.

     a.   This is the required argument data for 65Hz tone.

2. Write $45 to SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.

     a.   Task 5 command (No IRQs enabled)

3. Wait 250µs

4. Write $3F to SUBAUDIO TASK DATA ($95) register.

     a.   Task 4 argument data to disable all decoders.

5. Write $44 to SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register

     a.   Task 4 command (No IRQs enabled

6. Wait 250µs

7. Write $B3 to SUBAUDIO TASK DATA ($95) register.

     a.   Task 4 argument data to enable decoder 51 (The user-definable one)

8. Write $64 to SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.

     a.   Task 4 command (with IRQs enabled)

9.   Wait 250μs

10. Write $60 to SUBAUDIO PROCESSOR GENERAL CONTROL ($93) register.

   a.   Task 0 command (to place device normal run mode with IRQs enabled)
        NOTE: This command is not required if the device was already in normal run mode

5.3   Audio Filtering Examples

5.3.1   How to Generate "Melody Tone" And Process It with the CMX838

A common feature in FRS-type radios is the generation of 'melody tones' for use as call alerts. The CMX838 can generate audio tone square wave signals and filter them to make them suitable to drive the RXOUT (to an external speaker path) and TXMOD1 and TXMOD2 (to external RF transmitter paths) output pins, as desired. The CMX838 eliminates the need for external filtering. Simultaneous output to all of RXOUT, TXMOD1 and TXMOD2 pins supports playing a melody tone at the local loudspeaker while simultaneously transmitting it via RF.

For example, to configure the CMX838 to generate an audio "melody tone" and perform signal processing on it, a typical C-BUS sequence could be:

1.   Configure other registers as necessary for proper baseband clock generation, signal path routing, amplification, etc.

2.   Configure the tone signaling processor to generate the audio tone frequency of interest as described in sections 5.2.3.1, Set the Audio Tone Frequency, pg. 18, and 5.2.3.2, Internally Connect the AUX I/O Pin to the Audio Tone Generator, pg. 18.

3.   Write $DE to SETUP ($5F) register:

   a. Tx disabled (this is a 'don't care')

   b.   Audio signal input supplied from AUX I/O pin

   c.   Audio signal passed from HPF output to Limiter/LPF input

   d. Bit 0 is unused and must be set to logic 1.

4.   Write $50 to AUDIO CONTROL ($81) register:

   a.   First stage of audio filtering configured as LPF

   b.   Audio signal routed through HPF

   c.   Audio level (gain stage between pre-emphasis network and HPF) set to 0.0dB

5.   Write $97 to RX AUDIO LEVEL CONTROL ($82) register:

   a.   Deviation limiter bypassed.

   b.   Post-deviation limiter LPF not bypassed

   c.   De-emphasis network is not bypassed

   d.   Rx Audio Level (gain stage downstream of de-emphasis network) set to 0.0dB

6.   Connect speaker to connector J3 of the EV8380.

The desired tone should be audible at the speaker.

5.3.2   How to Use Pre-Emphasis/De-Emphasis

For the transmit path, pre-emphasis can be realized either through an appropriate combination of external components between the microphone and the CMX838 microphone input pin, or by activating the CMX838 internal pre-emphasis filter. The CMX838 internal pre-emphasis filter can be activated by setting bits 7 and 6 of the AUDIO CONTROL register ($81) to 0. **Please note that if pre-emphasis is performed externally, the internal pre-emphasis filter should not be utilized in the signal flow path.**

De-emphasis can be realized by activating the CMX838 internal de-emphasis filter; this is done by setting bit 5 of the RX VOLUME CONTROL register ($82) to 0.

5.4   RF Programming Examples

5.4.1   How to Change an FRS RF Channel

To change the RF channel (for FRS, GMRS, or PMR446), write a byte to the SYNTHESIZER CHANNEL SELECT ($8B) register that contains the desired channel number. If the device is in Tx mode, the new channel carrier frequency will automatically be synthesized by the CMX838 RF synthesizer.

5.4.2   How to Set the IF Offset For A FRS RX Channel

In the instances where it is desired to transmit on one channel but receive on another, the SYNTHESIZER 1ST IF OFFSET ($8D) register allows for a "high side" or "low side" IF offset to be programmed into the CMX838.

For example, suppose the CMX838-based FRS radio is transmitting on FRS channel 14 (467.7125MHz), but the radio user wishes to receive on FRS channel 1 (462.5625MHz). To accomplish this synthesizer manipulation, a value can be loaded into the SYNTHESIZER 1ST IF OFFSET ($8D) register that will automatically produce the desired offset when the CMX838 is switched from transmit mode to receive mode. For this example, the following calculations allow the determination of the appropriate setting for the SYNTHESIZER 1ST IF OFFSET ($8D) register:

1.   467.7125MHz – 462.5625MHz = 5.15MHz   Frequency IF shift required between FRS channels 14 & 1

2.   (5.15MHz / 12.5kHz) = 412d = $019C   Number of FRS channels (12.5kHz wide) in the required shift

3. $019C unsigned = $FE64 signed   Frequency shift is negative, so load $FE64 (the first four bytes of the signed version of $019C) into SYNTHESIZER 1st OFFSET ($8D) register

If the IF shift were a "high side" shift of 21.4MHz:

(21.4MHz / 12.5kHz) = 1712d = $06B0 unsigned

This means that for a "high side" IF shift of 21.4MHz, a value of $06B0 should be loaded into the SYNTHESIZER 1ST IF OFFSET ($8D) register.

Additionally, if the PMR446 service (6.25kHz channels) is being used and a "low side" IF shift of 45MHz is required:

(45MHz / 6.25kHz) = 7200d = $1C20 unsigned = $E3E0 signed

So, a value of $E3E0 should be loaded into the SYNTHESIZER 1ST IF OFFSET ($8D) register to accomplish a "low side" IF shift of 45MHz with the PMR446 radio service.

5.5   Modulation Section Programming Example

5.5.1   How to Manipulate the Tx MOD 1 & 2 Outputs

The TX MOD 1 & 2 outputs were designed with flexibility in mind. These output modulation drivers can independently be programmed to present either (1) Vbias, (2) audio, (3) inverted audio, (4) subaudio tone, (5) inverted subaudio tone, (6) audio + subaudio tone, (7) inverted (audio + subaudio tone).

The Tx MOD 1 & 2 outputs can be configured by writing two data bytes (containing the desired configuration settings) to the Tx MOD 1 & 2 CONTROL ($88) register.

5.6   How to Utilize the External Signal Processing Paths

The CMX838 supports the use of external signal companding and/or scrambling by way of integrated circuit-switched paths. For example, if it is desired to scramble the Tx audio signal, the signal can be routed off-chip by way of the AUDIO OUTPUT SELECT multiplexer (bits 2 and 1 of SETUP register, $80). The scrambler output can then be fed back into the CMX838 by the use of the AUDIO INPUT 2 SELECT multiplexer (bits 4 and 3 of SETUP register, $80).

6   Hardware

6.1   Hardware Description

6.1.1   Operating Voltage

There is no on-board voltage regulator on the EV8380. The board was designed in this manner to allow for ease of testing at a variety of 'real-world' operating voltages. As a result, the EV8380 operating voltage (V_{DD}) is set by adjustment of the board's external power supply.

6.1.2   Current Measurement

CMX838 supply current may be directly measured by removing jumpers E7 (computer interface current measurement), E8 (SV_{DD} measurement), or E10 (V_{DD} measurement) and replacing each removed jumper with a current meter.

6.1.3   RF Connections

There are three SMB connectors on the EV8380 to allow connections to external RF circuits:

1.   "CPOUT": (SM1 on schematic) Charge pump output current is available at this point. This connection should be connected to the external loop filter input.

   a.   Pads are available on the EV8380 board for the addition of user selected loop filter components and/or an amplifier for the charge pump output signal. Jumpers E14 and E15 allow various charge pump & loop filter configurations.

2.   "RF": (SM2 on schematic) This connector is the injection point for the synthesizer RF negative input (i.e. VCO output).

3.   "REFIN": (SM3 on schematic) The output of the external RF reference oscillator (i.e. VC/TCXO) should be connected here.

6.1.4   Clock/Oscillator

Flexible clock generation capabilities allow the CMX838 baseband clocks to be derived from the RF reference oscillator, a dedicated crystal oscillator circuit, or an externally applied signal of suitable frequency and amplitude. A 4.0MHz crystal is provided on the EV8380 for baseband clock generation. Two jumpers, E1 and E2, which are normally fitted, enable the on-board crystal oscillator.

For lowest cost, more synchronized, designs use an RF reference oscillator signal for baseband clock generation instead of an external crystal or external baseband clock signal. To do this:

1)   Connect an external RF reference oscillator to the "REFIN" SMB connector on the EV8380 board.

   a.   Please note that the reference oscillator frequencies compatible with the CMX838 are listed in the CMX838 data bulletin.

2)   Write a data byte to the SYNTHESIZER BASEBAND CLOCK CONTROL ($89) register to configure the CMX838 clock generation circuits. For example, if the external RF reference oscillator operates at 12.8MHz, a byte of $90 will enable baseband clock generation from the external 12.8MHz reference oscillator.

7   Circuit Schematics

Three components on the EV8380 are socketed for quick replacement, as necessary, to support customer testing. R3, a pullup resistor for the microphone, is socketed to allow for different impedance microphones to be used with the EV8380. R15 is user-selectable to allow for various charge pump current levels.

Jumpers E14 and E15 have solder links installed which direct the charge pump output current directly to the "CPOUT" SMB connector. These jumpers can be modified to support various user selected charge pump current schemes.
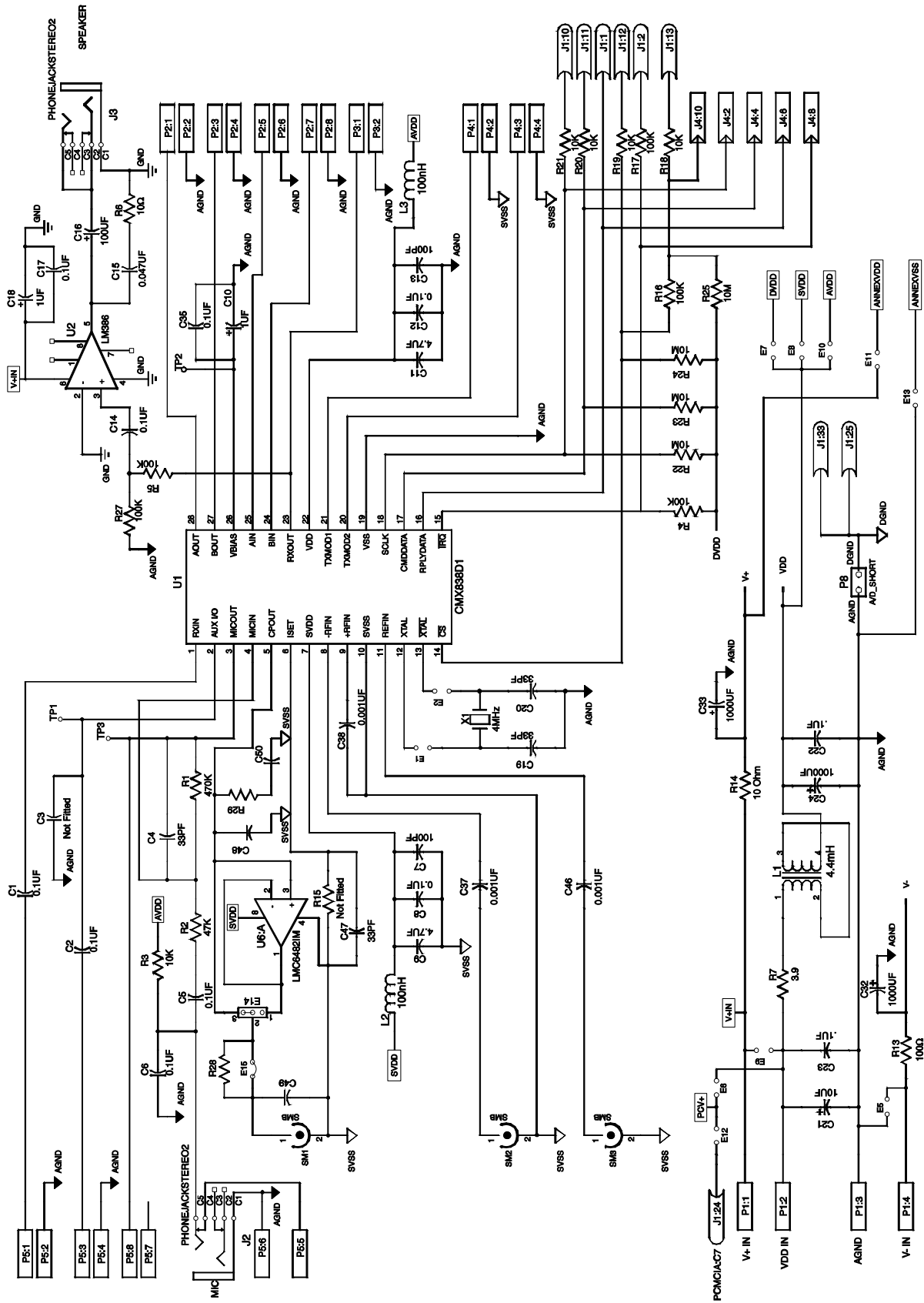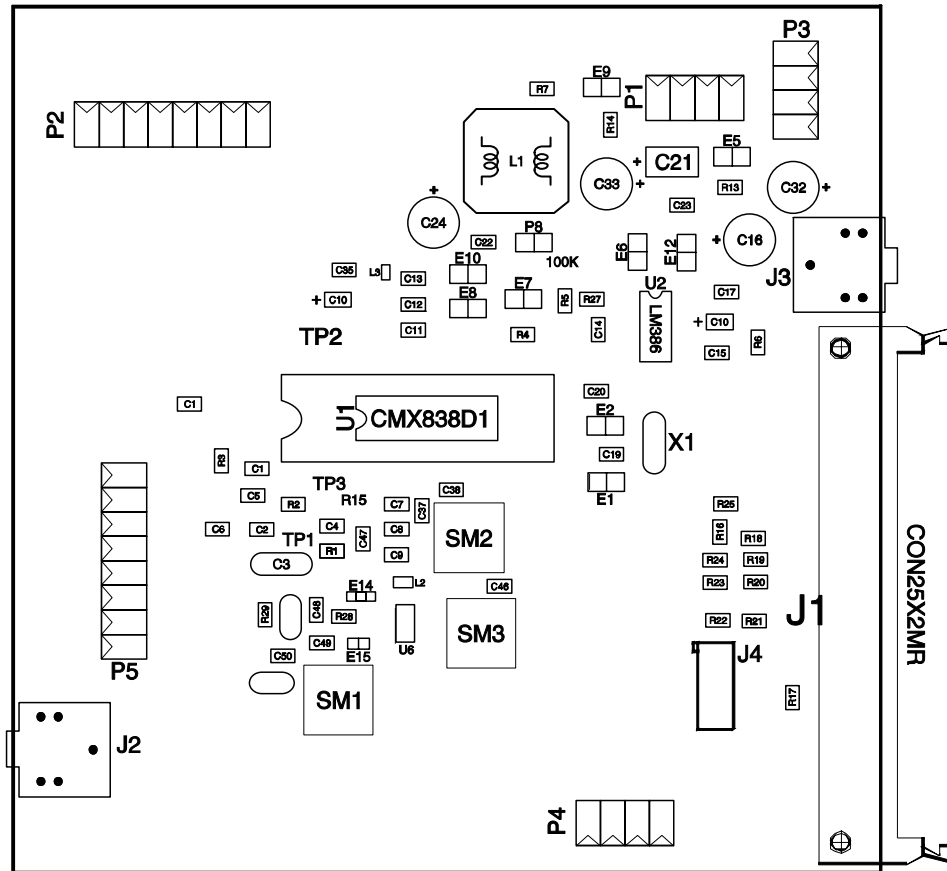
Figure 4: EV8380 Schematic

Figure 5: EV8380 Board Layout

8 Signal List

CONNECTOR PINOUT				
Connector Ref.	**Connector Pin No.**	**Signal Name**	**Signal Type**	**Description**
J1	32	REPLY DATA	Output	'C-BUS' Reply Data
	31	/IRQ	Output	'C-BUS' Interrupt Request
	39	SCLK	Input	'C-BUS' Serial Clock
	38	CMDDATA	Input	'C-BUS' Command Data
	37	/CS	Input	'C-BUS' Chip Select
	36	/CS AUX	Input	'C-BUS' Chip Select for second EV8380
	49	PCMCIA Vin	Input	+ve connector from PCMCIA card
	2	DGND	Ground	Digital ground
	50	DGND	Ground	Digital ground
J2	1	MIC INPUT	Input	Microphone input jack
J3	1	SPEAKER	Output	Speaker output jack
J4	2	SCLK	Input	'C-BUS' Serial Clock for second EV8380
	4	CMDDATA	Input	'C-BUS' Command Data for second EV8380
	6	REPLY DATA	Output	'C-BUS' Reply Data for second EV8380
	8	/IRQ	Output	'C-BUS' Interrupt Request for second EV8380
	10	/CS	Input	'C-BUS' Chip Select for second EV8380
P1	1	V+ IN	Power	+ve power (1) from external power supply
	2	VDD IN	Power	+ve power (2) from external power supply

CONNECTOR PINOUT				
Connector Ref.	**Connector Pin No.**	**Signal Name**	**Signal Type**	**Description**
	3	GND	Ground	Analog signal ground
	4	V- IN	Power	-ve power from external supply (not used)
P2	1	AOUT	Output	External signal processing path 'A' output
	2	AGND	Ground	Analog signal ground
	3	BOUT	Output	External signal processing path 'B' output
	4	AGND	Ground	Analog signal ground
	5	AIN	Input	External signal processing path 'A' input
	6	AGND	Ground	Analog signal ground
	7	BIN	Input	External signal processing path 'B' input
	8	AGND	Ground	Analog signal ground
P3	1	RXOUT	Output	Processed receive audio output
	2	AGND	Ground	Analog signal ground
	3	NC	NC	No connection
	4	NC	NC	No connection
P4	1	TXMOD1	Output	Transmit modulation output 1
	2	SVSS	Ground	Analog signal ground
	3	TXMOD2	Output	Transmit modulation output 2
	4	SVSS	Ground	Analog signal ground
P5	1	RXIN	Input	Receive input for audio & subaudio signals
	2	AGND	Ground	Analog signal ground
	3	AUX	Bi-directional	Auxiliary signal input/output
	4	AGND	Ground	Analog signal ground
	5	MIC IN	Input	Auxiliary microphone input
	6	AGND	Ground	Analog signal ground
	7	NC	NC	NC
	8	MIC OUT	Output	Microphone amplifier output
SM1	1	CP OUT	Output	Charge pump output current
	2	SVSS	Ground	
SM2	1	RFIN-	Input	Synthesizer RF positive input
	2	SVSS	Ground	
SM3	1	REF IN	Input	Reference oscillator input
	2	SVSS	Ground	

Table 5: Connector List

Test Point Ref.	Description
TP1	CMX838 Auxiliary input/output
TP2	CMX838 Vbias
TP3	CMX838 Microphone amplifier output

Table 6: Test Points

Jumper Ref.	Default Position	Description
E1	Installed	Enables crystal oscillator circuit
E2	Installed	Enables crystal oscillator circuit
E5	Removed	Connects V-IN to AGND
E6	Removed	Connects VDD IN to PCV+ (allows PCMCIA card to supply power to EV8380)
E7	Installed	Allows for current measurement for PC interface
E8	Installed	Allows for current measurement for SVDD
P8	Installed	Connects AGND to DGND
E9	Installed	Allows connection of VDD IN to V+IN
E10	Installed	Allows for current measurement for AVDD
E12	Removed	Connects VDD IN to PCV+ (allows PCMCIA card to supply power to EV8380)
E14	Installed	When used in conjunction with E15, this jumper allows various loop filter configurations.
E15	Installed	When used in conjunction with E14, this jumper allows various loop filter configurations.

Table 7: Jumper List

9 Performance Specification

9.1 Electrical Performance

9.1.1 Absolute Maximum Ratings

Exceeding these maximum ratings can result in damage to the Evaluation Kit.

	Min.	Max.	Units
Supply ($V_{+IN} - V_{-IN}$)	-15	+15	V
Supply ($V_{DD} - V_{SS}$)	V_{SS}+ 0.3	7.0	V
Voltage on any pin to V_{SS}		7	V
Current into or out of V_+ pin		100	mA
Current into or out of $V_{DD\ IN}$ pin		100	mA
Current into or out of V_- pin		35	mA
Current into or out of any other pin		30	mA
Storage Temperature	0	55	°C
Operating Temperature			°C

9.1.2 Operating Limits

	Notes	Min.	Max.	Units
Supply ($V_{+IN} - V_{-IN}$)	E9 jumper removed	2.7	15	V
Supply ($V_{DD} - V_{SS}$)		2.7	5.5	V
Operating Temperature		10	40	°C

9.1.3 Operating Characteristics

For the following conditions unless otherwise specified:
Evaluation Device Xtal Frequency = 4MHz
V_{DD} = 2.7V or 5.0V, T_{AMB} = +25°C.

	Notes	Min.	Typ.	Max.	Units
DC Parameters					
I_{DD} (evaluation board - idle)			17.5	30	mA

9.1.3.1 Timing Diagrams

For CMX838 timing information, refer to the current MX-COM CMX838 Data Bulletin.

CML Microcircuits

COMMUNICATION SEMICONDUCTORS

CML Product Data

In the process of creating a more global image, the three standard product semiconductor companies of CML Microsystems Plc *(Consumer Microcircuits Limited (UK), MX-COM, Inc (USA) and CML Microcircuits (Singapore) Pte Ltd)* have undergone name changes and, whilst maintaining their separate new names *(CML Microcircuits (UK) Ltd, CML Microcircuits (USA) Inc and CML Microcircuits (Singapore) Pte Ltd*), now operate under the single title **CML Microcircuits**.

These companies are all 100% owned operating companies of the CML Microsystems Plc Group and these changes are purely changes of name and do not change any underlying legal entities and hence will have no effect on any agreements or contacts currently in force.

CML Microcircuits Product Prefix Codes

Until the latter part of 1996, the differentiator between products manufactured and sold from MXCOM, Inc. and Consumer Microcircuits Limited were denoted by the prefixes MX and FX respectively. These products use the same silicon etc. and today still carry the same prefixes. In the latter part of 1996, both companies adopted the common prefix: CMX.

This notification is relevant product information to which it is attached.

Company contact information is as below:

CML Microcircuits (UK) Ltd

COMMUNICATION SEMICONDUCTORS

Oval Park, Langford, Maldon,
Essex, CM9 6WG, England
Tel: +44 (0)1621 875500
Fax: +44 (0)1621 875600
uk.sales@cmlmicro.com
www.cmlmicro.com

CML Microcircuits (USA) Inc.

COMMUNICATION SEMICONDUCTORS

4800 Bethania Station Road,
Winston-Salem, NC 27105, USA
Tel: +1 336 744 5050,
          0800 638 5577
Fax: +1 336 744 5054
us.sales@cmlmicro.com
www.cmlmicro.com

CML Microcircuits (Singapore) Pte Ltd

COMMUNICATION SEMICONDUCTORS

No 2 Kallang Pudding Road, 09-05/
06 Mactech Industrial Building,
Singapore 349307
Tel: +65 7450426
Fax: +65 7452917
sg.sales@cmlmicro.com
www.cmlmicro.com

D/CML (D)/1 February 2002