TEXAS INSTRUMENTS

TAS5508 GUI SOFTWARE

8 CHANNEL DIGITAL PWM PROCESSOR

Operating Instructions

Version 1.0

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TABLE OF CONTENTS

Topic **Page**

INDEX OF FIGURES

INDEX OF TABLES

1. Introduction

The purpose of this document is to describe the operation of TAS5508 Graphical User Interface (GUI) software. The TAS5508 is an eight channel Digital Audio Processor (DAP) and Pulse Width Modulator (PWM). The formal name of the TAS5508 GUI is DAP & PWM Configuration Tool or DPCT.

The TAS5508 GUI software is designed to demonstrate the features of the TAS5508 and to assist the applications programmer in generating and saving the desired operating characteristics. The GUI has an attractive appearance with the intent on emulating a DVD Receiver and TV.

The Volume, Tone, Loudness, Dynamic Range Compression, EQ, Bank Selection, Data Format, Input Mixers, Output Mixers and Bass Management can be programmed directly. The GUI also has a pop-up window which allows the user to read and write from any register in the device.

The ability to download a predefined device configuration or to save the existing configuration of the device also exists. This feature can assist the application programmer since they can get the device to operate properly with the GUI and then save this configuration to file. These values can then be ported into the applications programmers' micro-code.

The TAS5508 GUI has a feature where it can log the I2C data being sent to the device. This can be a big help to the programmer if they are concerned that their software is not sending the correct data to the device.

2. Initial Operation:

Before running the GUI software, the parallel port must be configured for EPP (enhanced parallel port). The parallel port configuration can be changed in the System Setup during system boot-up. Some computers have a BIRECTIONAL mode, which also works. (Some have PS2, which may also work.)

The GUI is designed to operate with the TAS5508 EVM and the paddle board. The paddle board is the IIC interface between the parallel port and the EVM.

3. The TAS5508 GUI

The GUI interface is partitioned into three sections. The ***lower*** section emulates a DVD receiver and it is used to display the operating characteristics of the device. The ***upper*** section emulates a TV screen and it supplies access to the predefined operations of the device. The ***menu*** section consists of a file menu, tools menu and help menu.

The initial screen of the GUI is shown below. Each section will be described later in this document.

Figure 1: The TAS5508 GUI

3.1 Lower Interface

The lower section emulates a DVD receiver. It has a ***display*** to show the operating characteristics of the device. It also contains a ***Master Volume*** adjust, ***Reinit*** Button, ***Reset*** Button and an ***Exit*** Button. Each of these features are explained in detail below.

3.1.1 Display

The display is intended to give a snapshot of the device operating parameters.

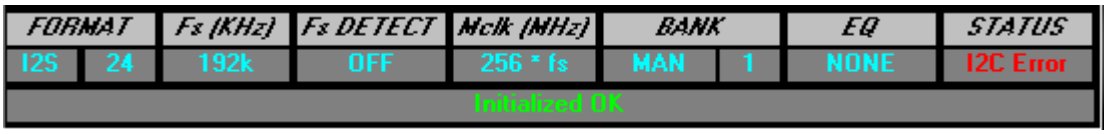

Figure 2: Lower Display

The ***FORMAT*** contains the data format and the number of bits. The data format can be I2S, Left Justified and Right Justified.

The ***Fs(KHz)*** is the sample rate of the device. The GUI polls the device every 5 seconds to update the Fs and the Mclk.

The ***Fs DETECT*** is the sample rate of the device. The GUI polls the device every 5 seconds to update the Fs and the Mclk.

The ***Mclk(MHz)*** is the sample rate of the device. The GUI polls the device every 5 seconds to update the Fs and the Mclk.

The ***BANK*** contains the information about the status of the Bank. The bank has three modes, Manual, Automatic and Update. In Manual Bank 1, 2 or 3, every GUI change is directed to that bank. Changes will be heard immediately. In automatic bank select, the stored bank values are selected automatically based on the sample rate. Bank values are typically not changed in this mode. When Update Bank is selected every change in the GUI is recorded but is not heard until Automatic Bank or Manual Bank is selected. Update Map-Only updates bits D0 through D15 of register 0x40. Bits D16 through D23 are left unchanged.

The ***EQ*** contains information about a particular type of EQ file that has been loaded. It keys on the name of the EQ file.

The caption will display "Country" if the file name equals any of the following:
"COUNTRY32.EQ550X",
"COUNTRY44.EQ550X",
"COUNTRY48.EQ550X",
"COUNTRY96.EQ550X",
"COUNTRY192.EQ550X",

The caption will display "H Rock" if the file name equals any of the following:
"HEAVYROCK32.EQ550X",
"HEAVYROCK44.EQ550X",
"HEAVYROCK48.EQ550X",
"HEAVYROCK96.EQ550X",
"HEAVYROCK192.EQ550X",

The caption will display "Jazz" if the file name equals any of the following:
"JAZZ32.EQ550X",
"JAZZ44.EQ550X",
"JAZZ48.EQ550X",
"JAZZ96.EQ550X",
"JAZZ192.EQ550X",

The caption will display "L Rock" if the file name equals any of the following:
"LITEROCK32.EQ550X",

"LITEROCK44.EQ550X",
"LITEROCK48.EQ550X",
"LITEROCK96.EQ550X",
"LiteRock192.EQ550X",

The caption will display "Rap" if the file name equals any of the following:

"RAP32.EQ550X",
"RAP44.EQ550X",
"RAP48.EQ550X",
"RAP96.EQ550X",
"RAP192.EQ550X".

The ***STATUS*** displays the status of the GUI. In this example the GUI did not communicate with the device therefore an I2C Error is displayed in this window.

3.1.2 Master Volume

The Master Volume allows the user to change the master volume regardless of which interface is visible in the upper section of the GUI. This Master Volume works in conjunction with the Master Volume in the upper section of the GUI.

Figure 3: Lower Master Volume

3.1.3 Reinit Button

The Re-init Button saves the current configuration of the device, sends a Reset command and then restores the contents of the device before the reset.

The GUI will rescan the default drive for new files when the Re-init button is pressed. Therefore, if the user creates a new .EQ550X file and stores it in the default directory, the Re-init button will load these new files in the EQ file windows. This is the same case for the DRC and loudness files.

Figure 4: REINIT Button

3.1.4 Reset Button

The Reset Button resets the device and then the GUI reinitializes itself to the device.

Figure 5: RESET Button

3.1.5 Exit Button

The Exit Button exits the program.

Figure 6: EXIT Button

3.2 Upper Interface

The upper section emulates a TV screen and it supplies access to the predefined operations of the device. The ***Volume***, ***Tone***, ***Loudness***, ***Compression***, ***EQ***, ***Bank Selection***, ***Data Format***, ***Mixers*** and ***Bass Management*** buttons are accessible here.

Figure 7: Upper Interface

3.2.1 Volume

When the user presses the Volume Button the Volume Control panel becomes visible. All eight channels of the Volume will be presented if the sample rate is less than or equal to 96 KHz. If the sample rate is greater than 96 kHz only channels 1, 2 and 8 will be visible. This is shown in the next two diagrams.

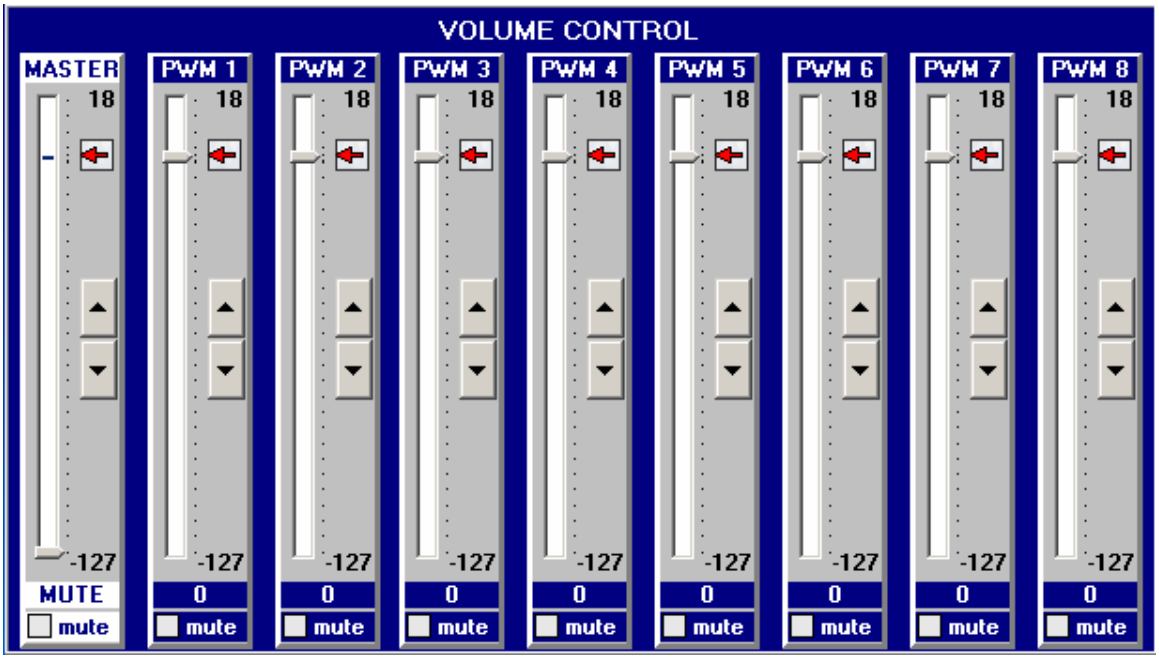

Figure 8: Full Volume Control

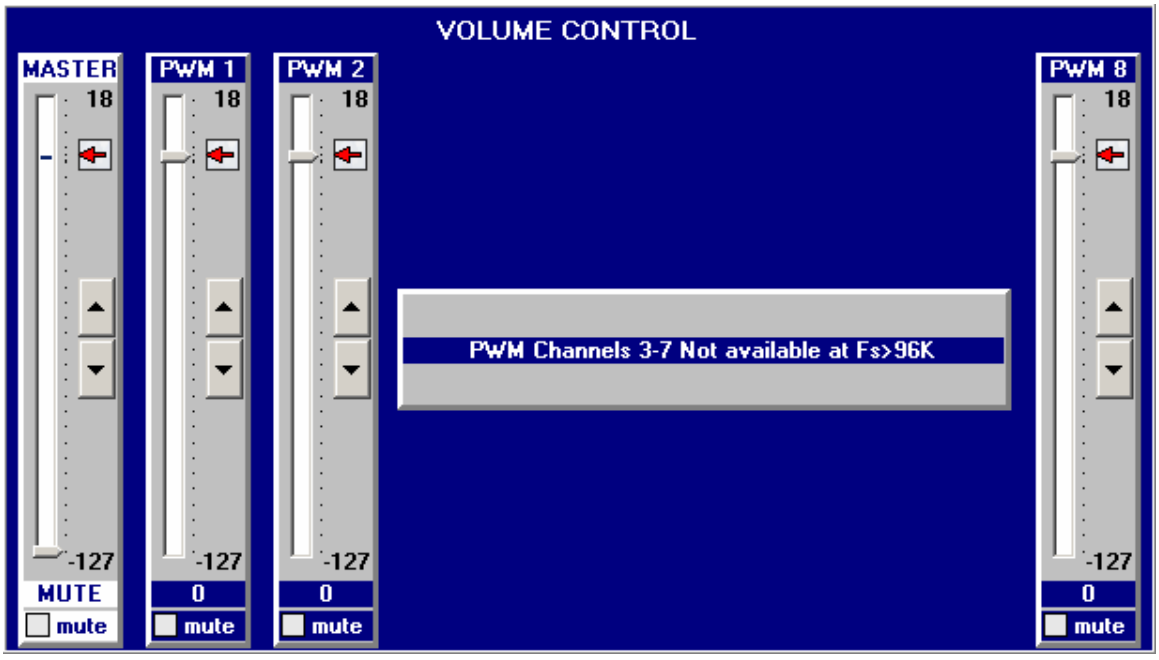

Figure 9: >96 kHz Volume Control

3.2.2 Tone

When the user presses the Tone Button the Tone Control panel becomes visible. All eight channels of the Tone will be presented if the sample rate is less than or equal to 96 KHz. If the sample rate is greater than 96 kHz only channels 1, 2 and 8 will be visible. The buttons along the bottom right side toggle the Bass and Treble inline and bypass mixers. This is shown in the next two diagrams.

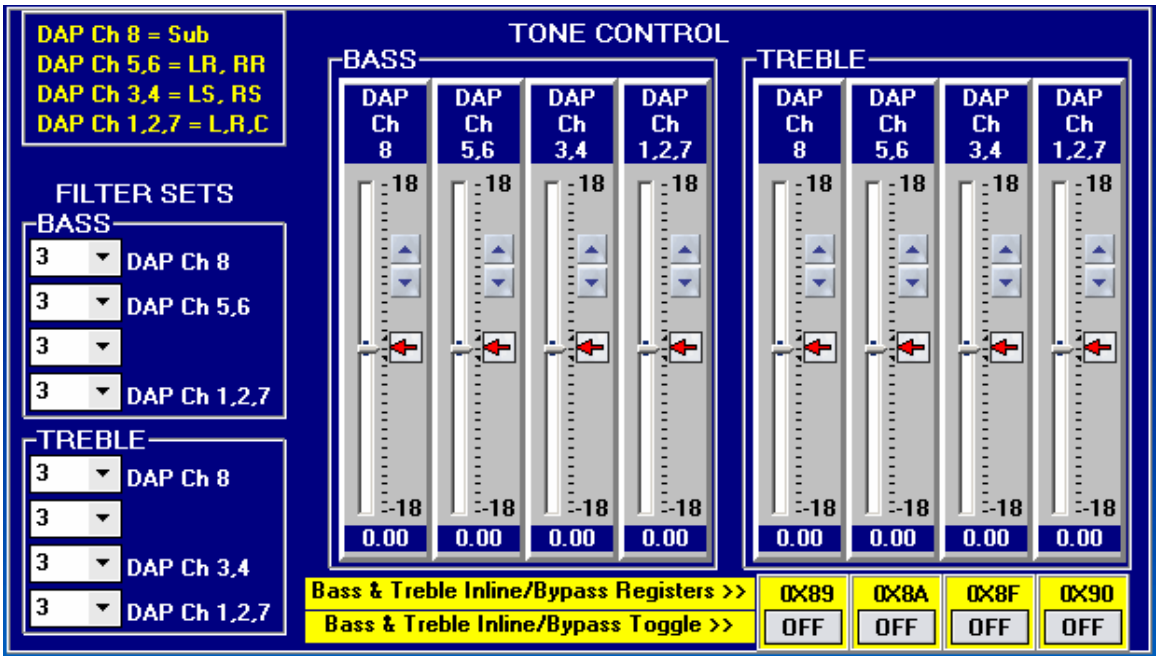

Figure 10: Full Tone Control

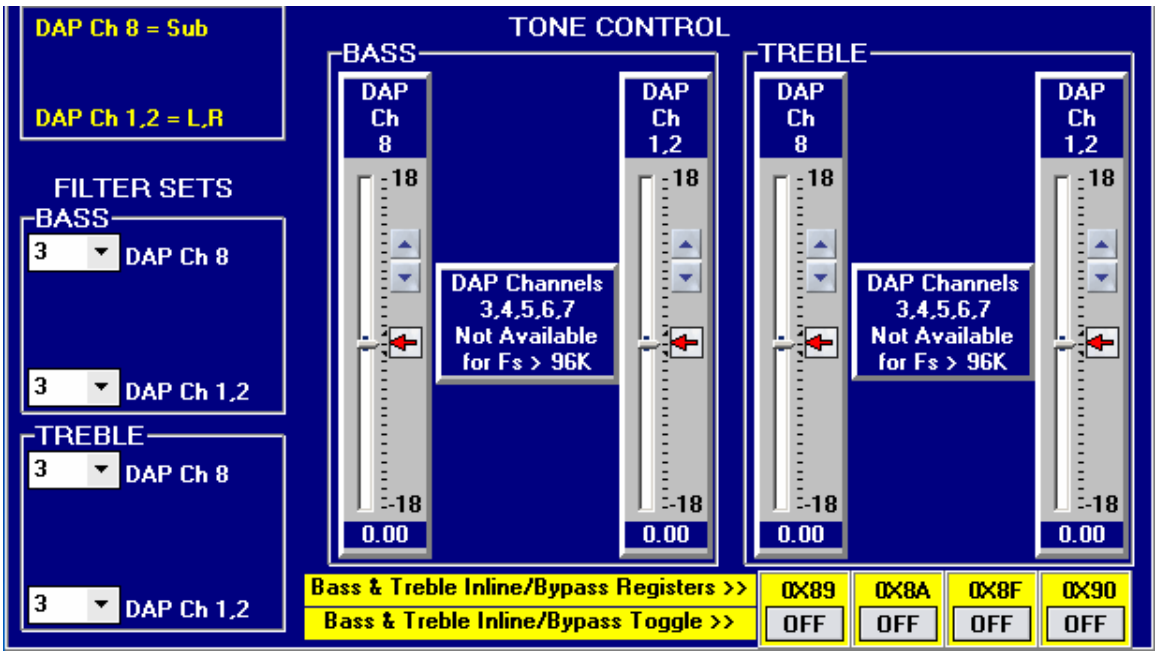

Figure 11: >96 kHz Tone Control

3.2.3 Loudness

When the user presses the Loudness Button the Loudness Control panel becomes visible.

The Loudness control allows the user to enter loudness parameters as either a Hex value or as a floating point value. The GUI will convert between the two values. The value is not written to the device until the APPLY button is pressed.

The Loudness Control also allows loading a loudness file. The program searches the default directory for files with the extension of ".LD550X" and loads them in the pull down window. Pressing the LOAD button will send the values in the file to the device. Loudness values are applied to all channels in the TAS5508.

The Loudness is turned off if the Gain Value is set to zero. The gain value must be greater than zero to turn the Loudness value on.

The Loudness Example Button describes the loudness files sent with the software.

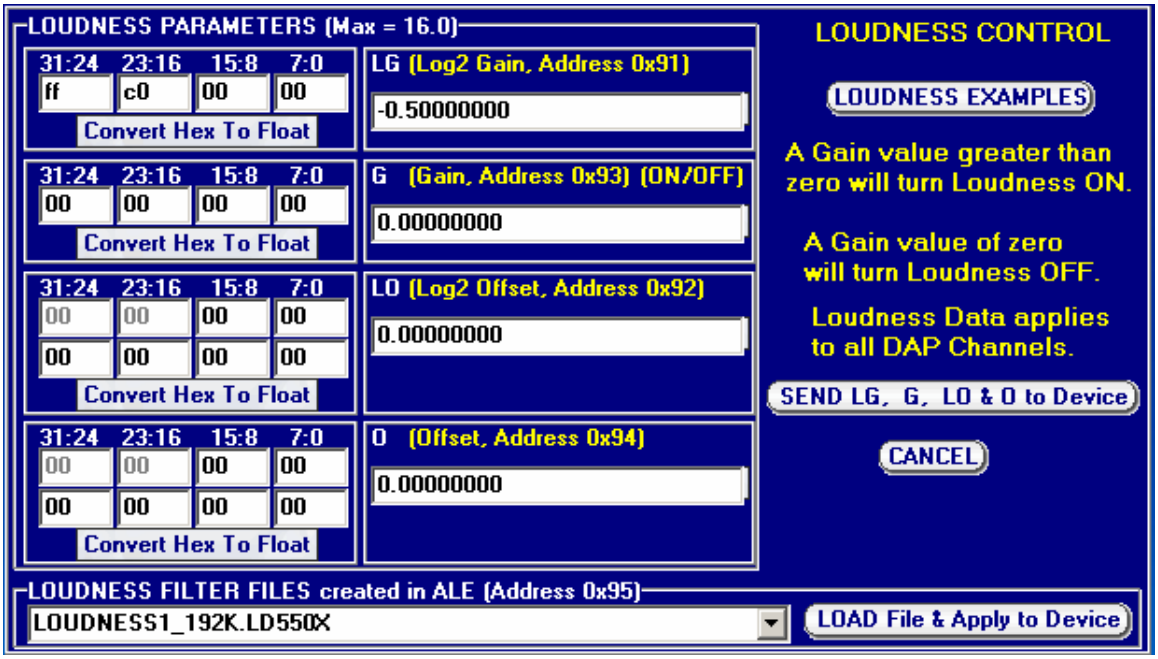

Figure 12: Loudness Control

3.2.4 Compression

When the user presses the Compression Button the Dynamic Range Compression & Expansion Control panel becomes visible.

The DRC control allows the user to enter DRC parameters as either a Hex value or as a floating point value. The GUI will convert between the two values. The value is not written to the device until the APPLY button is pressed.

The DRC Parameters Files created in ALE also allows loading a DRC file. The program searches the default directory for files with the extension of ".DRC550X" and loads them in the pull down window. Pressing the LOAD button will send the values in the file to the device. Loudness values are applied to all channels in the TAS5508.

The DRC (DAP Channel) Radio Buttons allow the user to change the channels which the DRC will be sent to.

The DRC control Radio Buttons allow the user to change the mode which the DRC will be configured.

The Alpha Calculator allows the user to create an alpha value. Pressing the E, A or D buttons will place the Alpha value into Alpha E, Alpha A or Alpha D respectively.

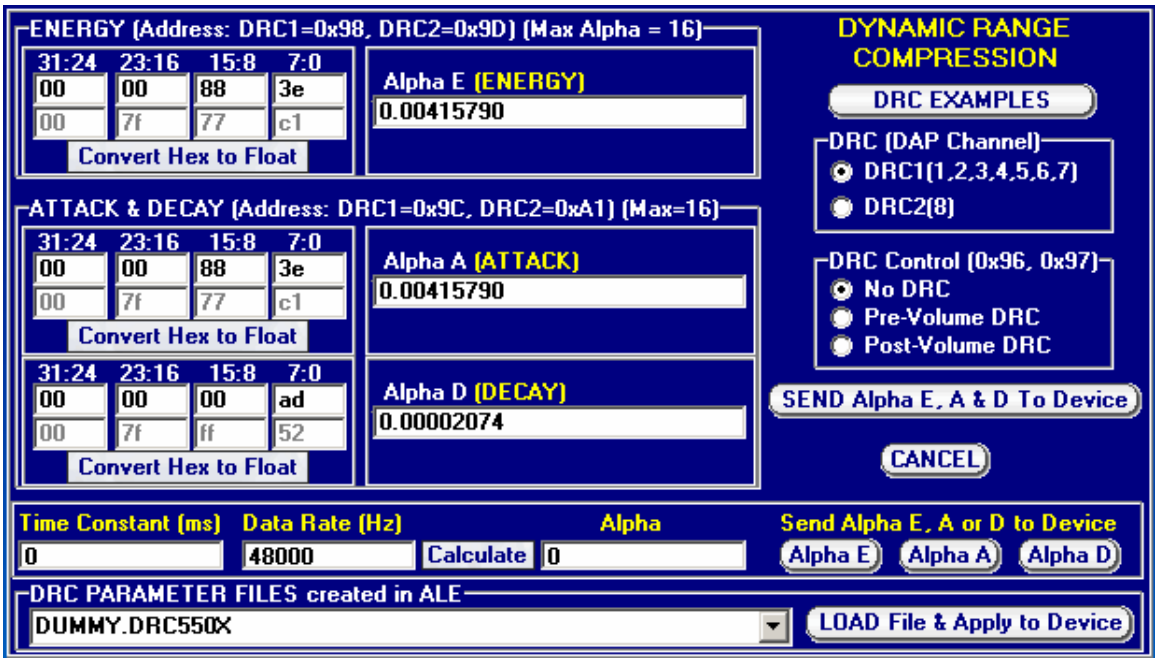

Figure 13: Compression Control

3.2.5 EQ

When the user presses the EQ Button the EQ Control panel becomes visible.

The EQ Files created in ALE also allows loading an EQ file. The program searches the default directory for files with the extension of ".EQ550X" and loads them in the pull down window. Pressing the LOAD button will send the values in the file to the device. EQ values are applied to the channel selected in the DAP Radio Button selection.

The DAP Radio Buttons allow the user to change the channels which the Biquad will be sent.

The EQ Radio Buttons allow the user to change the Biquad which is displayed in the Biquad Window. This will also be the Biquad sent to the device when the apply button is pressed.

The Graph BQ button will graph the magnitude response of the Biquad selected.

The Graph EQ button will graph the composite magnitude response of all seven Biquads.

The EQ Examples Button describes the EQ files sent with the software.

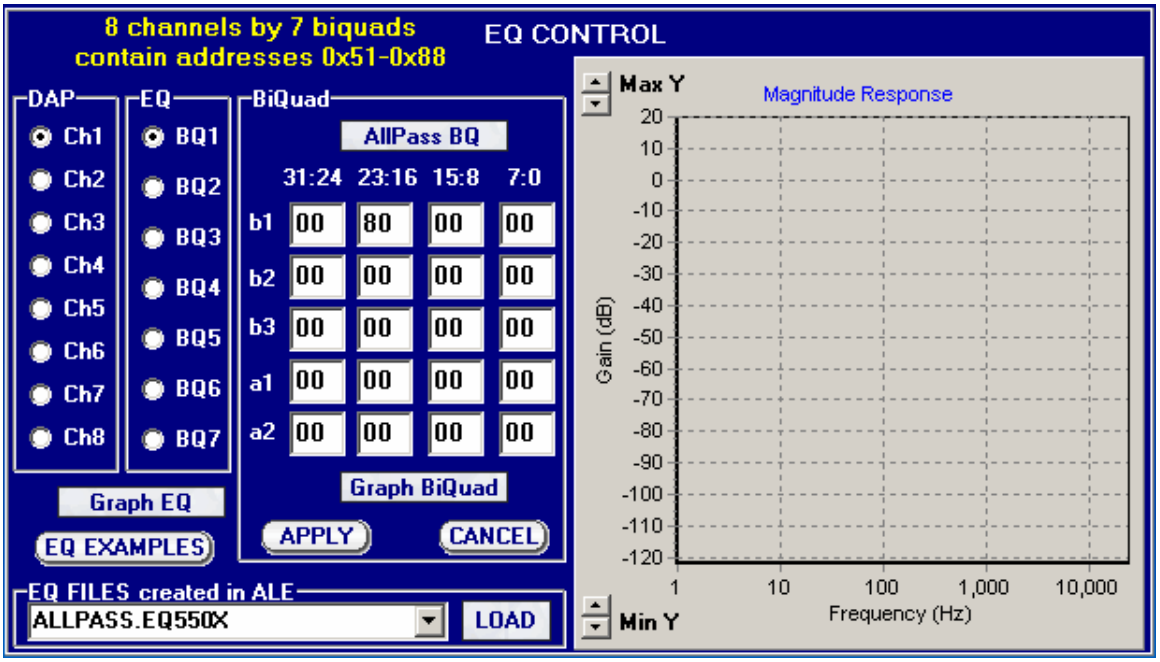

Figure 14: EQ Selection

3.2.6 Bank Selection

When the user presses the Bank Select Button the Bank Selection Control panel becomes visible.

Selecting any radio button will send that data to the device.

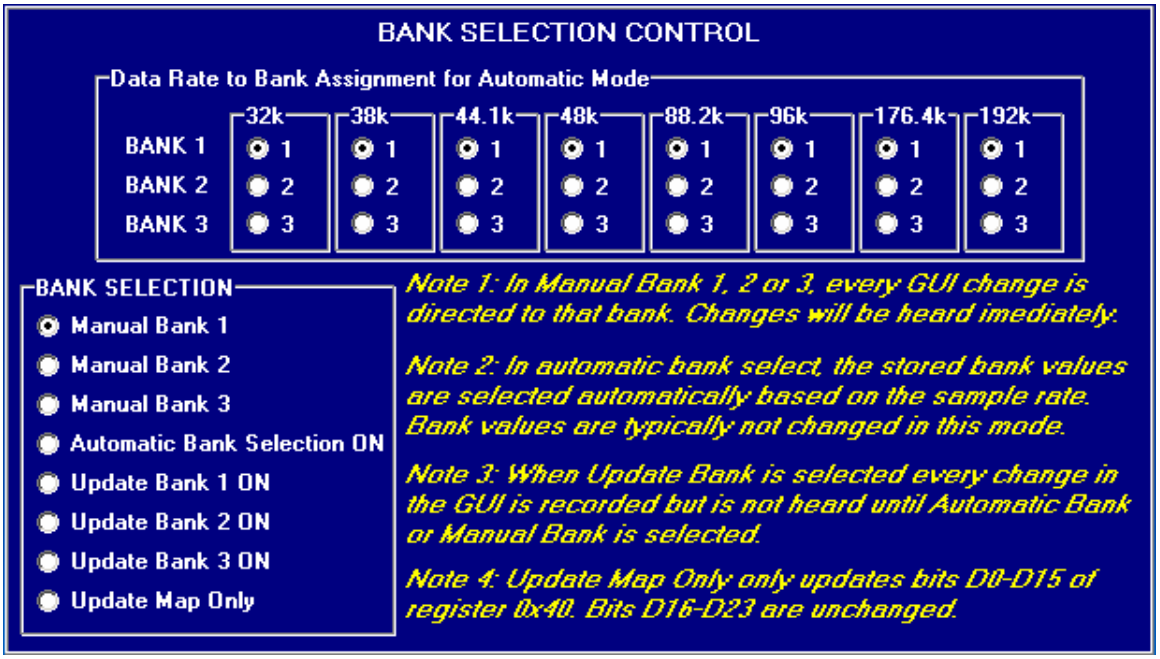

Figure 15: Bank Selection

3.2.7 Data Format

When the user presses the Data Format Button the Serial Interface & Clock Control panel becomes visible.

Selecting any radio button will send that data to the device.

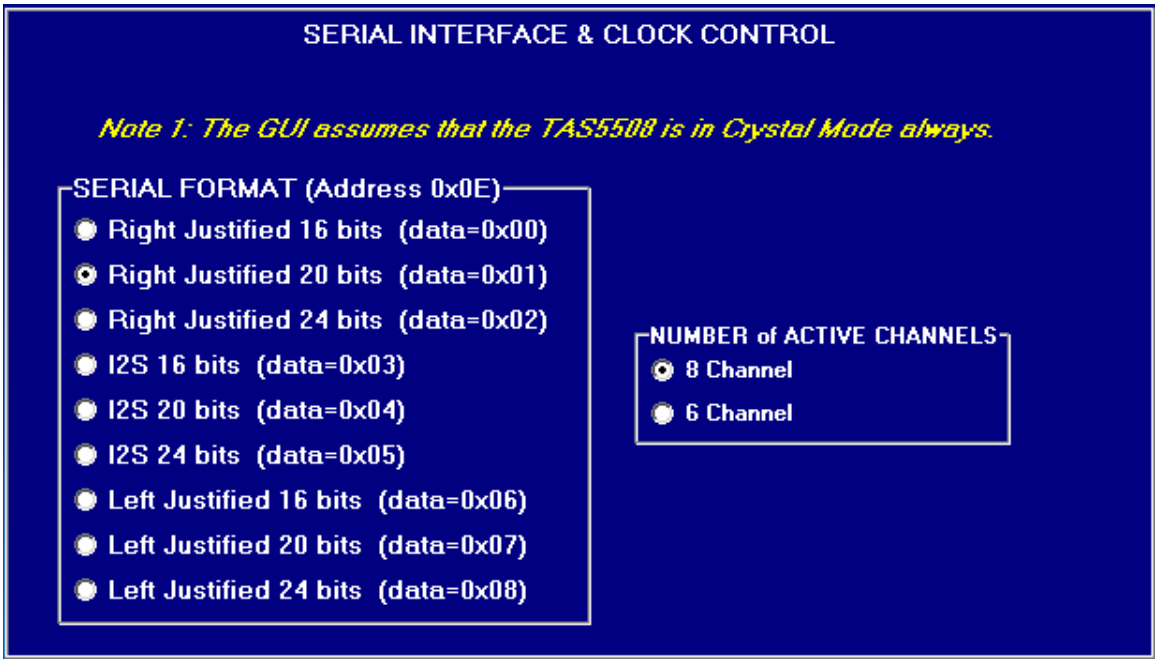

Figure 16: Data Format

3.2.8 Mixers

When the user presses the Mixer Button the Mixer Control panel becomes visible. From this panel the user can select between the input mixer and the output mixer.

The mixer buttons will change colour based on their value. When one of the buttons is pressed the mixer gain window will appear to allow the values to be changed.

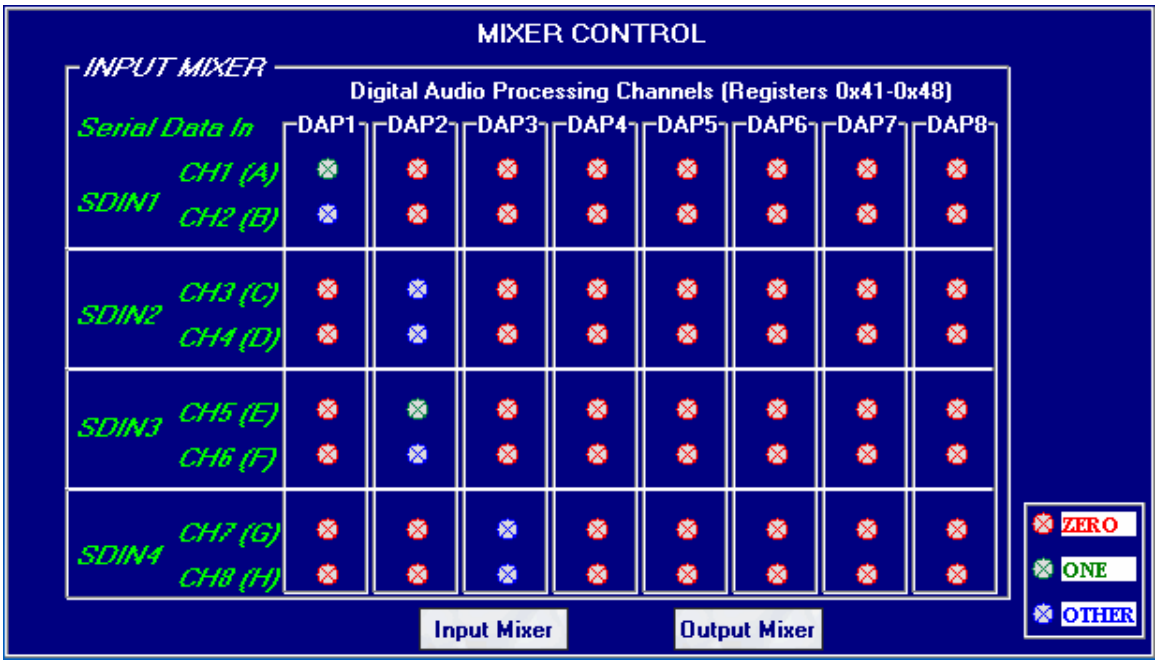

Figure 17: Input Mixer

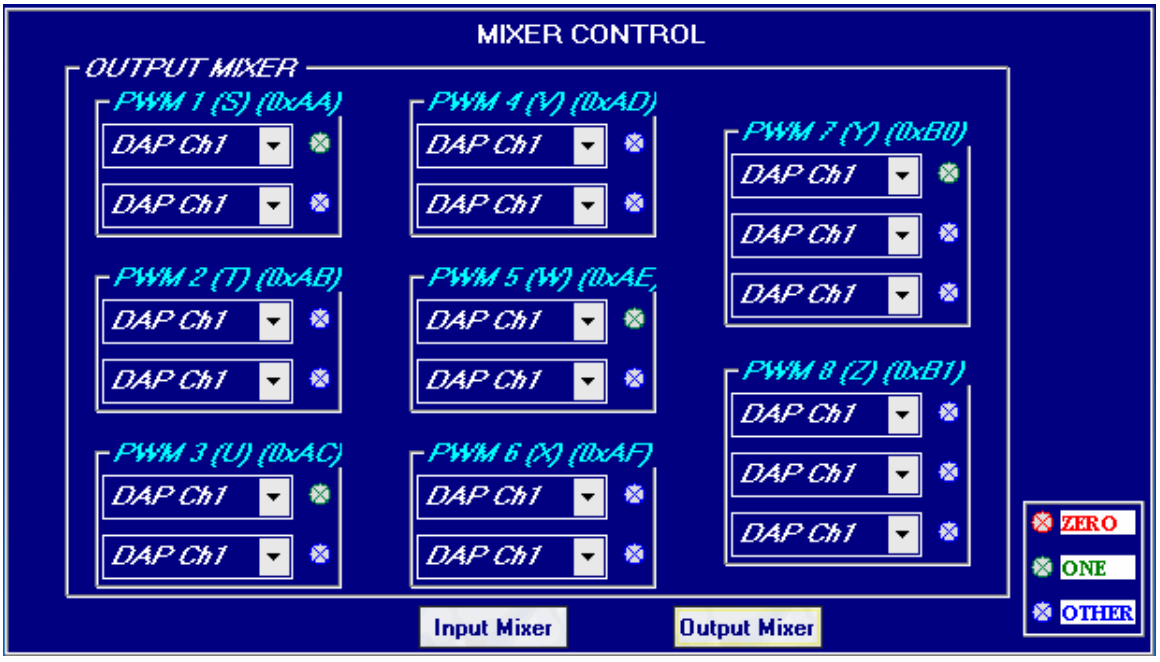

Figure 18: Output Mixer

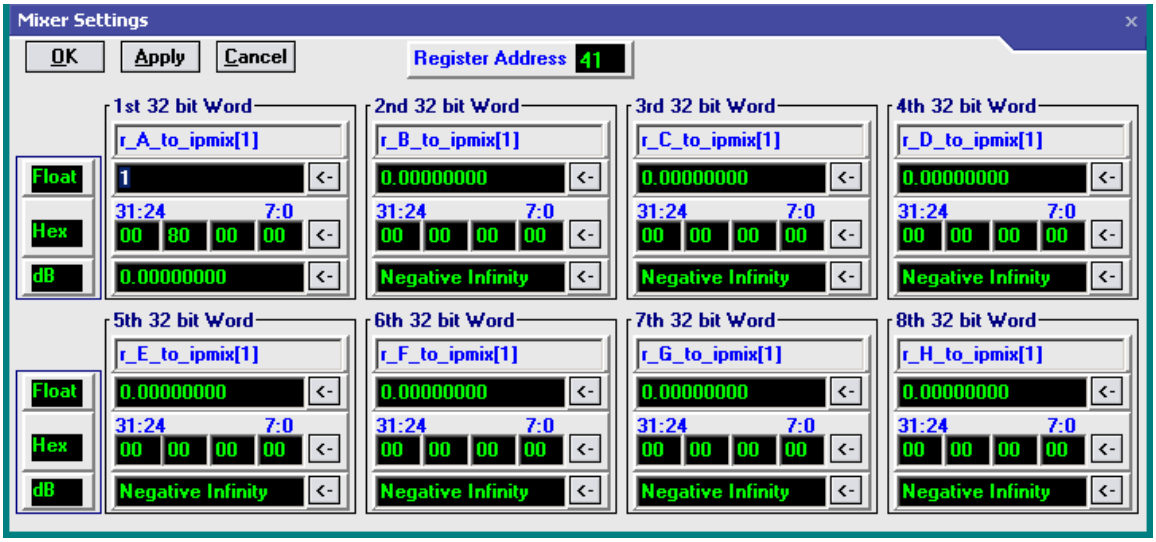

Figure 19: Mixer Gain Setting

3.2.9 Bass Management

When the user presses the Bass Manage Button the Bass Management Control panel becomes visible.

The Bass managements will consists of channels 1, 2, 7 and 8 the sample rate is less than or equal to 96 KHz. If the sample rate is greater than 96 kHz only channels 1, 2 and 8 will effect the bass management and channel 7 will be a pass through. This is shown in the next two diagrams.

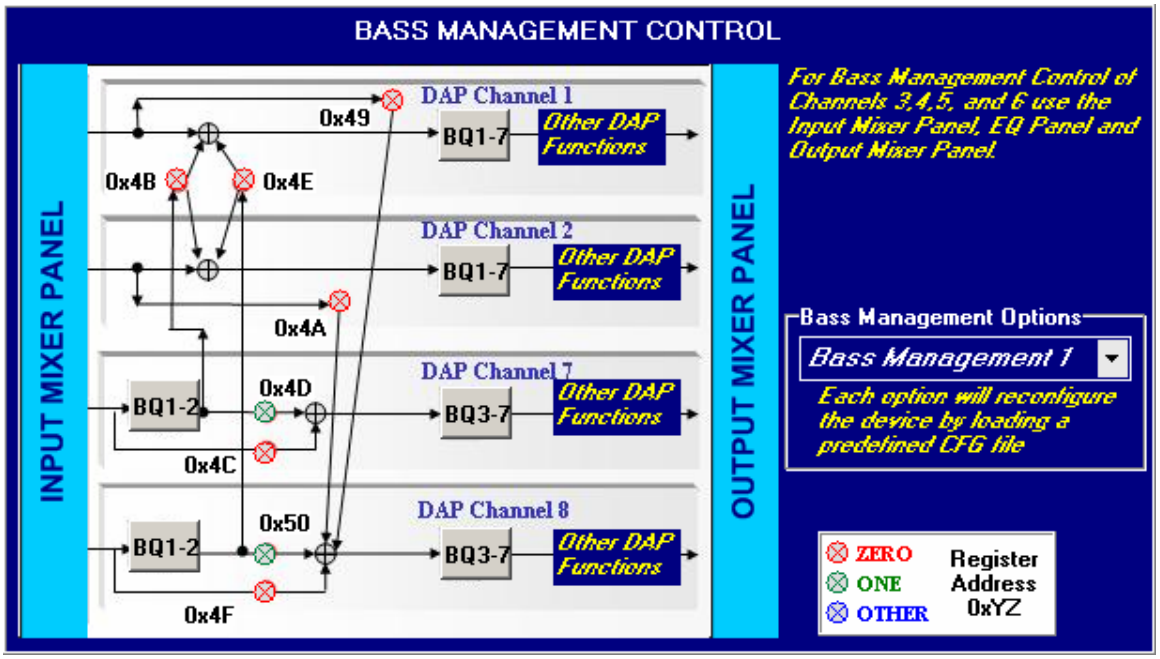

Figure 20: Bass Management

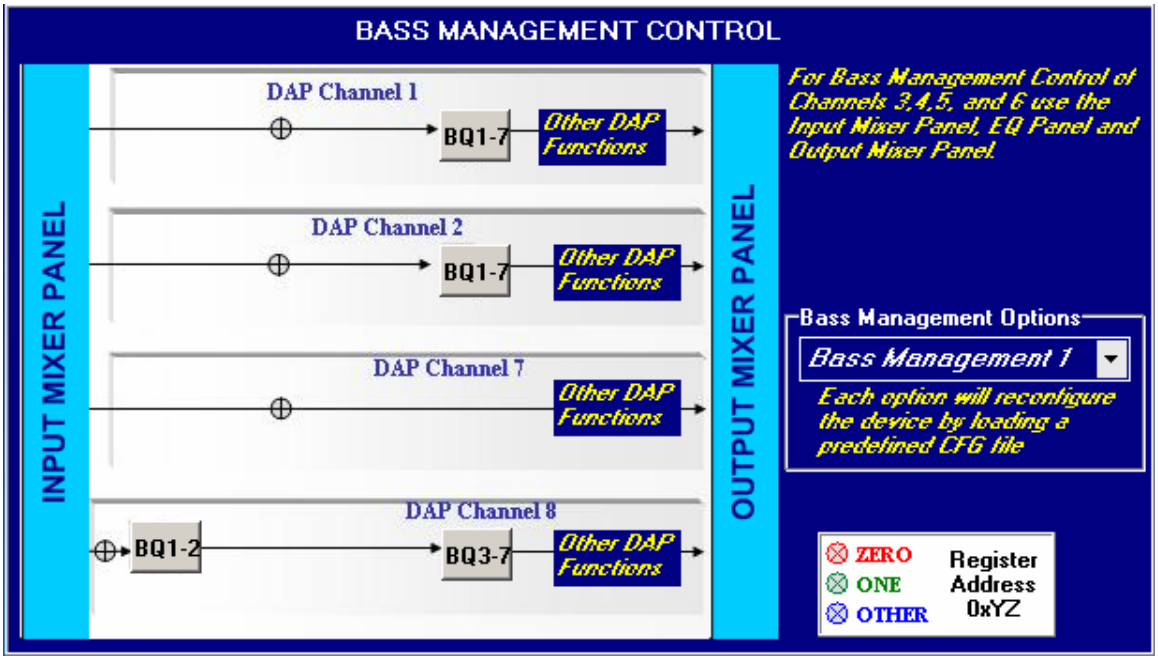

Figure 21: >96 kHz Bass Management

3.2.10 Block Diagram

When the user presses the Block Diagram Button the Block Diagram panel becomes visible. The font colour is representative of the font throughout the GUI. The line green is for the I2S data. The White is for the Digital Audio Processor (DAP) section and the blue is for the PWM section.

The Texas Instruments Inc is actually a hot link to the TI Web Site.

The writing in yellow throughout the GUI is for helpful comments. The helpful comments on the block diagram describe the logic for determining how many channels are available and when.

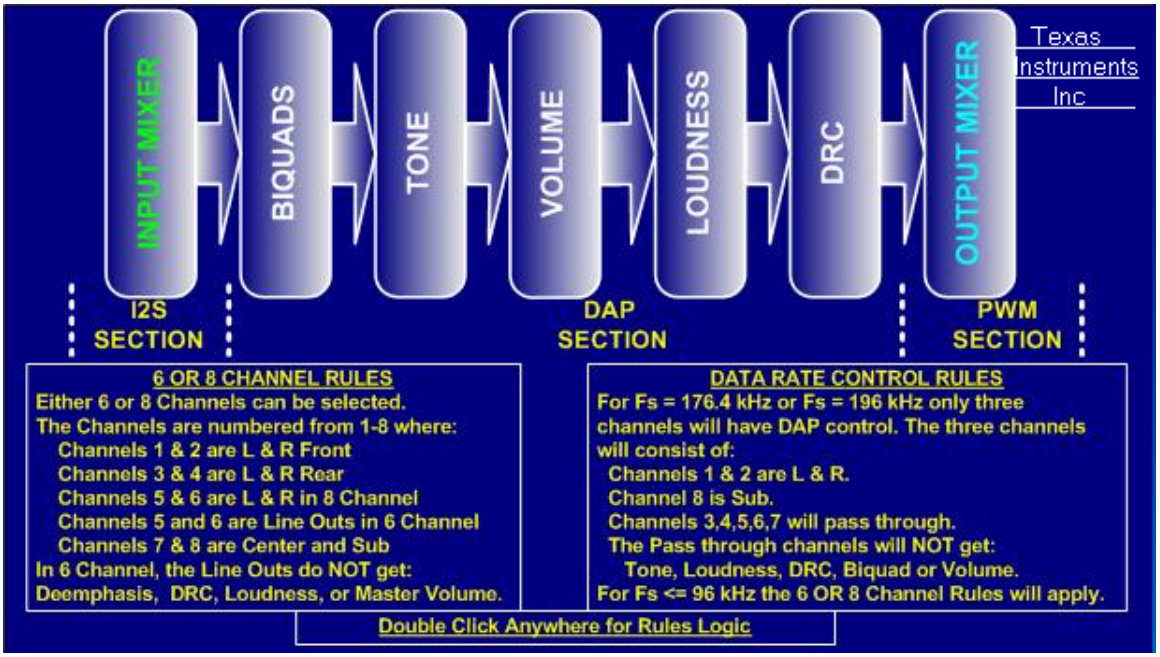

Figure 22: Block Diagram

3.3 Menu Interface

The menu section has file I/O, tools and help. The file menu allows the loading and saving of configuration files. The Tools menu has the SCL Adjust, Data Logger and Register Read/Write. The Help menu has an about box and access to the GUI Operating Instructions.

Figure 23: Menu Interface

3.3.1 File Menu

The file menu allows the loading and saving of configuration files. The configuration file format is defined in a later section.

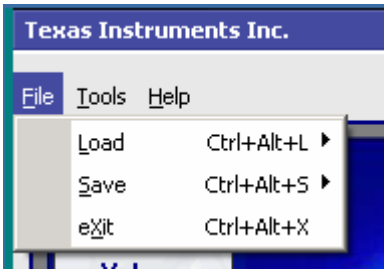

Figure 24: File Menu

3.3.2 Tools Menu

The Tools menu has the SCL Adjust, Data Logger and Register Read/Write.

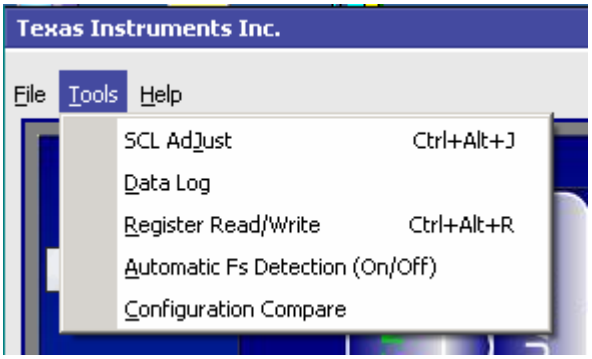

Figure 25: Tools Menu

3.3.2.1 SCL Adjust

The I2C commands are generated on the parallel port using a method referred to as Bit-banging. Since Windows does not have a readably accessible timers down into the microseconds, the I2C clocks are accomplished using C++ “For loops” to create the delay. The number of loops used in the “For Loop” is calibrated initially when the program is started. The following interface is used to assist the user in calibrating the timing on a particular computer.

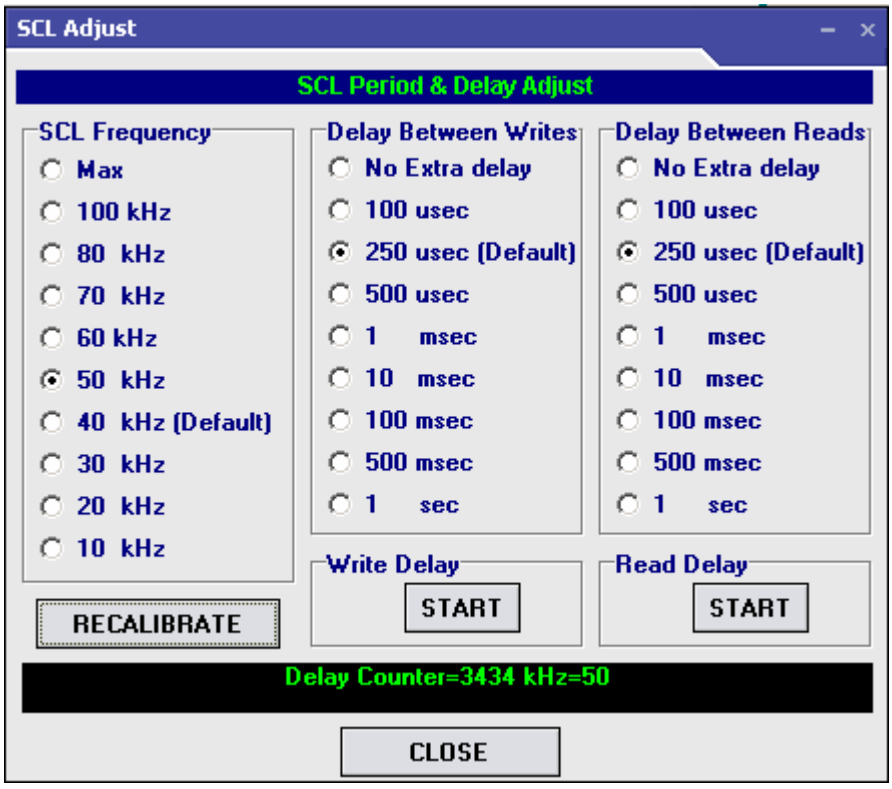

Figure 26: SCL Period & Delay Adjust

3.3.2.1.1 SCL Frequency

The looping variable in the For-loop can be changed by selecting a different SCL Frequency. The value of the SCL frequency may change between computers. To see the result of changing the frequency, it will require capturing the SCL output on as oscilloscope. The program will recalibrate SCL to the selected SCL frequency if the recalibrate button is pressed. The system clock cannot be used for this delay since the setup time for the system clock will take longer than the resultant delay will take.

3.3.2.1.2 Recalibrate Button

The recalibrate button will recalibrate the number of for loops required to achieve the SCL Frequency selected. The Recalibrate button will change to a lime green color while a calibration is in progress.

3.3.2.1.3 Delay Between Writes

The “Delay between Writes” add delays after a write transaction. For loops are not required for this delay since the system clock can be used for the longer delays. This is not a delay

between bytes but a delay between full write transactions. A transaction may consist of many bytes.

3.3.2.1.4 Write Delay Button

The write delay button will generate a series of clock pulses on the SCL line. The pulses will occur at the SCL Frequency. A delay equal to the “Delay Between Writes” will occur after every eight clock pulses. This is used to evaluate the actual timing of a particular computer.

3.3.2.1.5 Delay Between Reads

The “Delay between Reads” add delays after a read transaction. For loops are not required for this delay since the system clock can be used for the longer delays. This is not a delay between bytes but a delay between full read transactions. A transaction may consist of many bytes.

3.3.2.1.6 Read Delay Button

The read delay button will generate a series of clock pulses on the SCL line. The pulses will occur at the SCL Frequency. A delay equal to the “Delay Between Reads” will occur after every eight clock pulses. This is used to evaluate the actual timing of a particular computer.

3.3.2.2 Data Logger

Selecting the DataLog feature opens a file called “datalog.txt”. This allows the user to log the IIC command sent until the Datalog feature is selected again and the file is closed.

3.3.2.3 Register Read/Write Window

The Register Read and Write window gives the user access to every register in the TAS5508. This window is partitioned into two sections; The Byte Write/Read section and the Block Write/Read section. The Byte section allows the user to write or read a single byte to or from the TAS5508. The Block section allows the user to write or read multiple bytes to or from the TAS5508.

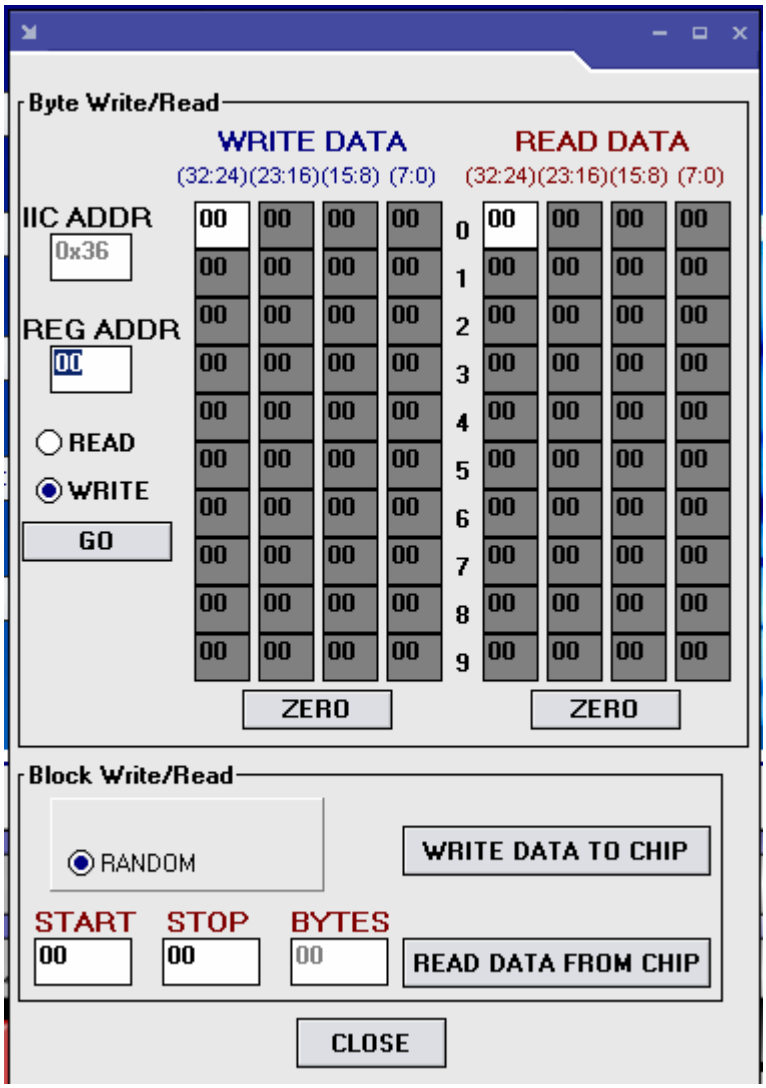

Figure 27: Register Read/Write Window

3.3.2.3.1 Byte Write/Read Section

A Register Address can be selected in the REG ADDR window. The Write Data and Read Data Windows highlight the number of bytes available for the register address selected.

When the Write Radio Button is selected, the data in the write data windows is written to the TAS5508 when the GO button is pressed.

When the Read Radio Button is selected, the data in the read data window displays the TAS5508 register value when the GO button is pressed.

The Zero buttons just zero the data in the window. It does not read or write to the device.

3.3.2.3.2 Block Write/Read Section

The START and STOP input boxes allow you to enter the starting register address and the ending register address from which the user wishes to read. The bytes will add up the total number of bytes available on the device between the start and stop address. The READ DATA FROM CHIP button reads all the registers between start and stop and allows the user to save the results to a file. The WRITE DATA TO CHIP allow the user to read a list of register values from a configuration file and writes to each register in the file. The Start and Stop register values are not considered when Writing from a files. It just writes every value in the file.

3.3.2.4 Automatic Fs Detection (On / Off)

Selecting the Fs Detection feature causes the program to read the device parameters every file seconds to verify that some parameters did not change. The GUI will update itself based on the parameters that it reads. The key parameter read is the Clock Control register (0x00). If this parameter is ON, the lower display (Figure 28) will be updated as follows:

- The Fs (KHz) window will display the current sample rate.
- The Fs Detect will be turned ON.
- The Mclk (MHz) window will display the current master clock.

If this parameter is OFF, the lower display (Figure 28) will be updated as follows:

- The Fs (KHz) window will display the last sample rate.
- The Fs Detect will be turned OFF.
- The Mclk (MHz) window will display the last master clock.

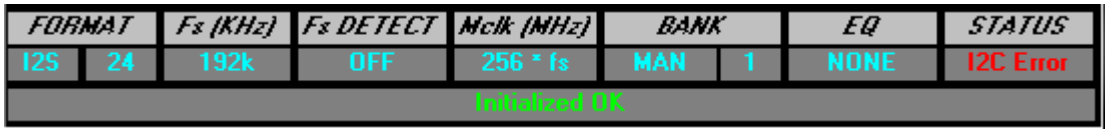

Figure 28: Lower Display

3.3.2.5 Configuration Compare

The Configuration Compare Tool pops up a window which allows the user to perform a comparison of up to three different Configuration Files. It also allows the user to call a third party file comparison tool called “ExamDiff.exe”. The following figure shows this window.

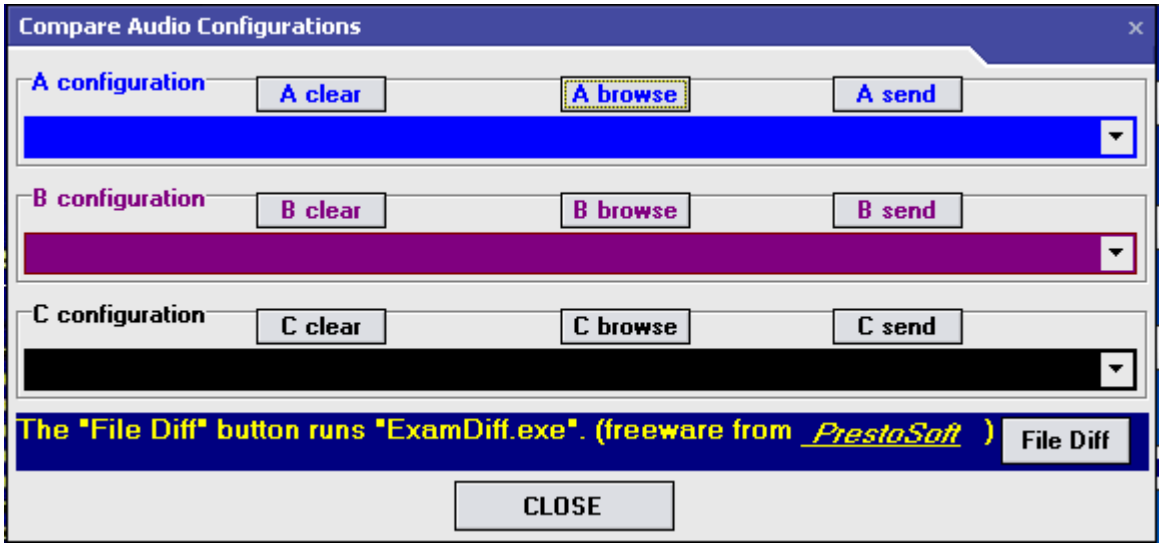

Figure 29: Configuration Compare Tool

3.3.3 Help Menu

The Help menu has an about box and access to the GUI Operating Instructions.

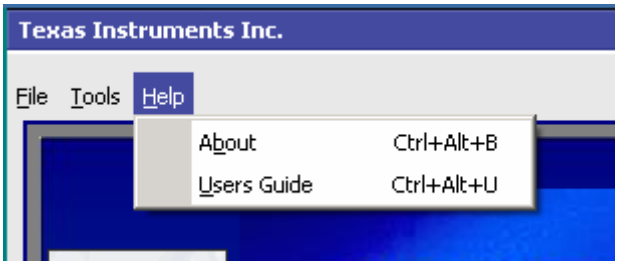

Figure 30: Help Menu

4. Support Files

The GUI has certain files that it uses to assist in the operation of the device. The file types are listed below with subsequent sections supplying more detail.

- .cfg
- .EQ550X
- .LD550X
- .DRC550X
- TAS5508.ADDR

4.1 Configuration Files

The GUI has the ability to load and save files which represent the configuration of the registers within the device. The files are called configuration files and have the extension of ".cfg". These are just text files and can be viewed using the Notepad editor available in Windows.

The Configuration files are used to setup the TAS5508 devices. The filenames are not case sensitive but the program converts them to all capitals. The configuration files can have different extensions but the format is consistent for all the configurations. The configuration file format follows closely with the TAS5508 Data Manual I2C Sub-Address Table

The configuration file format requires both Sub-Address and data. Each Sub-Address line must start with an "X". The Data starts on the line after the Sub-Address. The Data is partitioned into 4 byte words per line. The format allows a comment within the file. Each comment line must start with "//"or "!" and it must be on the line before the Sub-Address.

The following is an example configuration file format for setting the TAS5508 Loudness Parameters:

```
// Loudness Log2 LG
X91
00 00 00 00
// Loudness Log2 LO
X92
00 00 00 00
00 00 00 00
// Loudness G
X93
00 00 00 00
// Loudness O
X94
00 00 00 00
00 00 00 00
// Loudness BiQuad
X95
00 00 D5 13
00 00 00 00
0F FF 2A ED
00 FE 50 45
0F 81 AA 27
```

4.2 EQ550X File

A file with the extension .EQ550X is used to load the bi-quad coefficients directly into the device. The GUI will give an error if a file with this extension is not in the same directory as the GUI.

This file format requires both a Sub-address marker and data. The actual Sub-Address will not be specified explicitly, the GUI will decide the correct Sub-Address depending on the channel selected. The Data starts on the line after the Sub-Address Marker. The Data is partitioned into 4 byte words per line. The format allows a comment within the file. Each comment line must start with “//”or “!” and it must be on the line before the Sub-Address. The TAS5508 will accept from Q1 to Q7 bi-quads in the file. The following is an example file format for setting the TAS5508 bi-quad Parameters:

```
Q1
 C6 0E 14 00
 EE 5A 6D F0
 8A DE 66 00
 49 F1 EB F0
 96 C6 24 00
Q2
 5C 41 3F 00
 8B EA 88 F0
 E7 1E 38 00
 B3 BE C0 F0
 AC F6 37 00
Q3
 D0 D6 CF 00
 EF 3E 28 F0
 3A 47 F7 00
 3F 29 30 F0
 F5 7A D7 00
Q4
 EC 68 6A F0
 35 50 7C F0
 95 AA 29 00
 23 97 95 00
 45 05 62 00
Q5
 EC 68 6A F0
 35 50 7C F0
 95 AA 29 00
 23 97 95 00
 45 05 62 00
Q6
 EC 68 6A F0
 35 50 7C F0
 95 AA 29 00
 23 97 95 00
 45 05 62 00
```

4.3 LD550X File

The GUI has a file called “*.LD550X”. A file with this extension is used to load the Loudness coefficients directly into the device. The GUI will give an error if a file with this extension is not in the same directory as the GUI. The format of the loudness data is identical to the .cfg file format.

4.4 DRC550X File

The GUI has a file called “*.DRC550X”. A file with this extension is used to load the Dynamic Range Compression coefficients directly into the device. The GUI will give and error if a file with this extension is not in the same directory as the GUI.

This file format requires both a Sub-address marker and data. The actual Sub-Address will not be specified explicitly, the GUI will decide the correct Sub-Address depending on the channel number. The Data starts on the line after the Sub-Address Marker. The Data is partitioned into 4 byte words per line. The format allows a comment within the file. Each comment line must start with “//” or “!” and it must be on the line before the Sub-Address.

The following is an example file format for setting the TAS5508 DRC Parameters:

```
T
 00 00 0C 16
 D1 F0 00 00
 00 00 FE 8E
 E4 70 00 00
O
 00 00 89 3C
 81 EF FF FF
 00 00 00 00
 00 20 00 00
k
 06 66 66 00
 00 00 00 CF
 0A 99 99 9F
```

4.5 ADDR File

The GUI has a file called “TAS5508.addr”. This file tells the program how many registers there are in the device, how many bytes there are per register and the register address of the device. This file initializes the internal memory of the software. The GUI will not work without this file resident in the same directory as the GUI. A snapshot of the TAS5508.addr file is listed below.

```
//  Addr Bytes
#  0  0x00  1
#  1  0x01  1
#  2  0x02  1
#  3  0x03  1
#  4  0x04  1
#  5  0x05  1
#  6  0x06  1
#  7  0x07  1
#  8  0x08  1
#  9  0x09  1
```

5. IIC Error Messages

The GUI will issue an error message if it does not receive an acknowledge from the device after sending an IIC byte. The GUI will continue to send data to the device even with ACK Errors but it will no longer look for ACK's.

Figure 31: No Device Found Error Message

Error Message	Cause
"No Device Found"	The TAS5508 is not sending an Acknowledge command back to the host processor. This is referred to as an ACK in I2C terminology.
	Pin 1 (SDA) or pin 14 (SCL) on the parallel cable not allowing the I2C commands to reach the EVM.
	Pin 15 on the parallel cable not allowing the ACK command to return back to the host.
	The LPT port is not connected and configured for EPP or Bi-directional Mode. (PS-2 in some cases)
	The actual LPT port connected does not match the LPT1 port.
	The LPT1 is the default for the LPT port. The hexadecimal address for each port defined in the software is LPT1=0x378, LPT2=0x278, and LPT3=0x3BC.

Table 1: No Device Found Causes

6. Missing File Error

The program will pop up an error window if one of the following files or file extensions does not reside in the same directory as the GUI.

Error Message	Cause
“No .EQ550X files exist	**Cause**: No files with the extension of “*.EQ550X” are present in the directory. **Solution:** Create a “dummy.EQ550X” file.
“No .LD550X files exist	**Cause**: No files with the extension of “*.LD550X” are present in the directory. **Solution:** Create a “dummy.LD550X” file.
“No .DRC550X files exist	**Cause**: No files with the extension of “*.DRC550X” are present in the directory. **Solution:** Create a “dummy.DRC550X” file.
“No TAS5508.ADDR files exist	**Cause**: No files with the name of “TAS5508.ADDR” are present in the directory. **Solution:** Get the TAS5508.ADDR which was delivered with the software.

Table 2: File error causes