ISD61S00 Evaluation System

User's Manual

Publication Release Date: May 5, 2010
Revision 61.0031

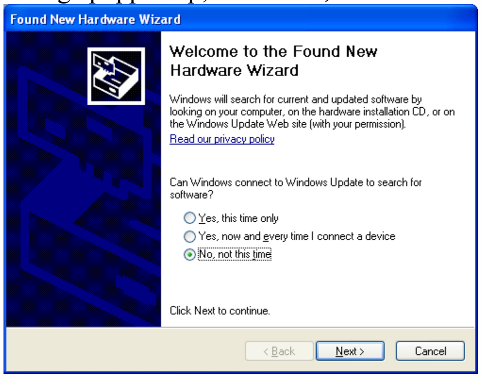
1. Getting Started

The ISD61S00 is a feature chip for the security and telephony industry. The ISD61S00 evaluation system is an integrated tool kit which enables users to evaluate the features of ISD61S00 including: a powerful macro scripting ability to facilitate audio prompting in a multi-language environment and simple address-free recording and playback; and telephony based data communications including DTMF detection and generation, FSK modem functions from 75-1200 baud, CAS and CPT (Call Progress Tone) detection and ring detection.

1.1 Install the ISD61S00 Evaluation System

Please follow the steps below to set up the evaluation system:

1. Install the ISD61S00 software utility before plugging the EV board onto PC.
2. After the ISD61S00 software utility is properly installed, plug the EV board onto PC.
    1. Users shall see a message popped up; select ‘No, not this time’, and then click ‘Next’.



Welcome to the Found New Hardware Wizard

Windows will search for current and updated software by looking on your computer, on the hardware installation CD, or on the Windows Update Web site (with your permission).

Read our privacy policy

Can Windows connect to Windows Update to search for software?

○ Yes, this time only

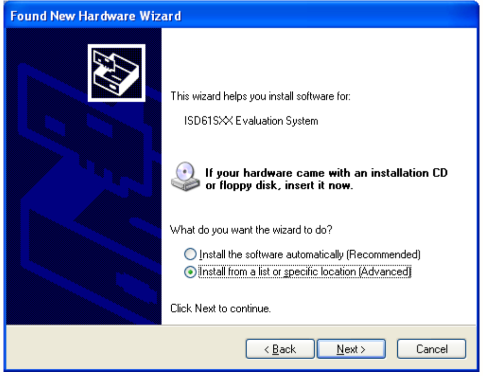
○ Yes, now and every time I connect a device

● No, not this time

Click Next to continue.

Figure 1-1 Install USB Driver - 1

2. Select ‘install from a list or specific location (Advanced)’, and then click ‘Next’.



This wizard helps you install software for:

ISD61SXX Evaluation System

If your hardware came with an installation CD or floppy disk, insert it now.

What do you want the wizard to do?

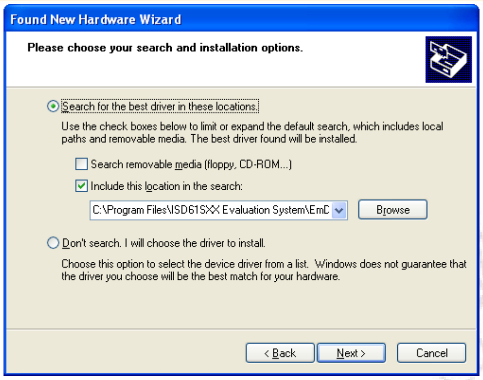
○ Install the software automatically (Recommended)

● Install from a list or specific location (Advanced)

Click Next to continue.

Figure 1-2 Install USB Driver - 2

3. Click ‘Browse’.



Found New Hardware Wizard

Please choose your search and installation options.

● Search for the best driver in these locations.

Use the check boxes below to limit or expand the default search, which includes local paths and removable media. The best driver found will be installed.

□ Search removable media (floppy, CD-ROM...)

☑ Include this location in the search:

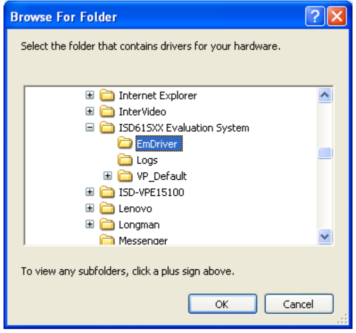
C:\Program Files\ISD61SXX Evaluation System\EmD

○ Don't search. I will choose the driver to install.

Choose this option to select the device driver from a list. Windows does not guarantee that the driver you choose will be the best match for your hardware.

Figure 1-3 Install USB Driver - 3

4. USB driver is located in the ISD61S00 software utility installation folder, which by default is "c:\Program Files\ISD61SXX Evaluation System\Driver"; Click ‘OK’.



Browse For Folder

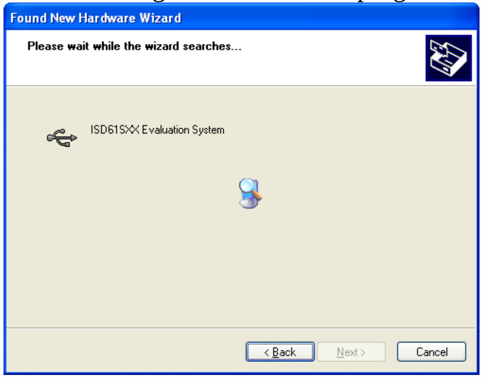
Select the folder that contains drivers for your hardware.

- Internet Explorer
- InterVideo
- ISD61SXX Evaluation System
  - EmDriver
  - Logs
  - VP_Default
- ISD-VPE15100
- Lenovo
- Longman
- Messenger

To view any subfolders, click a plus sign above.

Figure 1-4 Install USB Driver - 4

5. Users shall then see the installing of USB driver in progress.



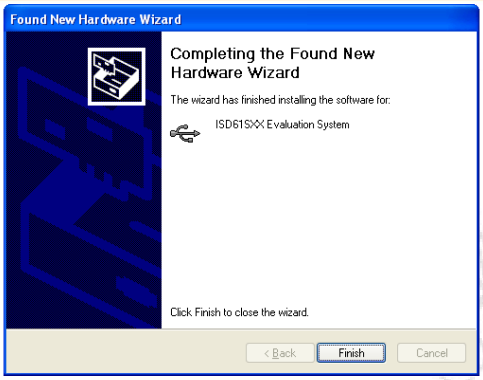
Found New Hardware Wizard

Please wait while the wizard searches...

ISD61SXX Evaluation System

Figure 1-5 Install USB Driver - 5

6. After the installation of USB driver, reset the USB by unplugging and plugging the board.



Found New Hardware Wizard

Completing the Found New Hardware Wizard

The wizard has finished installing the software for:

ISD61SXX Evaluation System

Click Finish to close the wizard.

Figure 1-6 Install USB Driver - 6

2. Evaluation Board Overview

Figure 2-1 below is the ISD61S00 Evaluation board. The board has 4 main areas:

1. USB Controller and Power Sources
2. ISD61S00 and Related Circuitry
3. Clare DAA
4. Discrete DAA

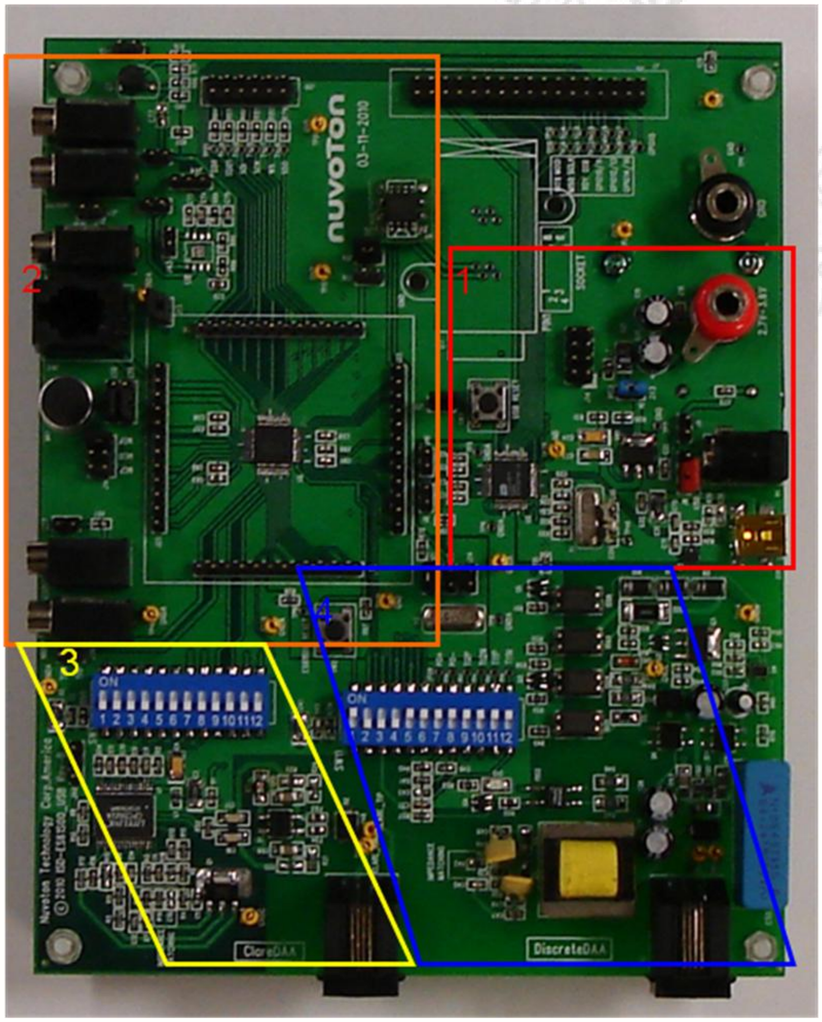


Figure 2-1 ISD61S00 Evaluation Board

2.1 USB Controller and Power Sources

The board could be powered by USB power (**A**) or an external 5V power (**B**), as determined by the jumper J6 (**C**) or 3 AAA batteries by the jumper J9 (?). VR1 (**D**) is a 5V-to-3V regulator to provide the needed 3V for the ISD61S00. Or, users could also power the board by a 3V power source via the on-board banana jack (**E**); the jumper J5 (**F**) must be removed if this is the case.

The evaluation system is controlled by an on-board USB controller W681308 (**G**). Users can capture the controlling signals between the USB controller and the ISD61S00 on the on-board headers (**H**). To reset the USB controller, users could either press the RESB button (**I**) or unplug and plug the USB cable (**A**). The on-board USB controller has its own microphone (**J**) and speaker (**K**) interfaces.

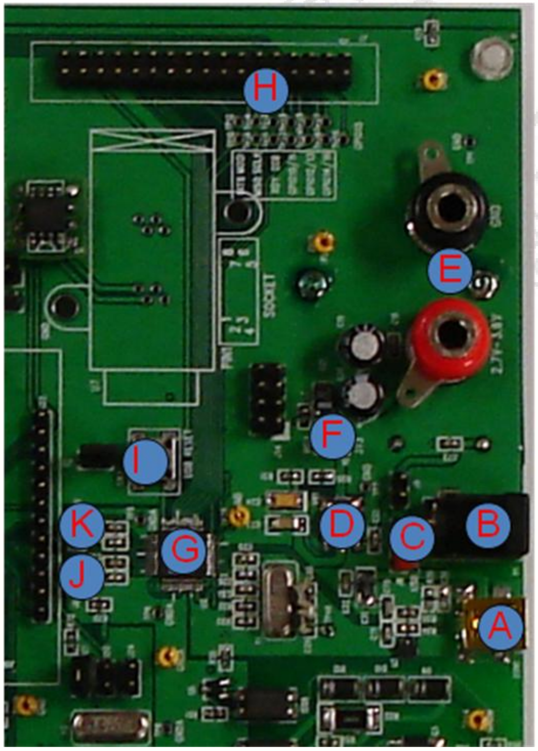


Figure 2-2 USB Controller and Power Sources

2.2 ISD61S00 and Related Circuitry

ISD61S00 (**A**) is clocked by the crystal or the external clock source. There is a 12.288MHz crystal (**B**) on the board. The jumper J21 (**C**) must be removed if users want to clock the ISD61S00 by an external clock source.

ISD61S00 is a ROM-less part, which uses external serial flash as the storage media. On board there are a DIP-8 socket (**E**) and a flash socket footprint (**F**, socket not included); users can select either one by the jumpers J29 (**G**) and J30 (**H**). For the DIP-8 socket, flash memory will be soldered on an SOIC adaptor board so that users can replace the flash easier. For the flash socket, users can turn the board into a flash programmer by soldering the flash socket.

Around the ISD61S00 are the related connectors and circuitries:

- I2S interface (**I**) shares the GPIO pins 4 ~9; by default these pins are pull-up input pins; to set them as I2S users have to set the corresponding PORT_AF (port alternate function) registers.
- J31 and JAUX (**J**) are the pin-header and jack for Aux-out. Aux-out is a single-ended voltage output, which needs an external amplifier to drive the speaker.
- J25 and JAUXSPK (**S**) are the pin-header and jack for Aux-out amplified. The ISD8101 U10 takes the single end voltage as input and drives the speaker with differential voltage by JAUXSPK.

  - If user wants to drive speaker by external amplifier ISD8101, user may put jumpers on J43 and J44 2-3 to turn on the ISD8101.
- J27 and 3.5mm Jack (**K**) are the pin-header and jack for the device speaker driver outputs.
- MK1 (**L**) is the on-board microphone, which is configured as a single-ended microphone input. To use the microphone, user can put jumper on J19 (**N**) 5-6, J38 1-2 and J39 1-2 (**R**).
  - If users want to record messages via Aux-in, users can put jumpers on J19 (**N**) 1-2 & 3-4, remove the jumper on J19 (**N**) 5-6, and then record through the jack phone connector JADCIN (**O**).
- J36 and JTI1IN (**P**) are the pin-header the jack connector of the TI1P/TI1N.

- J14 is the keypad connector for standalone operation. The keypad consists of
  - A 4 x 4 kay pad.
  - A uC to talk to the main board.
  - A display for Caller ID.

To reset the ISD61S00, users can press the RESET button (**Q**)

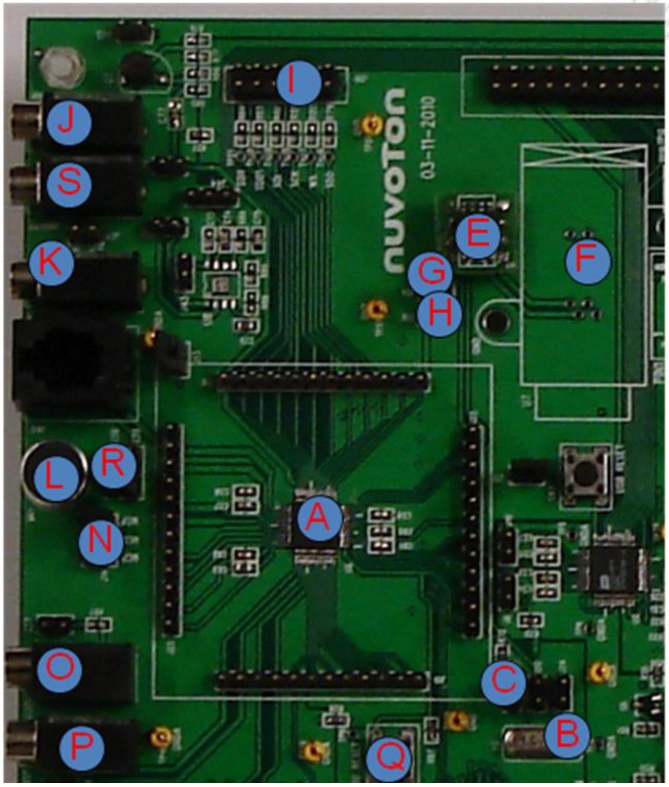


Figure 2-3 ISD61S00 and Related Circuitry

2.3 Clare DAA

With the Clare DAA configuration, the phone signal (**A**) goes to the on-board Clare CPC5622 (**B**) to talk to the ISD61S00. The Clare CPC5622 (**B**) converts the 2-wire phone signal (TIP and RING) to 4 wires (TI and PO) and processes the controlling signals including the Hook Control, Hook Detection, Loop Detection, Ring Detection, and Pulse Dial. This simple configuration reduces the number of discrete components as in the Discrete DAA.

To use the Clare DAA, users need to switch SW1 (**C**) as in Figure 2-5 . The switch SW1 (**C**) of Clare DAA is a counterpart of the switch SW11 of Discrete DAA.

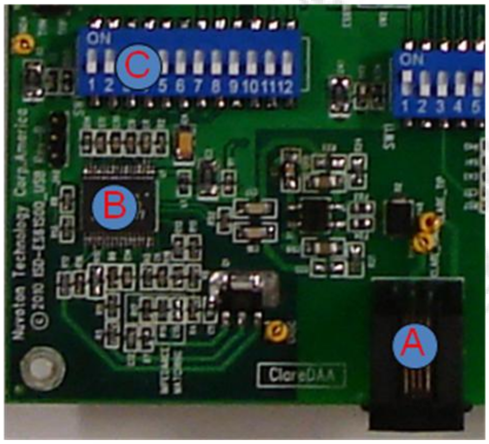


Figure 2-4 Clare DAA

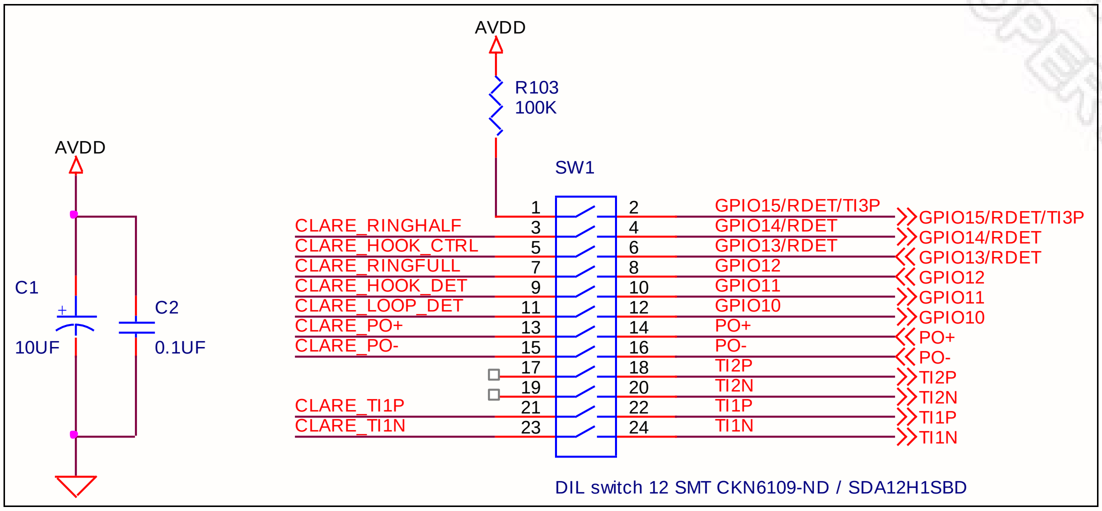


Figure 2-5 Switch of Clare DAA

2.4 Discrete DAA

Unlike the Clare DAA, the Discrete DAA configuration uses discrete components to convert the phone line (**A**) and process the controlling signals including the Hook Control, Hook Detection, Loop Detection, Ring Detection, and Pulse Dial. For details of Discrete DAA configuration, please refer to section 5.4 *Discrete DAA.*

To use the Discrete DAA, users need to swicth SW11 (**B**). The switch SW11 (**B**) of Discrete DAA is a counterpart of the switch SW1 of Clare DAA.

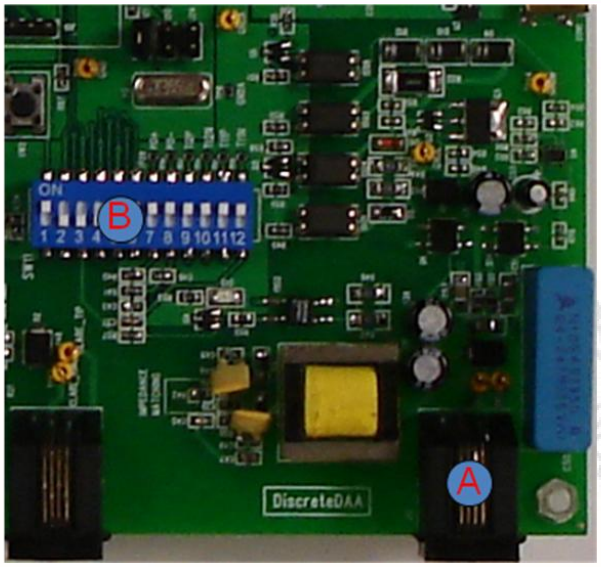


Figure 2-6 Discrete DAA

3. ISD61S00 Software Utility Overview

There is an example project named I61SXX_Sample. This section introduces the ISD61S00 software utility panel by panel based on this example project.

To open the example project: launch the ISD61S00 software utility. On the menu bar, click ‘Project’ -> ‘Open’; users can find the example project in "c:\Program Files\ISD61SXX Evaluation System\I61SXX_Sample".

There are three subfolders in the project folder:

- BinFiles: where the compressed Voice Prompts are stored.
- WavFiles: where the re-sampled and de-compressed Voice Prompts are stored.
- WavSrc: where wave sources of Voice Prompts are stored.

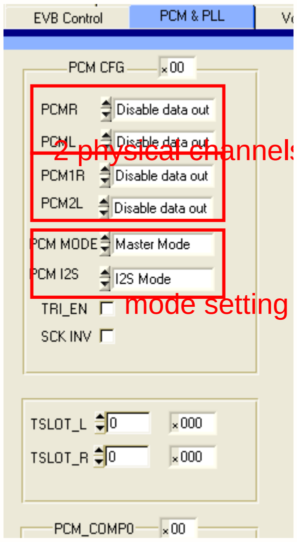
3.1 PCM & PLL

Users configure the Digital Audio Interface (PCM/I2S) and the Clock Generation on this panel.

ISD61S00 has a digital audio interface consisting of two physical transitive/receive channels; both channels share common clock and synchronization. With digital audio, ISD61S00 can be configured as master (ISD61S00 generates clock and frame sync) or slave (ISD61S00 accepts clock and frame sync). The digital audio interface has two modes;

- In I2S mode, PCM interface conforms to the I2S digital audio interface convention.
- In PCM mode, data transfer is synchronized to the rising edge of the frame sync signal.

For register details please refer to the design guide, section 8.4 Digital Audio Interface Control.



PCM CFG x00

PCMR Disable data out

PCML Disable data out

PCM1R Disable data out

PCM2L Disable data out

PCM MODE Master Mode

PCM I2S I2S Mode

TRI_EN □

SCK INV □

TSLOT_L 0 x000

TSLOT_R 0 x000

PCM_COMP0 x00

Figure 3-1 PCM/I2S Control

Figure 3-2 below shows the block diagram of the clock generation. The PLL is mainly used to multiply the external input clock by a fractional number (the $\mathbf{f_2 = R * f_1}$ in the diagram). The dividers before and after the PLL are optional for users to get the frequency within a certain range. The target IMCLK is 24.576MHz.

For register details please refer to the design guide, section 8.3 Clock Generation and PLL.

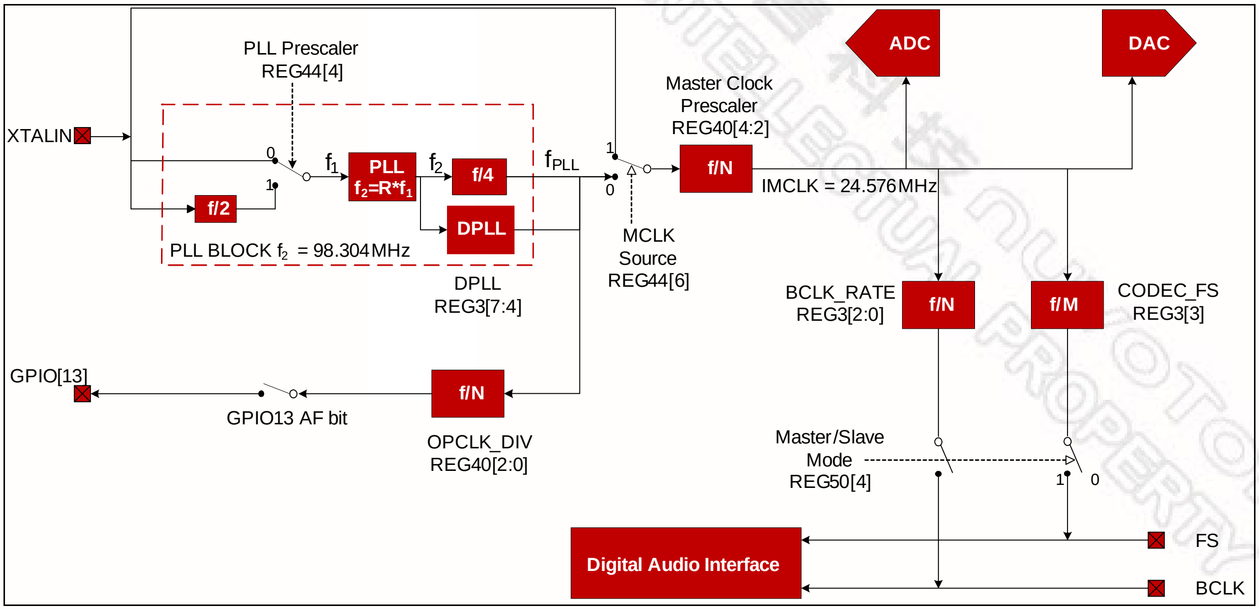
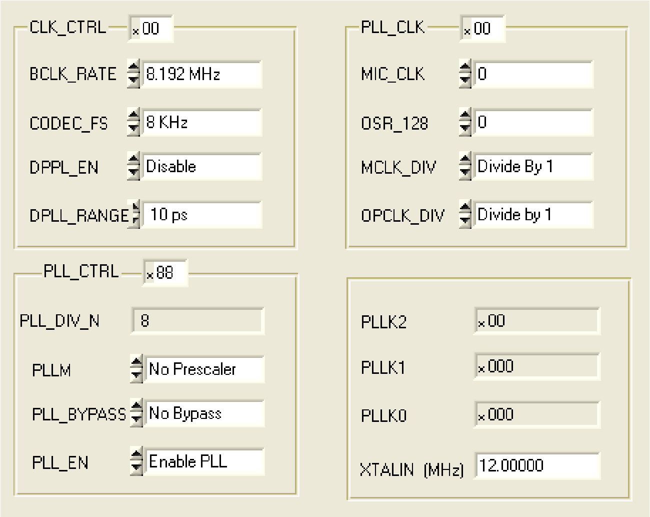


Figure 3-2 Clock Generation and PLL



CLK_CTRL	x00
BCLK_RATE	8.192 MHz
CODEC_FS	8 KHz
DPPL_EN	Disable
DPLL_RANGE	10 ps
PLL_CLK	x00
MIC_CLK	0
OSR_128	0
MCLK_DIV	Divide By 1
OPCLK_DIV	Divide by 1
PLL_CTRL	x88
PLL_DIV_N	8
PLLM	No Prescaler
PLL_BYPASS	No Bypass
PLL_EN	Enable PLL
PLLK2	x00
PLLK1	x000
PLLK0	x000
XTALIN (MHz)	12.00000

Figure 3-3 Clock and PLL Setting

3.2 Voice Prompt

Voice prompts are pre-recorded audio of any length, from short words, phrases or sound effects to long passages of music. These Voice Prompts can be played back in any order as determined by the users and applications. A voice prompt consists of two components:

(1) An index pointing to the pre-recorded audio.
(2) Pre-recorded audio.

Before adding voice prompts, first set the default sampling rate (A) and compression algorithm (B).

To add voice prompts, follow the steps below:

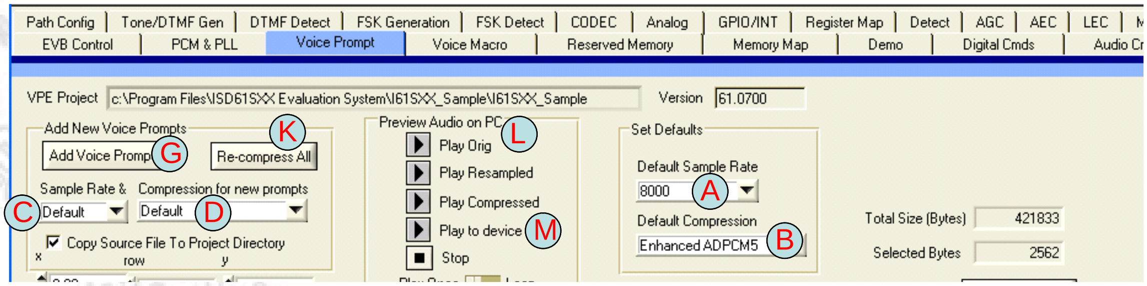
(1) Set sampling rate (C) and compression algorithm (D) for the VPs you are going to add. Users can change the individual VP's sampling rate (E) and/or compression algorithm (F) later when evaluating the quality.
(2) Add voice prompts (G).
(3) Repeat steps (1) & (2).
(4) Change the VP tags (H) if needed; by default, the tags are the filenames.
(5) Re-order the VPs if needed; select VP by clicking on the square (I), and click on the up/down arrows (J) to re-order the VPs.

If users change the clock configuration, they must re-compress all VPs (K) because clock setting is part of the VP compression.

Users can evaluate the VPs through PC or EV board:

- Through PC (L):
  - i. Original voice prompts before re-sampling & compression.
  - ii. Re-sampled but not yet compressed voice prompts.
  - iii. Compressed voice prompts.
- Through EV board:
  1. Plug in the EV board and the 'Play to device' button (M) shows up.

The added voice prompts will appear on the 'Voice Macro' screen for voice macro script programming.



VPE Project: c:\Program Files\ISD61SXX Evaluation System\I61SXX_Sample\I61SXX_Sample
Version: 61.0700

Sample Rate & Compression for new prompts: Default / Default
☑ Copy Source File To Project Directory

Default Sample Rate: 8000
Default Compression: Enhanced ADPCM5

Total Size (Bytes): 421833
Selected Bytes: 2562

Figure 3-4 Voice Prompt screen

3.2.1 Keystrokes to Re-order VPs

Besides the up/down arrows (J), users can also use the keystrokes to re-order the VPs:

- Select VP by clicking on the square (I)
- Press M on keyboard and a pop-up box shows up.
- Enter the new VP # to re-order the VP.

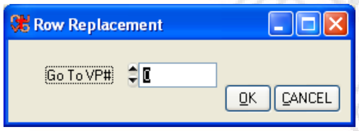


Figure 3-5 Re-order VPs

3.3 Reserved Memory

Recording Memory Pointer (RMP) divides the ISD61S00 memory space into two blocks:

- Reserved Memory,
- Recording Memory.

RMP is a two-byte address pointer pointing to a 4kbyte memory sector which is the first sector available for message recordings. Memory between address zero and the RMP pointer is considered the Reserved Memory for pre-recorded audio (Voice Prompts), pre-programmed macro scripts (Voice Macros), digital read/write access for other applications (User Data), and memory sectors reserved for the first sector of the reserved messages (Reserved Sectors). The memory between the RMP and the end of memory is considered the Recording Memory allocated for recording messages (Message Recordings).

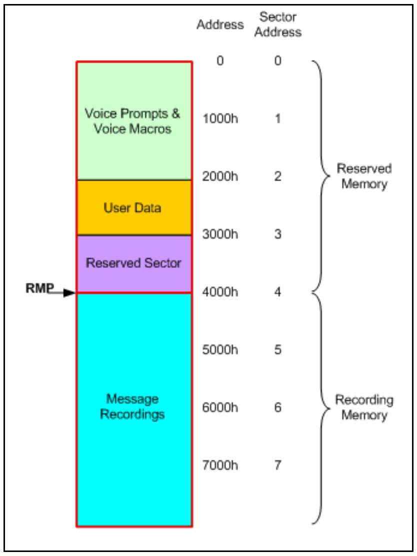


Figure 3-6 Example of ISD61S00 Memory Map

On the ‘Reserved Memory’ screen, users reserve memory sectors for message recordings:

- Empty messages (A):
  - An Empty Message is an empty 4kByte memory sector reserved for message recordings. An SPI record command pointing to the Empty Message starts the message recording and will continue to

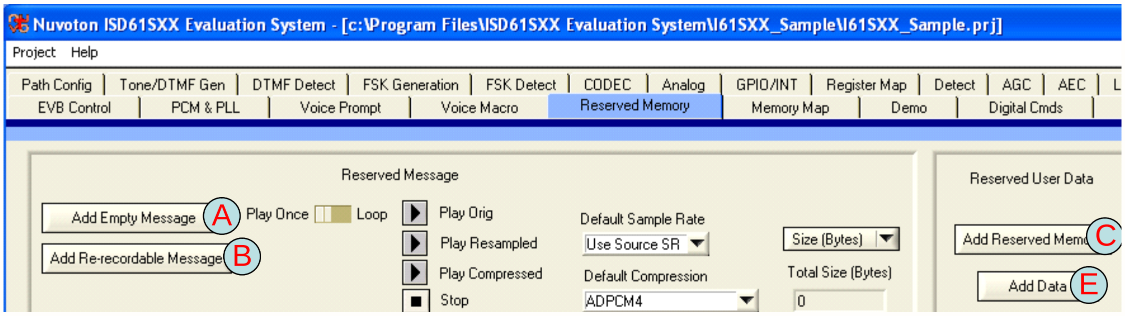
record to the free memory sectors located in Recording Memory if the message recording is over 4kByte.

- Re-recordable messages (B):
  - ○ Users can have a default message when reserving the memory sectors for a message recording. For a re-recordable message, the first sector will be allocated in 'Reserved Sectors'; the rest of the message will be allocated in 'Recording Memory'.

Please note only ADPCM, (D)uLaw, and PCM compressions are supported for reserved messages, not Enhanced-ADPCM or VBR.

To record (or playback) a message, the microcontroller simply issues a record (or playback) command with starting address pointing to the reserved initial address. The reserved messages are also listed on the 'Voice Macro' screen for voice macro script programming.

Memory sectors can also be assigned in Reserved Memory for digital read/write access for other application tasks (User Data). To reserve empty memory sectors, click the button 'Add Reserved Memory' (C) and then adjust the 'Number of Sectors' (D). To reserve (import) existing data, click the button 'Add Data' (E); the file size will be shown on 'number of sectors' (F).



Nuvoton ISD61SXX Evaluation System - [c:\Program Files\ISD61SXX Evaluation System\I61SXX_Sample\I61SXX_Sample.prj]

Default Sample Rate: Use Source SR

Default Compression: ADPCM4

Size (Bytes)

Total Size (Bytes): 0

Figure 3-7 Reserved Memory

3.4 Voice Macro

Voice Macro is a powerful scripting feature integrated into the ISD61S00. Voice Macro editor enables users to create scripted commands to play voice prompts (A), play reserved messages (B), insert silence (C), power down the ISD61S00 (D), and configure the ISD61S00 (E) such as setting play or record paths. As mentioned earlier, the voice prompts (A) are from the 'Voice Prompt' panel, and the reserved messages (B) are from the 'Reserved Memory' panel, respectively. Each VM is assigned a tag and an index, where it can be executed with a single SPI command referring to the index.

There are 3 special Voice Macros:

- Index 0: VM0, Power-On Initialization (POI).
  - ○ Executed automatically when power supplied or when a high pulse applied on RESET pin.
- Index 1: VM1, Power-Up (PU).

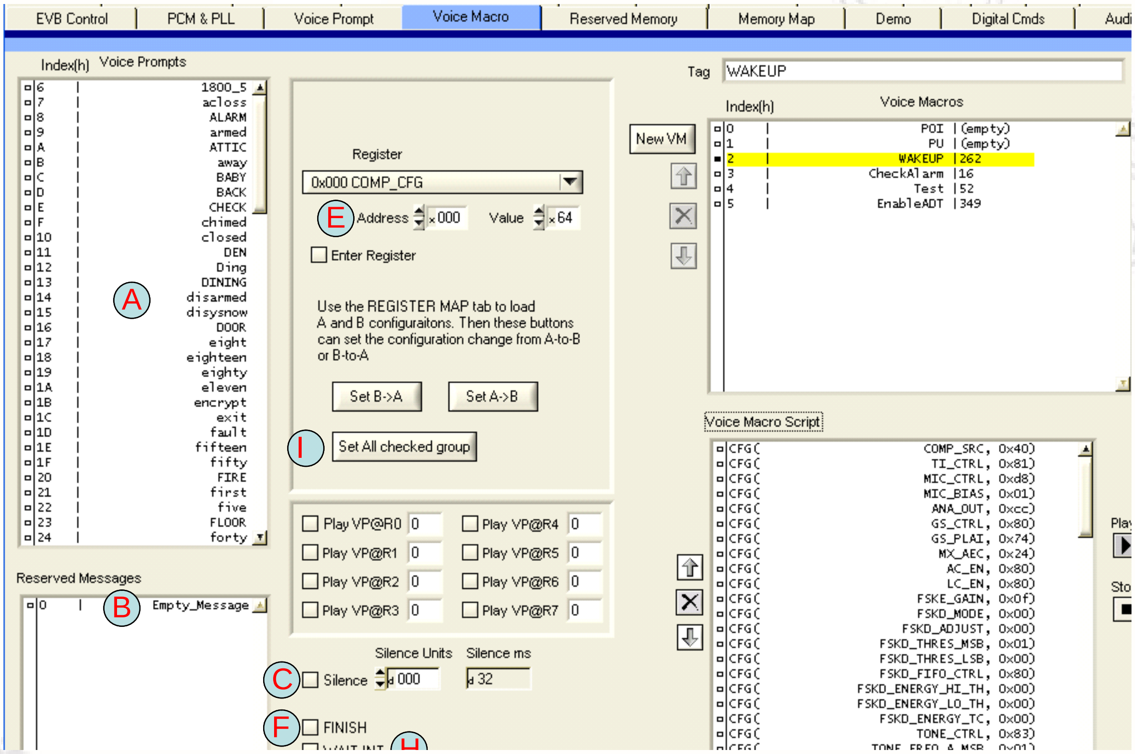
- Executed automatically when received power-up command.

- Index 2: VM2, Power-Up (WAKEUP).
  - Executed automatically when woke up from a GPIO interrupt.

POI, PU, and WAKEUP Macros allow users to customize how the part emerges from power-on, power-up, and wake-up conditions.

There are two special VM commands: FINISH (**F**) and WAIT INT (**G**).

- FINISH (**F**):
  - Finish the voice macro and exit. Voice Macros must be ended in either FINISH or PD (power down).
- WAIT INT (**G**):
  - Wait until current play command finishes before executing next macro instruction.



```
6    |   1800_5
7    |   acloss
8    |   ALARM
9    |   armed
A    |   ATTIC
B    |   away
C    |   BABY
D    |   BACK
E    |   CHECK
F    |   chimed
10   |   closed
11   |   DEN
12   |   Ding
13   |   DINING
14   |   disarmed
15   |   disysnow
16   |   DOOR
17   |   eight
18   |   eighteen
19   |   eighty
1A   |   eleven
1B   |   encrypt
1C   |   exit
1D   |   fault
1E   |   fifteen
1F   |   fifty
20   |   FIRE
21   |   first
22   |   five
23   |   FLOOR
24   |   forty
```

Reserved Messages:

```
0    |   Empty_Message
```

Register: 0x000 COMP_CFG

Address: x 000 Value: x 64

Use the REGISTER MAP tab to load A and B configuraitons. Then these buttons can set the configuration change from A-to-B or B-to-A

□ Play VP@R0 0 □ Play VP@R4 0
□ Play VP@R1 0 □ Play VP@R5 0
□ Play VP@R2 0 □ Play VP@R6 0
□ Play VP@R3 0 □ Play VP@R7 0

Silence Units: d 000 Silence ms: d 32

Tag: WAKEUP

```
0    |        POI  | (empty)
1    |         PU  | (empty)
2    |     WAKEUP  | 262
3    | CheckAlarm  | 16
4    |       Test  | 52
5    |  EnableADT  | 349
```

Voice Macro Script

```
CFG(            COMP_SRC, 0x40)
CFG(             TI_CTRL, 0x81)
CFG(            MIC_CTRL, 0xd8)
CFG(            MIC_BIAS, 0x01)
CFG(             ANA_OUT, 0xcc)
CFG(             GS_CTRL, 0x80)
CFG(             GS_PLAI, 0x74)
CFG(              MX_AEC, 0x24)
CFG(               AC_EN, 0x80)
CFG(               LC_EN, 0x80)
CFG(           FSKE_GAIN, 0x0f)
CFG(           FSKD_MODE, 0x00)
CFG(         FSKD_ADJUST, 0x00)
CFG(      FSKD_THRES_MSB, 0x01)
CFG(      FSKD_THRES_LSB, 0x00)
CFG(      FSKD_FIFO_CTRL, 0x80)
CFG(  FSKD_ENERGY_HI_TH, 0x00)
CFG(  FSKD_ENERGY_LO_TH, 0x00)
CFG(      FSKD_ENERGY_TC, 0x00)
CFG(           TONE_CTRL, 0x83)
CFG(    TONE_FREQ_A_MSB, 0x01)
```

Figure 3-8 Voice Macro

The ISD61S00 has a built-in command buffer so that consecutive commands can be executed one after another seamlessly. However, some commands won't go through command buffer; therefore, they won't wait. Users may refer to the design guide; section 10 SPI Commands, Table 10-2, for details. In table 10-2 users may see that some commands can be executed regardless the status bit-1 CBUF_FULL (command buffer full). WAIT_INT (**H**) is used whenever a command in a voice macro intends to wait to be executed until previous play command finishes. A typical example will be adjusting volume in a voice macro. Commands for adjusting volume won't go through the command buffer; therefore, they will be executed immediately. Below is an example where the volume is immediately decreased when VP_A starts to play.

- Play VP_A
- Decrease Volume

- Play VP_B
- Finish.

If users want to decrease volume after VP_A finishes, the voice macro should be modified as below:

- Play VP_A
- WAIT_INT
- Decrease Volume
- Play VP_B
- Finish.

Please note that the commands in a Voice Macro are executed sequentially, and all voice macros shall be ended with PD or FINISH to finish the voice macro. Users shall see a warning message if the software detects a voice macro not ended with PD or FINISH.

3.5 Path Config

This panel helps users to configure the ISD61S00 Signal Path. The path diagram has 4 main parts:

1. Air input/output (microphone and speaker)
2. Phone-line input/output, and the DTMF/FSK generation and detection.
3. Mixers
4. Compression and de-compression

Figure 3-9 below shows a typical phone signal path:

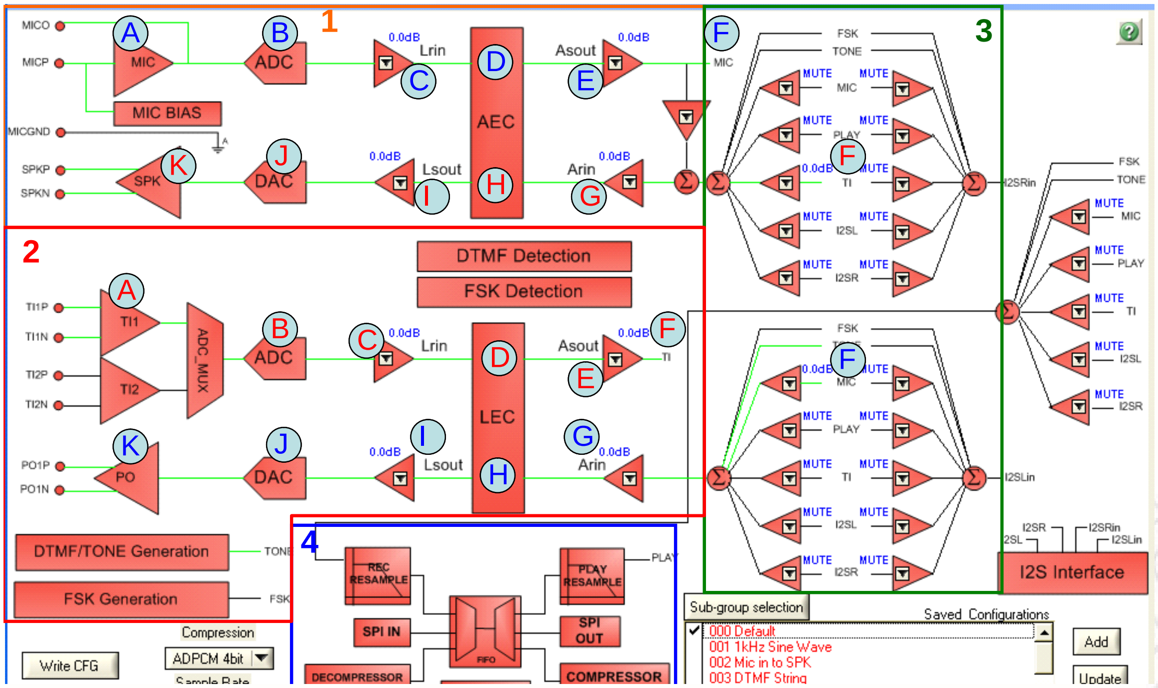
- This site to the other site:
  - Microphone (**A**) -> ADC (**B**) -> Lrin (**C**) -> AEC (**D**) ->Asout (**E**) -> MIC (**F**) -> Arin (**G**) -> LEC (**H**) -> Lsout (**I**) -> DAC (**J**) -> PO (**K**)
- The other site to this site:
  - TI1 (**A**) -> ADC (**B**) -> Lrin (**C**) -> LEC (**D**) ->Asout (**E**) -> TI (**F**) -> Arin (**G**) -> AEC (**H**) -> Lsout (**I**) -> DAC (**J**) -> SPK (**K**)

To enable or disable a path portion, users simply click on the related function block, and the software automatically switches to the corresponding panel. For example, to disable the ADC (**B**), users just click on the ADC block, and then the software will switch to the CODEC panel as in Figure 3-10 . Users then uncheck the Air CODEC (**L**) to disable the ADC. (Please note that ADC and DAC share the hardware circuitry, so the ADC and DAC will be turned on or off at the same time).

Similarly, to tune the AEC (**D**) users just click on the AEC block, and the software utility automatically switches to the AEC panel as in Figure 3-11 . For register details of AEC and LEC, please refer to design guide, section 8.18 Air and Line CODEC.

To change the of gain of Lrin (**C**), Asout (**E**), MIC (**F**) … etc, users just click on the function block and then a gain setting bar shows up as the Figure 3-12 .

The software utility has built in some default path settings as in Figure 3-13 for users' reference. To try a setting, users simply double-click on it. Or, users can click on it, modify the setting as needed, and then click the 'Write CFG'.



Compression ADPCM 4bit

Saved Configurations

☑ 000 Default
001 1kHz Sine Wave
002 Mic in to SPK
003 DTMF String

Figure 3-9 Path Config

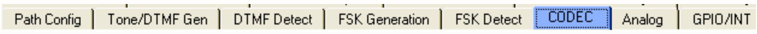
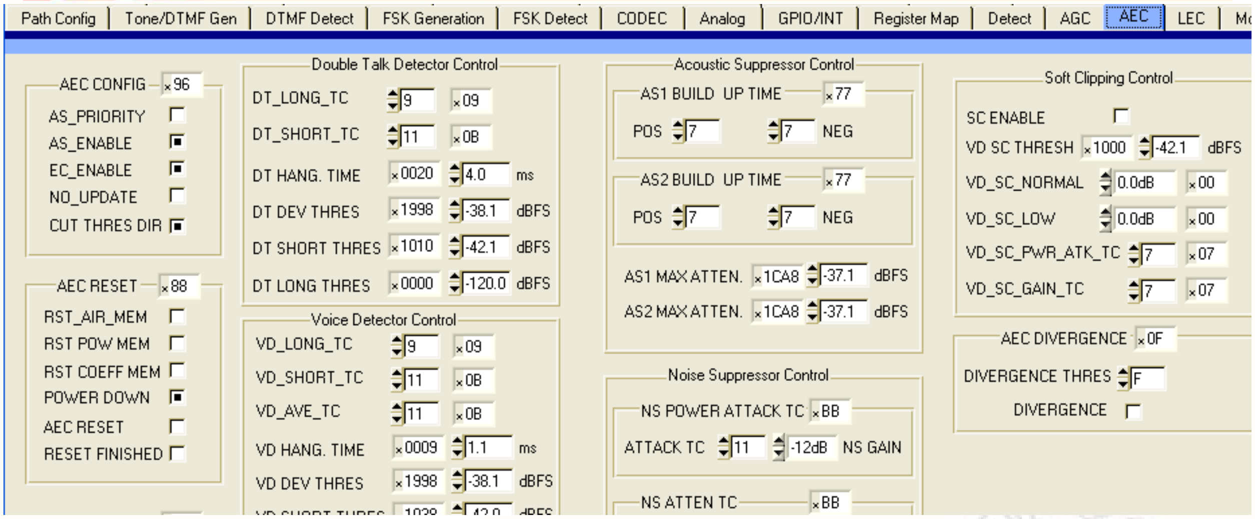


Figure 3-10 CODEC



AEC CONFIG x96
AS_PRIORITY □
AS_ENABLE ☑
EC_ENABLE ☑
NO_UPDATE □
CUT THRES DIR ☑

AEC RESET x88
RST_AIR_MEM □
RST POW MEM □
RST COEFF MEM □
POWER DOWN ☑
AEC RESET □
RESET FINISHED □

Double Talk Detector Control
DT_LONG_TC 9 x09
DT_SHORT_TC 11 x0B
DT HANG. TIME x0020 4.0 ms
DT DEV THRES x1998 -38.1 dBFS
DT SHORT THRES x1010 -42.1 dBFS
DT LONG THRES x0000 -120.0 dBFS

Voice Detector Control
VD_LONG_TC 9 x09
VD_SHORT_TC 11 x0B
VD_AVE_TC 11 x0B
VD HANG. TIME x0009 1.1 ms
VD DEV THRES x1998 -38.1 dBFS

Acoustic Suppressor Control
AS1 BUILD UP TIME x77
POS 7 7 NEG
AS2 BUILD UP TIME x77
POS 7 7 NEG
AS1 MAX ATTEN. x1CA8 -37.1 dBFS
AS2 MAX ATTEN. x1CA8 -37.1 dBFS

Noise Suppressor Control
NS POWER ATTACK TC xBB
ATTACK TC 11 -12dB NS GAIN
NS ATTEN TC xBB

Soft Clipping Control
SC ENABLE □
VD SC THRESH x1000 -42.1 dBFS
VD_SC_NORMAL 0.0dB x00
VD_SC_LOW 0.0dB x00
VD_SC_PWR_ATK_TC 7 x07
VD_SC_GAIN_TC 7 x07

AEC DIVERGENCE x0F
DIVERGENCE THRES F
DIVERGENCE □

Figure 3-11 AEC Parameters

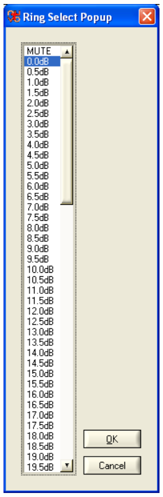
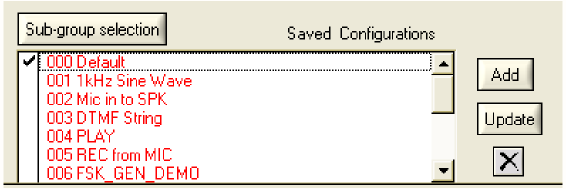


Figure 3-12 Gain Setting



000 Default
001 1kHz Sine Wave
002 Mic in to SPK
003 DTMF String
004 PLAY
005 REC from MIC
006 FSK_GEN_DEMO

Figure 3-13 Default Path Settings

3.6 Memory Map

Memory Map generates the programming file to be burned into the ISD61S00 memory. The programming file is a binary file with .mem extension. On this panel users will see how each VP, VM, and message is assigned an address and/or index. To activate the process, click on ‘Create Programming File’ (**A**). The left hand side displays the memory map (**B**).

On the right hand side, the Memory Protection (**C**) decides how you want the memory contents protected. ISD61S00 provides three kinds of protection mechanisms:

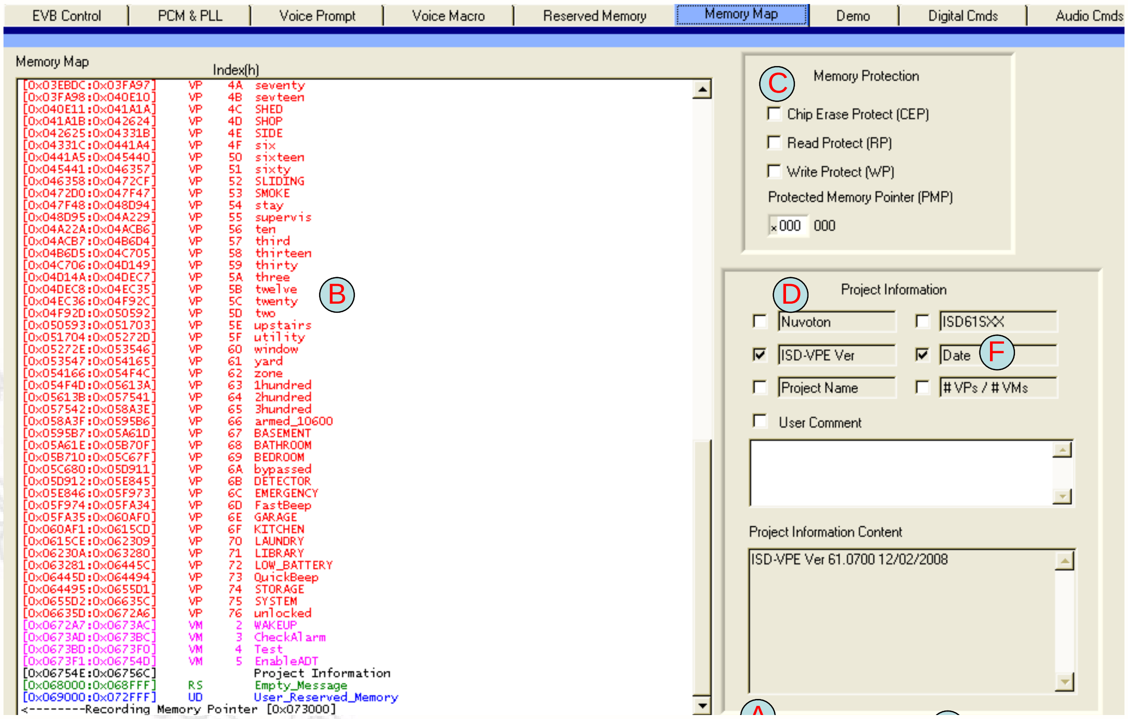
- Read Protection:
  - Once enabled, digital-read command is not effective in those protected area.
- Write Protection:
  - Once enabled, digital-write and memory-erase commands are not effective in those protected area.
- Chip-Erase Protection:
  - Once enabled, chip-erase command is not effective.

If both Chip Erase Protect (CEP) and Write Protect (WP) are checked, it will lock the content from beginning of the memory to PMP, preventing ISD61S00 from any further modification.

The check boxes under 'Project Information' (**D**) allow users to add project information to the programming file. If users check the users comment box, whatever they type in the comment window will be appended to the end of the project information.

If the EV board is connected to PC, the 'Burn Device' button (**E**) will be effective. Once the programming file is created, users can click 'Burn Device' to download the programming file into the ISD61S00.

Please note that 'Date' (**F**) is part of the project, so clicking on the 'Create Programming File' (**A**) may change the project contents even though users don't change anything. When users open an existing project, the memory map will be loaded automatically.



Memory Map

```
                          Index(h)
[0x03EBDC:0x03FA97]   VP   4A  seventy
[0x03FA98:0x040E10]   VP   4B  sevteen
[0x040E11:0x041A1A]   VP   4C  SHED
[0x041A1B:0x042624]   VP   4D  SHOP
[0x042625:0x04331B]   VP   4E  SIDE
[0x04331C:0x0441A4]   VP   4F  six
[0x0441A5:0x045440]   VP   50  sixteen
[0x045441:0x046357]   VP   51  sixty
[0x046358:0x0472CF]   VP   52  SLIDING
[0x0472D0:0x047F47]   VP   53  SMOKE
[0x047F48:0x048D94]   VP   54  stay
[0x048D95:0x04A229]   VP   55  supervis
[0x04A22A:0x04ACB6]   VP   56  ten
[0x04ACB7:0x04B6D4]   VP   57  third
[0x04B6D5:0x04C705]   VP   58  thirteen
[0x04C706:0x04D149]   VP   59  thirty
[0x04D14A:0x04DEC7]   VP   5A  three
[0x04DEC8:0x04EC35]   VP   5B  twelve
[0x04EC36:0x04F92C]   VP   5C  twenty
[0x04F92D:0x050592]   VP   5D  two
[0x050593:0x051703]   VP   5E  upstairs
[0x051704:0x05272D]   VP   5F  utility
[0x05272E:0x053546]   VP   60  window
[0x053547:0x054165]   VP   61  yard
[0x054166:0x054F4C]   VP   62  zone
[0x054F4D:0x05613A]   VP   63  1hundred
[0x05613B:0x057541]   VP   64  2hundred
[0x057542:0x058A3E]   VP   65  3hundred
[0x058A3F:0x0595B6]   VP   66  armed_10600
[0x0595B7:0x05A61D]   VP   67  BASEMENT
[0x05A61E:0x05B70F]   VP   68  BATHROOM
[0x05B710:0x05C67F]   VP   69  BEDROOM
[0x05C680:0x05D911]   VP   6A  bypassed
[0x05D912:0x05E845]   VP   6B  DETECTOR
[0x05E846:0x05F973]   VP   6C  EMERGENCY
[0x05F974:0x05FA34]   VP   6D  FastBeep
[0x05FA35:0x060AF0]   VP   6E  GARAGE
[0x060AF1:0x0615CD]   VP   6F  KITCHEN
[0x0615CE:0x062309]   VP   70  LAUNDRY
[0x06230A:0x063280]   VP   71  LIBRARY
[0x063281:0x06445C]   VP   72  LOW_BATTERY
[0x06445D:0x064494]   VP   73  QuickBeep
[0x064495:0x0655D1]   VP   74  STORAGE
[0x0655D2:0x06635C]   VP   75  SYSTEM
[0x06635D:0x0672A6]   VP   76  unlocked
[0x0672A7:0x0673AC]   VM    2  WAKEUP
[0x0673AD:0x0673BC]   VM    3  CheckAlarm
[0x0673BD:0x0673F0]   VM    4  Test
[0x0673F1:0x06754D]   VM    5  EnableADT
[0x06754E:0x06756C]            Project Information
[0x068000:0x068FFF]   RS       Empty_Message
[0x069000:0x072FFF]   UD       User_Reserved_Memory
<--------Recording Memory Pointer [0x073000]
```

Memory Protection

☐ Chip Erase Protect (CEP)
☐ Read Protect (RP)
☐ Write Protect (WP)

Protected Memory Pointer (PMP)

x000 000

Project Information

☐ Nuvoton	☐ ISD61SXX
☑ ISD-VPE Ver	☑ Date
☐ Project Name	☐ # VPs / # VMs

☐ User Comment

Project Information Content

ISD-VPE Ver 61.0700 12/02/2008

Figure 3-14 Memory Map

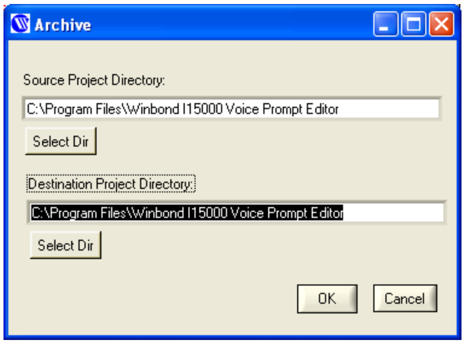
3.7 Archive and Clean

Under project menu there are two very useful functions.

1. Archive and,
2. Clean

These two functions are used to backup your project files and clean up unnecessary files. They are useful when you want to send an ISD61S00 project to other people.

The archiving function duplicates a project to an archive directory with exactly same project name and structure, and changes the project links to point to this directory so that all paths are stored relative to the project directory. For example, if you create an ISD61S00 project in a network share folder which is, say, mapped to T drive, you cannot just send this project to other people because “T” drive means nothing to other people. By archiving the project, that project is now self contained and can just be copied to another machine or to backup.



Source Project Directory: C:\Program Files\Winbond I15000 Voice Prompt Editor

Destination Project Directory: C:\Program Files\Winbond I15000 Voice Prompt Editor

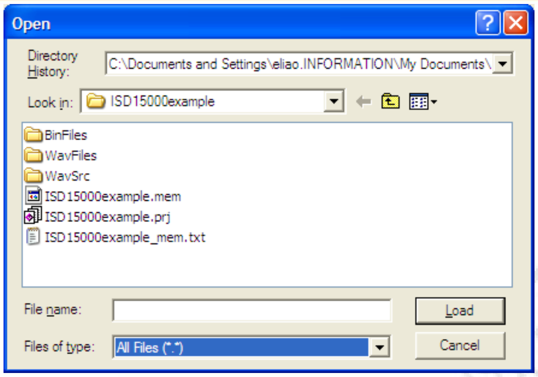
Figure 3-15 Archive

The Clean function deletes the contents in the BinFile and WavFile directories as they can be regenerated. The WavFile directory contains wave files of re-sampled and de-compressed voice prompts. The BinFile directory contains files of compressed voice prompts.

When a project is created, three sub-directories are created:

- BinFiles: where the compressed VPs are stored.
- WavFiles: where the re-sampled and de-compressed VPs are stored.
- WavSrc: where wave source of VPs are stored.

A project file has the prj extension. The binary programming file for the project has mem extension. A header file with txt extension is also included for C language firmware programming.



Directory History: C:\Documents and Settings\eliao.INFORMATION\My Documents\

Look in: ISD15000example

File name:

Files of type: All Files (*.*)

Figure 3-16 Open a Project

3.8 DTMF Detect

DTMF Detect can decode the digits in off-hook state and display them in (A).

To run the function, follow the steps below and refer to Figure 3-17:

(1) Press the "GO" (B) button to load the default registers and the new registers will update to the panel immediately. Then the EVB will decode the default digits automatically and display them in (A).

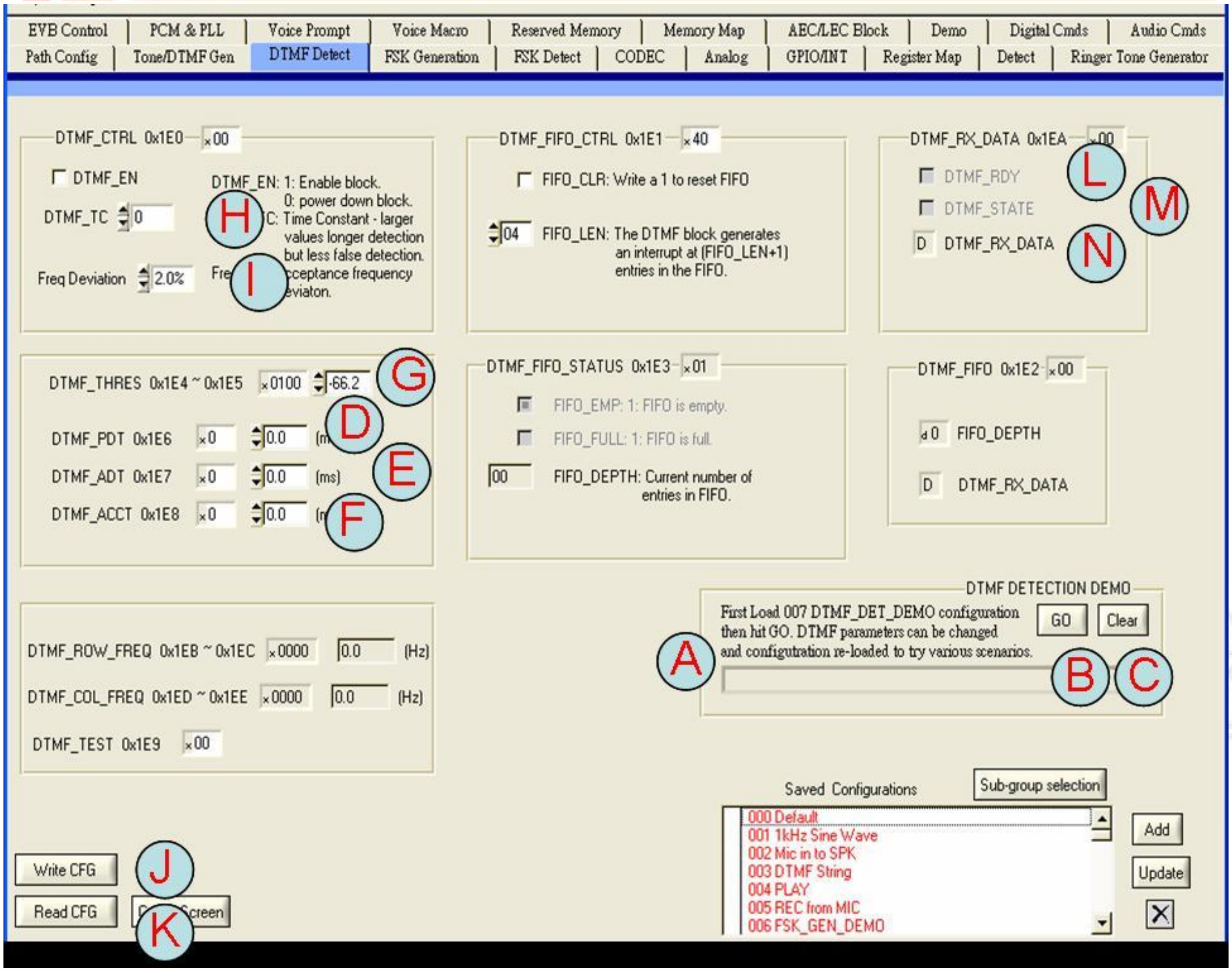
(2) Press the "Clear" (C) button to clear the decoded digits in (A) if needed; send more digits from other telephone through telephone line. And the decoded digits are also shown in (A).

(3) If the decoded digits are not correct, tune the timing (D) (E) (F) according to Figure 3-18, tune the threshold (G), DTMF_TC (H) and Freq Deviation(I). And then Press "Write CFG" (J) button to change the conditions. Press "Read CFG" (K) to confirm the conditions if needed.

(4) Send the digits again.

(5) If the decoded digits are still not correct, repeat steps (3) & (4).

According to Figure 3-18, users can also monitor the immediate states: DTMF_RDY(L), DTMF_STATE(M) and DTMF_RX_DATA(N).



DTMF_CTRL 0x1E0 x00

☐ DTMF_EN

DTMF_TC 0

Freq Deviation 2.0%

DTMF_EN: 1: Enable block.
0: power down block.
DTMF_TC: Time Constant - larger values longer detection but less false detection.
Freq Deviation: Acceptance frequency deviaton.

DTMF_FIFO_CTRL 0x1E1 x40

☐ FIFO_CLR: Write a 1 to reset FIFO

04 FIFO_LEN: The DTMF block generates an interrupt at (FIFO_LEN+1) entries in the FIFO.

DTMF_RX_DATA 0x1EA x00

☐ DTMF_RDY

☐ DTMF_STATE

D DTMF_RX_DATA

DTMF_THRES 0x1E4 ~ 0x1E5 x0100 -66.2

DTMF_PDT 0x1E6 x0 0.0 (ms)

DTMF_ADT 0x1E7 x0 0.0 (ms)

DTMF_ACCT 0x1E8 x0 0.0 (ms)

DTMF_FIFO_STATUS 0x1E3 x01

☑ FIFO_EMP: 1: FIFO is empty.

☐ FIFO_FULL: 1: FIFO is full.

00 FIFO_DEPTH: Current number of entries in FIFO.

DTMF_FIFO 0x1E2 x00

d 0 FIFO_DEPTH

D DTMF_RX_DATA

DTMF_ROW_FREQ 0x1EB ~ 0x1EC x0000 0.0 (Hz)

DTMF_COL_FREQ 0x1ED ~ 0x1EE x0000 0.0 (Hz)

DTMF_TEST 0x1E9 x00

DTMF DETECTION DEMO

First Load 007 DTMF_DET_DEMO configuration then hit GO. DTMF parameters can be changed and configutration re-loaded to try various scenarios.

Saved Configurations

000 Default
001 1kHz Sine Wave
002 Mic in to SPK
003 DTMF String
004 PLAY
005 REC from MIC
006 FSK_GEN_DEMO

Figure 3-17 DTMF Detect screen

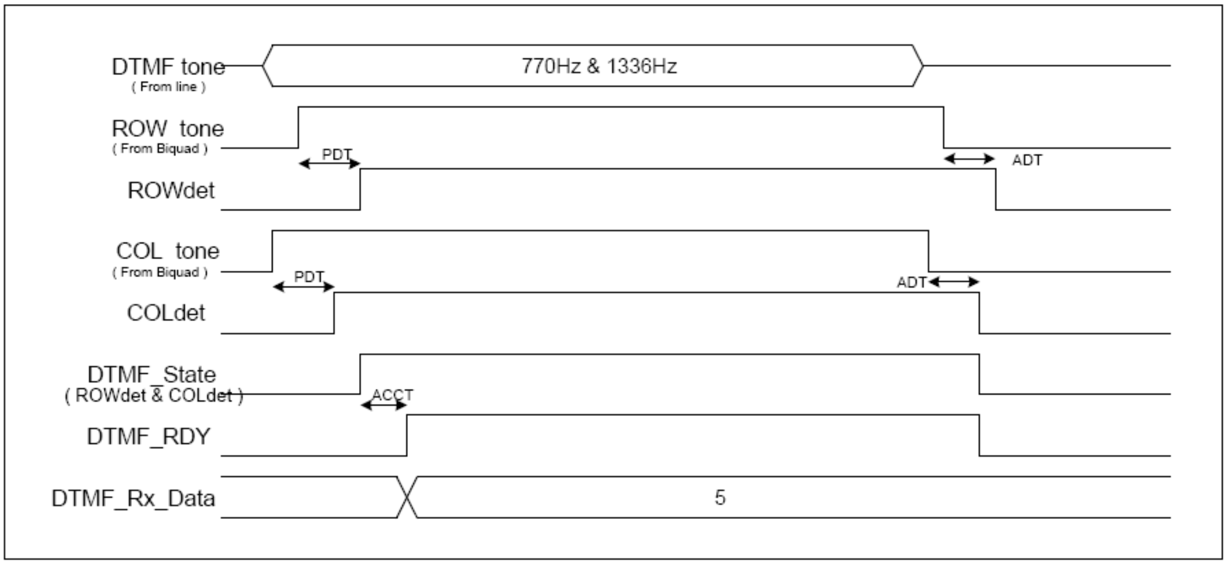


Figure 3-18 DTMF Detector Acquisition timing

3.9 Tone/DTMF Gen

Tone/DTMF generator operates in three different modes:

1. DTMF mode
2. Programmable mode
3. Continuous mode

If user set the DTMF_MODE to 1, the function is in the DTMF mode. If user set the DTMF_MODE to 0, the function is in the programmable and continuous mode. The setting of DTMF_MODE bit explained as following table:

Mode	TONE_FREQ_A TONE_FREQ_B	TONE_LEVEL_A TONE_LEVEL_B	TONE_ON_TIME TONE_OFF_TIME	TONE_LENGTH	TONES TO GEN
Continuous	User define	User define	Set: 0	Set: 0	Don't care
DTMF	Don't care	User define	User define	Number of digits in tone index registers.	Digits to send
Programmable	User define	User define	User define	Number of tones to generate	Don't care

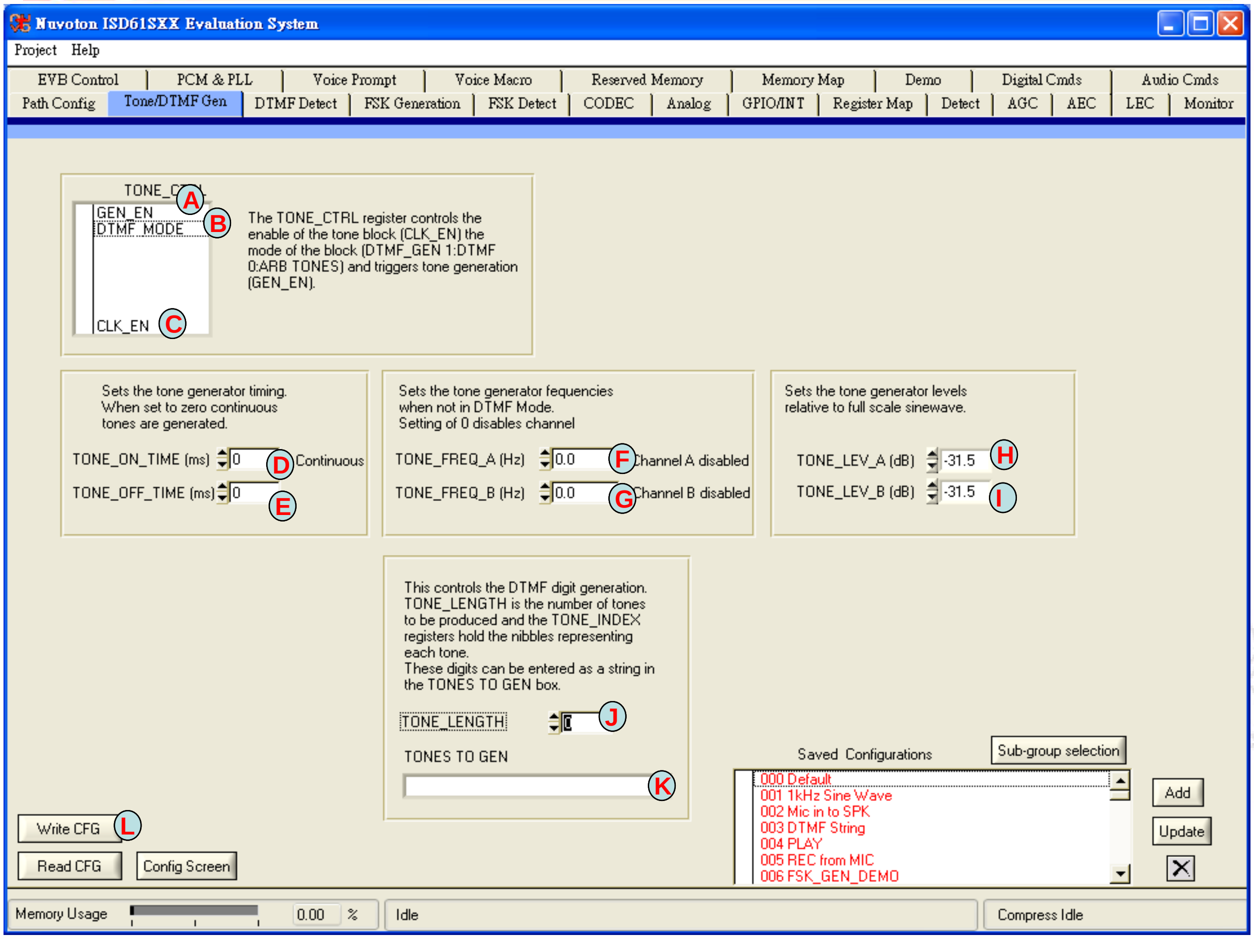
For the DTMF mode,

(1) Checking GEN_EN (A), DTMF_MODE (B) and CLK_EN (C)

(2) Setting TONE_ON_TIME (D), TONE_OFF_TIME (E), TONE_LEV_A (H) and TONE_LEV_B (I).

(3) To input the number of DTMF digits in the TONE_LENGTH (J) and the numeral you want to generate in TONES TO GEN (K).

(4) Clicking Write CFG (L), the DTMF tone of the TONES TO GEN (K) would be generated.



The TONE_CTRL register controls the enable of the tone block (CLK_EN) the mode of the block (DTMF_GEN 1:DTMF 0:ARB TONES) and triggers tone generation (GEN_EN).

Sets the tone generator timing. When set to zero continuous tones are generated.

TONE_ON_TIME (ms) 0 Continuous

TONE_OFF_TIME (ms) 0

Sets the tone generator fequencies when not in DTMF Mode. Setting of 0 disables channel

TONE_FREQ_A (Hz) 0.0 Channel A disabled

TONE_FREQ_B (Hz) 0.0 Channel B disabled

Sets the tone generator levels relative to full scale sinewave.

TONE_LEV_A (dB) -31.5

TONE_LEV_B (dB) -31.5

This controls the DTMF digit generation. TONE_LENGTH is the number of tones to be produced and the TONE_INDEX registers hold the nibbles representing each tone. These digits can be entered as a string in the TONES TO GEN box.

TONE_LENGTH 0

TONES TO GEN

000 Default
001 1kHz Sine Wave
002 Mic in to SPK
003 DTMF String
004 PLAY
005 REC from MIC
006 FSK_GEN_DEMO

Memory Usage 0.00 %

Idle

Figure 3-19 Tone/DTMF Gen

The frequency of DTMF is defaulted as following table:

		Column frequency			
		1209 Hz	**1336 Hz**	**1477 Hz**	**1633 Hz**
Row Frequency	**697 Hz**	0x01 hex	0x02 hex	0x03 hex	0x0D hex
		1	2	3	A
	770 Hz	0x04 hex	0x05 hex	0x06 hex	0x0E hex
		4	5	6	B
	852 Hz	0x07 hex	0x08 hex	0x09 hex	0x0F hex
		7	8	9	C
	941 Hz	0x0B hex	0x0A hex	0x0C hex	0x00 hex
		*	0	#	D

For Programmable mode,

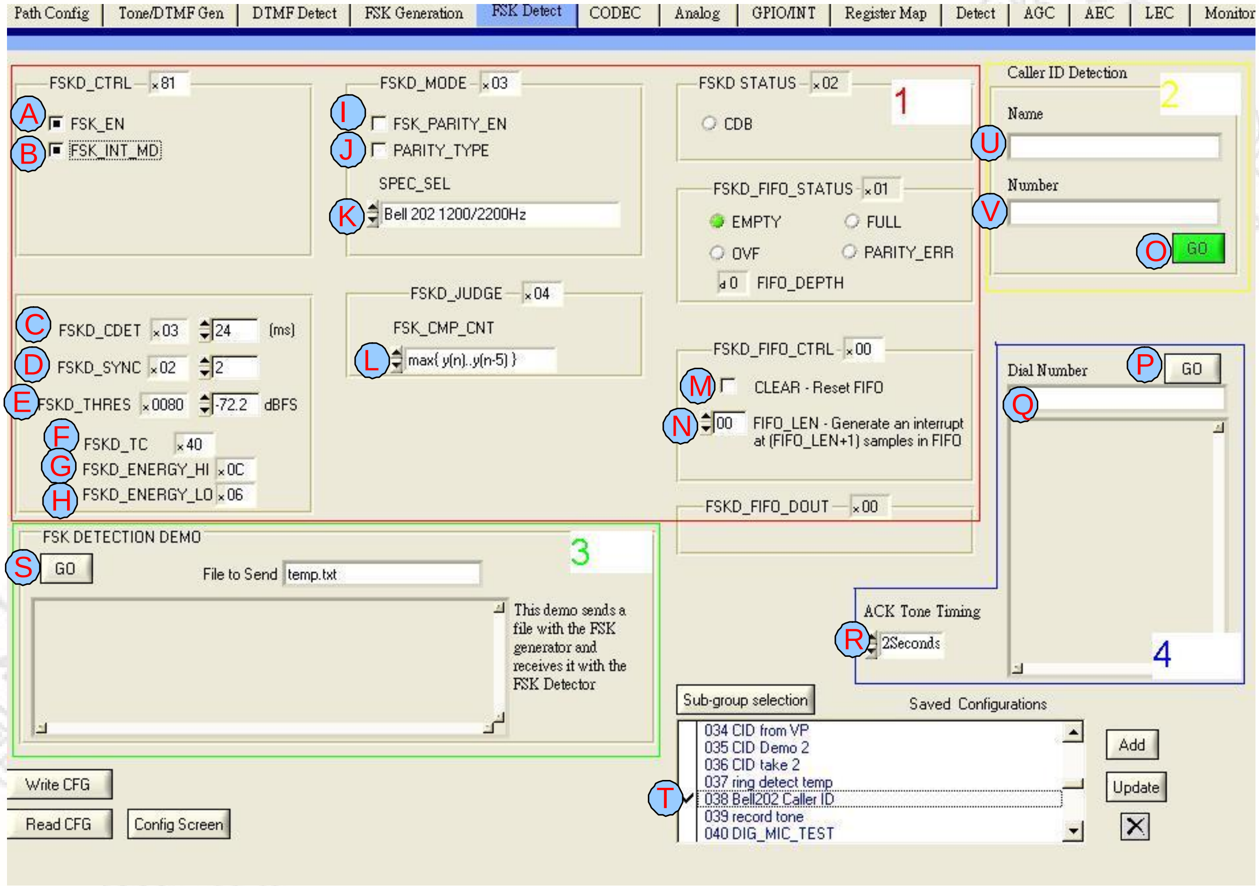
(1) Checking GEN_EN (A) and CLK_EN (B)

(2) Setting TONE_ON_TIME (D), TONE_OFF_TIME (E), TONE_FREQ_A (F), TONE_FREQ_B (G), TONE_LEV_A (H) and TONE_LEV_B (I).
(3) To input the number of tones you want to generate in TONE_LENGTH (J).
(4) Click the Write CFG (L), the tone would be generated for TONE_LENGTH (J) times

For continuous mode,
(1) Checking GEN_EN (A) and CLK_EN (B)
(2) Setting TONE_ON_TIME (D) and TONE_OFF_TIME (E) to zero
(3) Setting TONE_FREQ_A (F), TONE_FREQ_B (G), TONE_LEV_A (H) and TONE_LEV_B (I)
(4) Clicking Write CFG (L). The continuous tone would be generated.

3.10 FSK Detect

The ISD61S00 has a FSK Detector supporting Bell202, Bell103, ITU-T V.21 and ITU-T V.23 standard FSK. The control panel is



FSKD_CTRL x81
☒ FSK_EN
☒ FSK_INT_MD

FSKD_MODE x03
☐ FSK_PARITY_EN
☐ PARITY_TYPE
SPEC_SEL
Bell 202 1200/2200Hz

FSKD STATUS x02
○ CDB

FSKD_FIFO_STATUS x01
● EMPTY ○ FULL
○ OVF ○ PARITY_ERR
d 0 FIFO_DEPTH

FSKD_CDET x03 24 (ms)
FSKD_SYNC x02 2
FSKD_THRES x0080 -72.2 dBFS
FSKD_TC x40
FSKD_ENERGY_HI x0C
FSKD_ENERGY_LO x06

FSKD_JUDGE x04
FSK_CMP_CNT
max{ y(n)..y(n-5) }

FSKD_FIFO_CTRL x00
☐ CLEAR - Reset FIFO
00 FIFO_LEN - Generate an interrupt at (FIFO_LEN+1) samples in FIFO

FSKD_FIFO_DOUT x00

Caller ID Detection
Name
Number

Dial Number

ACK Tone Timing
2Seconds

FSK DETECTION DEMO
File to Send temp.txt
This demo sends a file with the FSK generator and receives it with the FSK Detector

Saved Configurations
034 CID from VP
035 CID Demo 2
036 CID take 2
037 ring detect temp
✓ 038 Bell202 Caller ID
039 record tone
040 DIG_MIC_TEST

Figure 3-20 FSK Detect

Area 1 is for general FSK detect control register setting. Area 2, 3 and 4 are for demo only.
(1) FSK Detect control registers:

- FSK detection parameter:

FSKD_CDET (C), FSK_SYNC(D), FSKD_THRES(E), FSKD_TC(F), FSKD_ENERGY_HI(G), FSKD_ENERGY_LO(H), FSKD_MODE(K), FSKD_JUDGE(L).

- FSKD_FIFO_CTRL: control the FSK detection FIFO, to clear the FIFO, click 'CLEAR'(M) and Press 'Write CFG', the FSK detector FIFO will reset to a clean state.
- FSKD_Status: shows the FSK detect CDB and current demodulated data bit.
- FSKD_FIFO_STATUS: shows current FSK detection FIFO status.
- FSKD_FIFO_DOUT: current FIFO head.

To receive type 1 FSK CID message, the procedure is:
Step1: click FSK_EN(A) and FSK_INT_MD(B) to checked state.
Step2: configure FSKD_CDET(C), this value at least greater than 3.
Step3: configure FSKD_SYNC(D), this vale at least greater than 2.
Step4: configure FSKD_THRES(E), control the FSK detector sensitivity. The smaller, the more sensitive.
Step5: configure FSKD_TC(F), FSKD_ENERGY_HI(G) and FSKD_ENERGY_LO(H). These 3 parameters only used when Bell202 or V.23 is chose. For other FSK spec (Bell103 and V.21), these 3 parameters has no effect.
Step6: configure FSK_PARITY_EN(I), PARITY_TYPE(J), SPEC_SEL(K).
Step7: configure FSK_CMP_CNT(L)
Step8: clear FIFO by click 'CLEAR'(M), set the FIFO_LEN(N) to decide INT occur depth.
Step9: Press 'GO'.
Step10: configure SPEC_SEL(K) again.
Step11: Press 'Write CFG'
Step12: When FSK detector receives the message, it will display the message in the text box.

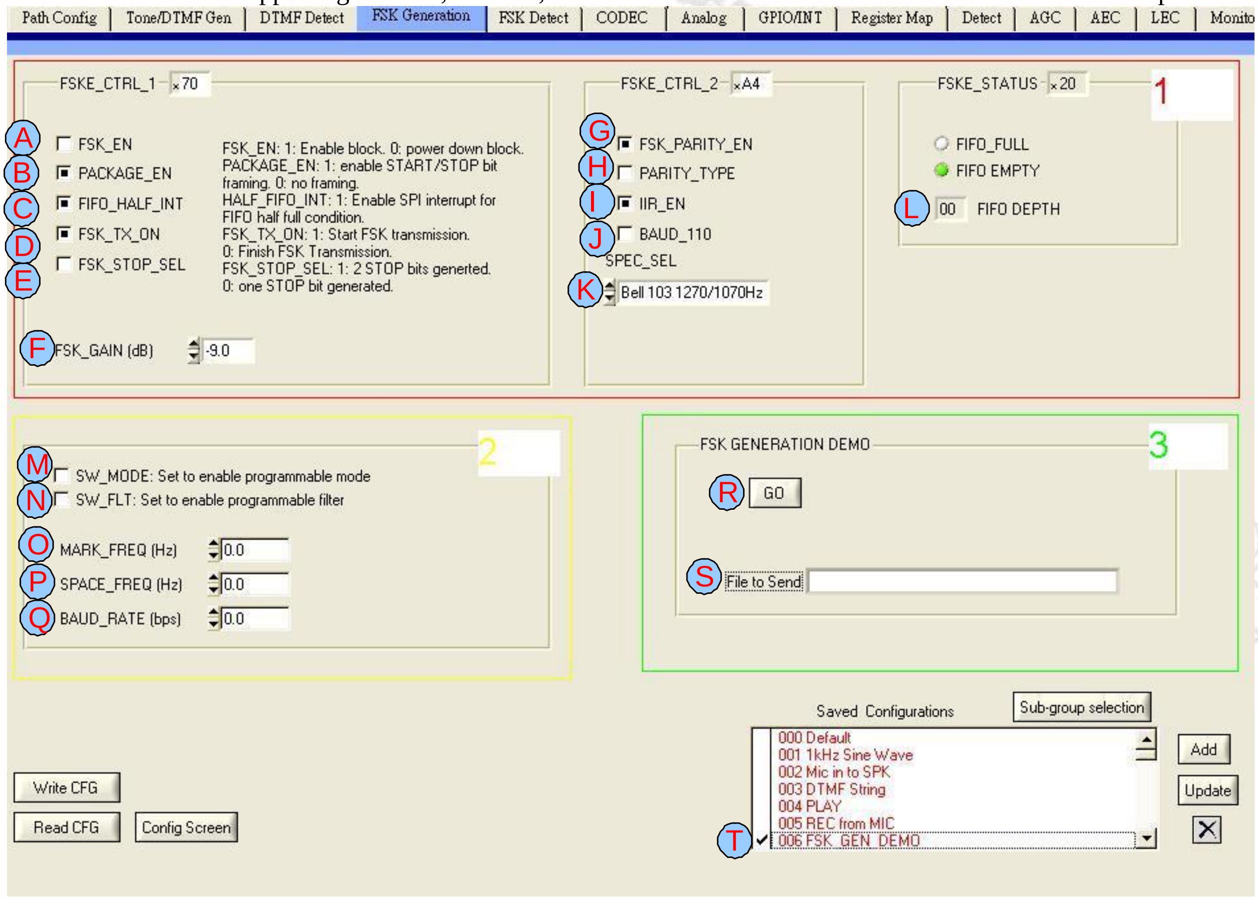
(2) Caller ID Detection : Demo area for Bell202 CID detection, the procedure is:
Step1: click Bell202 Caller ID(T) in the right down configuration area.
Step2: Press 'GO' button in 2nd area.
Step3: When ISD61S receive a session of FSK CID signal, the Name(U) and Number(V) will display.

(3) FSK DETECION DEMO : Demo for FSK generator and receive with Detector, the procedure is:
Step1: select FSK spec in the SPEC_SEL(K) text box and press 'Write CFG'.
Step2: prepare a text file with message in it.
Step3: Press 'GO'(S) in 3rd area. The GUI will ask user to provide a file and then control ISD61S FSK generator to send this text in FSK spec selected in SPEC_SEL(K). When FSK detector receives the message, it will display the message in the text box.

(4) Dedicate area for security product. The procedure is:
Step1: Select ACK Tone timing(R).
Step2: Key in the dial number in the Dial Number text box(Q).
Step3: Press 'GO'(P) and GUI will control ISD61S to communicate with a main machine.

3.11 FSK Generation

The FSK Generation supporting Bell202, Bell103, ITU-T V.21 and ITU-T V.23 standard FSK. The control panel is



FSKE_CTRL_1 x70

- ☐ FSK_EN
- ☑ PACKAGE_EN
- ☑ FIFO_HALF_INT
- ☑ FSK_TX_ON
- ☐ FSK_STOP_SEL

FSK_EN: 1: Enable block. 0: power down block.
PACKAGE_EN: 1: enable START/STOP bit framing. 0: no framing.
HALF_FIFO_INT: 1: Enable SPI interrupt for FIFO half full condition.
FSK_TX_ON: 1: Start FSK transmission. 0: Finish FSK Transmission.
FSK_STOP_SEL: 1: 2 STOP bits generted. 0: one STOP bit generated.

FSK_GAIN (dB) -9.0

FSKE_CTRL_2 xA4

- ☑ FSK_PARITY_EN
- ☐ PARITY_TYPE
- ☑ IIR_EN
- ☐ BAUD_110

SPEC_SEL: Bell 103 1270/1070Hz

FSKE_STATUS x20

- ○ FIFO_FULL
- ● FIFO EMPTY

00 FIFO DEPTH

- ☐ SW_MODE: Set to enable programmable mode
- ☐ SW_FLT: Set to enable programmable filter

MARK_FREQ (Hz) 0.0
SPACE_FREQ (Hz) 0.0
BAUD_RATE (bps) 0.0

FSK GENERATION DEMO

File to Send

Saved Configurations

000 Default
001 1kHz Sine Wave
002 Mic in to SPK
003 DTMF String
004 PLAY
005 REC from MIC
✓ 006 FSK_GEN_DEMO

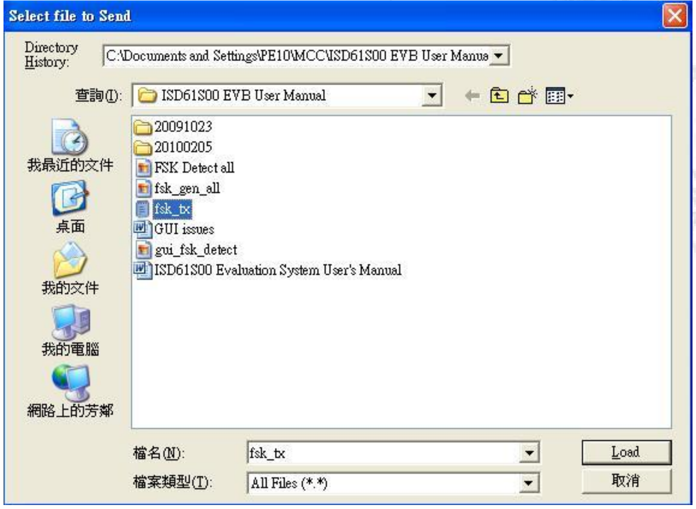
Figure 3-21 FSK Generation

Area 1 is FSK Encode control setting and status. Area 2 is specific customized mark and space baud rate setting. When SW_MODE(M) is checked, the SPEC_SEL(K) has no effect. Area 3 is for demo the transmit text with area 1 and area 2 setting.

- To quickly select a FSK_GEN_DEMO:
  Double click (T), if speaker is on, user should hear 1270Hz tone.

- To generate type 2 FSK signal:
  1. Click FSK_EN(A)
  2. Click PACKAGE_EN(B) to 1, this will insert start bit before data and stop bit behind data to construct a frame.
  3. Click FIFO_HALF_INT(C), INT will occur when Transmit FIFO depth reaches half full condition.
  4. Click FSK_TX_ON(D)
  5. Configure FSK_STOP_SEL(E). 1 for two stop bits, 0 for 1 stop bit.
  6. Configure FSK generation gain(F).
  7. Configure FSK_PARITY_EN(G). 1 add parity bit. 0 no parity bit.
  8. Configure FSK PARITY_TYPE(H). 0 for odd parity. 1for even parity.
  9. Click IIR_EN(I) to 1

10. Configure BAUD_110(J) to 1 if baud rate 110 is required.
11. Configure SPEC_SEL(K) to target FSK standard.
12. Prepare a text file with transmit content.
13. Press GO(R) a File input dialog will show



Directory History: C:\Documents and Settings\PE10\MCC\ISD61S00 EVB User Manua

查詢(I): ISD61S00 EVB User Manual

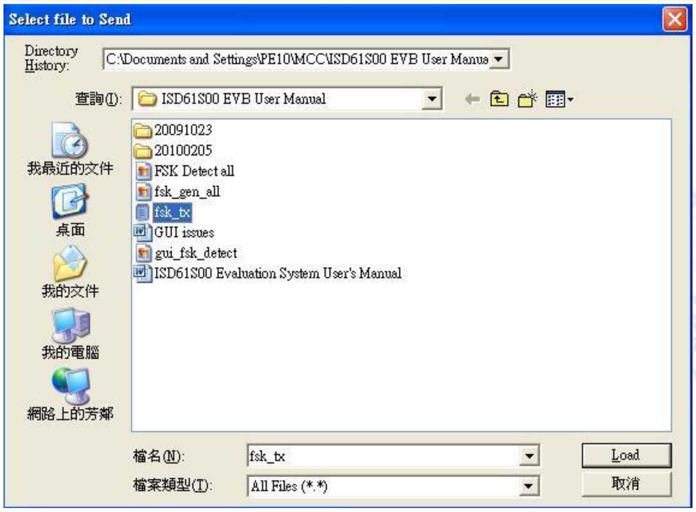
20091023
20100205
FSK Detect all
fsk_gen_all
fsk_tx
GUI issues
gui_fsk_detect
ISD61S00 Evaluation System User's Manual

檔名(N): fsk_tx

檔案類型(T): All Files (*.*)

14. Select the file and Press ‘Load’. The line interface will send FSK signal based on the text file.
15. To repeat transmit, press GO(R) again.

- To use SW Mode to generate type 2 FSK signal:
  1. Click FSK_EN(A)
  2. Click PACKAGE_EN(B) to 1, this will insert start bit before data and stop bit behind data to construct a frame.
  3. Click FIFO_HALF_INT(C), INT will occur when Transmit FIFO depth reaches half full condition.
  4. Click FSK_TX_ON(D)
  5. Configure FSK_STOP_SEL(E). 1 for two stop bits, 0 for 1 stop bit.
  6. Configure FSK generation gain(F).
  7. Configure FSK_PARITY_EN(G). 1 add parity bit. 0 no parity bit.
  8. Configure FSK PARITY_TYPE(H). 0 for odd parity. 1for even parity.
  9. Click IIR_EN(I) to 1
  10. Click SW_MODE(M)
  11. Configure MARK_FREQ(O)
  12. Configure SPACE_FREQ(P).
  13. Configure BAUDRATE(Q).
  14. Prepare a text file with transmit content.
  15. Press GO(R) a File input dialog will show



Directory History: C:\Documents and Settings\PE10\MCC\ISD61S00 EVB User Manua

查詢(I): ISD61S00 EVB User Manual

20091023
20100205
FSK Detect all
fsk_gen_all
fsk_tx
GUI issues
gui_fsk_detect
ISD61S00 Evaluation System User's Manual

檔名(N): fsk_tx

檔案類型(T): All Files (*.*)

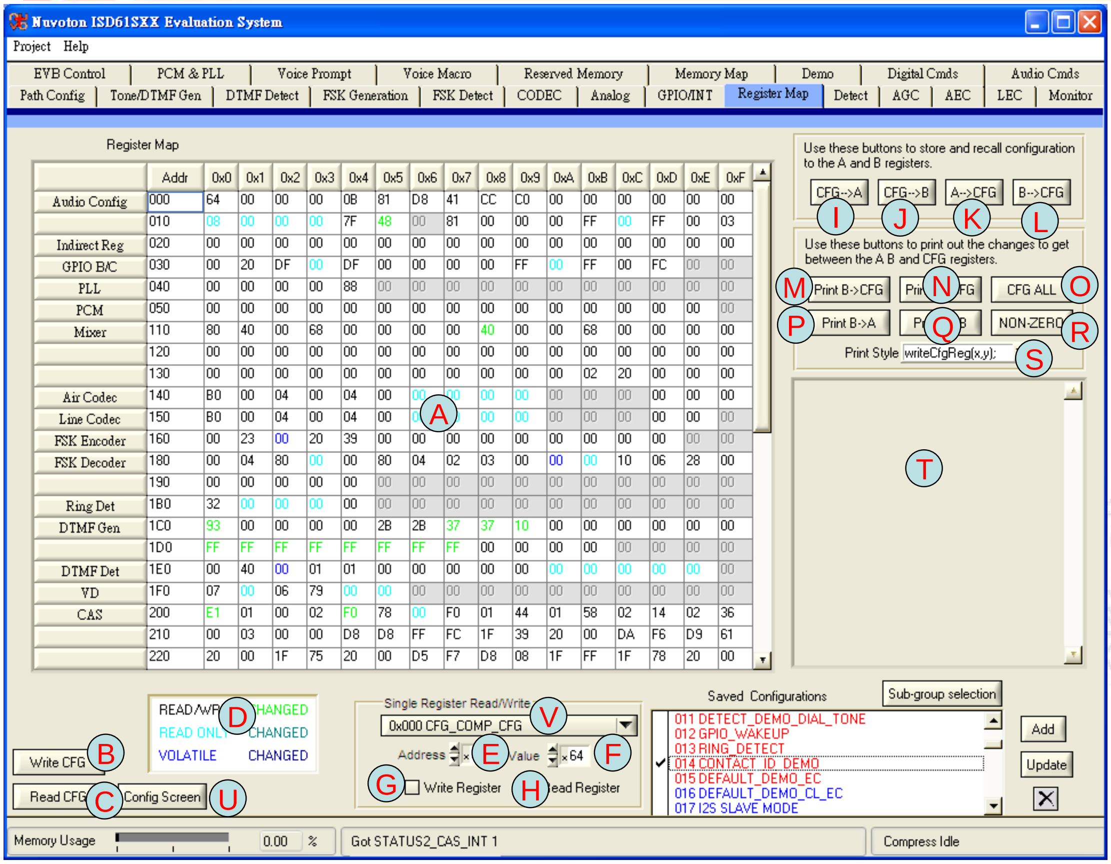
16. Select the file and Press ‘Load’. The line interface will send FSK signal based on the text file.
17. To repeat transmition, press GO(R) again.

3.12 Register Map

The Register Map shows the register status. There are two ways to modify the register values:

(1) Modify many register values at the same time: To modify the value in the register map (A) at different addresses, click Write CFG (B), the modified value would be written into these registers. Clicking Read CFG (C), the register map shows the value you just modified. The color of modified value would be changed. The illustration is in the textbox (D).

(2) Modify one register value at a time: to choose one address (E) and modify the value (F), button (V) would show the address and register name. Clicking Write Register (G), the register map (A) shows the changed value. Clicking Read Register (H) to check if the value is written into register.



	Addr	0x0	0x1	0x2	0x3	0x4	0x5	0x6	0x7	0x8	0x9	0xA	0xB	0xC	0xD	0xE	0xF
Audio Config	000	64	00	00	00	0B	81	D8	41	CC	C0	00	00	00	00	00	00
	010	08	00	00	00	7F	48	00	81	00	00	00	FF	00	FF	00	03
Indirect Reg	020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
GPIO B/C	030	00	20	DF	00	DF	00	00	00	00	FF	00	FF	00	FC	00	00
PLL	040	00	00	00	00	88	00	00	00	00	00	00	00	00	00	00	00
PCM	050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Mixer	110	80	40	00	68	00	00	00	00	40	00	00	68	00	00	00	00
	120	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
	130	00	00	00	00	00	00	00	00	00	00	00	02	20	00	00	00
Air Codec	140	B0	00	04	00	04	00	00	00	00	00	00	00	00	00	00	00
Line Codec	150	B0	00	04	00	04	00	00	00	00	00	00	00	00	00	00	00
FSK Encoder	160	00	23	00	20	39	00	00	00	00	00	00	00	00	00	00	00
FSK Decoder	180	00	04	80	00	00	80	04	02	03	00	00	00	10	06	28	00
	190	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
Ring Det	1B0	32	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
DTMF Gen	1C0	93	00	00	00	00	2B	2B	37	37	10	00	00	00	00	00	00
	1D0	FF	FF	FF	FF	FF	FF	FF	FF	00	00	00	00	00	00	00	00
DTMF Det	1E0	00	40	00	01	01	00	00	00	00	00	00	00	00	00	00	00
VD	1F0	07	00	06	79	00	00	00	00	00	00	00	00	00	00	00	00
CAS	200	E1	01	00	02	F0	78	00	F0	01	44	01	58	02	14	02	36
	210	00	03	00	00	D8	D8	FF	FC	1F	39	20	00	DA	F6	D9	61
	220	20	00	1F	75	20	00	D5	F7	D8	08	1F	FF	1F	78	20	00

011 DETECT_DEMO_DIAL_TONE
012 GPIO_WAKEUP
013 RING_DETECT
✓ 014 CONTACT_ID_DEMO
015 DEFAULT_DEMO_EC
016 DEFAULT_DEMO_CL_EC
017 I2S SLAVE MODE

Figure 3-22 Register Map

There are two register storages in the panel. Clicking the CFG→A (I) button, the register values can be saved to register A. Clicking the CFG→B (J) button, the register values can be saved to register B. If you want to recover register A status, click A→CFG (K) button. If you want to recover register B status, click B→CFG (L) button.

Clicking the Print B→CFG (M) button, the text area (T) shows the different register values from register B to current configurations. Clicking the Print A→CFG (N) button, the text area (T) shows the different register values from register B to current configurations. The CFG ALL (O) button shows the register values in the text area (T) except the read only registers. The Print B→A (P) button shows the changed register values from register B to register A in the text area (T). The Print A→B (Q) button shows the changed register values from register A to register B. The NON-ZERO (R) button shows the non-zero register values. The Print Style (S) button can choose the style of address and register value showing in the text area (T). Clicking the Config Screen (U) button, GUI would jump to Path Config screen.

3.13 Analog

The Analog section contains the setting of MIC, SPK, PSTN line input and output, and AUX out.

- The microphone interface can operate on 3 modes: 1) Microphone Voltage gain mode; 2) Microphone Current gain mode; 3) Auxiliary input mode. The reference voltage and internal resistor can be configured by GUI.

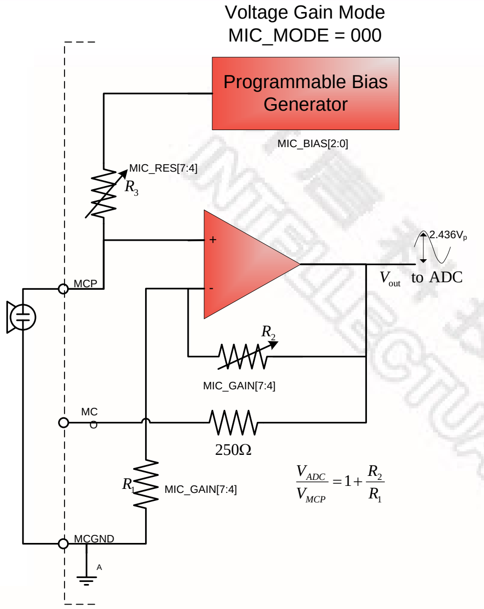


Figure 3-23 Voltage Gain Mode

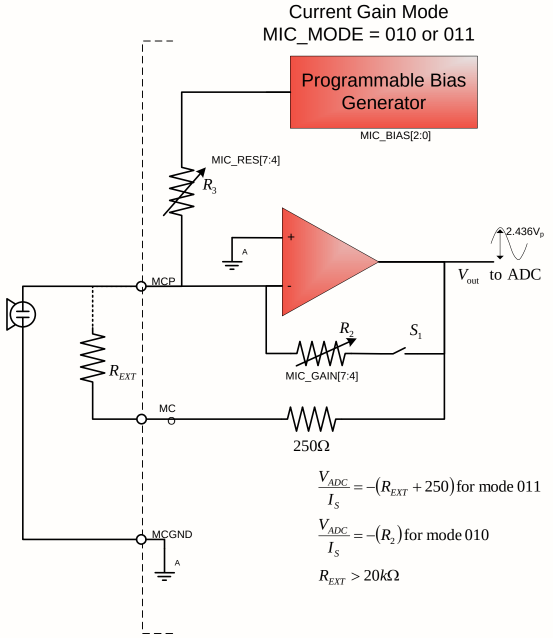


Figure 3-24 Current Gain Mode

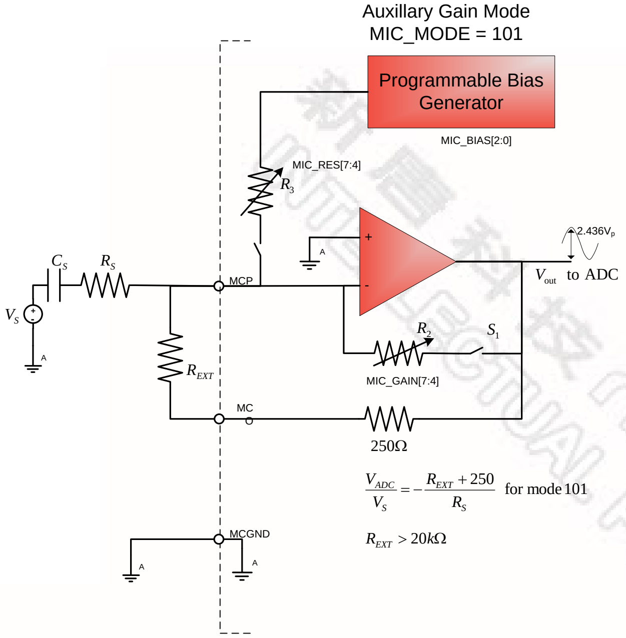


Figure 3-25 Auxiliary Gain Mode

- The PSTN line input contains two stage amplifier with independently settable gains and multiplexed input as described in figure 3-26.

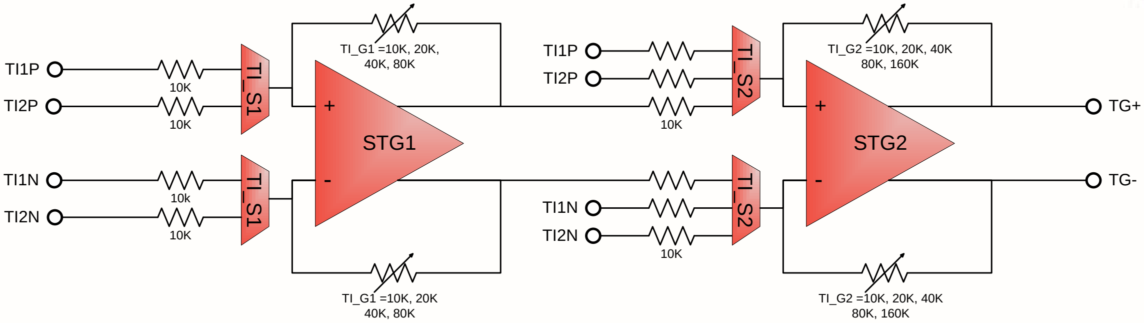


Figure 3-26 Two stage amplifier

- The analog output consists of speaker driver capable of driving 8 ohm and a differential line driver for PSTN interface capable of driving 120 ohm. These buffers are connected to the two DAC paths incorporating digital gain control. The ANA_OUT register controls the analog function of these two drivers allowing power control via SPK_EN and PO_EN and gain setting (including MUTE) via PO_GAIN and SPK_GAIN.

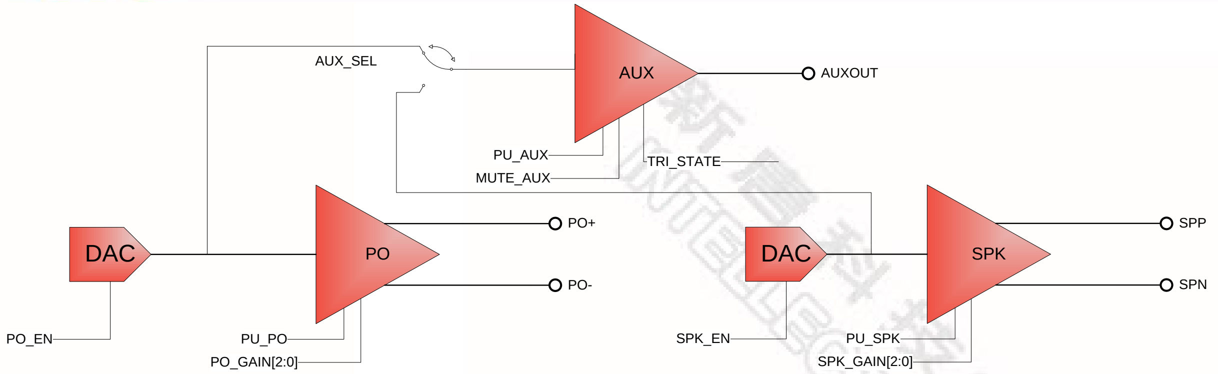
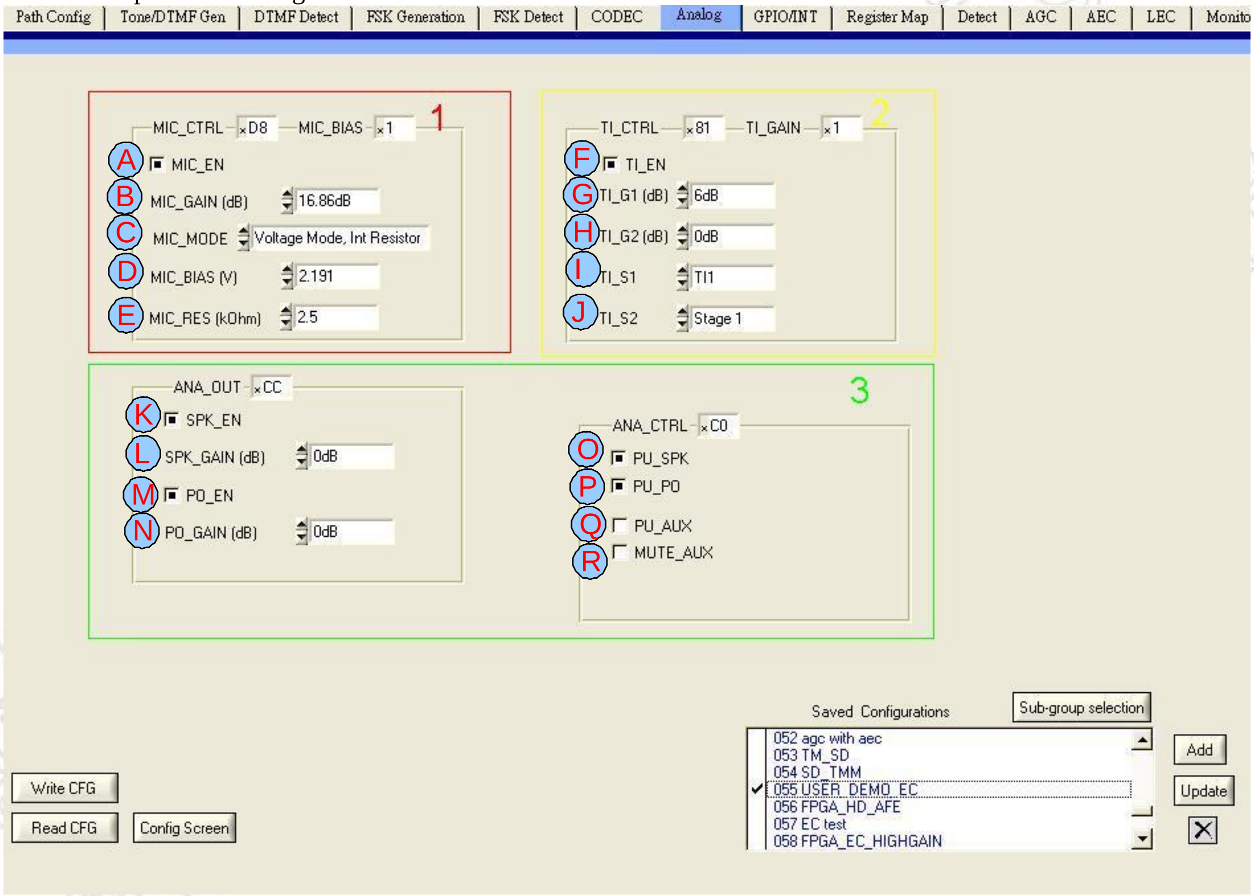


Figure 3-27 Analog output

The control panel of Analog section is:



MIC_CTRL x D8 MIC_BIAS x 1

A ☑ MIC_EN

B MIC_GAIN (dB) 16.86dB

C MIC_MODE Voltage Mode, Int Resistor

D MIC_BIAS (V) 2.191

E MIC_RES (kOhm) 2.5

TI_CTRL x 81 TI_GAIN x 1

F ☑ TI_EN

G TI_G1 (dB) 6dB

H TI_G2 (dB) 0dB

I TI_S1 TI1

J TI_S2 Stage 1

ANA_OUT x CC

K ☑ SPK_EN

L SPK_GAIN (dB) 0dB

M ☑ PO_EN

N PO_GAIN (dB) 0dB

ANA_CTRL x C0

O ☑ PU_SPK

P ☑ PU_PO

Q ☐ PU_AUX

R ☐ MUTE_AUX

Saved Configurations

052 agc with aec
053 TM_SD
054 SD_TMM
✓ 055 USER_DEMO_EC
056 FPGA_HD_AFE
057 EC test
058 FPGA_EC_HIGHGAIN

Figure 3-28 Analog setting

Figure 3-28 shows the analog input/output setting. Area 1 is for Microphone interface configuration. Area2 is for PSTN line input configuration. Area 3 is for PSTN line output and Speaker and AUX out.

- To configure the MIC interface
  1. Click MIC_EN(A) to 1
  2. Configure MIC Gain(B)
  3. Configure MIC Mode(C)

4. Configure MIC_BIAS(D)
5. Configure MIC_RES(E)
6. Press 'Write CFG'

- To configure PSTN line input
  1. Click TI_EN(F) to 1
  2. Configure T1_G1(G), the gain of first stage amplifier
  3. Configure T1_G2(H), the gain of second stage amplifier
  4. Configure T1_S1(I), choose the input source of first stage amplifier
  5. Configure T2_S2(J), choose the input source of second stage amplifier
  6. Press 'Write CFG'

- To configure Speaker output
  1. Click SPK_EN(K) to 1
  2. Configure speaker gain(L)
  3. Click PU_SPK(O)
  4. Press 'Write CFG'

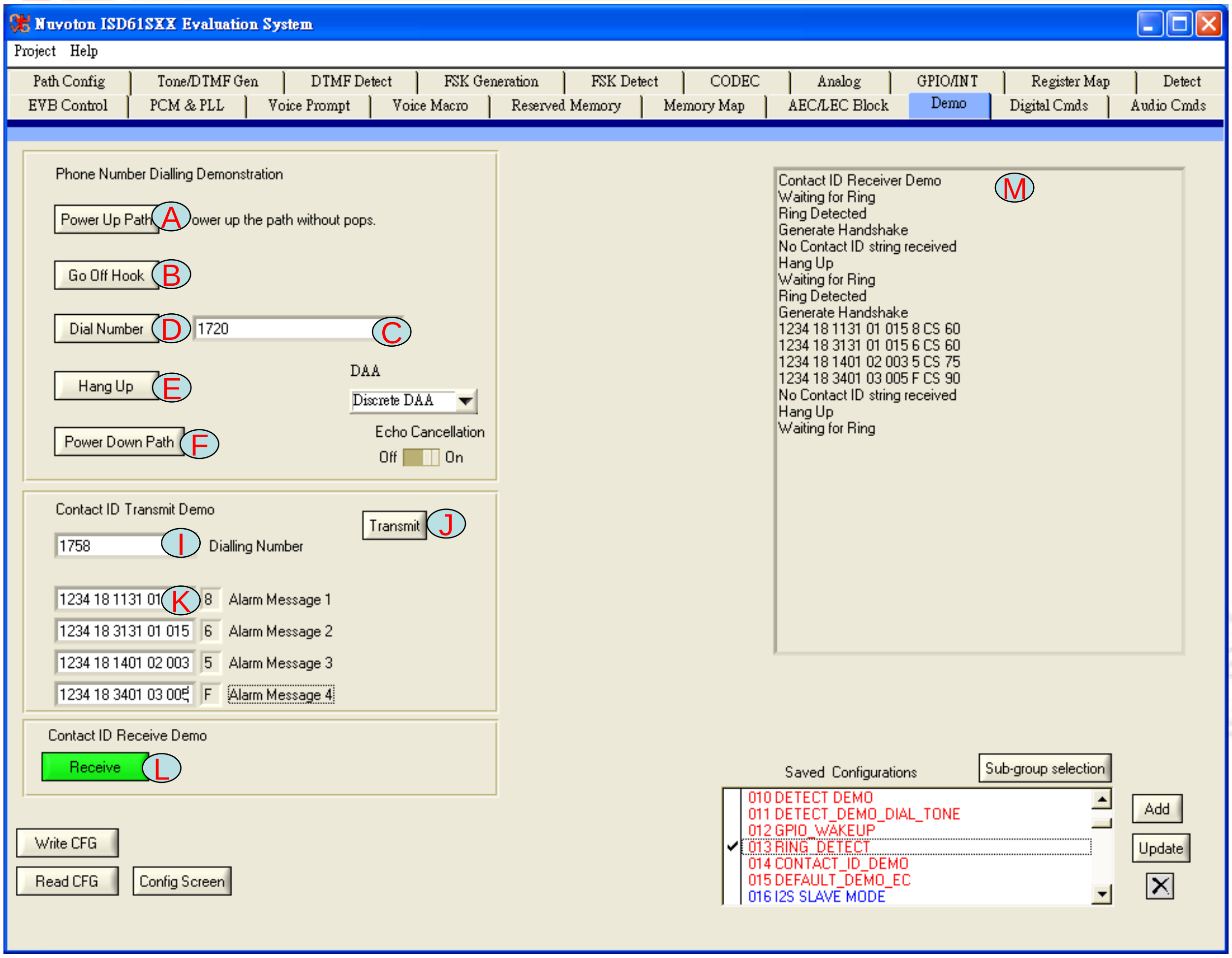
- To configure PSTN line output
  1. Click PO_EN(M) to 1
  2. Configure PO gain(N)
  3. Click PU_PO(P)
  4. Press 'Write CFG'

- To configure AUX output
  1. Click PO_EN(M) to 1
  2. Configure PO gain(N)
  3. Click PU_AUX(Q) to 1
  4. Press 'Write CFG'

3.14 Demo

The following procedure demonstrates phone numbers dialing:

(1) Clicking the Power Up Path (A) button: the path would be power up.
(2) Clicking Go off Hook (B) button: the 61S board would go to off hook status. If you don't click Power Up Path (A) before Go off Hook (B) button, the Go off Hook button would execute Power Up Path automatically.
(3) To input the dial numbers in the text area (C)
(4) Clicking the Dial Number (D) button. The 61S board would make a phone call according the number you input.
After finishing the communication:
(5) Clicking Hang Up (E) button.
(6) The Power Down Path (F) button is power down the device.


Nuvoton ISD61SXX Evaluation System
Project Help
Path Config | Tone/DTMF Gen | DTMF Detect | FSK Generation | FSK Detect | CODEC | Analog | GPIO/INT | Register Map | Detect
EVB Control | PCM & PLL | Voice Prompt | Voice Macro | Reserved Memory | Memory Map | AEC/LEC Block | Demo | Digital Cmds | Audio Cmds
Phone Number Dialling Demonstration
Power Up Path
A
B
Go Off Hook
Dial Number
D
C
Hang Up
E
Power Down Path
F
Contact ID Transmit Demo
Transmit
J
I
K
Contact ID Receive Demo
Receive
L
M
Write CFG
Read CFG
Config Screen
Saved Configurations
Sub-group selection
Add
Update

Power up the path without pops.

Dial Number: 1720

DAA: Discrete DAA

Echo Cancellation: Off / On

Dialling Number: 1758

1234 18 1131 01	8	Alarm Message 1
1234 18 3131 01 015	6	Alarm Message 2
1234 18 1401 02 003	5	Alarm Message 3
1234 18 3401 03 005	F	Alarm Message 4

```
Contact ID Receiver Demo
Waiting for Ring
Ring Detected
Generate Handshake
No Contact ID string received
Hang Up
Waiting for Ring
Ring Detected
Generate Handshake
1234 18 1131 01 015 8 CS 60
1234 18 3131 01 015 6 CS 60
1234 18 1401 02 003 5 CS 75
1234 18 3401 03 005 F CS 90
No Contact ID string received
Hang Up
Waiting for Ring
```

Saved Configurations:

010 DETECT DEMO
011 DETECT_DEMO_DIAL_TONE
012 GPIO_WAKEUP
✓ 013 RING_DETECT
014 CONTACT_ID_DEMO
015 DEFAULT_DEMO_EC
016 I2S SLAVE MODE

The Contact ID Transmit and Receive Demo:
This is the major security panel reporting method as specified by the SIA (security industry association) spec.

- For Contact ID Transmit Demo:

(1). Input the dial number in the text area (I)
(2). Clicking Transmit (J) button. The device would be waiting for the response of the other side.
If the other side sends a handshake response, the alarm messages would be transmitted. After transmitting alarm messages, the device would be hang up. The statuses are described in the text area (M).

- For Contact ID Receive Demo:

Clicking Receive (L) button, the device goes into the waiting for ring status.
If the device detects ringing, it will generate handshake signal and start to receive the messages from transmitter.
The messages are showing in the text area (M).

3.15 AEC/LEC

The Acoustic possessing block performs a variety of functions to implement the full or half duplex echo cancellation. The ISD61SXX incorporates an Acoustic Echo Cancellation (AEC) block for the microphone and speaker CODEC and a Line Echo Cancellation (LEC) block for the PSTN interface. These blocks are very similar, with the main differences being that the AEC includes an AGC function and has a greater echo cancellation length.

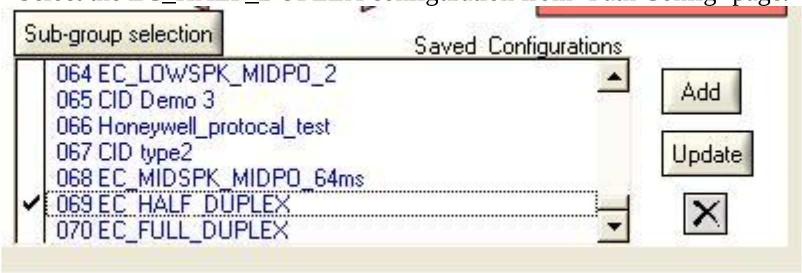
The functions performed by the acoustic processing blocks include:

- **Echo Cancellation**: This is an adaptive estimation of the characteristics of the echo path to cancel echo introduced by speaker/microphone acoustic coupling and echo return paths from the environment.
- **Double Talk (DT) Detection**: When both line and acoustic paths are active it is important to detect this double talk conditions to halt adaptation of the acoustic echo.
- **Voice Detection (VT)**: Estimates when the voice path is active.
- **AGC**: Automatic gain control to provide an optimal level from the microphone input.
- **Soft Clipping**: limits the input signal path to prevent hard saturation of the output.
- **Acoustic Suppression**: Used in half duplex mode to switch from one path to the other.
- **Noise Suppression**: Detect noise and suppress.

Full acoustic echo cancellation unit removes the echo signal inserted by the speaker coupling and space reflections. **Half duplex AEC** smoothly transitions the acoustic suppressors from one direction to the other based on the power estimations of the voice and double talk detectors.

- To use half duplex AEC:

Select the EC_HALF_DUPLEX configuration from ‘Path Config’ page.

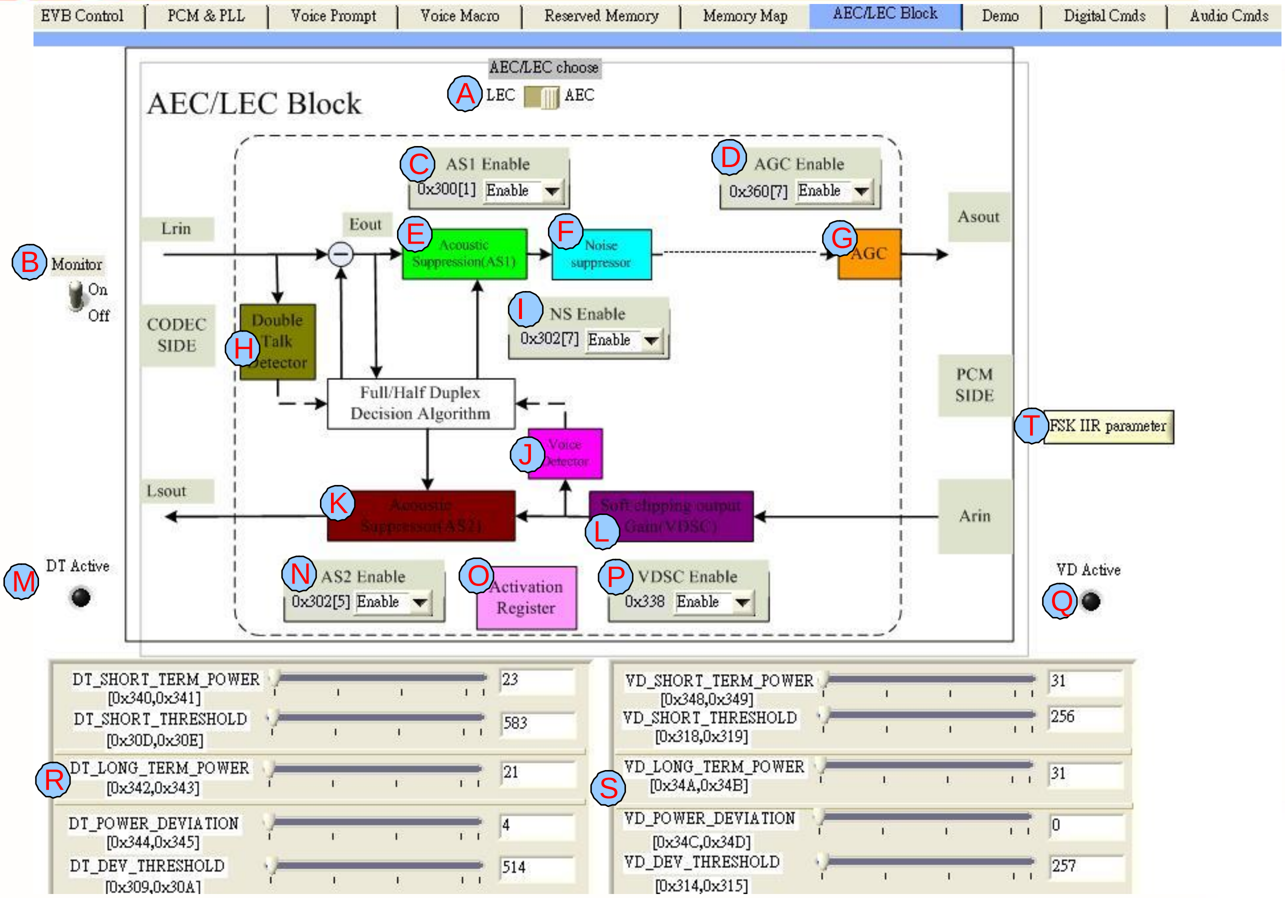


064 EC_LOWSPK_MIDPO_2
065 CID Demo 3
066 Honeywell_protocal_test
067 CID type2
068 EC_MIDSPK_MIDPO_64ms
✓ 069 EC_HALF_DUPLEX
070 EC_FULL_DUPLEX

This will load the default values for the AEC/LEC half duplex mode setting.

- To tune the performance of AEC block

Switch to AEC/LEC block page and make sure the switch (A) is selected to AEC position.



AS1 Enable 0x300[1] Enable

AGC Enable 0x360[7] Enable

NS Enable 0x302[7] Enable

AS2 Enable 0x302[5] Enable

VDSC Enable 0x338 Enable

DT_SHORT_TERM_POWER [0x340,0x341] 23

DT_SHORT_THRESHOLD [0x30D,0x30E] 583

DT_LONG_TERM_POWER [0x342,0x343] 21

DT_POWER_DEVIATION [0x344,0x345] 4

DT_DEV_THRESHOLD [0x309,0x30A] 514

VD_SHORT_TERM_POWER [0x348,0x349] 31

VD_SHORT_THRESHOLD [0x318,0x319] 256

VD_LONG_TERM_POWER [0x34A,0x34B] 31

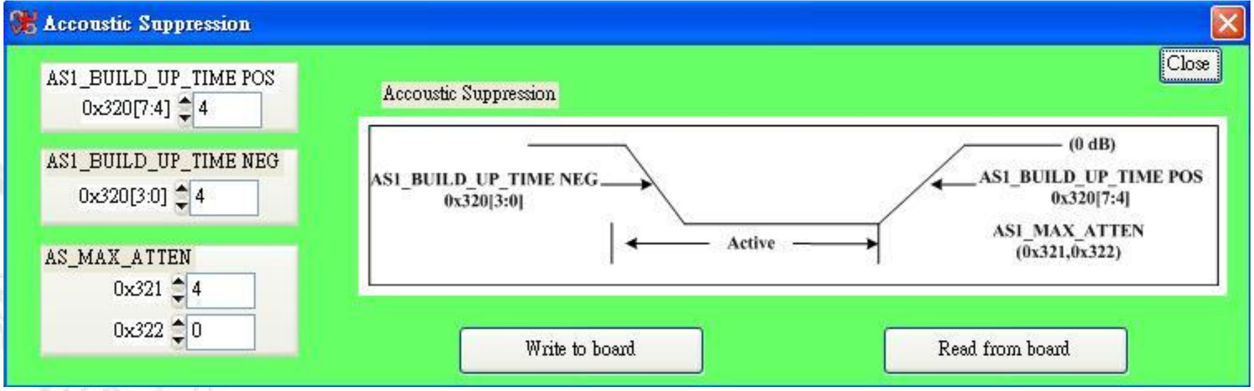
VD_POWER_DEVIATION [0x34C,0x34D] 0

VD_DEV_THRESHOLD [0x314,0x315] 257

- To tune the Acoustic Suppressor 1 (AS1)

(1) Click (C) to enable AS1.

(2) Click (E) to open the AS1 configuration panel.



AS1_BUILD_UP_TIME POS 0x320[7:4] 4

AS1_BUILD_UP_TIME NEG 0x320[3:0] 4

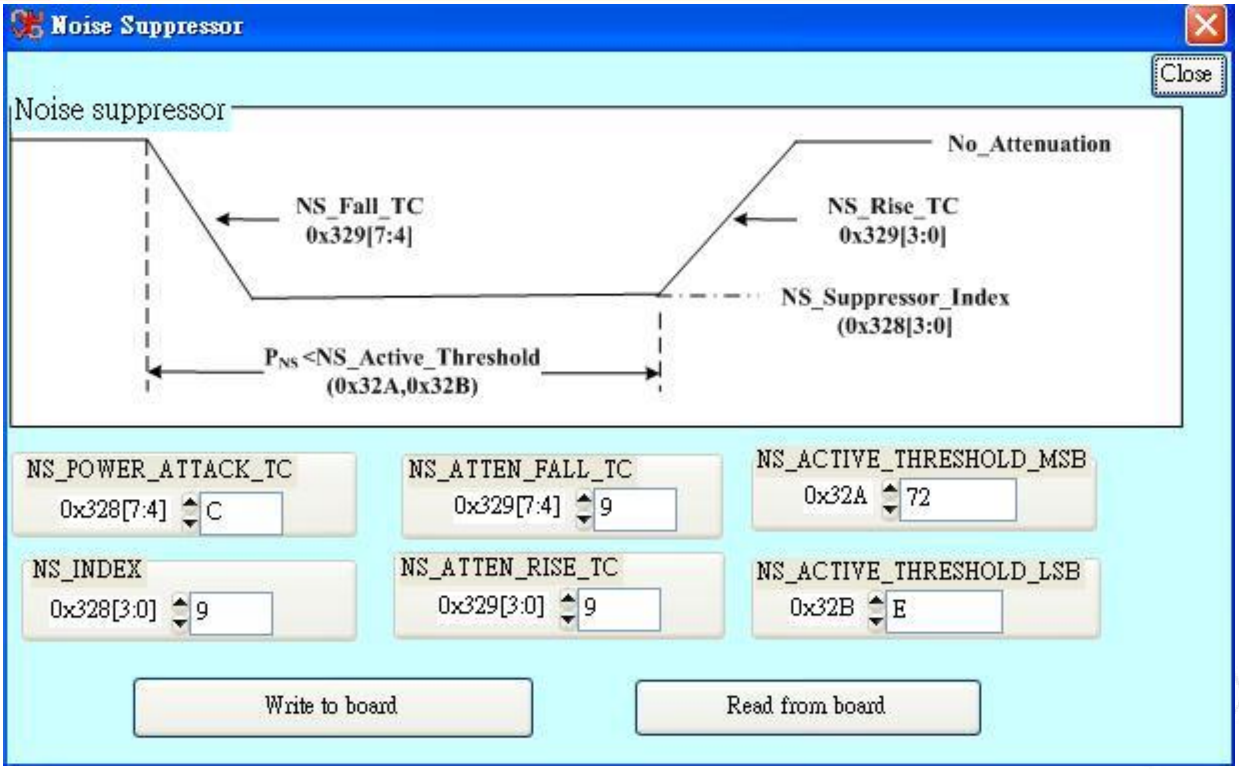
AS_MAX_ATTEN 0x321 4 0x322 0

(3) Configure the parameters in the panel and Press ‘Write to board’.

- To tune the Noise Suppressor (NS)

(1) Click (I) to enable NS.

(2) Click (F) to open the NS configuration panel.



NS_POWER_ATTACK_TC 0x328[7:4] C

NS_ATTEN_FALL_TC 0x329[7:4] 9

NS_ACTIVE_THRESHOLD_MSB 0x32A 72

NS_INDEX 0x328[3:0] 9

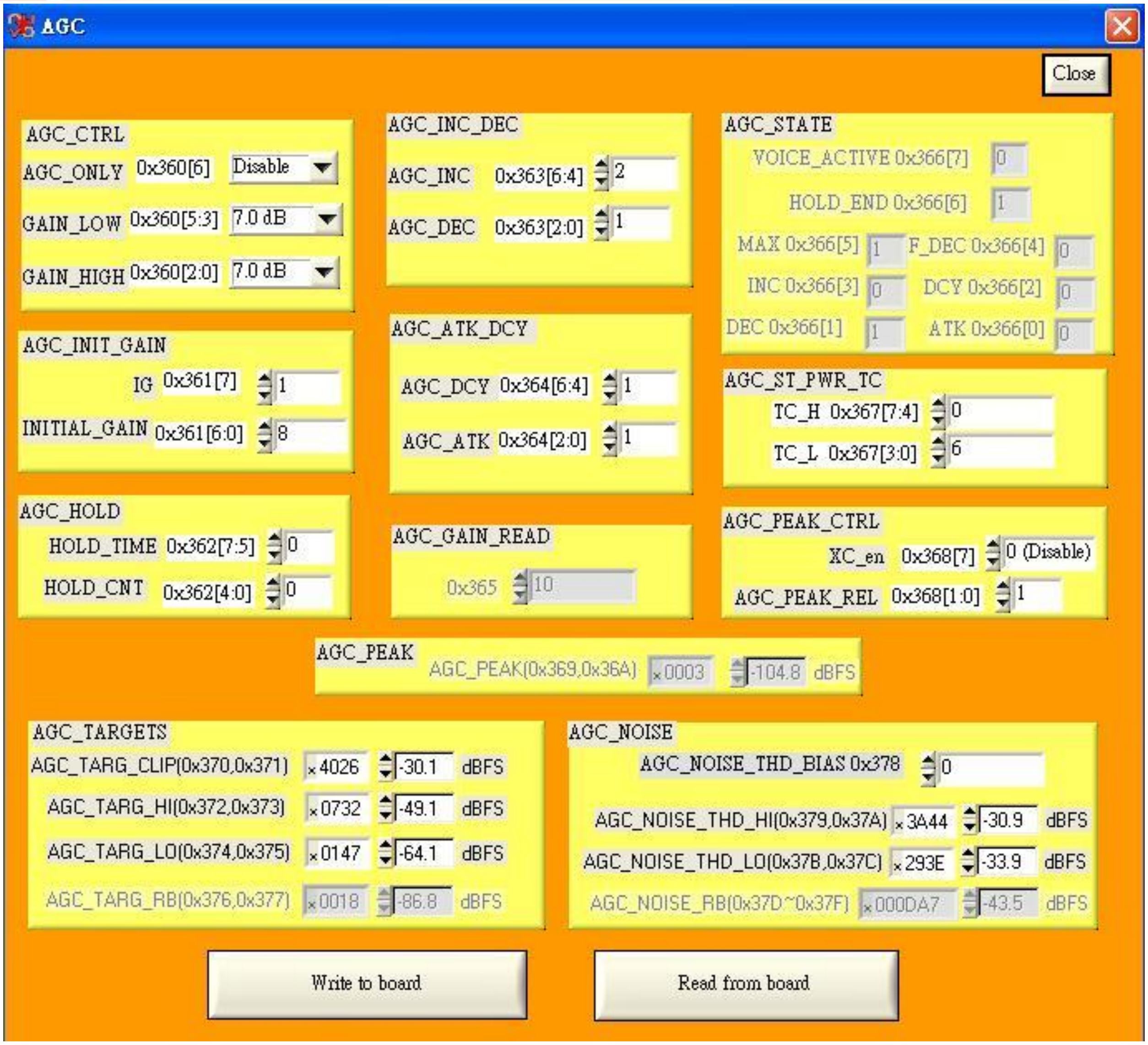
NS_ATTEN_RISE_TC 0x329[3:0] 9

NS_ACTIVE_THRESHOLD_LSB 0x32B E

(3) Configure the parameters in the panel and Press ‘Write to board’.

- To tune the Auto Gain Control (AGC)

(1) Click (D) to enable AGC.
(2) Click (G) to open the AGC configuration panel.



AGC

AGC_CTRL
AGC_ONLY 0x360[6] Disable
GAIN_LOW 0x360[5:3] 7.0 dB
GAIN_HIGH 0x360[2:0] 7.0 dB

AGC_INC_DEC
AGC_INC 0x363[6:4] 2
AGC_DEC 0x363[2:0] 1

AGC_STATE
VOICE_ACTIVE 0x366[7] 0
HOLD_END 0x366[6] 1
MAX 0x366[5] 1 F_DEC 0x366[4] 0
INC 0x366[3] 0 DCY 0x366[2] 0
DEC 0x366[1] 1 ATK 0x366[0] 0

AGC_INIT_GAIN
IG 0x361[7] 1
INITIAL_GAIN 0x361[6:0] 8

AGC_ATK_DCY
AGC_DCY 0x364[6:4] 1
AGC_ATK 0x364[2:0] 1

AGC_ST_PWR_TC
TC_H 0x367[7:4] 0
TC_L 0x367[3:0] 6

AGC_HOLD
HOLD_TIME 0x362[7:5] 0
HOLD_CNT 0x362[4:0] 0

AGC_GAIN_READ
0x365 10

AGC_PEAK_CTRL
XC_en 0x368[7] 0 (Disable)
AGC_PEAK_REL 0x368[1:0] 1

AGC_PEAK
AGC_PEAK(0x369,0x36A) x 0003 -104.8 dBFS

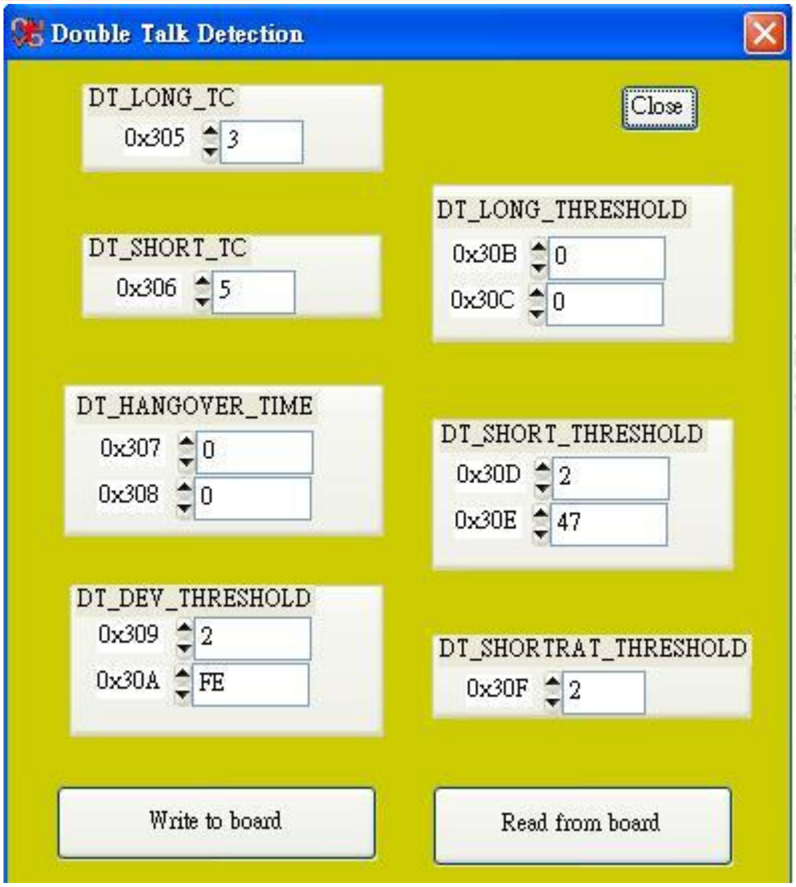
AGC_TARGETS
AGC_TARG_CLIP(0x370,0x371) x 4026 -30.1 dBFS
AGC_TARG_HI(0x372,0x373) x 0732 -49.1 dBFS
AGC_TARG_LO(0x374,0x375) x 0147 -64.1 dBFS
AGC_TARG_RB(0x376,0x377) x 0018 -86.8 dBFS

AGC_NOISE
AGC_NOISE_THD_BIAS 0x378 0
AGC_NOISE_THD_HI(0x379,0x37A) x 3A44 -30.9 dBFS
AGC_NOISE_THD_LO(0x37B,0x37C) x 293E -33.9 dBFS
AGC_NOISE_RB(0x37D~0x37F) x 000DA7 -43.5 dBFS

(3) Configure the parameters in the panel and Press ‘Write to board’.

- To tune the Double Talk Detector (DT)

(1) Click (H) to open the DT configuration panel.



DT_LONG_TC
0x305 3

DT_SHORT_TC
0x306 5

DT_HANGOVER_TIME
0x307 0
0x308 0

DT_DEV_THRESHOLD
0x309 2
0x30A FE

DT_LONG_THRESHOLD
0x30B 0
0x30C 0

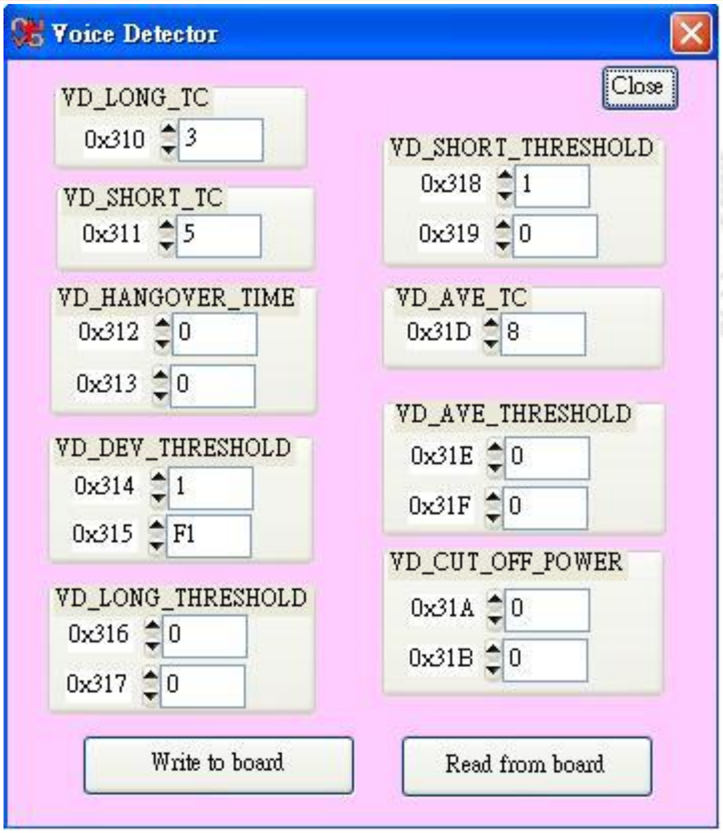
DT_SHORT_THRESHOLD
0x30D 2
0x30E 47

DT_SHORTRAT_THRESHOLD
0x30F 2

(2) Configure the parameters in the panel and Press ‘Write to board’.

- To tune the Voice Detector (VD)

(1) Click (J) to open the VD configuration panel.



VD_LONG_TC
0x310 3

VD_SHORT_TC
0x311 5

VD_HANGOVER_TIME
0x312 0
0x313 0

VD_DEV_THRESHOLD
0x314 1
0x315 F1

VD_LONG_THRESHOLD
0x316 0
0x317 0

VD_SHORT_THRESHOLD
0x318 1
0x319 0

VD_AVE_TC
0x31D 8

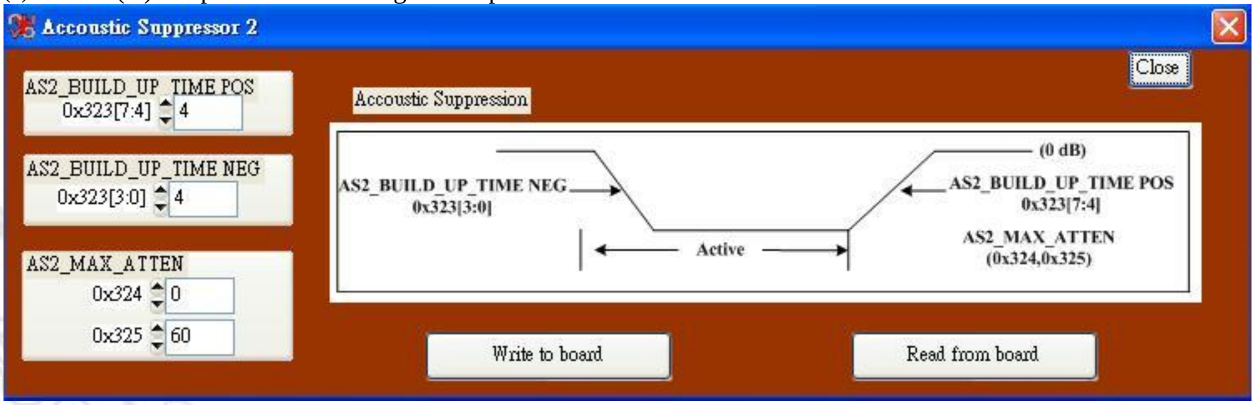
VD_AVE_THRESHOLD
0x31E 0
0x31F 0

VD_CUT_OFF_POWER
0x31A 0
0x31B 0

(2) Configure the parameters in the panel and Press ‘Write to board’.

- To tune the Acoustic Suppressor 2 (AS2)

(1) Click (N) to enable AS2.
(2) Click (K) to open the AS2 configuration panel.



AS2_BUILD_UP_TIME POS
0x323[7:4] 4

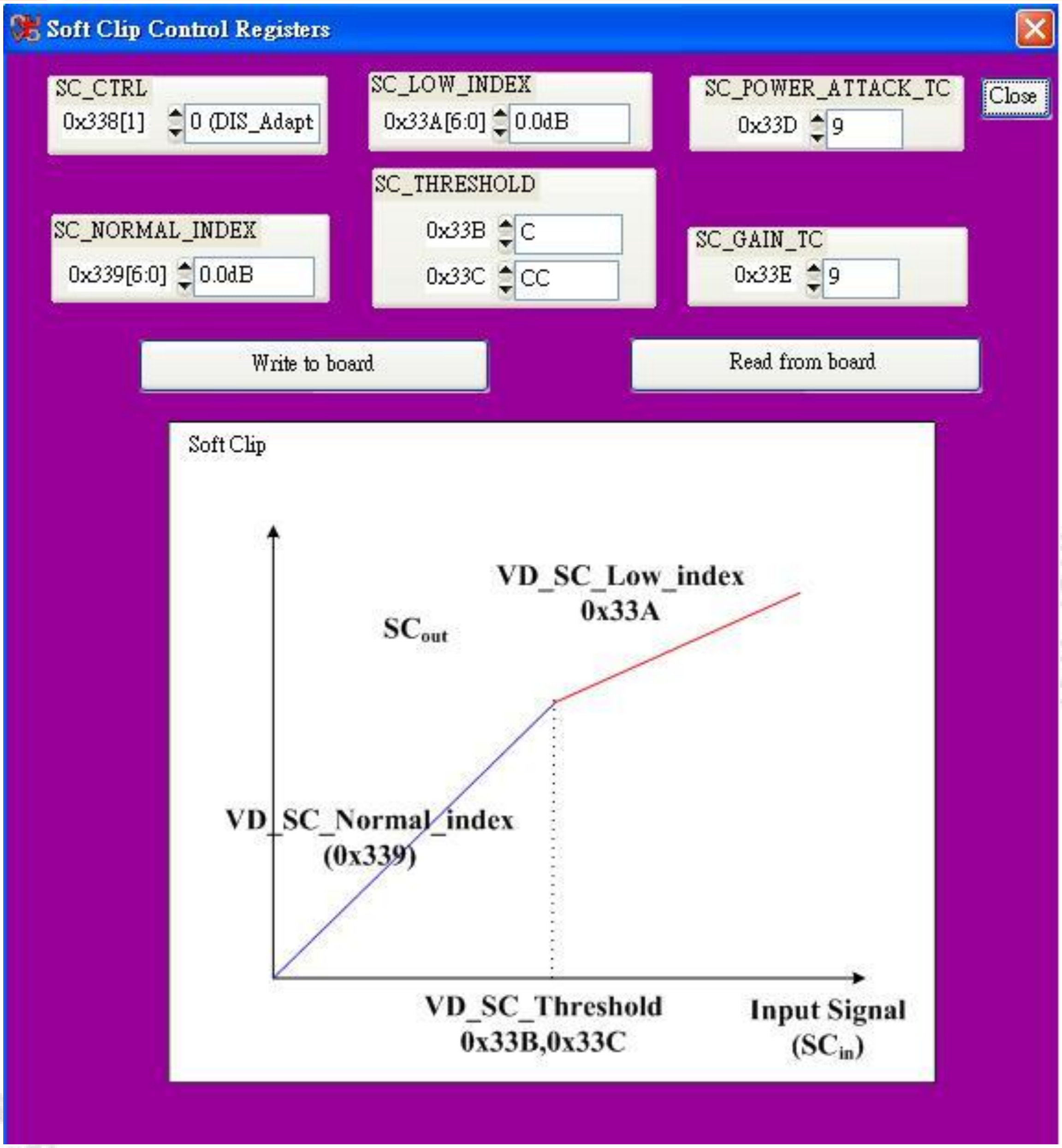
AS2_BUILD_UP_TIME NEG
0x323[3:0] 4

AS2_MAX_ATTEN
0x324 0
0x325 60

(3) Configure the parameters in the panel and Press ‘Write to board’.

- To tune the Soft Clipping(SC)

(1) Click (P) to enable SC.
(2) Click (L) to open the SC configuration panel.



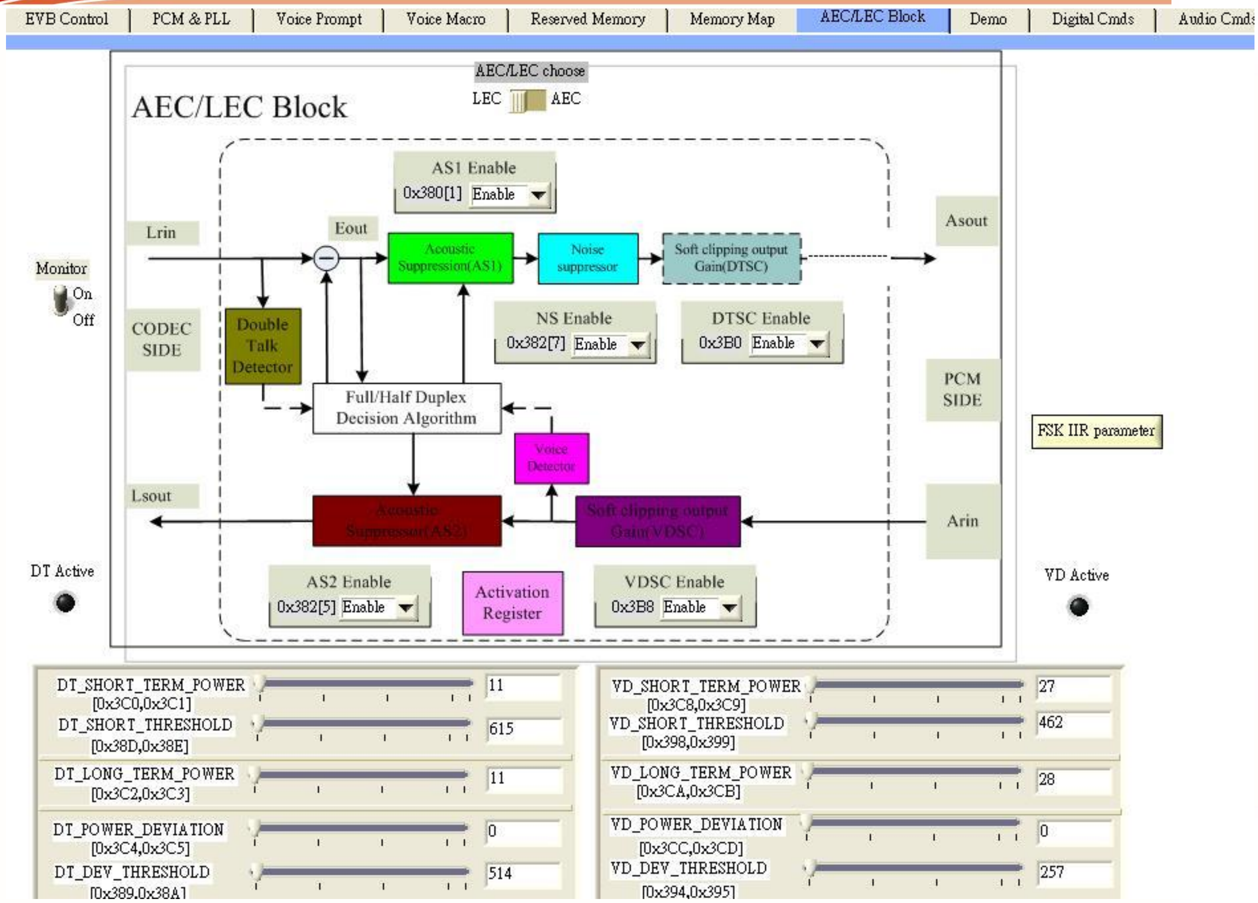
Soft Clip Control Registers

SC_CTRL	0x338[1]	0 (DIS_Adapt
SC_LOW_INDEX	0x33A[6:0]	0.0dB
SC_POWER_ATTACK_TC	0x33D	9
SC_NORMAL_INDEX	0x339[6:0]	0.0dB
SC_THRESHOLD	0x33B	C
	0x33C	CC
SC_GAIN_TC	0x33E	9

(3) Configure the parameters in the panel and Press ‘Write to board’.

- To tune the performance of LEC block

Switch to AEC/LEC block page and make sure the LEC switch is selected.



AS1 Enable 0x380[1] Enable

NS Enable 0x382[7] Enable

DTSC Enable 0x3B0 Enable

AS2 Enable 0x382[5] Enable

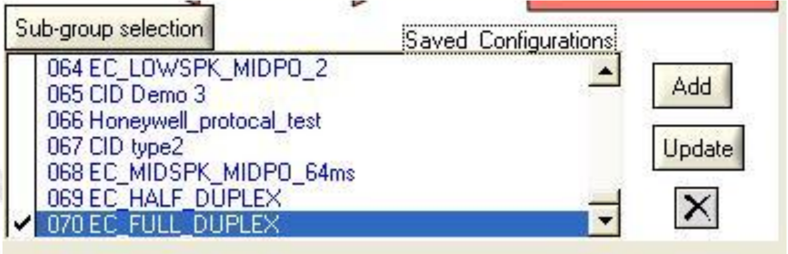
VDSC Enable 0x3B8 Enable

DT_SHORT_TERM_POWER [0x3C0,0x3C1]	11
DT_SHORT_THRESHOLD [0x38D,0x38E]	615
DT_LONG_TERM_POWER [0x3C2,0x3C3]	11
DT_POWER_DEVIATION [0x3C4,0x3C5]	0
DT_DEV_THRESHOLD [0x389,0x38A]	514
VD_SHORT_TERM_POWER [0x3C8,0x3C9]	27
VD_SHORT_THRESHOLD [0x398,0x399]	462
VD_LONG_TERM_POWER [0x3CA,0x3CB]	28
VD_POWER_DEVIATION [0x3CC,0x3CD]	0
VD_DEV_THRESHOLD [0x394,0x395]	257

The function block operations are similar to previous AEC procedure.

- To use full duplex AEC/LEC:

Select the EC_FULL_DUPLEX configuration from ‘Path Config’ page.

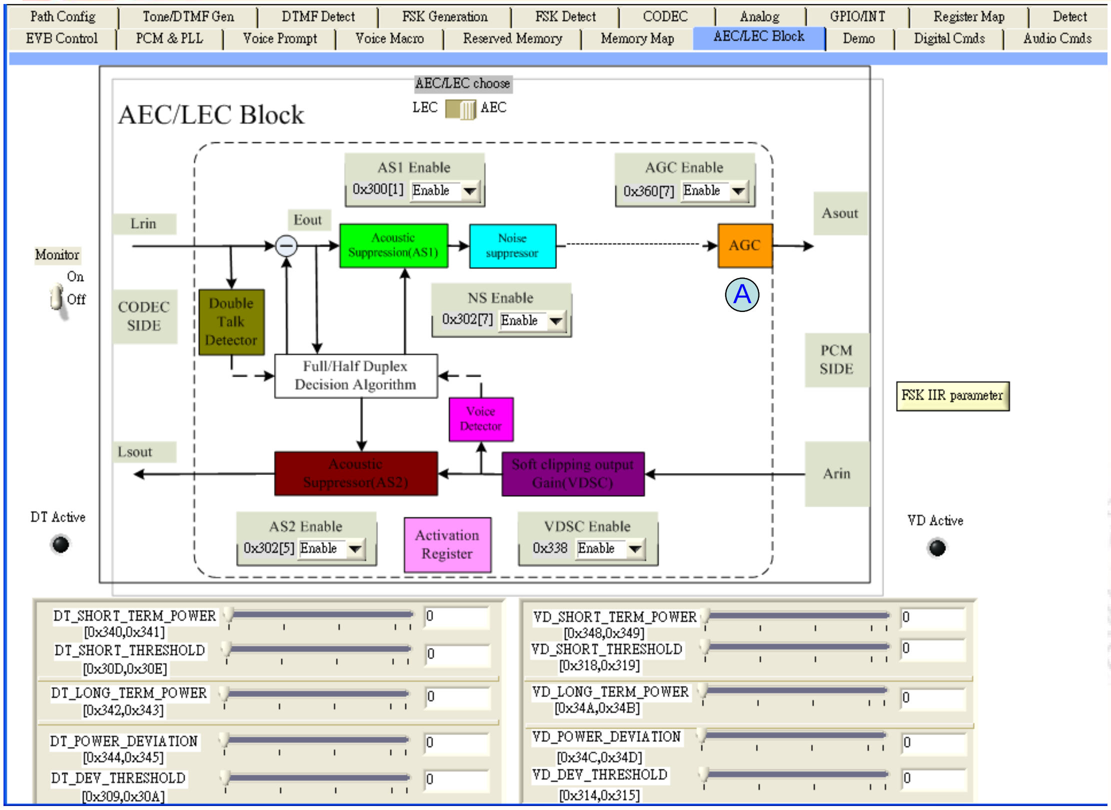


064 EC_LOWSPK_MIDPO_2
065 CID Demo 3
066 Honeywell_protocal_test
067 CID type2
068 EC_MIDSPK_MIDPO_64ms
069 EC_HALF_DUPLEX
✓ 070 EC_FULL_DUPLEX

This will load the default values for the AEC/LEC full duplex mode setting.
The AEC/LEC detail tuning is similar to previous procedure.

3.16 AGC

The AGC function is in the AEC/LEC block (A). To set the AGC Enable, and double click the AGC icon, the AGC setting shows on the screen.



AEC/LEC choose: LEC / AEC

AS1 Enable 0x300[1] Enable

AGC Enable 0x360[7] Enable

NS Enable 0x302[7] Enable

AS2 Enable 0x302[5] Enable

VDSC Enable 0x338 Enable

DT_SHORT_TERM_POWER [0x340,0x341] 0

DT_SHORT_THRESHOLD [0x30D,0x30E] 0

DT_LONG_TERM_POWER [0x342,0x343] 0

DT_POWER_DEVIATION [0x344,0x345] 0

DT_DEV_THRESHOLD [0x309,0x30A] 0

VD_SHORT_TERM_POWER [0x348,0x349] 0

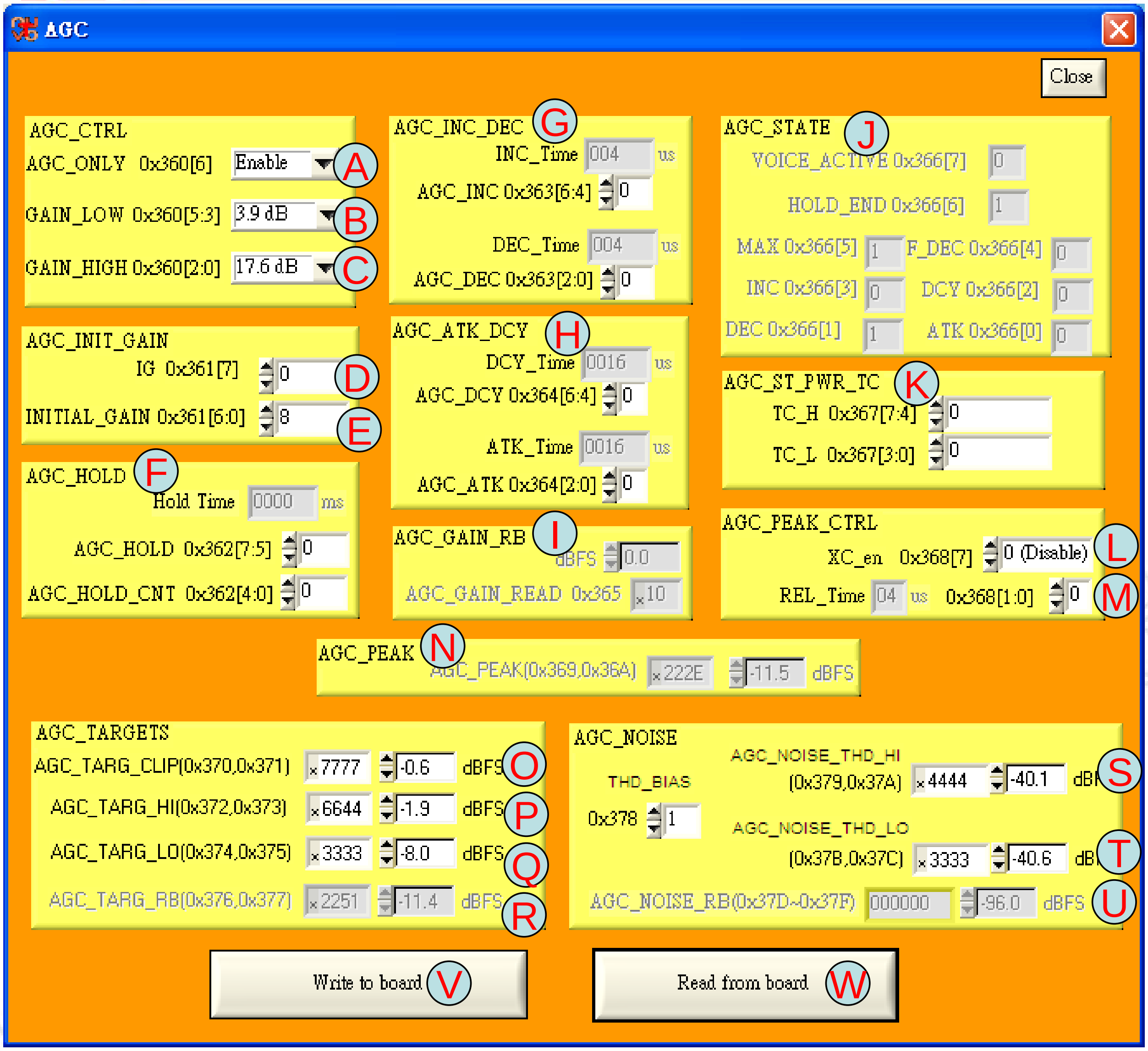
VD_SHORT_THRESHOLD [0x318,0x319] 0

VD_LONG_TERM_POWER [0x34A,0x34B] 0

VD_POWER_DEVIATION [0x34C,0x34D] 0

VD_DEV_THRESHOLD [0x314,0x315] 0

Figure 3-29 AEC/LEC screen



AGC

Close

AGC_CTRL
AGC_ONLY 0x360[6] Enable
GAIN_LOW 0x360[5:3] 3.9 dB
GAIN_HIGH 0x360[2:0] 17.6 dB

AGC_INIT_GAIN
IG 0x361[7] 0
INITIAL_GAIN 0x361[6:0] 8

AGC_HOLD
Hold Time 0000 ms
AGC_HOLD 0x362[7:5] 0
AGC_HOLD_CNT 0x362[4:0] 0

AGC_INC_DEC
INC_Time 004 us
AGC_INC 0x363[6:4] 0
DEC_Time 004 us
AGC_DEC 0x363[2:0] 0

AGC_ATK_DCY
DCY_Time 0016 us
AGC_DCY 0x364[6:4] 0
ATK_Time 0016 us
AGC_ATK 0x364[2:0] 0

AGC_GAIN_RB
dBFS 0.0
AGC_GAIN_READ 0x365 x10

AGC_STATE
VOICE_ACTIVE 0x366[7] 0
HOLD_END 0x366[6] 1
MAX 0x366[5] 1 F_DEC 0x366[4] 0
INC 0x366[3] 0 DCY 0x366[2] 0
DEC 0x366[1] 1 ATK 0x366[0] 0

AGC_ST_PWR_TC
TC_H 0x367[7:4] 0
TC_L 0x367[3:0] 0

AGC_PEAK_CTRL
XC_en 0x368[7] 0 (Disable)
REL_Time 04 us 0x368[1:0] 0

AGC_PEAK
AGC_PEAK(0x369,0x36A) x222E -11.5 dBFS

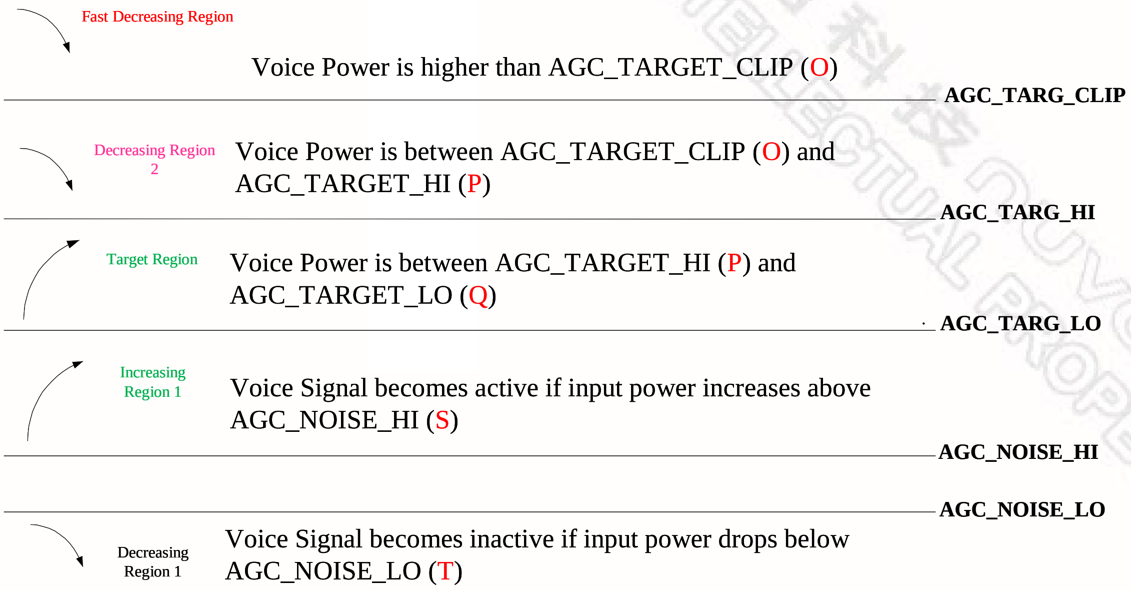
AGC_TARGETS
AGC_TARG_CLIP(0x370,0x371) x7777 -0.6 dBFS
AGC_TARG_HI(0x372,0x373) x6644 -1.9 dBFS
AGC_TARG_LO(0x374,0x375) x3333 -8.0 dBFS
AGC_TARG_RB(0x376,0x377) x2251 -11.4 dBFS

AGC_NOISE
THD_BIAS 0x378 1
AGC_NOISE_THD_HI (0x379,0x37A) x4444 -40.1 dB
AGC_NOISE_THD_LO (0x37B,0x37C) x3333 -40.6 dB
AGC_NOISE_RB(0x37D~0x37F) 000000 -96.0 dBFS

Figure 3-30 AGC screen

1. If the AGC_ONLY (A) is enabled, the AEC block is bypassed and AGC operates from AEC input.
2. The GAIN_LOW (B) and GAIN_HIGH (C) are setting the min and max gain.
3. The INITIAL_GAIN (E) determines the initial gain if the IG (D) set to1.
4. The AGC_HOLD (F) determines how long the AGC will hold gain constant after entering target region before increasing gain.
5. The AGC_INC and AGC_DEC of AGC_INC_DEC (G) determine the time constant at which the gain increase and decrease.
6. The AGC_ATK_DCY (H) controls the decay time constant (AGC_DCY) and attack time constant (AGC_ATK) when in the target region.
7. The AGC_GAIN_RB (I) shows the current AGC gain. The AGC_STATE (J) shows some states of AGC function.
8. The AGC_ST_PWR_TC (K) determines the time constant for calculating the short term noise energy. The short term noise energy responds faster to signal change when the time constant larger.

9. The REL_Time (M) is the release time of peak detector. If the XC_en (L) is enabled, the AGC_GAIN would be changed when the input signal is in the zero crossing.
10. The AGC_PEAK (N) reads back the peak detector value.
11. The AGC_TARGETS and AGC_NOISE set the high or low limit of target and noise region. When the signal short term energy increases above AGC_NOISE_THD_HI (S), the AGC is activated. The illustration is as below figure:



Voice Power is higher than AGC_TARGET_CLIP (O)

Voice Power is between AGC_TARGET_CLIP (O) and AGC_TARGET_HI (P)

Voice Power is between AGC_TARGET_HI (P) and AGC_TARGET_LO (Q)

Voice Signal becomes active if input power increases above AGC_NOISE_HI (S)

Voice Signal becomes inactive if input power drops below AGC_NOISE_LO (T)

12. The AGC_TARG_RB (R) is read only for the current AGC output signal level.
13. The AGC_NOISE_RB (U) shows the current noise power estimated.
14. Clicking Write to board (V), the settings would be written into registers.
15. Clicking Read from board (W) can read the register settings and check if the settings written into registers or not.

3.17 GPIO/INT

GPIO and interrupt are the two functions of the panel and are described as below, respectively.

3.17.1 GPIO

The GPIO pins are divided into three port group:
(1) PORTA consists of GPIO [7:0]
(2) PORTB consists of GPIO [15:8]
(3) PORTC consists of GPIO [16]
You can set the configurations in “GPIO configuration Bits” (A) or “GPIO configuration Registers” (B).

The GPIO pin can be either an input pin or an output pin by enabling or disabling “OE” (C).

1. The GPIO pin is an input pin: If the “OE” (C) is enabled, the corresponding GPIO pin is an output pin. And you can change the output state by setting “DO” (D) and the driving strength by setting “DS” (E).

2. On the other hand, if the “OE” (C) is disabled, the corresponding GPIO pin is an input pin. And you can change the pin state by enabling “PE” (F) and setting “PS” (G).

The port state can be checked by the value of “DI” (H).

If “AF” (I) is enabled, the corresponding GPIO pin may have an alternate function and are shown as below.

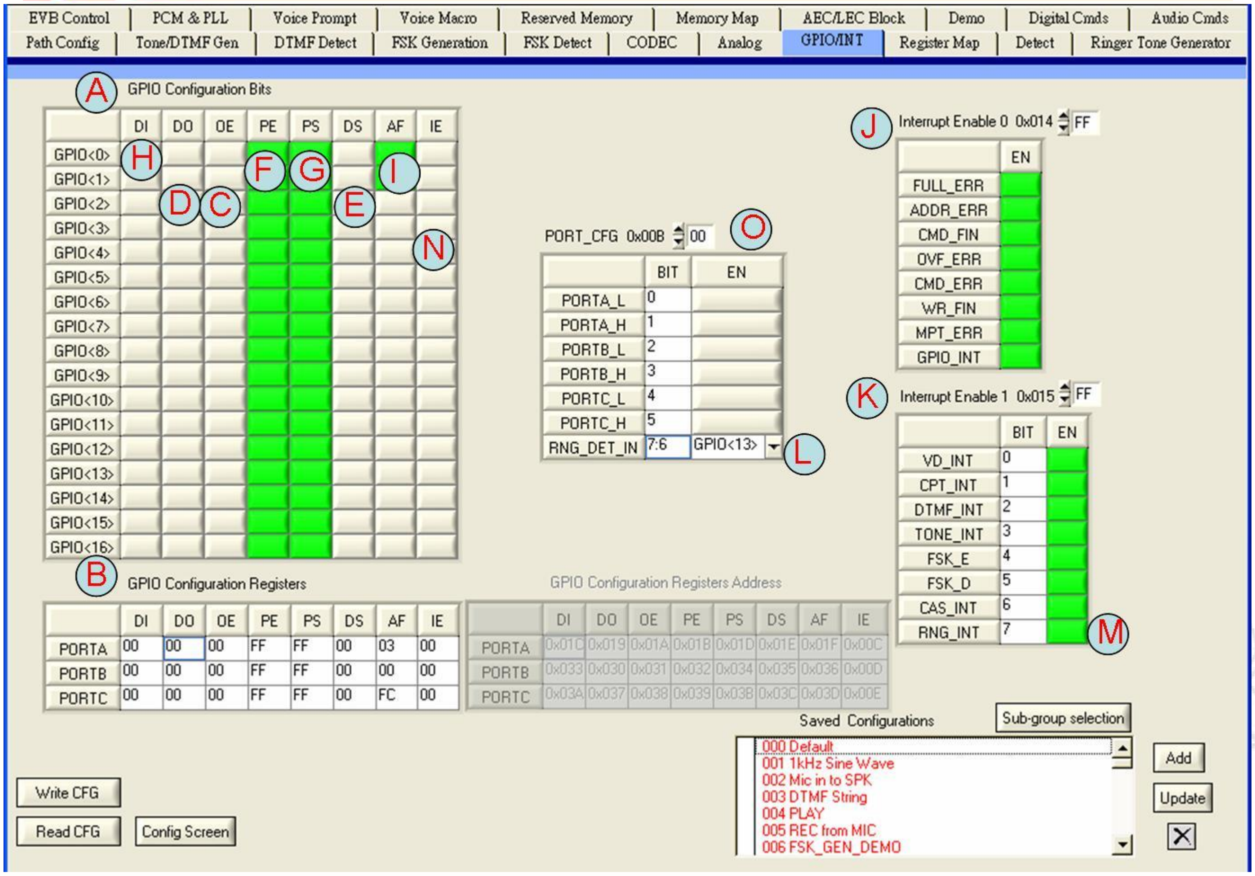
Port	GPIO Pin	Alternate Function	Description
PORTA<4>	GPIO<4>	SDO	I2S Serial Data Out
PORTA<5>	GPIO<5>	WS	I2S WS
PORTA<6>	GPIO<6>	SCK	I2S Serial Clock
PORTA<7>	GPIO<7>	SDI	I2S Serial Data In
PORTB<0>	GPIO<8>	SDO1	PCM1 Tx Data
PORTB<1>	GPIO<9>	SDI1	PCM1 Rx Data

3.17.2 Interrupt

There are sixteen enables and they are in “Interrupt Enable 0” (J) and “Interrupt Enable 1” (K). If you enable the interrupt, then the corresponding interrupt state will be shown in “Audio Cmds” panel.

If the interrupt state shows the detected signal, the GPIO should be an input pin and the interrupt is enabled. For example, the ringing interrupt detect the ringing signal from GPIO 14 in J1 or J11 on the EVB board, so the GPIO14 should be an input pin; “RINT_DET_IN” (L) should be GPIO14; RING_INT (M) should be enabled, then the ringing interrupt state will be shown in “Audio Cmds” panel.

The ISD61SXX can be waked up by a voltage level change. To do it, you should set the GPIO pin as an input pin; enable the interrupt (N) of the corresponding pin and enable the corresponding wakeup interrupt (O).



GPIO Configuration Bits

	DI	DO	OE	PE	PS	DS	AF	IE
GPIO<0>				🟩	🟩		🟩	
GPIO<1>				🟩	🟩		🟩	
GPIO<2>				🟩	🟩			
GPIO<3>				🟩	🟩			
GPIO<4>				🟩	🟩			
GPIO<5>				🟩	🟩			
GPIO<6>				🟩	🟩			
GPIO<7>				🟩	🟩			
GPIO<8>				🟩	🟩			
GPIO<9>				🟩	🟩			
GPIO<10>				🟩	🟩			
GPIO<11>				🟩	🟩			
GPIO<12>				🟩	🟩			
GPIO<13>				🟩	🟩			
GPIO<14>				🟩	🟩			
GPIO<15>				🟩	🟩			
GPIO<16>				🟩	🟩			

PORT_CFG 0x00B 00

	BIT	EN
PORTA_L	0	
PORTA_H	1	
PORTB_L	2	
PORTB_H	3	
PORTC_L	4	
PORTC_H	5	
RNG_DET_IN	7:6	GPIO<13>

Interrupt Enable 0 0x014 FF

	EN
FULL_ERR	🟩
ADDR_ERR	🟩
CMD_FIN	🟩
OVF_ERR	🟩
CMD_ERR	🟩
WR_FIN	🟩
MPT_ERR	🟩
GPIO_INT	🟩

Interrupt Enable 1 0x015 FF

	BIT	EN
VD_INT	0	🟩
CPT_INT	1	🟩
DTMF_INT	2	🟩
TONE_INT	3	🟩
FSK_E	4	🟩
FSK_D	5	🟩
CAS_INT	6	🟩
RNG_INT	7	🟩

GPIO Configuration Registers

	DI	DO	OE	PE	PS	DS	AF	IE
PORTA	00	00	00	FF	FF	00	03	00
PORTB	00	00	00	FF	FF	00	00	00
PORTC	00	00	00	FF	FF	00	FC	00

GPIO Configuration Registers Address

	DI	DO	OE	PE	PS	DS	AF	IE
PORTA	0x01C	0x019	0x01A	0x01B	0x01D	0x01E	0x01F	0x00C
PORTB	0x033	0x030	0x031	0x032	0x034	0x035	0x036	0x00D
PORTC	0x03A	0x037	0x038	0x039	0x03B	0x03C	0x03D	0x00E

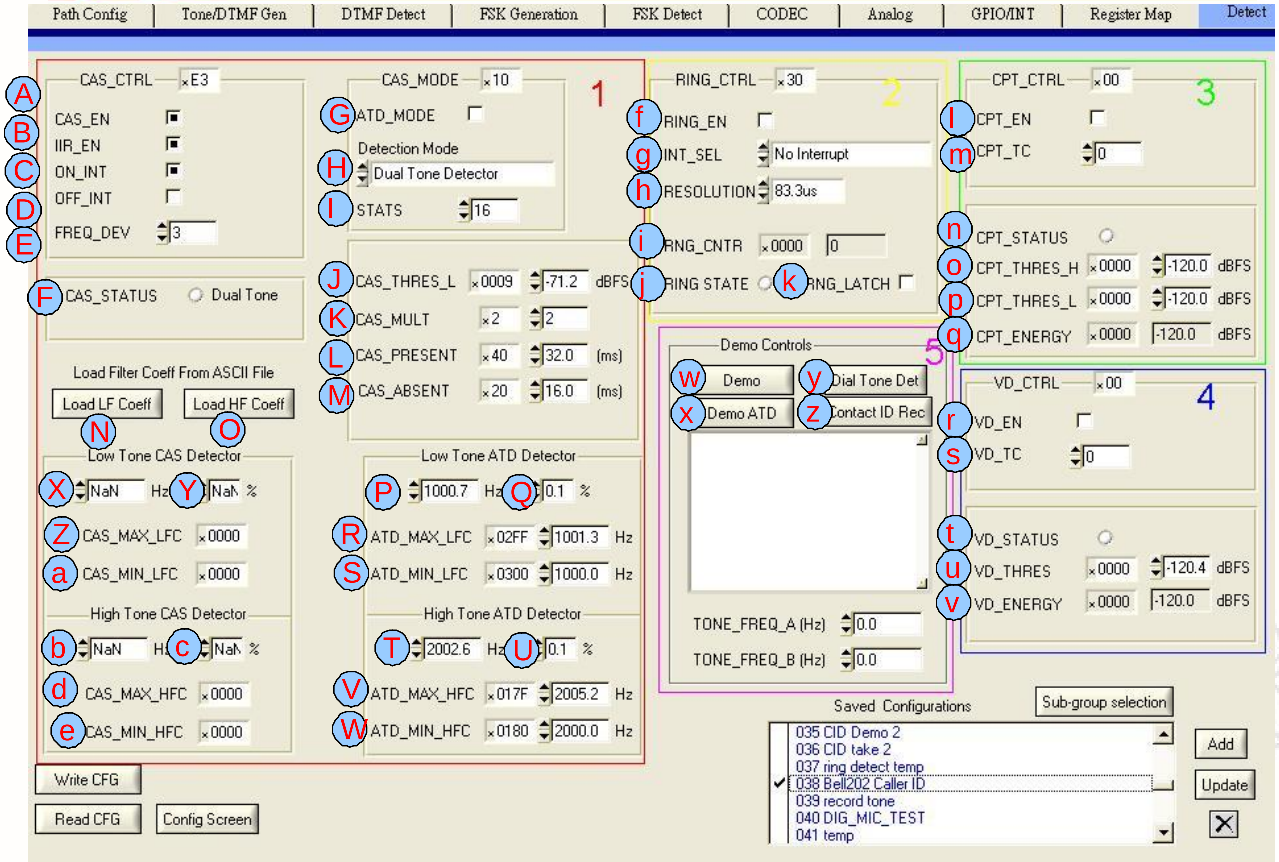
Saved Configurations

000 Default
001 1kHz Sine Wave
002 Mic in to SPK
003 DTMF String
004 PLAY
005 REC from MIC
006 FSK_GEN_DEMO

Figure 3-31 GPIO/INT screen

3.18 Detect

The detect panel has several function settings including CAS detect, ATD, RING detect, CPT detect and VD.



CAS_CTRL x E3
CAS_EN ☒
IIR_EN ☒
ON_INT ☒
OFF_INT □
FREQ_DEV 3
CAS_STATUS ○ Dual Tone

Low Tone CAS Detector
NaN Hz NaN %
CAS_MAX_LFC x 0000
CAS_MIN_LFC x 0000

High Tone CAS Detector
NaN Hz NaN %
CAS_MAX_HFC x 0000
CAS_MIN_HFC x 0000

CAS_MODE x 10
ATD_MODE □
Detection Mode
Dual Tone Detector
STATS 16

CAS_THRES_L x 0009 -71.2 dBFS
CAS_MULT x 2 2
CAS_PRESENT x 40 32.0 (ms)
CAS_ABSENT x 20 16.0 (ms)

Low Tone ATD Detector
1000.7 Hz 0.1 %
ATD_MAX_LFC x 02FF 1001.3 Hz
ATD_MIN_LFC x 0300 1000.0 Hz

High Tone ATD Detector
2002.6 Hz 0.1 %
ATD_MAX_HFC x 017F 2005.2 Hz
ATD_MIN_HFC x 0180 2000.0 Hz

RING_CTRL x 30
RING_EN □
INT_SEL No Interrupt
RESOLUTION 83.3us
RNG_CNTR x 0000 0
RING STATE ○ RNG_LATCH □

CPT_CTRL x 00
CPT_EN □
CPT_TC 0
CPT_STATUS ○
CPT_THRES_H x 0000 -120.0 dBFS
CPT_THRES_L x 0000 -120.0 dBFS
CPT_ENERGY x 0000 -120.0 dBFS

VD_CTRL x 00
VD_EN □
VD_TC 0
VD_STATUS ○
VD_THRES x 0000 -120.4 dBFS
VD_ENERGY x 0000 -120.0 dBFS

Demo Controls
TONE_FREQ_A (Hz) 0.0
TONE_FREQ_B (Hz) 0.0

Saved Configurations
035 CID Demo 2
036 CID take 2
037 ring detect temp
✓ 038 Bell202 Caller ID
039 record tone
040 DIG_MIC_TEST
041 temp

Figure 3-32: The Detect panel has 4 setting area and one demo area.

Area 1 is for CAS detect and ATD setting. The ATD_MODE(G) and Detection Mode(H) will select the Tone detect engine to operate as dual tone detect(ie CAS detect) or ATD (arbitrary tone detect). Area 2 is for ring detect. Area 3 is for CPT (Call Progress Tone) detect. Area 4 is for VD (Voice detect). Area 5 demo the function of tone detects with tone generation to generate the desired tones and feedback to tone detect.

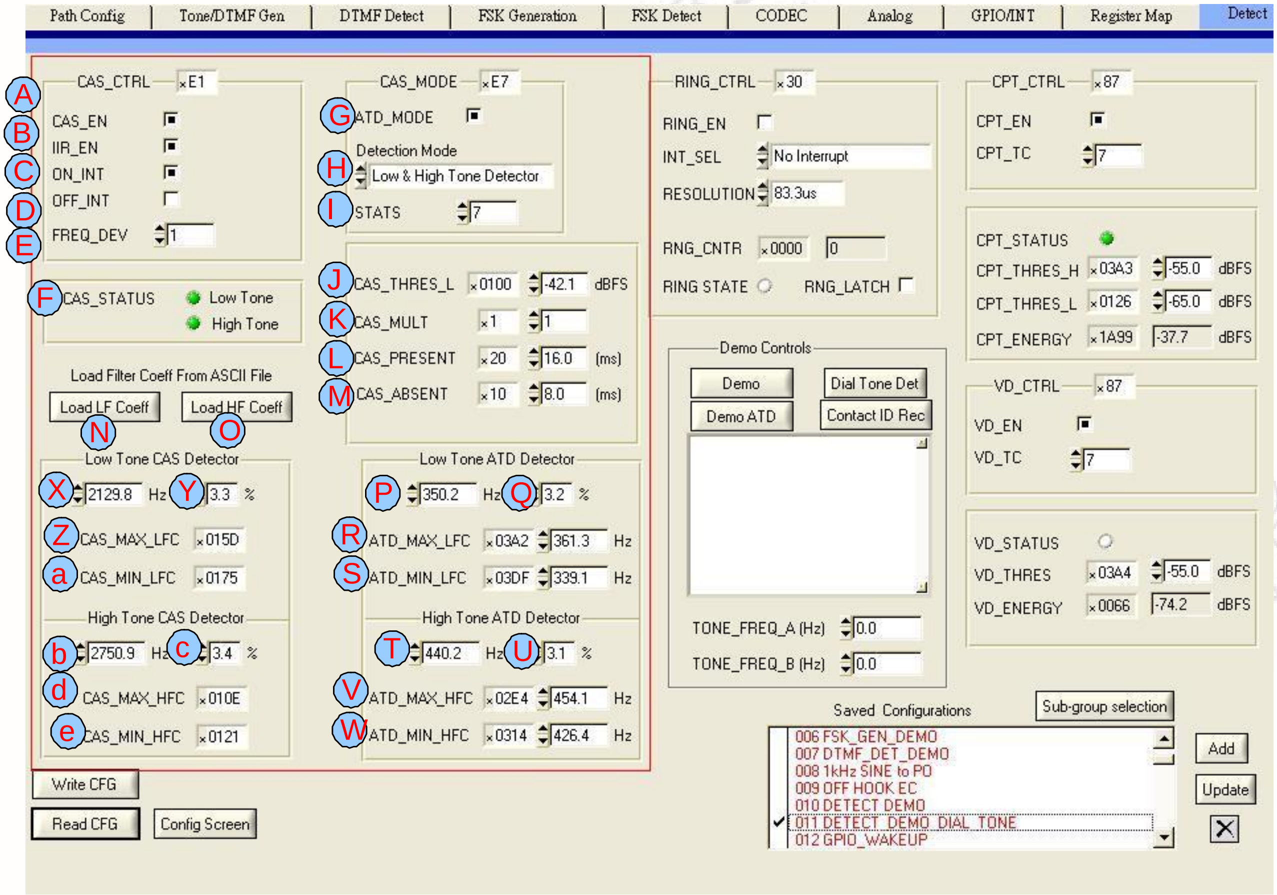
3.18.1 CAS detect

To use CAS detect:

1. Click CAS_EN (A) to 1, enable the CAS module.
2. Click IIR_EN (B) to 1, enable the IIR filter.
3. Click ON_INT (C) to 1, this will issue interrupt when CAS detected.
4. Click OFF_INT (D) to zero. If OFF_INT is 1, another interrupt will issued when CAS is absent.
5. Configure FREQ_DEV (E) to 3, select 3% deviation.
6. Click ATD_Mode (G) to 0. For CAS mode, this check box must be 0.
7. Configure Detection Mode (H) to Dual Tone Detector.
8. Configure the STATS (I) to decide number of sine wave cycles to judge.
9. Configure CAS_THRES_L (J) to decide low threshold
10. Configure CAS_MULT (K) to decide high threshold multiplier to low threshold.
11. Configure CAS_PRESENT (L) to decide present time.
12. Configure CAS_ABSENT (M) to decide absent time.
13. Press ‘Write CFG’

If CAS tone is detected, press ‘Read CFG’, the Dual Tone status (F) will change to green.

3.18.2 ATD



CAS_CTRL x E1
CAS_EN ☒
IIR_EN ☒
ON_INT ☒
OFF_INT □
FREQ_DEV 1
CAS_STATUS ● Low Tone ● High Tone

CAS_MODE x E7
ATD_MODE ☒
Detection Mode: Low & High Tone Detector
STATS 7

CAS_THRES_L x 0100 -42.1 dBFS
CAS_MULT x 1 1
CAS_PRESENT x 20 16.0 (ms)
CAS_ABSENT x 10 8.0 (ms)

Low Tone CAS Detector: 2129.8 Hz 3.3 %
CAS_MAX_LFC x 015D
CAS_MIN_LFC x 0175

High Tone CAS Detector: 2750.9 Hz 3.4 %
CAS_MAX_HFC x 010E
CAS_MIN_HFC x 0121

Low Tone ATD Detector: 350.2 Hz 3.2 %
ATD_MAX_LFC x 03A2 361.3 Hz
ATD_MIN_LFC x 03DF 339.1 Hz

High Tone ATD Detector: 440.2 Hz 3.1 %
ATD_MAX_HFC x 02E4 454.1 Hz
ATD_MIN_HFC x 0314 426.4 Hz

RING_CTRL x 30
RING_EN □
INT_SEL No Interrupt
RESOLUTION 83.3us
RNG_CNTR x 0000 0
RING STATE ○ RNG_LATCH □

TONE_FREQ_A (Hz) 0.0
TONE_FREQ_B (Hz) 0.0

CPT_CTRL x 87
CPT_EN ☒
CPT_TC 7
CPT_STATUS ●
CPT_THRES_H x 03A3 -55.0 dBFS
CPT_THRES_L x 0126 -65.0 dBFS
CPT_ENERGY x 1A99 -37.7 dBFS

VD_CTRL x 87
VD_EN ☒
VD_TC 7
VD_STATUS ○
VD_THRES x 03A4 -55.0 dBFS
VD_ENERGY x 0066 -74.2 dBFS

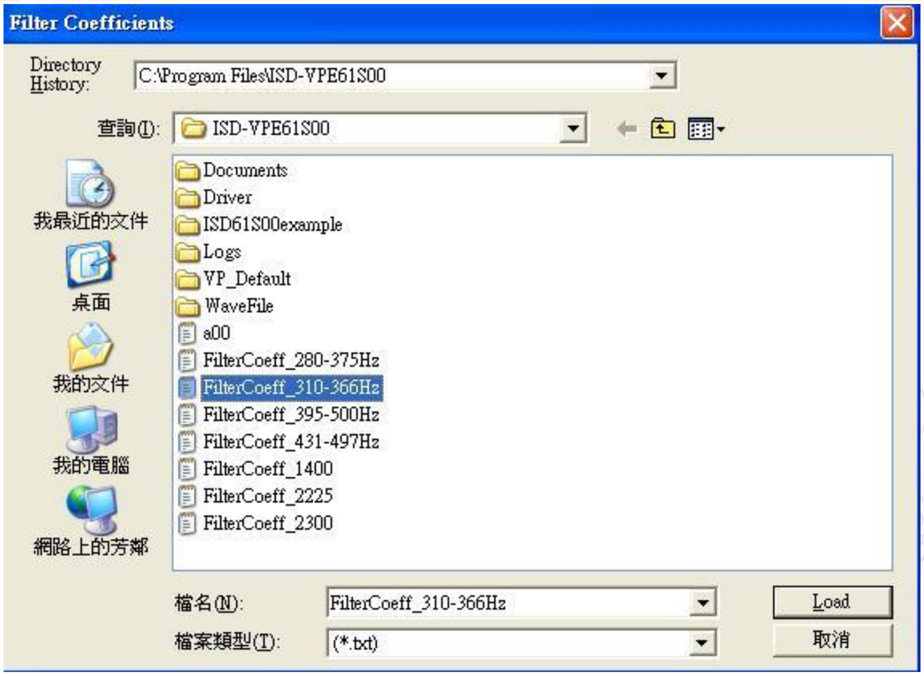
Saved Configurations
006 FSK_GEN_DEMO
007 DTMF_DET_DEMO
008 1kHz SINE to PO
009 OFF HOOK EC
010 DETECT DEMO
✓ 011 DETECT_DEMO_DIAL_TONE
012 GPIO_WAKEUP

Figure 3-33: An example of ATD setting and detected result.

To use ATD:

- Phase1: load band pass filter coefficients

1. Click CAS_EN (A) to 1, enable the CAS module.
2. Click IIR_EN (B) to 0, disable the IIR filter.
3. Click ATD_Mode(G) to 1. For ATD mode, this check box must be 1.
4. Configure Detection Mode (H) to Low&High Tone Detector.
5. Click Load LF Coeff (N), a file input dialog will ask user to select the coefficient file.



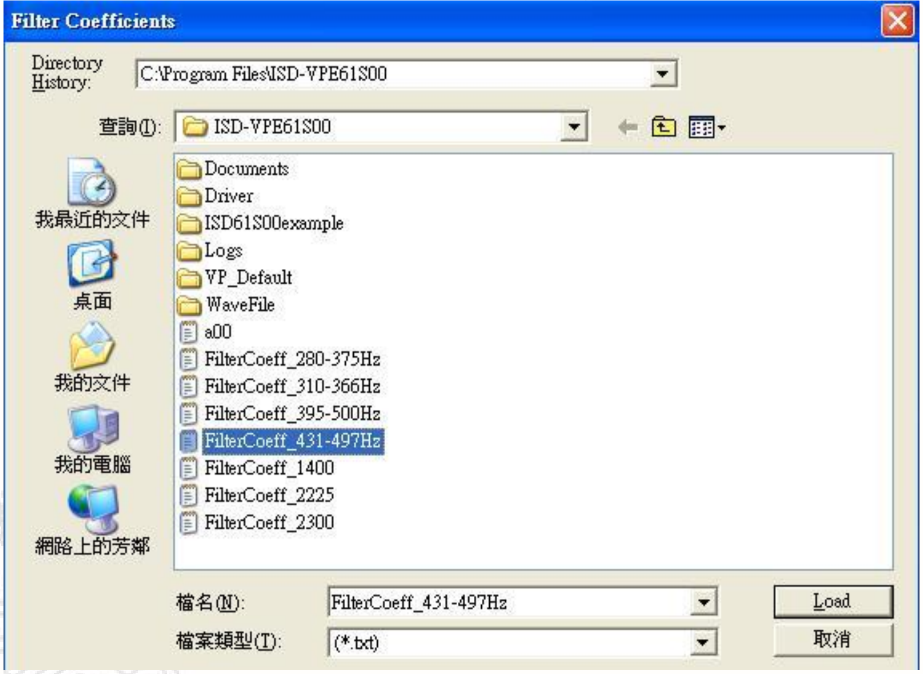
Directory History: C:\Program Files\ISD-VPE61S00

檔名(N): FilterCoeff_310-366Hz

檔案類型(T): (*.txt)

Choose the desired filter coefficient file and press 'Load'.

6. Click Load HF Coeff (O), another file input dialog will ask user to select the coefficient file.



Directory History: C:\Program Files\ISD-VPE61S00

檔名(N): FilterCoeff_431-497Hz

檔案類型(T): (*.txt)

Choose the desired filter coefficient file and press 'Load'.

7. Press 'Write CFG' and the Band pass Filter coefficients will write in the register.

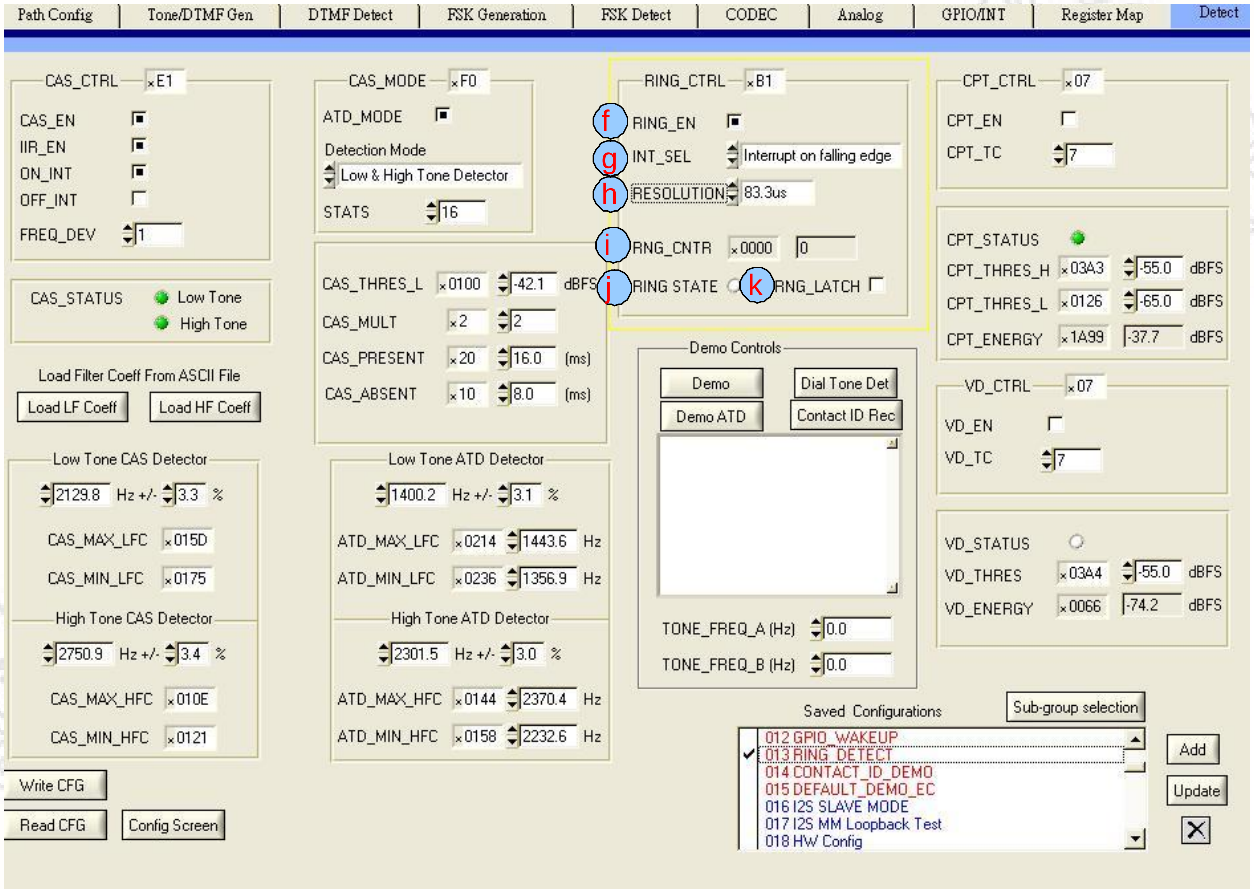
- Phase2: configure the detect parameter

1. Click IIR_EN (B) to 1, enable the IIR filter.

2. Click ON_INT (C) to 1, this will issue interrupt when tone detected.
3. Click OFF_INT (D) to zero. If OFF_INT is 1, another interrupt will issued when tone is absent.
4. Configure the STATS (I) to decide number of sine wave cycles to judge.
5. Configure CAS_THRES_L (J) to decide low threshold
6. Configure CAS_MULT (K) to 1
7. Configure CAS_PRESENT (L) to decide present time.
8. Configure CAS_ABSENT (M) to decide absent time.
9. Configure desired low tone frequency (P) and deviation percentage (Q). The ATD_MAX_LFC (R) and ATD_MIN_LFC (S) will be calculated by GUI automatically.
10. Configure desired high tone frequency (T) and deviation percentage (U). The ATD_MAX_HFC (V) and ATD_MIN_HFC (W) will be calculated by GUI automatically.
11. Press ‘Write CFG’

If low or high tone is detected, press ‘Read CFG’, the low tone or high tone status (F) will change to green.

3.18.3 Ring Detect



CAS_CTRL x E1
CAS_EN ☑
IIR_EN ☑
ON_INT ☑
OFF_INT □
FREQ_DEV 1

CAS_STATUS ● Low Tone ● High Tone

Load Filter Coeff From ASCII File

Low Tone CAS Detector
2129.8 Hz +/- 3.3 %
CAS_MAX_LFC x 015D
CAS_MIN_LFC x 0175

High Tone CAS Detector
2750.9 Hz +/- 3.4 %
CAS_MAX_HFC x 010E
CAS_MIN_HFC x 0121

CAS_MODE x F0
ATD_MODE ☑
Detection Mode: Low & High Tone Detector
STATS 16

CAS_THRES_L x 0100 -42.1 dBFS
CAS_MULT x 2 2
CAS_PRESENT x 20 16.0 (ms)
CAS_ABSENT x 10 8.0 (ms)

Low Tone ATD Detector
1400.2 Hz +/- 3.1 %
ATD_MAX_LFC x 0214 1443.6 Hz
ATD_MIN_LFC x 0236 1356.9 Hz

High Tone ATD Detector
2301.5 Hz +/- 3.0 %
ATD_MAX_HFC x 0144 2370.4 Hz
ATD_MIN_HFC x 0158 2232.6 Hz

RING_CTRL x B1
RING_EN ☑
INT_SEL Interrupt on falling edge
RESOLUTION 83.3us
RNG_CNTR x 0000 0
RING STATE ○ RNG_LATCH □

Demo Controls
TONE_FREQ_A (Hz) 0.0
TONE_FREQ_B (Hz) 0.0

CPT_CTRL x 07
CPT_EN □
CPT_TC 7

CPT_STATUS ●
CPT_THRES_H x 03A3 -55.0 dBFS
CPT_THRES_L x 0126 -65.0 dBFS
CPT_ENERGY x 1A99 -37.7 dBFS

VD_CTRL x 07
VD_EN □
VD_TC 7

VD_STATUS ○
VD_THRES x 03A4 -55.0 dBFS
VD_ENERGY x 0066 -74.2 dBFS

Saved Configurations
012 GPIO_WAKEUP
✓ 013 RING_DETECT
014 CONTACT_ID_DEMO
015 DEFAULT_DEMO_EC
016 I2S SLAVE MODE
017 I2S MM Loopback Test
018 HW Config

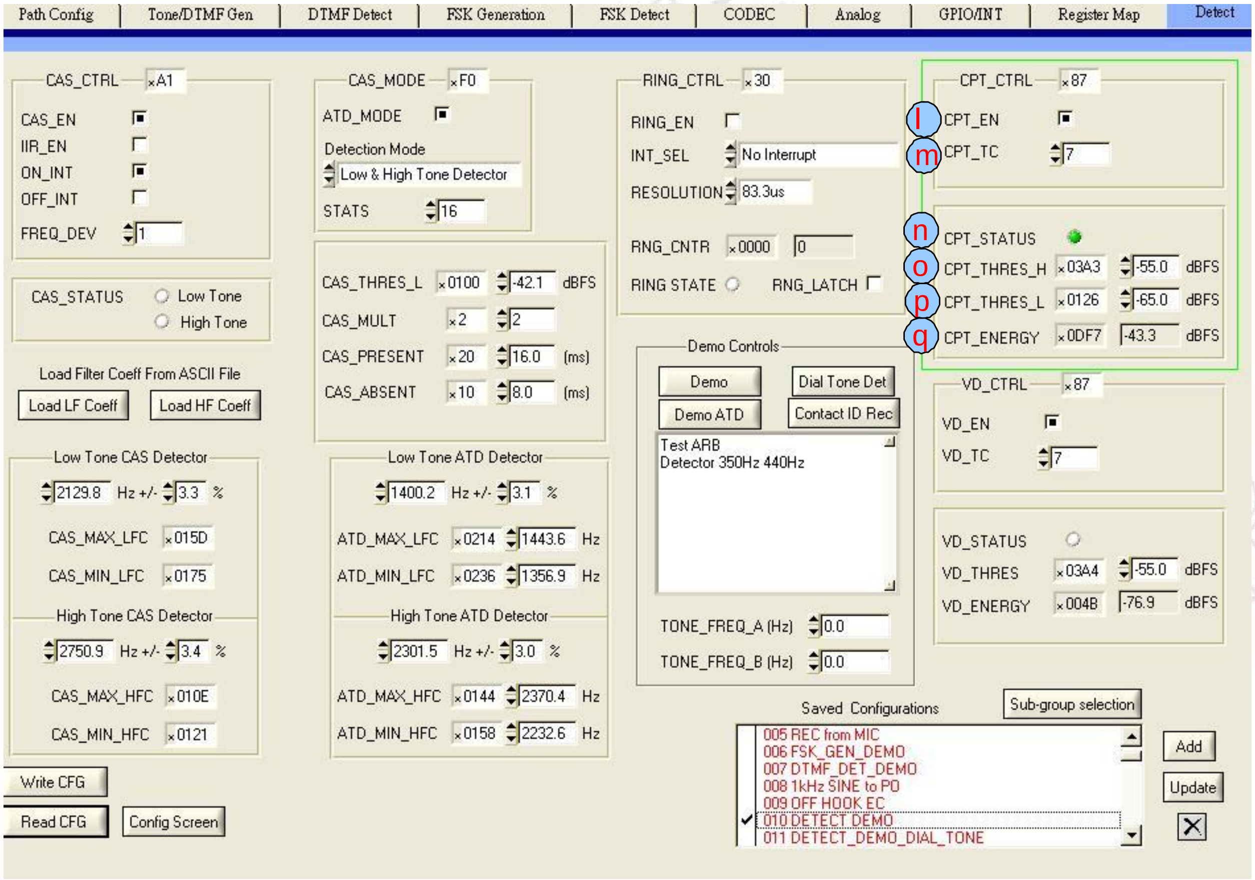
Figure 3-34: An example of Ring Detect setting.

To use Ring Detect:

1. Click RING_EN (f) to 1.
2. Configure INT_SEL (g) to choose INT occur edge.
3. Configure the RESOLUTION (h) to decide the counter increasing step.
4. Press ‘Write CFG’

If ring is applied in the Tip/Ring, the RING_INT will occur.

3.18.4 CPT Detect



CAS_CTRL x A1
CAS_EN ☒
IIR_EN □
ON_INT ☒
OFF_INT □
FREQ_DEV 1

CAS_STATUS ○ Low Tone ○ High Tone

Low Tone CAS Detector
2129.8 Hz +/- 3.3 %
CAS_MAX_LFC x 015D
CAS_MIN_LFC x 0175

High Tone CAS Detector
2750.9 Hz +/- 3.4 %
CAS_MAX_HFC x 010E
CAS_MIN_HFC x 0121

CAS_MODE x F0
ATD_MODE ☒
Detection Mode: Low & High Tone Detector
STATS 16

CAS_THRES_L x 0100 -42.1 dBFS
CAS_MULT x 2 2
CAS_PRESENT x 20 16.0 (ms)
CAS_ABSENT x 10 8.0 (ms)

Low Tone ATD Detector
1400.2 Hz +/- 3.1 %
ATD_MAX_LFC x 0214 1443.6 Hz
ATD_MIN_LFC x 0236 1356.9 Hz

High Tone ATD Detector
2301.5 Hz +/- 3.0 %
ATD_MAX_HFC x 0144 2370.4 Hz
ATD_MIN_HFC x 0158 2232.6 Hz

RING_CTRL x 30
RING_EN □
INT_SEL No Interrupt
RESOLUTION 83.3us
RNG_CNTR x 0000 0
RING STATE ○ RNG_LATCH □

Test ARB
Detector 350Hz 440Hz

TONE_FREQ_A (Hz) 0.0
TONE_FREQ_B (Hz) 0.0

CPT_CTRL x 87
CPT_EN ☒
CPT_TC 7
CPT_STATUS ●
CPT_THRES_H x 03A3 -55.0 dBFS
CPT_THRES_L x 0126 -65.0 dBFS
CPT_ENERGY x 0DF7 -43.3 dBFS

VD_CTRL x 87
VD_EN ☒
VD_TC 7
VD_STATUS ○
VD_THRES x 03A4 -55.0 dBFS
VD_ENERGY x 004B -76.9 dBFS

Saved Configurations
005 REC from MIC
006 FSK_GEN_DEMO
007 DTMF_DET_DEMO
008 1kHz SINE to PO
009 OFF HOOK EC
✓ 010 DETECT DEMO
011 DETECT_DEMO_DIAL_TONE

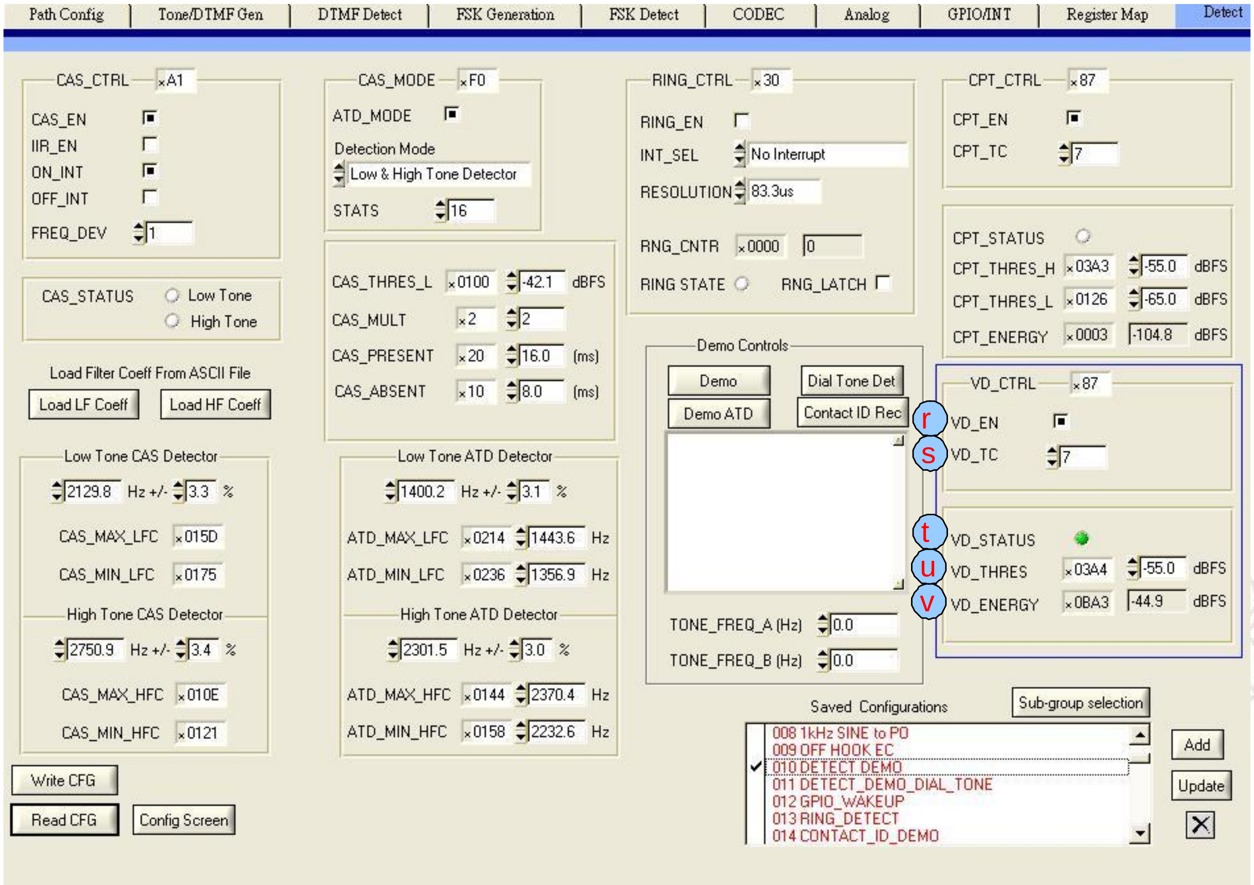
Figure 3-35: An example of CPT Detect setting.

To use CPT:

1. Click CPT_EN (l)to 1
2. Configure CPT_TC (m). Larger values imply a large time constant, greater stability but less response to change in energy level.
3. Configure CPT_THRES_H (o).
4. Configure CPT_THRES_L (p).
5. Press ‘Write CFG‘.

If signal energy (q) is over the CPT_THRES_H (o), Press ‘Read CFG’, the CPT_STATUS (n) will change to green.

3.18.5 VD



CAS_CTRL x A1
CAS_EN ☑
IIR_EN □
ON_INT ☑
OFF_INT □
FREQ_DEV 1

CAS_STATUS ○ Low Tone ○ High Tone

Load Filter Coeff From ASCII File

Low Tone CAS Detector
2129.8 Hz +/- 3.3 %
CAS_MAX_LFC x 015D
CAS_MIN_LFC x 0175

High Tone CAS Detector
2750.9 Hz +/- 3.4 %
CAS_MAX_HFC x 010E
CAS_MIN_HFC x 0121

CAS_MODE x F0
ATD_MODE ☑
Detection Mode
Low & High Tone Detector
STATS 16

CAS_THRES_L x 0100 -42.1 dBFS
CAS_MULT x 2 2
CAS_PRESENT x 20 16.0 (ms)
CAS_ABSENT x 10 8.0 (ms)

Low Tone ATD Detector
1400.2 Hz +/- 3.1 %
ATD_MAX_LFC x 0214 1443.6 Hz
ATD_MIN_LFC x 0236 1356.9 Hz

High Tone ATD Detector
2301.5 Hz +/- 3.0 %
ATD_MAX_HFC x 0144 2370.4 Hz
ATD_MIN_HFC x 0158 2232.6 Hz

RING_CTRL x 30
RING_EN □
INT_SEL No Interrupt
RESOLUTION 83.3us
RNG_CNTR x 0000 0
RING STATE ○ RNG_LATCH □

Demo Controls
TONE_FREQ_A (Hz) 0.0
TONE_FREQ_B (Hz) 0.0

CPT_CTRL x 87
CPT_EN ☑
CPT_TC 7

CPT_STATUS ○
CPT_THRES_H x 03A3 -55.0 dBFS
CPT_THRES_L x 0126 -65.0 dBFS
CPT_ENERGY x 0003 -104.8 dBFS

VD_CTRL x 87
VD_EN ☑
VD_TC 7

VD_STATUS ●
VD_THRES x 03A4 -55.0 dBFS
VD_ENERGY x 0BA3 -44.9 dBFS

Saved Configurations
008 1kHz SINE to PO
009 OFF HOOK EC
✓ 010 DETECT DEMO
011 DETECT_DEMO_DIAL_TONE
012 GPIO_WAKEUP
013 RING_DETECT
014 CONTACT_ID_DEMO

Figure 3-36: An example of VD setting.

To use VD:

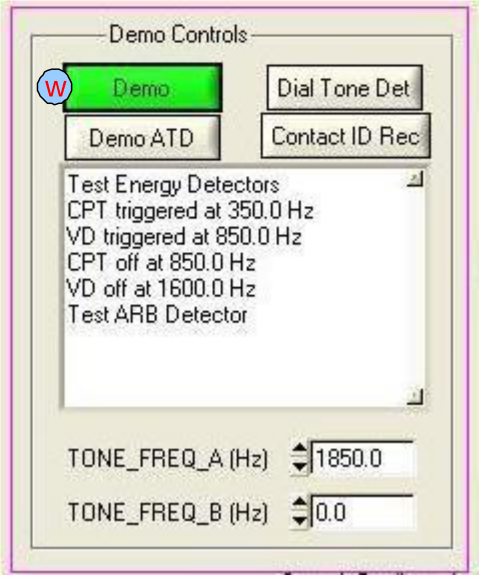
1. Click VD_EN (r) to 1
2. Configure VD_TC (s)
3. Configure VD_THRES (u)
4. Press 'Write CFG'

If signal energy (v) between 750~1400Hz is detected and over the VD_THRES (u), Press 'Read CFG', the VD_STATUS (t) will change to green.

3.18.6 Demo

The Demo area shows some demos.

1. Demo: Press ‘Demo’ (w) GUI will generate a series of tones and demonstrate the CPT and VD

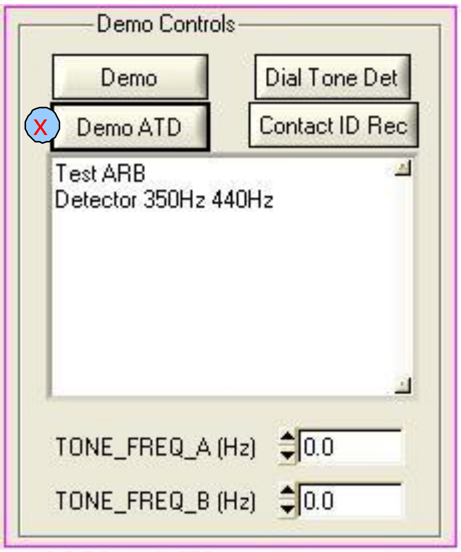


Test Energy Detectors
CPT triggered at 350.0 Hz
VD triggered at 850.0 Hz
CPT off at 850.0 Hz
VD off at 1600.0 Hz
Test ARB Detector

TONE_FREQ_A (Hz) 1850.0

TONE_FREQ_B (Hz) 0.0

2. Demo ATD: Press ‘Demo ATD’ (x), GUI will generate a series of two tones and the CAS_STATUS (?) will show detected tones.

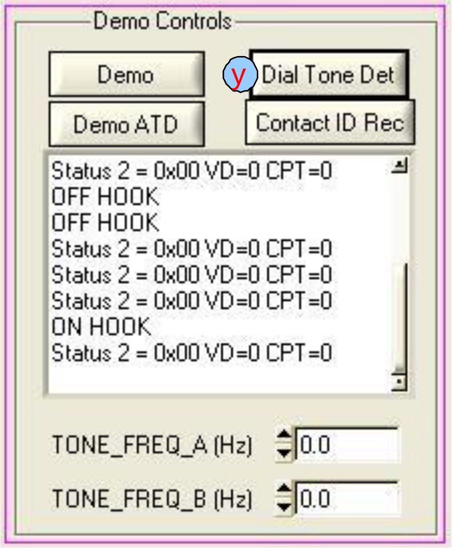


Test ARB
Detector 350Hz 440Hz

TONE_FREQ_A (Hz) 0.0

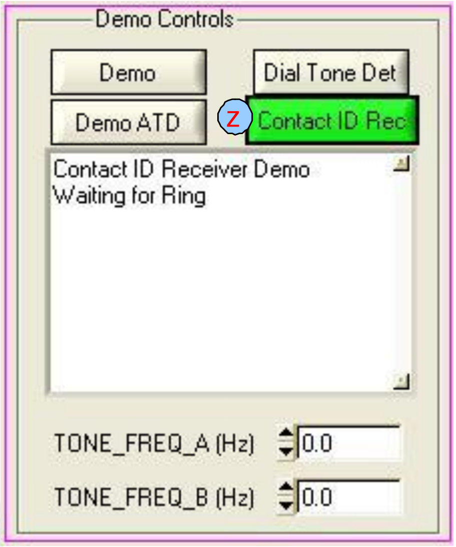
TONE_FREQ_B (Hz) 0.0

3. Dial Tone Det: Press ‘Dial Tone Det’ (y).



```
Status 2 = 0x00 VD=0 CPT=0
OFF HOOK
OFF HOOK
Status 2 = 0x00 VD=0 CPT=0
Status 2 = 0x00 VD=0 CPT=0
Status 2 = 0x00 VD=0 CPT=0
ON HOOK
Status 2 = 0x00 VD=0 CPT=0
```

4. Contact ID Rec: Press ‘Contact ID Rec’ (z).



```
Contact ID Receiver Demo
Waiting for Ring
```

4. Evaluation Board Control

When the ISD61S00 EV board is connected to PC, three extra screens show up:

- EVB Control: Allows control of Evaluation board, such as floating the PCM and SPI buses for connection of external drivers or setting SPI speed, setting USB device sample rate.
- Audio Cmds: Controls the ISD61S00 play and record commands such as executing voice macros, erasing recorded messages etc.
- Digital Cmds: Controls the ISD61S00 digital commands such as downloading a project, reading back for verification etc.

4.1 Audio Cmds

ISD61S00 provides two message recording mechanisms:

- 'Record Msg' (**A**):
  - Record a message without specifying address. ISD61S00 automatically searches for available memory sectors starting from the RMP (Recording Memory Pointer) for the message recordings. When a 'Stop' command is issued to finish the recording, or the recording is stopped due to memory full, VPE automatically reads back the recorded message address and length.
- 'Record Msg At' (**B**):
  - Record a message at the user-specified address, which is reserved in advance. By default the reserved messages are allocated before the RMP. Since the addresses of Reserved Messages are allocated in advance, there is no need to read back the recorded message address. Reserved Message is especially useful for a predefined scenario.

The 'Play Msg At' command (**C**) plays a message at the user-specified address. Users can play a message from the middle by specifying an offset (**D**). One offset is one memory sector, which is about one second based on 8kHz 4-bit ADPCM. ISD61S00 does not provide a PAUSE command; however, a PAUSE command can be easily done by toggling between 'Stop' and 'Play Msg At' with an offset.

To erase a message, simply specify the address and then click on 'Erase Msg At' (**E**). Addresses of recorded messages are shown on the Message List. (**F**)

'Get Msg List' (**G**) is not an ISD61S00 command. It is indeed a bunch of digital-read commands used to scan the memory for messages. The first byte of each memory sector shows whether or not a beginning of message is inside the memory sector.

VM (**H**) & VP (**I**) sub-screens show users the VM/VP tags and indexes that they are working on.

To execute a VM or play a VP, simply click on the VM/VP sub-screens to get the index, and then click on 'Execute VM' (**J**) or 'Play VP' (**K**).

VM and VP are index based, which means to execute a VM or play a VP the microcontroller only sends a simple command followed by the index. It is address-free! Please note that VM and VP share the indexes, and it always starts from VM. For example, if a project has three VMs and four VPs, then the indexes 0 ~ 2 belong to VMs, and the indexes 3 ~ 6 belong to VPs.

'Indirect Voice Prompts' (**L**) is to make voice macro flexible. For example, it could be used to announce time and temperature in users' system. To announce a sentence like '***It is 7:15 am; the outside temperature is 70 degree Fahrenheit***', do the following:

- Microcontroller gets the time (*7:15 am*) and temperature (*70* degree Fahrenheit) from the system's timer and sensor.
- Write the corresponding voice prompt indexes of *7*, *15*, *am*, and *70* into R0, R1, R2, and R3.
- Execute the voice macro of that sentence.

To emulate it on VPE, click on VP sub-screen (**I**) to get the index, then right-click on Rn, n = 0 ~ 7, to update Rn, then click on 'Play VP@Rn' (**L**).

ISD61S00 provides four 'SPI Data Transfer' commands. For details, please refer to design guide. Take 'SPI Decode' (**M**) as an example, microcontroller sends compressed audio data (VPs or recorded messages) to ISD61S00 to decode. Double-click on '004 PLAY' (**N**) to set a playback path for 'SPI Decode'. Check 'CTS SPI' (**O**) if users want a continuous SPI transaction, which means VPE ignores all other tasks before the 'SPI Data Transfer' command finishes. Right click on the Filename blank (**P**) to assign a compressed audio file, and then click on 'SPI Decode' (**M**).

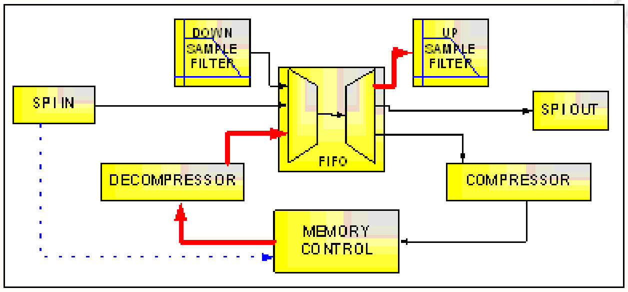


Figure 4-1 SPI Send Compressed Data to Decode

'Read ID' (**Q**) reads back the four-byte ISD61S00 ID.
'Read Status' and 'Read INT' (**R**) both read back the 4-byte ISD61S00 status. The only difference is that 'Read INT' clears the pending interrupt signal as well. ISD61S00 will not generate a new interrupt signal if there is a pending one.

'Saved Configurations' (**N**) is a copy from the 'Configuration' screen. Double-click on a setting to write to the ISD61S00.

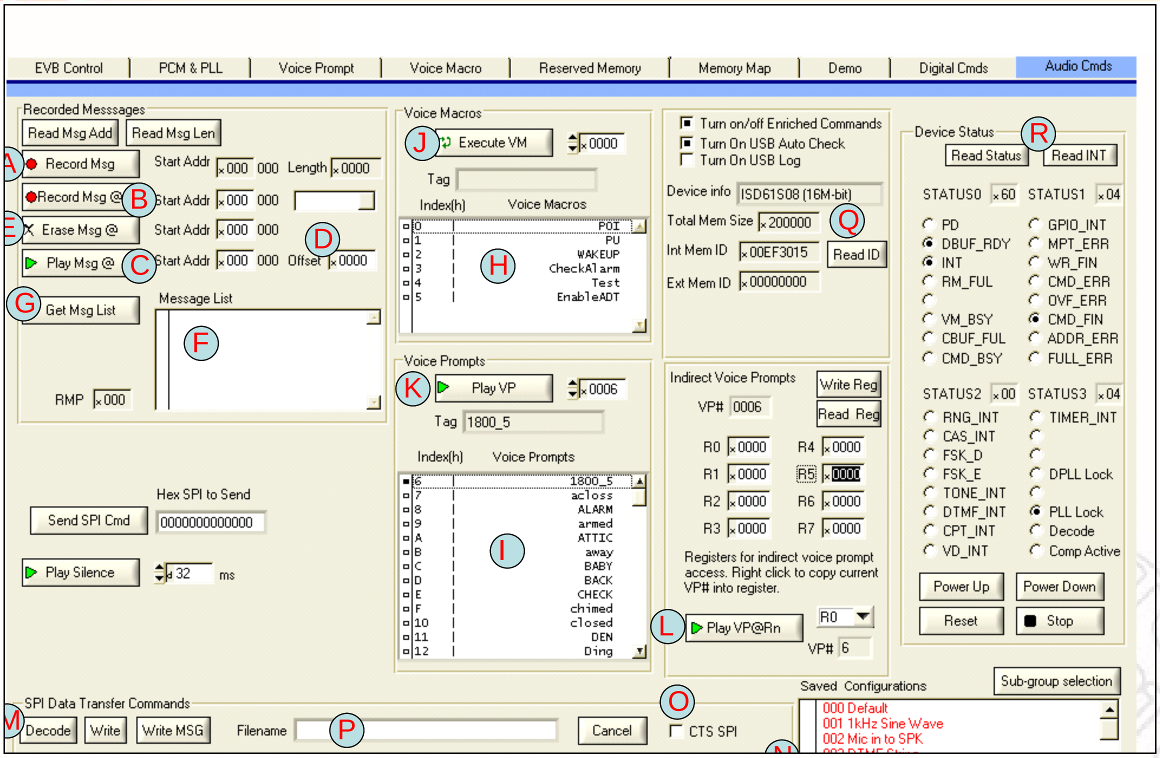


Figure 4-2 Audio Cmds

4.2 Digital Cmds

ISD61S00 provides four digital commands: digital-read (**A**), digital-write (**B**), erase-memory (**C**), and chip-erase (**D**).

To erase the whole memory, click on ‘Chip Erase’ (**D**); while the erasing is in progress, status bit CMD_BSY (**E**) goes high. Time for erasing the whole chip depends on the memory density. Below is a table for reference.

Device	Memory Density	Typical Erasing Time
ISD61S02	4Mbit	5 sec
ISD61S04	8Mbit	10 sec
ISD61S08	16Mbit	25 sec
ISD61S16	32Mbit	40 sec
ISD61S32	64Mbit	80 sec

Table 4-1 Chip Erasing Time

To erase partial memory, specify the start (**F**) and end addresses (or start address and number of sectors to erase), and then click on ‘Erase Memory’ (**C**) ; while the erasing is in progress, status bit CMD_BSY (**E**) goes high. The minimum erasable memory size is one sector (4k bytes).

Memory contents can be read back and either displayed on screen or saved as a file on PC.
To display the memory contents on screen:

(3) Specify the start (**G**) and end addresses (or start address and number of bytes to read).

(4) Specify to display on screen (**H**) (the default).
(5) Click on ‘Digital Read’ (**A**).

To save the memory contents on PC:

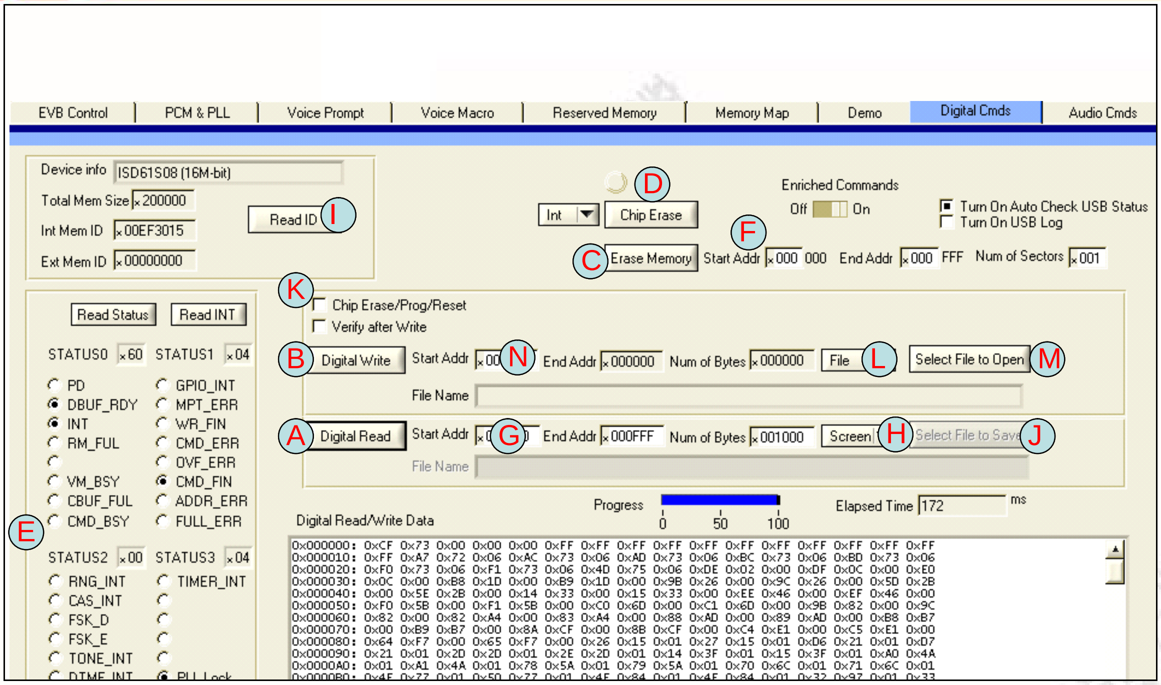
- Specify the start (**G**) and end addresses (or start address and number of bytes to read).
    - Users can click on ‘Read ID’ (**I**) to get the memory size.
- Specify to save as a file (**H**). Click on ‘Select File to Save’ (**J**) to specify the path and filename.
- Click on ‘Digital Read’ (**A**).

Memory contents can be updated by digital-write from either a file on PC or from the screen.
To update the memory contents from a file:

1. Want to erase the whole chip and then download a file from PC:
    - Check ‘Mass Erase/Prog/Reset’ (**K**).
    - Specify to update from a file (**L**) (the default). Click on ‘Select File to Open’ (**M**) to specify the path and filename.
    - The start (**N**) and end addresses will be loaded automatically.
    - Click on ‘Digital Write’ (**B**), and VPE will chip-erase the whole memory and then download the file to the ISD61S00.
2. Just overwrite partial memory contents by a file from PC (the default):
    - Uncheck ‘Mass Erase/Prog/Reset’ (**K**).
    - Similar procedure as the above. VPE accomplishes this task via the help of PC:
        1. If needed,
            - Read back one 4kbyte sector and save on PC buffer.
            - Overwrite the buffer by the corresponding portion of the file.
        2. Sector-erase the sector.
        3. Download the buffer to the sector.
        4. Repeat the above steps.

To update the memory contents from the screen:

- Read out the ISD61S00 data and have it displayed on screen as described above.
- Edit the data contents on screen.
    - The first sector (sector 0) is about the ISD61S00 configuration. Do not mess it up.
- Specify to update the memory contents from screen (**L**).
- The start (**N**) and end addresses will be automatically matched.
- Click on ‘Digital Write’ (**B**); VPE accomplishes this task via the help of PC:
    1. If needed,
        - Read back one 4kbyte sector and save on PC buffer.
        - Overwrite the buffer by the corresponding portion of the file.
    2. Sector-erase the sector.
    3. Download the buffer to the sector.
    4. Repeat the above steps.



Device info ISD61S08 (16M-bit)
Total Mem Size x 200000
Int Mem ID x 00EF3015
Ext Mem ID x 00000000

Int
Enriched Commands: Off / On
☒ Turn On Auto Check USB Status
□ Turn On USB Log

Start Addr x 000 000 End Addr x 000 FFF Num of Sectors x 001

□ Chip Erase/Prog/Reset
□ Verify after Write

Digital Write: Start Addr x 00 End Addr x 000000 Num of Bytes x 000000 File
File Name

Digital Read: Start Addr x 0 0 End Addr x 000FFF Num of Bytes x 001000 Screen
File Name

Elapsed Time 172 ms

STATUS0 x 60 STATUS1 x 04

STATUS0	STATUS1
○ PD	○ GPIO_INT
● DBUF_RDY	○ MPT_ERR
● INT	○ WR_FIN
○ RM_FUL	○ CMD_ERR
○	○ OVF_ERR
○ VM_BSY	● CMD_FIN
○ CBUF_FUL	○ ADDR_ERR
○ CMD_BSY	○ FULL_ERR

STATUS2 x 00 STATUS3 x 04

STATUS2	STATUS3
○ RNG_INT	○ TIMER_INT
○ CAS_INT	○
○ FSK_D	○
○ FSK_E	○
○ TONE_INT	○
○ DTME_INT	● PLL Lock

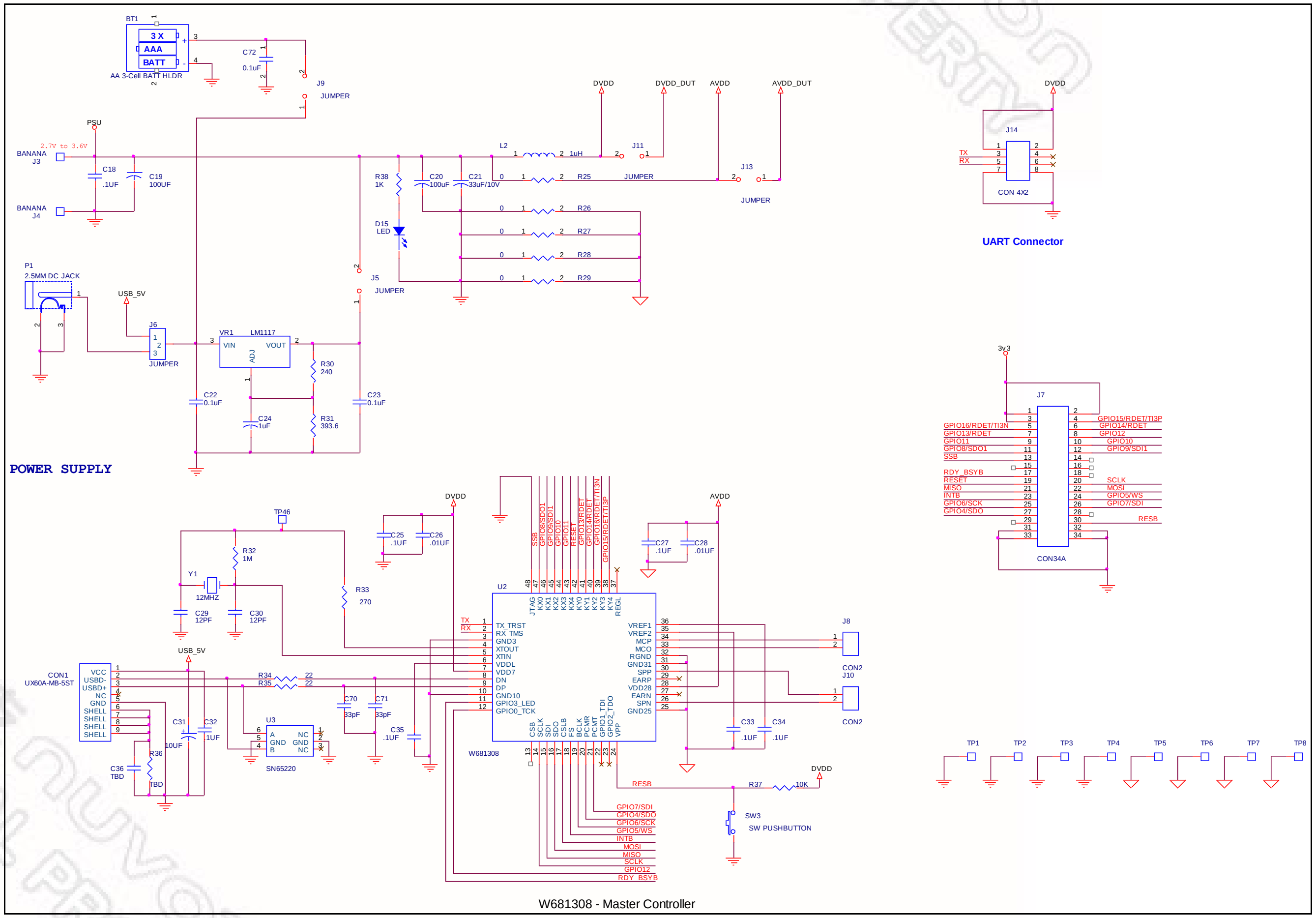
Digital Read/Write Data

```
0x000000: 0xCF 0x73 0x00 0x00 0x00 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF
0x000010: 0xFF 0xA7 0x72 0x06 0xAC 0x73 0x06 0xAD 0x73 0x06 0xBC 0x73 0x06 0xBD 0x73 0x06
0x000020: 0xF0 0x73 0x06 0xF1 0x73 0x06 0x4D 0x75 0x06 0xDE 0x02 0x00 0xDF 0x0C 0x00 0xE0
0x000030: 0x0C 0x00 0xB8 0x1D 0x00 0xB9 0x1D 0x00 0x9B 0x26 0x00 0x9C 0x26 0x00 0x5D 0x2B
0x000040: 0x00 0x5E 0x2B 0x00 0x14 0x33 0x00 0x15 0x33 0x00 0xEE 0x46 0x00 0xEF 0x46 0x00
0x000050: 0xF0 0x5B 0x00 0xF1 0x5B 0x00 0xC0 0x6D 0x00 0xC1 0x6D 0x00 0x9B 0x82 0x00 0x9C
0x000060: 0x82 0x00 0x82 0xA4 0x00 0x83 0xA4 0x00 0x88 0xAD 0x00 0x89 0xAD 0x00 0xB8 0xB7
0x000070: 0x00 0xB9 0xB7 0x00 0x8A 0xCF 0x00 0x8B 0xCF 0x00 0xC4 0xE1 0x00 0xC5 0xE1 0x00
0x000080: 0x64 0xF7 0x00 0x65 0xF7 0x00 0x26 0x15 0x01 0x27 0x15 0x01 0xD6 0x21 0x01 0xD7
0x000090: 0x21 0x01 0x2D 0x2D 0x01 0x2E 0x2D 0x01 0x14 0x3F 0x01 0x15 0x3F 0x01 0xA0 0x4A
0x0000A0: 0x01 0xA1 0x4A 0x01 0x78 0x5A 0x01 0x79 0x5A 0x01 0x70 0x6C 0x01 0x71 0x6C 0x01
0x0000B0: 0x4F 0x77 0x01 0x50 0x77 0x01 0x4F 0x84 0x01 0x4F 0x84 0x01 0x32 0x97 0x01 0x33
```

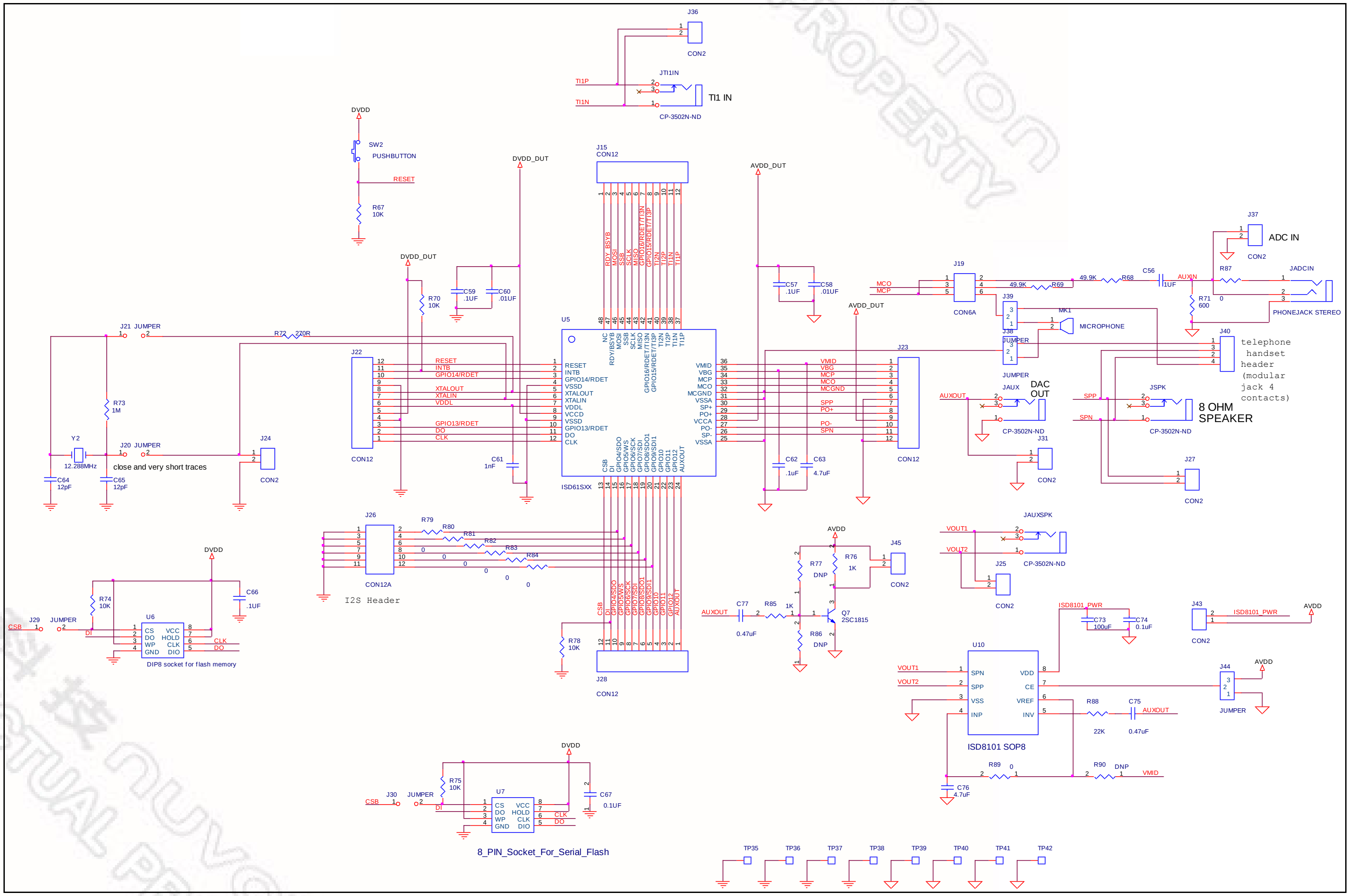
Figure 4-3 Digital Cmds

5. Board Schematic

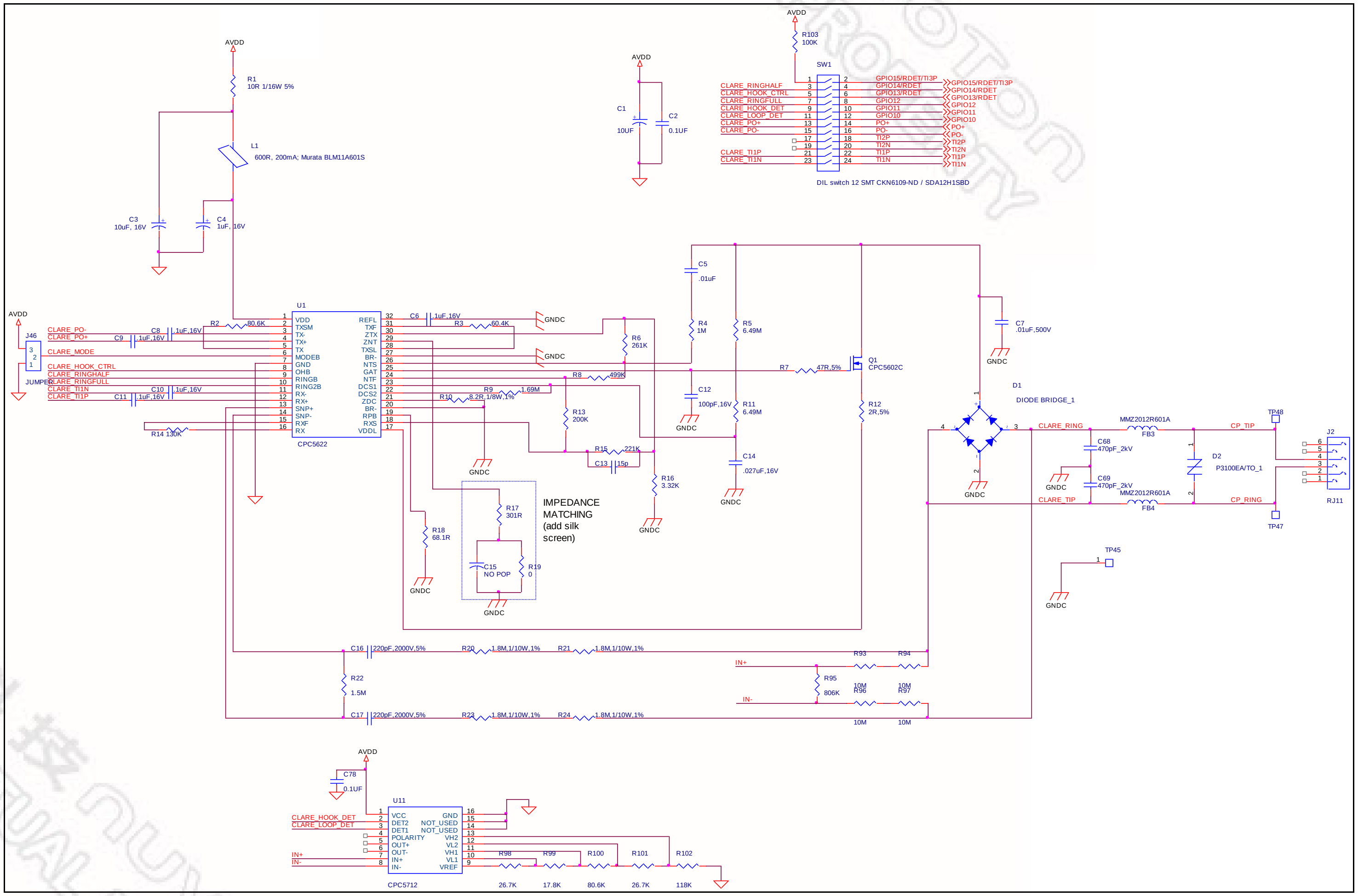
5.1 USB Controller and Power Sources



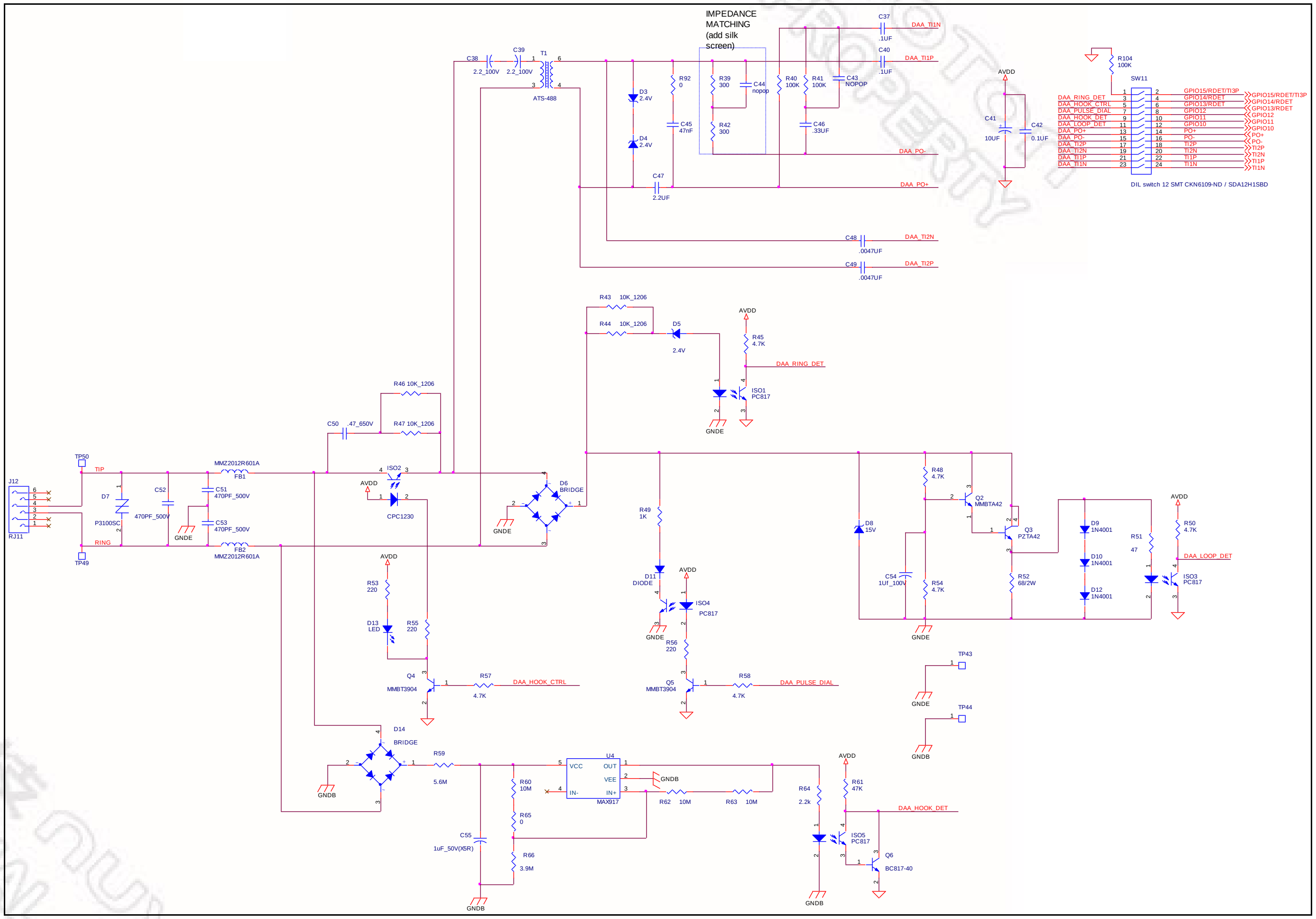
5.2 ISD61S00 and Related Circuitry



5.3 Clare DAA


AVDD
R1
10R 1/16W 5%
L1
600R, 200mA; Murata BLM11A601S
C3
10uF, 16V
C4
1uF, 16V
AVDD
C1
10UF
C2
0.1UF
AVDD
R103
100K
SW1
CLARE_RINGHALF
CLARE_HOOK_CTRL
CLARE_RINGFULL
CLARE_HOOK_DET
CLARE_LOOP_DET
CLARE_PO+
CLARE_PO-
CLARE_TI1P
CLARE_TI1N
GPIO15/RDET/TI3P
GPIO14/RDET
GPIO13/RDET
GPIO12
GPIO11
GPIO10
PO+
PO-
TI2P
TI2N
TI1P
TI1N
DIL switch 12 SMT CKN6109-ND / SDA12H1SBD
AVDD
J46
JUMPER
CLARE_PO-
CLARE_PO+
CLARE_MODE
CLARE_HOOK_CTRL
CLARE_RINGHALF
CLARE_RINGFULL
CLARE_TI1N
CLARE_TI1P
C8 .1uF,16V
C9 .1uF,16V
C10 .1uF,16V
C11 .1uF,16V
R2 80.6K
R14 130K
U1
VDD TXSM TX- TX+ TX MODEB GND OHB RINGB RING2B RX- RX+ SNP+ SNP- RXF RX
REFL TXF ZTX ZNT TXSL BR- NTS GAT NTF DCS1 DCS2 ZDC BR- RPB RXS VDDL
CPC5622
C6 .1uF,16V
R3 60.4K
GNDC
R6 261K
R8 499K
R9 1.69M
R10 8.2R,1/8W,1%
R13 200K
R15 221K
C13 15p
R16 3.32K
R17 301R
R18 68.1R
C15 NO POP
R19 0
IMPEDANCE MATCHING (add silk screen)
C5 .01uF
R4 1M
R5 6.49M
C12 100pF,16V
R11 6.49M
C14 .027uF,16V
R7 47R,5%
Q1 CPC5602C
R12 2R,5%
C7 .01uF,500V
D1
DIODE BRIDGE_1
CLARE_RING
CLARE_TIP
MMZ2012R601A FB3
MMZ2012R601A FB4
C68 470pF_2kV
C69 470pF_2kV
D2 P3100EA/TO_1
CP_TIP
CP_RING
TP48
TP47
TP45
J2
RJ11
C16 220pF,2000V,5%
R20 1.8M,1/10W,1%
R21 1.8M,1/10W,1%
R22 1.5M
C17 220pF,2000V,5%
R23 1.8M,1/10W,1%
R24 1.8M,1/10W,1%
IN+
IN-
R93 10M
R94 10M
R95 806K
R96 10M
R97 10M
AVDD
C78 0.1UF
U11
VCC DET2 DET1 POLARITY OUT+ OUT- IN+ IN-
GND NOT_USED NOT_USED VH2 VL2 VH1 VL1 VREF
CLARE_HOOK_DET
CLARE_LOOP_DET
CPC5712
R98 26.7K
R99 17.8K
R100 80.6K
R101 26.7K
R102 118K

5.4 Discrete DAA



6. Revision History

Version	Date	Description
61.0010	Nov 10, 2008	Initial release:
61.0031	May 5, 2010	Added telephony description.