![Texas Instruments logo]

DM81xx DVR RDK Demo Guide

Version 1.0

For DVRRDK Ver 02.80.00.XX

Revision History

Version	Date	Revision History
0.1	29 Aug 2011	Demo Details Added
0.2	14 Sep 2011	Added details on McFW DVR Demo for new features
0.3	29 Sep 2011	Updated McFW DVR Demo details, Removed Chains Demo
0.4	28 Oct 2011	Updated as per new user inputs
0.5	02 Dec 2011	Updated as per new user inputs
0.6	30 Dec 2011	DM816x & DM814x demo guides merged. A single demo guide applicable for DVR RDK release for all platforms
0.7	07 Feb 2012	Decoder only Application section updated
0.8	08Feb 2012	SCD/LMD user interface changes updated
0.9	16 Feb 2012	Snapshots removed, document cleaned up.
1.0	25 May 2012	Updated Menu Options, descriptions

TABLE OF CONTENTS

1   Overview

DVR RDK comprise of multiple applications depicting the use cases required for implementing DVR products.

The document provides details on how to use the applications provided in this release package.

The applications included in the package are:

1. Multi Channel Framework DVR Application

Multi Channel FW Application is based on McFW APIs and demonstrates the data flow for 16 CH Encode and Decode DVR

2. Multi Channel Decode Only Application

3. Multi Channel Capture, Encode Application

4. Multi Channel Capture, Display Application

Note

o Several applications are made available to exercise the capability of DVR Multi Channel Framework. Few applications might not be supported in different release versions / platforms

o Demo guide is common for DM81xx platforms. All usecase / options are not valid in all platforms. Refer release notes on application specification supported on specific platforms.

2   Running the Main Applications

- On the console screen, go to the directory

  /opt/dvr_rdk/ti816x/ **(or)** /opt/dvr_rdk/ti814x

  and run the scripts as below:

```
$cd /opt/dvr_rdk/ti816x/
$./init.sh


"init.sh" initializes the shared regions and loads the required
kernel modules
```

- Then run the load.sh script and the output would be as shown after the command:

```
$./load.sh
```

```
"load.sh" uses firmware load utility to load the VPSS and Video
M3 executables. Also, it sets up the remote debug print utility
to receive the prints from M3 and DSP cores on A8.
```

2.1 Multi-Channel Framework DVR Application

This application displays a sample use of the TI Multi-channel framework APIs

Steps to run this particular application are as below:

1. On the console screen, and run the script below

```
$./run.sh
```

2. On running the above script you should see a user menu

   1: VCAP + VENC + VDEC + VDIS - Progressive SD Encode + Decode

   3: VCAP + VENC   + VDIS - SD Encode ONLY

   4: VCAP + VENC   + VDIS - HD Encode ONLY

   5:   VDEC + VDIS - SD/HD Decode ONLY

   6: VCAP   + VDIS - NO Encode or Decode

   There will be a set of Custom demos also shown on the user menu.

3. To execute any application enter as input. If you want to run a Progressive Capture / Encode / Decode / Display demo, press 1.

    This user guide explains supported controls with respect to use case 1.

4. After choosing option 1 of the above options, you will be shown the default channel details showing the number of primary <D1> & secondary channels <CIF>. For Example

   --------------- CHANNEL DETAILS-------------

   Capture Channels => 16

   Enc Channels => Primary 16, Secondary 16

   Dec Channels => 16

   Disp Channels => 32

5. After each of the above step you will be asked whether to enable file write mode or not. To enable press 'y' and then you will be asked the location where u want to save the file.

   First user would be asked to enable file write of SCD block Meta data. User can enable file write of a particular channel. Once user has chosen this option, user would be asked to enable file write of bit stream.

   This provides option of writing captured video to a location in file system <hard disk or nfs>. User can choose to write the captured video data for a later analysis. No read back is provided. By default, only 1 channel write for each stream type is enabled.

> Number of channels getting stored can be controlled with MCFW_IPC_BITS_FWRITE_ENABLE_BITMASK_DEFAULT macro in demos/mcfw_api_demos/mcfw_demo/demo_vcap_venc_vdec_vdis.h file.
>
> The maximum file size allowed for writing video data is predefined. This can be controlled with MCFW_IPC_BITS_MAX_FILE_SIZE macro in ti_mcfw_ipcbits.c file. Both these options are build time options.
>
> In current application, the maximum file size is the available space on the partition used for saving the file. File writes stop once the maximum file size is reached.

> **Note**: For two demos namely
> DEMO_VCAP_VENC_VDEC_VDIS_PROGRESSIVE_NON_D1 &
> DEMO_VCAP_VENC_VDEC_VDIS_PROGRESSIVE_16CH_NRT on **814X & 810x** platforms
> File write options are overridden in the demo file. Hence to enable file write options you need to modify in the demo file demos\mcfw_api_demos\mcfw_demo\Demo_vcap_venc_vdec_vdis.c

To disable file write mode just enter 'n'.

6. If you choose to give a input file of your own stream to see the decoder box functionality, enter 5 and fill the location of the streams as per instructions required to be decoded as shown below: This use-case has been described in Section 2.2

7. After either of the above choices, you should be able to see the outputs.

2.1.1 Generic Notes on Options Selection

- Channel numbers will vary depending upon the usecase executed. On UI input, min~max range of channels specific to usecase will be displayed. User should select the channels according to that range.

- Netra HD encoder demo is the 4 channel Vcap+VEnc+VDis demo application (Demo option 4 in user menu). In this demo 4 channels are captured via 4 different TVPs. In this demo 0th channel of each TVP is feed video input to 4 capture channels. Below list shows the logical to physical channel mapping.

HD Enc Demo Channel No.	Channel Mapping to Physical Channel
Channel 0	0th channel of TVP-0 (Physical Channel 0)
Channel 1	0th channel of TVP-1 (Physical Channel 4)
Channel 2	0th channel of TVP-2 (Physical Channel 8)
Channel 3	0th channel of TVP-3 (Physical Channel 12)

- Channel numbers might vary in your actual run depending on use case executed.

- Few UI options, explicitly display message informing the availability of the feature on specific usecase. Selecting an invalid option will end up in undefined behavior.

- The data flow of individual usecases will be part of McFW user guide (or) release notes. Also, the specific usecase files in mcfw source directory have an ASCII data flow diagram at the top of the file. For e.g., mcfw\src_linux\mcfw_api\usecases\ti814x\multich_progressive_4d1_vcap_venc_vdec_vdis.c has the data flow diagram for 814x – 4D1 progressive usecase.

2.1.2   Capture Settings

Capture settings can be controlled using "Capture Settings" UI option as explained below.

Note: These settings are not applicable for use cases not involving CAPTURE

1. Disable channel

It disables a particular channel at the capture; hence you would not be able to see the channels live output and the decoded output on the displays.

2. Enable channel

It enables a previously disabled channel at the capture; hence you would be able to see the channels live output and the decoded output on the displays.

3. OSD Window position

Position of the OSD window can be varied by specifying the window number (0 or 1 or 2), and the new position coordinates(X, Y) for that window. After which you will have the OSD window at your desired position.

Note: OSD settings are not valid on all use cases.

4. OSD Window size

Note: Please refer to the source code to get the usage of VCAP_OSDWINPRM

5. OSD Window transparency

User can enable or disable transparency of the OSD window by selecting this option.

Currently, the transparency value is 0 (Luma value is 0 and Chroma value is 0). This means if the OSD overlaying content has value of 0 and transparency is enabled, the pixels with value 0 will become transparent.

Note: In current application, the sample OSD content does not have any 0x0 pixels, so the effect of enable/disable transparency is not noticed.

6. OSD Window global alpha

The OSD Window global alpha can be set by this option, entered value will be valid for the complete overlay content.

User can set the value from 0 to 128. The value entered is U1Q7 format where the MSB is integer value and other seven bits are fraction.

User Input	Hex Equivalent	Decimal Interpretation	Alpha Blending Percentage	Output seen on Display
0	0x00	0.0	0%	OSD window is seen 100% and video behind the OSD window is not seen
64	0x40	0.5	50%	OSD window is blended 50%
128	0x80	1.0	100%	OSD window is not seen and video is fully seen

7. OSD Window enable/disable

OSD window itself can be enabled\disabled using this option.

8. Detect video

The input video format (PAL/NTSC) will be detected and notified with this option.

9. Dynamic resolution change

This option allows changing the resolution of primary/secondary channels to pre-defined resolutions (D1, VGA, HALD-D1, CIF and QCIF)

This option is not valid on ALL use-cases.

a. SCD mode selection

To select different SCD modes on a particular channel, this option can be used. Following different modes are supported.

SCD mode	Remark
0	Disable both Tamper and Motion detection.
1	Enable Tamper detection.
2	Enable Motion detection.
3	Enable both Tamper and Motion detection.

In the dvr_rdk release version **02.80.00.10**, motion detection has to be enabled with tamper detection. SCD motion detection only mode i.e. ALG_LINK_SCD_DETECTMODE_MONITOR_BLOCKS, is not supported. Motion detection works fine when it is enabled with tamper detection i.e. SCD mode ALG_LINK_SCD_DETECTMODE_MONITOR_BLOCKS_AND_FRAME.

b. SCD frame sensitivity

Internal threshold for tamper tolerance. Takes values in the range of 0-6 where 0 is lowest and 6 is the maximum sensitivity.

c. SCD ignore lights OFF flag

Set to 1 to ignore sudden, significantly darker avg. scene luminance

d. SCD ignore lights ON flag

Set to 1 to ignore sudden, significantly brighter avg. scene luminance.

e. Live Motion Detection (LMD) Configuration

To activate Live Motion Detection (LMD), CIF stream path should be always activated as LMD works on CIF stream only.

In the SCD/LMD algorithm, motion detection happens on (32x12)/ (32x10) pixel block in PAL/NTSC input respectively.

In the current demo application, SCD/LMD configuration is done at tile level. Each frame is divided in to a set of 3x3 tiles (Max 9-tiles).

Tile 0 | Tile 1 | Tile 2

Block

Tile 8

User can select any of the tiles starting from 0 - 8 (topLeft - bottomRight). Once the tile is selected, user can update block config of all the blocks within that tile.

To enable motion detection in any of the tiles of the frame, user can select option "e" in the capture settings.

Condition checks have been added to take care of left and bottom boundary. Once the block boundaries are decided, user is asked to input SCD tile enable/disable option. If the tile is enabled, user is asked to input sensitivity. Refer SCD reference document for further details on SCD.

Note: Option "p" can be used to navigate to a preview menu.

f. BlindArea/PrivacyMask Configuration

User is required to input chanId and MaskWindowId. Windows Co-ordinates, Color and Size are default. Please refer demo_blindArea.c file for default values

g. Print SCD frame status

Prints the Tamper detect status for all the channels

2.1.3 Encoder Settings

Encode settings are not applicable for VDEC+VDIS based use cases.

Encoder settings of MJPEG channels <included in some use cases> might not be controlled in some cases.

1. Disable channel

It disables encode of a particular channel; hence you would not be able to see the channel's decoded output on the display.

2. Enable channel

It enables a previously disabled encode of a particular channel, hence you would be able to see the channel's decoded output on the display.

3. Change bit-rate

This option allows you to modify the output bit-rate of a particular encode channel.

4. Change frame-rate

This option allows you to modify the output frame-rate (fps) of a particular encode channel.

5. Change I-frame interval (GOP ratio)

This option allows you to modify the Interval between two consecutive intra frames.

Example Values:

   0 - Only first frame to be intra coded

   1 - No inter frames (all intra frames)

   N - One intra frame and N-1 inter frames, where N > 1

6. Force I-frame

Force the current (immediate) frame to be encoded as an Intra coded frame for a particular channel.

7. Change QP I-frame/JPEG Quality Factor

This option handles two cases,

i)   To set the Quantization parameters for the I/IDR frames, for the specific H264 channels

   This option allows you to modify the initial/Maximum/Minimum quantization parameter for I/IDR frames.

ii)   JPEG quality factor can be controlled for the JPEG specific channels also

   This option allows you to modify the quality of the JPEG output. Set this parameter to 2 for lowest quality and 97 for best quality.

8. Change QP P-frame

To set the Quantization parameters for the P frames, for the specific H264 channels

This option allows you to modify the initial/Maximum/Minimum quantization parameter for P frames.

9. Change RateControl Algorithm(CBR/VBR)

This option allows you to change the rate control algorithm from the default VBR to CBR for a particular channel.

0. Take a JPEG Snapshot

By using this option, you would be able to take the MJPEG snapshot of the current (immediate) frame. For Example: If the MJPEG target framerate is set to 0 and file writes are enabled. Then this option will dump a single frame every time pressed for a particular channel

a. Bitrate Switching Interval: VBRDuration

Increasing this value will make VBR wait for longer time before   switching bit-rate state. This parameter is used only when maxBitRate is greater than 1.5 times of targetBitRate

Note: the above option is just for debugging purposes.

b. VBR Sensitivity

It controls the amount by which the bit-rate increases over the average bitrate when transition happens to high complexity state

Note: the above option is just for debugging purposes

i.   To print the current values of Bit-rate, Frame rate and the I-Frame interval for a particular channel.

2.1.4   Decoder Settings

Decoder Settings are not valid in use cases not using Decoder and for the Decode Display Demo.

1. Disable channel

It disables decode of a particular channel hence you would not be able to see the decoded display.

2. Enable channel

It enables a previously disabled decode of a particular channel after which you will be able to see the decoded display of a particular channel

3. Create Channel

It enables you to create a channel dynamically, explained in detail in section <u>2.2.1</u>

4. Delete Channel

It enables you to delete a channel dynamically, explained in detail in section <u>2.2.1</u>

5. Switch File

It enables you to switch to a different file for playback dynamically, explained in detail in section <u>2.2.1</u>

This option enables us to:

a. Switch channel resolution runtime

Dec link and Demo app supports runtime resolution change. For example, A channel created with D1 can be closed and open for HD resolution

b. Dynamically switch codec Alg type

Dec link and Demo app supports runtime codec type change. For example, A channel created with H264 can be closed and open for MPEG codec Type

<u>Note – 3,4,5 options is not valid for all use cases.</u>

6. Enable/Disable Decoder Error Reporting

It allows you to enable/disable error reporting console prints per channel wise.

7. b: Get decoder buffer stats

Prints decoder buffer statistics

Note: Support 32ch H264 HD @ lower FPS

"numBufPerCh" in DecLink_ChCreateParams need to be configured with lower values such as 2 or 3 here

Please use an ini file to start the demo app, or set by App, with following channel configuration
- ✓ First 6ch -> HD
- ✓ Next 6ch -> 720p
- ✓ Remaining 20 -> D1
- ✓ This will help to allocate all different size buffers (HD, 720p, D1 & CIF) on Linux side
- ✓ Set DecLink_ChCreateParams. algCreateStatus = 0 for the unwanted channels

A new parameter "algCreateStatus" has been added in the DecLink_ChCreateParams(), where it can take values either 0 or 1. This will allow the App to **DO NOT** create the codec instance and output buffers in certain channels
- ✓ If algCreateStatus = 0: Only the partial channel gets created, neither codec instance nor the output buffers will be getting created. App will be able to make this channel fully functional dynamically by calling DEC_LINK_CMD_CREATE_CHANNEL API
- ✓ If algCreateStatus =1: Fully functional channel will be created with Both codec instance and the output buffers created, ready to operate

Dynamically create/delete channels supported only in NON-TILED mode as Ducati Tiler memory allocation doesn't support run time memory allocation & free

2.1.5 Display Settings

- Both live & encode channels are sent to display.

- 0~Max Capture channel numbers correspond to Live Channels <e.g. – 0~15 in case of 16 CH progressive demo>

- Max Capture Ch Num~(2*Max Capture Ch Num – 1) correspond to Playback Channels <e.g. – 16~31 in case of 16 CH progressive demo>

- All channels might not be visible. Number of channels displayed is dependent on Layout being selected.

- In some usecases, HDMI outputs are tied – so same display will come in both displays. SDTV might show different channels if selected specifically.

1. Disable channel

It allows the user to hide the display of a particular channel.

2. Enable channel

It enables the user to show back a previously disabled/hidden display of a particular channel.

3. Switch Layout

This option allows the user to change the display layout of display of the channels.

On selecting a particular layout, you will notice that the layout has changed as per the layout selected at all displays Live.

4. Switch Channels

This option allows you to re-map of the channels in the present layout. The user needs to only give the start channel number that should get displayed on the top left window (Window 0) for each display; other channels to the rest of the windows are calculated by incrementing the channel number.

5. Change resolution

This option allows you to change the display output resolution for all the mentioned output displays. Supported resolution changes are 1080P60, 720P60, XGA, SXGA, NTSC or PAL

Live-Bypass based settings

Under display setting the following options are provided. This option is valid only for few use cases & using this option on unsupported use case will have no effect or undefined behavior.

6. Switch Queue(ONLY FOR SD Display)

User can switch the content to be displayed by the display link by selecting the queueID. In the demo,

The display queue 0 corresponds to mosaic view (SwMs output)

The display queue 1 corresponds to live bypass view (Capture output)

By selecting QueueID 0 or QueueID 1 user can switch at runtime between Mosaic and Live views.

7. Switch Channel(ONLY FOR Enc HD Usecase)

User can switch the content to be displayed by the display link by selecting the chanID in the demo. User can switch channel of either HDMI 0 or 1 by selecting respective HDMI.

8. Switch SDTV channel (ONLY for progressive demo)

When in live bypass view, the user can additionally select the capture channel to be displayed and switch between capture channels at runtime.

This option is applicable only for SD display link and only when in "Live bypass view" mode.

9. 2x digital zoom in top left

This option can magnify the top left (from point 0%, 0% to point 50%, 50%) corner of the window for 2 times. It applies on the window 0. Please reference the API to see how to apply on other window.

a. 2x digital zoom in center

This option can magnify the center (from point 25%, 25% to point 75%, 75%) of the window for 2 times. It applies on the window 0. Please reference the API to see how to apply on other window.

b. Avsync Pause

This option enables a pause on all channels being displayed

c. Avsync Timescale Play

This option is for trickplay mode of playing slow or faster than usual speed.

1000 = 1x

d. Avsync Step Fwd

Pause & display frame by frame every-time this option is pressed

e. Avsync Normal Play

Undo a pause, and return to normal playback

f. Avsync Seek Play

Seek forward based on entering time

g. Avsync Reset Player timer

This option can be ignored. It is used to API debugging purposes only.

h. Avsync Set Player State Play

Reset, Step Fwd play or Scan Play to normal

i. Avsync Scan Play

Playback based on playing by only I-Frames

j. SwMS flush frames

This option can be ignored. It is used to API debugging purposes only.

k. Run Grpx fbdev demo

Settings Changed For	Channel Numbers Interpretation	Operation	Channel Numbers		Behavior			
			816x[16CH D1 Capture]	814x[4Ch D1 Capture]	Live Display	Playback Display	D1 File Output <if enabled>	CIF File Output <if enabled>
VCAP	0-15 <816x>, 0-3 <814x> represent both D1 & CIF channels. Other numbers are invalid	Chan Disable	0-15	0-3	Blank Display for ch	Blank Display for ch	No increase in file size	No increase in file size
			>15	>4	No Change	No Change	No Change	No Change
		Chan Enable	0-15	0-3	Ch display gets reenabled	Ch display gets reenabled	File write will restart	File write will restart
			>15	>4	No Change	No Change	No Change	No Change
VENC	0-15 <816x>, 0-3 <814x> represent D1 channels. 16-31 <816x>, 4-7 <814x> represent CIF channels	Chan Disable	0-15	0-3	Blank Display for ch	Blank Display for ch	No increase in file size	No Change
			16-31	4-7	No Change	No Change	No Change	No increase in file size
			>31	>7	No Change	No Change	No Change	No Change
		Chan Enable	0-15	0-3	Ch display gets reenabled	Ch display gets reenabled	File write will restart	No Change
			16-31	4-7	No Change	No Change	No Change	File write will restart
			>31	>7	No Change	No Change	No Change	No Change
VDEC	0-15 <816x>, 0-3 <814x> represent D1 channels. Other numbers are invalid	Chan Disable	0-15	0-3	No Change	Blank Display for ch	No Change	No Change
			>15	>4	No Change	No Change	No Change	No Change
		Chan Enable	0-15	0-3	No Change	Ch display gets reenabled	No Change	No Change
			>15	>4	No Change	No Change	No Change	No Change
VDIS	0-15 <816x>, 0-3 <814x> represent live channels. 16-31 <816x>, 4-7 <814x> represent playback channels	Chan Disable	0-15	0-3	Blank Display for ch	No Change	No Change	No Change
			16-31	4-7	No Change	Blank Display for ch	No Change	No Change
			>31	>8	No Change	No Change	No Change	No Change
		Chan Disable	0-15	0-3	Ch display gets reenabled	No Change	No Change	No Change
			16-31	4-7	No Change	Ch display gets reenabled	No Change	No Change
			>31	>8	No Change	No Change	No Change	No Change

Table: Interpreting Channel numbers for 8168 16D1 & 8148 4D1 progressive demos

2.1.6 Audio Capture <TVP5158> & Encode <AAC-LC> demo

AAC-LC File In -> Encode -> AAC File Out

User need not provide any parameter.

Example Usage:
To enable capture & encode & store aac output of captured channels
- Select Option 6 after starting custom encode demo
- ch*.aac files will be generated in current execution directory

2.1.7 Audio encode demo <AAC-LC File In/Out>

PCM File In -> Encode -> AAC-LC File Out

Parameters to provide – bitrate, sample rate, num of channels

Bit rate depends on encoder's allowed bitrate range for a specific sample rate <Check Codec User Guide>

Example Usage:

To enable file based encode.
- Select Option 7 after starting custom encode demo
- Give input file name. Eg - input_16k_mono.pcm
- Give output file name. E.g – out.aac
- Give Bitrate – E.g – 24000
- Give Num Of channels – E.g – 1
- Give Sample Rate – E.g – 16000
- Encoding will start & message will get printed after encode completes
- Encode can be stopped by selecting same option.

2.1.8 Audio decode demo <AAC-LC File In/Out>

PCM File In -> Decode -> PCM File Out

Parameter – DesiredChannelMode

Example Usage:

To enable file based decode.
- Select Option 8 after starting custom encode demo
- Give input file name. Eg - input_16k_stereo.aac
- Give output file name. E.g – out.pcm
- Give Num Of channels – E.g – 2
- Decode will start & message will get printed after decode completes
- Decode can be stopped by selecting same option.

2.1.9 Change Modes (8CH Usecase)

In 8 CH usecase, there are different modes of captured resolutions like 2D1 + 6 CIF, 8 D1 etc. These modes can be selected using "Change modes" .

2.1.10 Audio Capture / Playback Settings

Audio capture and playback are integrated with Progressive SD Encode + Decode demo. Sub menu available for setting audio storage path, audio capture start/stop, audio play start/stop & statistics. User need to set the valid file storage path before proceeding for Audio capture/playback.

Following audio configuration is available on DM816X EVM Board:

➢ Upto 16 channels audio capture and 1 channel playback is supported in this version.

➢ Audio capture/playback with G711 encoding/decoding.

➢ Audio is captured at 16 KHz and encoded using 8-bits ulaw. Playback happens at same sampling rate using 8-bits ulaw decode. If testing with any PC software at audio out end, configure the PC software for 16 KHz and 8-bit ulaw.

➢ By default, 16 channels capture is configured in the linux kernel as well as the application. **Need to ensure that DVRRDK hardware ver 0.3A or above (Or DM8168 EVM VS Daughter Card Rev C) is used**

Note - Any operation with audio is possible only when video capture is enabled and running. Audio settings will fail if tried before video capture.

Audio settings menu screen shot is shown below –

```
Enter Choice: 5

==================
Audio Settings Menu
==================

1: Set   Audio Storage Path
2: Start Audio Capture
3: Stop  Audio Capture
4: Start Audio Playback
5: Stop  Audio Playback
6: Set Audio Capture Configuration

p: Previous Menu

Enter Choice: ▮
```

Capture, playback operations will fail as shown below until a proper storage path is provided. So, the first operation should be setting audio storage path.

User should set audio recording directory - can point to nfs or hdd. Application will try to create this directory if not existing already. If creation fails for some reason, it prints error message. User should again select storage path setting option and provide a valid path. This is one time operation on app execution. Still user can switch to a different storage path on same session.

```
Enter Choice: 1

Enter file write path : /home

Selected file write path [/home]

Store path set to /home
Trying to set storage path to /home
```

There is 1 playback instance & 1 capture instance. Each can operate on different channels. If user gives a channel which is already being captured or played then the already running capture / playback will be terminated and new request will be started.

Capture can be selected as shown below. Channel number will be the input.

```
Enter Choice: 2

Select AUDIO CAPTURE CH ID [0 .. 3] : 0

Selected CH ID = 0


Removing /home/01/record01.pcm
Removing /home/01/record02.pcm
Removing /home/01/record03.pcm
Removing /home/01/record04.pcm
Removing /home/01/record05.pcm
Removing /home/01/record06.pcm
Removing /home/01/record07.pcm
Removing /home/01/record08.pcm
Removing /home/01/record09.pcm
Removing /home/01/record10.pcm



==================
Audio Settings Menu
==================

1: Set   Audio Storage Path
2: Start Audio Capture
3: Stop  Audio Capture
4: Start Audio Playback
5: Stop  Audio Playback
6: Set Audio Capture Configuration

p: Previous Menu

Enter Choice: Opened /home/01/record01.pcm for recording..
```
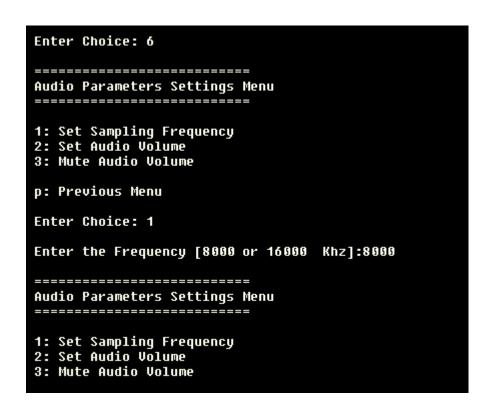
When a channel is selected for audio capture, already existing audio files for that channel will be deleted. This is done to avoid mixing up of a previous recording. Max of 10 files <each of max size around 20MB> is allowed to record. If capture continues beyond 10th file, data will be overwritten starting from first file. Capture & playback can be independently started / stopped.

Capture Stop can be done using respective option. Both capture & playback simultaneously on same channel is not allowed.

Playback of a particular channel can be started / stopped in similar way.

Audio parameters like sampling frequency, audio volume, mute option can be selected as shown below.

Note - Explicit delays are added on selection of menu option <so wait till the operation completes>.

```
Enter Choice: 6

===========================
Audio Parameters Settings Menu
===========================

1: Set Sampling Frequency
2: Set Audio Volume
3: Mute Audio Volume

p: Previous Menu

Enter Choice: 1

Enter the Frequency [8000 or 16000  Khz]:8000

===========================
Audio Parameters Settings Menu
===========================

1: Set Sampling Frequency
2: Set Audio Volume
3: Mute Audio Volume
```

```
--------------------------------
Audio Parameters Settings Menu
==============================

1: Set Sampling Frequency
2: Set Audio Volume
3: Mute Audio Volume

p: Previous Menu

Enter Choice: 2

Enter the Audio Volume[0..8] :6
```

```
Enter Choice: 3

Audio capture is muted. Increase audio capture volume to unmute
============================
Audio Parameters Settings Menu
============================

1: Set Sampling Frequency
2: Set Audio Volume
3: Mute Audio Volume
```

2.2   Decoder Only Application (VDEC + VDIS – SD/HD Decode Only)

In VDEC+VDIS demo input stream files details are provided through *ini* files. User can see sample *ini* files available in *dvr_rdk/bin/ti816x/demo_ini* folder.

Before using, please edit the *ini* files to specify the actual input file name according to your data files.

After entering correct detail in *ini* files user can give the path of *ini* files after starting VDEC+VDIS demo. By default, the *ini* file specifies /data/<filename> as the place for the data files. Follow below instruction while creating your own *ini* file.

- User has to mention stream path, resolution and channel enable flag in the file.

- User can individually enable or disable each channel via *ini* file by setting enable flag.

- User needs to mention the number of decode buffers required for particular channel.

- Desired display delay can be set by the User by setting displaydelay param in the ini file. Decoder shall start displaying of frames not later than displayDelay number of frames are decoded.

displayDelay param value	Its meaning
-1	Decoder decides the display delay (Auto mode)
0	Display frames are in decoded

	order without delay
1	Display the frames with 1 frame delay
2	Display the frames with 2 frame delay
3	Display the frames with 3 frame delay
4	Display the frames with 4 frame delay
5	Display the frames with 5 frame delay
6	Display the frames with 6 frame delay
7	Display the frames with 7 frame delay
8	Display the frames with 8 frame delay
9	Display the frames with 9 frame delay
10	Display the frames with 10 frame delay
11	Display the frames with 11 frame delay
12	Display the frames with 12 frame delay
13	Display the frames with 13 frame delay
14	Display the frames with 14 frame delay
15	Display the frames with 15 frame delay
16	Display the frames with 16 frame delay

Next, user is asked to choose if he/she wants to generate header file. For every new stream, user must select "Y" and generated information header files can be re-used again. If user already has header information then he/she can select "N". For huge files header file generation will take very long time.

Note:

- Multiple references frames is supported but only 1 reference frame(B-frame) has been tested extensively, hence limited Generic Decoder testing has been conducted for H264
- The numbuf should always be greater than the displaydelay and **preferably the difference between them is five or more**
- Keep the numbuf = 16 and displaydelay = 10 in the ini file, for generic streams which have multiple reference buffers
- Limitted error stream testing has been done

2.2.1 Decode ->Display App and Demo options for Delete/Create decode channels

1. **Delete channel:** Select "Decode Settings" from the main menu -> choose "Delete channel" option and enter the channel number

2. **To switch the input file:** Select "Decode Settings" from the main menu -> choose "Switch File" and enter chId & file sequence number (file ID in the .ini file). A few important points to remember here
   a. File switch option works only for the Decode->Display demo
   b. Enter all the files in the .ini file affront and remember the file sequence number (file ID)
   c. Enable generate header option if the HDR files for all these bit streams are not present along with the data files
   d. We need only one .ini file in the beginning and all the bit stream files used while switch file option should be listed here
3. **Create channel:** Select "Decode Settings" from the main menu -> choose "Create channel" option. Enter chId, Codec Type and Resolution
   a. We have limited the number of "channel creates parameters" to 3 for ease of use. Actually ~10 parameters are there in the MCFW level and even more in the Link level
   b. Only 5 resolutions types (HMP, HD, 720P, D & CIF) are supported in the demo level. Actually, any arbitrary resolutions can be supported at MCFW API level
   c. If memory is not sufficient to create a new channel, then the create cmd exit gracefully. You will be able to try again after deleting a few existing channels
4. Please use the ini file to start the demo app, or set by App, with following channel configuration
   a. First 6ch -> HD **(resolution > 720p works only in these first 6 channels)**
   b. Next 6ch -> 720p **(resolution > D1 works only in these first 12 channels)**
   c. Remaining 20 -> D1 **(resolution <= D1 works for all 32 channels)**
   d. This will help to allocate all different size buffers (HD, 720p, D1 & CIF) on Linux side
   e. Set DecLink_ChCreateParams. algCreateStatus = 0 for the unwanted channels or all channels in the beginning as 6HD + 6 720p + 20 D1 will not be in real time.

Those inactive channels are be dynamically get activated by "**Create channel**" API