Texas Instruments

Digital Video Test Bench (DVTB) User's Guide

Contents

1 Introduction 1
1.1 Availability of the DVTB 1
1.2 Purpose of the DVTB 1
1.3 Features 1
2 Using the DVTB 6
2.1 Environment setup 6
2.2 Installing DVTB 6
2.3 Updating DVTB Configuration to support Evaluation Combos 12
2.4 Using DVTB 13
2.4.1 Commands 14
2.4.2 DVTB configuration parameters 20
2.4.3 DVTB scripting 114
2.4.4 DVTB Debug Mode 115
2.4.5 Rebuilding DVTB for different codec servers 116
3 DVTB Use Cases 119
3.1 DM6446(Linux) Usecases 119
3.1.1 Video Usecases 119
3.1.2 Audio Usecases 123
3.1.3 Speech Usecases 124
3.1.4 Simultaneous Operations - Usecases 127
3.2 DM355(Linux) Usecases 129
3.2.1 Video Usecases 129
3.2.2 Speech Usecases 131
3.2.3 Simultaneous Operations - Usecases 132
3.2.4 Image Usecases 133
3.2.5 JPEG Usecases 134
3.2.6 MPEG4 Usecases 136
3.2.7 IPNC Usecases 138
3.2.8 DVR Usecases 139
3.3 DM6467(Linux) Usecases 141
3.3.1 Video Usecases 141
3.3.2 Audio Usecases 143
3.3.3 Speech Usecases 143
3.3.4 Image Usecases 145
3.3.5 Extended parameters AACHE Usecases 146
3.3.6 Extended parameters H264 Video Usecases 147
3.3.7 Extended parameters MPEG2 Video Usecases 152
3.3.8 Extended parameters MPEG4 Video Usecases 152
3.3.9 Extended parameters Image Usecases 155
3.4 DM365(Linux) Use-cases 156
3.4.1 Video Usecases 156
3.4.2 Audio Usecases 158
3.4.3 Speech Usecases 160
3.4.4 Image Usecases 161
3.4.5 Audio Extended parameter Usecases 162
3.4.6 Extended parameters MPEG4 Video Usecases 164
3.4.7 Extended parameters MPEG4 HDVICP Video Usecases 168

3.4.8 Extended parameters Image Usecases 170
3.4.9 Extended parameters H264 Video Usecases 172
3.5 OMAP3530(Linux) Usecases 178
3.5.1 Video Use Cases 178
3.5.2 Video Encode 183
3.5.3 Audio Use Cases 191
3.5.4 Speech Use Cases 193
3.5.5 Image Use Cases 195
3.5.6 Simultaneous Operations - Use Cases 196
3.5.7 Power Manager- Use Cases 199
3.6 DM3730(Linux) Usecases 199
3.6.1 Video Use Cases 199
3.6.2 Video Encode 205
3.6.3 Audio Use Cases 212
3.6.4 Speech Use Cases 214
3.6.5 Image Use Cases 216
3.6.6 Simultaneous Operations - Use Cases 217
3.7 OMAP3530(WinCE) Usecases 220
3.7.1 Video Use Cases 220
3.8 Performance measurement for the use-cases 222

1 Introduction

1.1 Availability of the DVTB

The DVTB is available with the DVSDK package. For more information about the installation of the DVSDK the reader is referred to the DVSDK Getting started guide.

1.2 Purpose of the DVTB

The DVTB is a Linux/WinCE utility that was developed to execute end to end data flows using the DaVinci DVSDK product. The DVTB uses the Codec Engine VISA APIs and Linux/WinCE driver peripheral APIs to encode / decode audio and video streams.

The user can configure the codecs and/or peripherals before starting a data flow. This enables the user to exercise different use case scenarios and evaluate the system by configuring the codecs and/or peripherals as required.

1.3 Features

The package for this product support the following platforms

DM6446(Linux)

DM355(Linux)

DM6467(Linux)

DM365(Linux)

OMAP3530(Linux)

DM3730(Linux)

OMAP3530(WinCE)

Following are the platform-wise high level features supported by DVTB:

DM6446(Linux)

- Encode A/V stream from camera, line-in, etc
- Decode A/V stream and playback on display, line-out, etc
- Encode from a raw uncompressed (YUV/PCM) A/V file
- Decode compressed streams and store as raw uncompressed A/V file
- Supports YUV 4:2:2 Interleaved and YUV 4:2:0 Planar formats
- Configuration of all parameters supported by XDM classes => Configuration of all base parameters supported by codecs

- Configuration of audio and video peripherals
- Encode/Decode stream with any of the codecs available in the codec servers
    - Video Playback of MPEG2 MP, H.264 BP, MPEG4 SP
    - Video Capture of MPEG4 SP, H.264 BP
    - Audio Playback of AAC, MPEG1-L2-L3, MP3
    - Speech Playback of G.711
    - Speech Capture of G.711

DM355(Linux)

- Encode from a raw uncompressed (YUV/PCM) video, image and speech file
- Decode compressed streams and store as raw uncompressed YUV and PCM file
- Supports YUV 4:2:2 Interleaved and YUV 4:2:0 Planar formats
- Supports XDM 1.0 classes for video encoder/decoder and image encoder/decoder
- Supports XDM 1.0 classes for speech encoder/decoder
- Encode/Decode stream with any of the codecs available in the codec servers
    - Video Decode of MPEG4 SP
    - Video Encode of MPEG4 SP
    - Speech Decode of G.711
    - Speech Encode of G.711
    - Image encode – JPEG Encode
    - Image decoder – JPEG Decoder

DM6467(Linux)

- Decode compressed streams and store as raw uncompressed file
- Read raw uncompressed file and store as compressed streams
- Supports encode from YUV 4:2:0P planar format
- Supports decode to YUV 4:2:0P planar format
- Configuration of all parameters supported by XDM classes => Configuration of all base parameters supported by codecs
- Encode/Decode stream with any of the codecs available in the codec servers

- Video Decode of MPEG2 , H.264 , MPEG4, H2641080p60vdec
- Video Encode of H.264, MPEG4, H264fhdvenc
- Audio Decode of AAC HE
- Speech Decode of G.711
- Speech Encode of G.711
- Image encode – JPEG Encode
- Image decoder – JPEG Decoder

DM365(Linux)

- Encode from a raw uncompressed (YUV/PCM) video, image and speech files
- Decode compressed streams and store as raw uncompressed YUV and PCM file
- Supports YUV 4:2:2 Interleaved and YUV 4:2:0 Semi-Planar formats
- Supports XDM 1.0 classes for video encoder/decoder and image encoder/decoder
- Supports XDM 1.0 classes for speech encoder/decoder
- Encode/Decode stream with any of the codecs available in the codec servers
    - Video Decode of MPEG4, MPEG2, H264 and VC1
    - Video Encode of MPEG4, MPEG2 and H264
    - Image encode – JPEG Encode
    - Image decoder – JPEG Decoder
    - Speech Decode of G.711
    - Speech Encode of G.711
    - Audio Decoder of AAC
    - Audio Encoder of AAC

OMAP3530(Linux)

- Decode A/V and Image compressed stream and store as raw uncompressed file
- Decode A/V and Image compressed stream and playback on display (LCD), line-out.
- Encode from a raw uncompressed (YUV) video file
- Supports video and Image decode to YUV 4:2:2 interleaved format
- Supports video decode to YUV 4:2:0 planar format
- Supports video and Image display of YUV 4:2:2 interleaved format

- Supports encode from YUV 4:2:0 planar format
- Supports encode from YUV 4:2:2 interleaved format
- Configuration of all parameters supported by XDM classes => Configuration of all base parameters supported by codecs
- Configuration of audio and video peripherals
- Encode/Decode streams with any of the codecs available in codec servers
    - Video decode and Playback of MPEG2 MP, MPEG4 ASP and H264BP
    - Video encode of MPEG4 SP and H264 BP
    - Audio decode and playback of AAC-HE and AAC-LC
    - Image decode and display of JPEG
    - Image encode and dump of JPEG
    - Speech decode and Playback of G.711
    - Speech capture and encode of G.711
    - Video encode, decode and dump of H264 BP

DM3730(Linux)

- Decode A/V and Image compressed stream and store as raw uncompressed file
- Decode A/V and Image compressed stream and playback on display (LCD), line-out.
- Encode from a raw uncompressed (YUV) video file
- Supports video and Image decode to YUV 4:2:2 interleaved format
- Supports video decode to YUV 4:2:0 planar format
- Supports video and Image display of YUV 4:2:2 interleaved format
- Supports encode from YUV 4:2:0 planar format
- Supports encode from YUV 4:2:2 interleaved format
- Configuration of all parameters supported by XDM classes => Configuration of all base parameters supported by codecs
- Configuration of audio and video peripherals
- Encode/Decode streams with any of the codecs available in codec servers
    - Video decode and Playback of MPEG2 MP, MPEG4 ASP and H264BP
    - Video encode of MPEG4 SP and H264 BP
    - Audio decode and playback of AAC-HE and AAC-LC

- Image decode and display of JPEG
- Image encode and dump of JPEG
- Speech decode and Playback of G.711
- Speech capture and encode of G.711
- Video encode, decode and dump of H264 BP

OMAP3530(WinCE)

- Decode Video compressed stream and store as raw uncompressed file
- Encode from a raw uncompressed (YUV) video file
- Supports video decode to YUV 4:2:2 interleaved format
- Supports video decode to YUV 4:2:0 planar format
- Supports encode from YUV 4:2:0 planar format
- Supports encode from YUV 4:2:2 interleaved format
- Configuration of all parameters supported by XDM classes => Configuration of all base parameters supported by codecs
- Encode/Decode streams with any of the codecs available in codec servers
    - Video decode and dump of MPEG2 MP, MPEG4 ASP and H264BP
    - Video encode of MPEG4 SP and H264 BP

2 Using the DVTB

The following sections provide the directions to use the dvtb.

2.1 Environment setup

Before using the DVTB, the DVSDK environment must be set up.

For more information about setting up the environment the reader is referred to the DVEVM Hardware Setup and the DVEVM Software Setup chapters in the *DVEVM Getting Started Guide*.

2.2 Installing DVTB

The DVTB is installed in a $(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx folder during the installation of the dvsdk_setuplinux_#_##_##_##.bin file.

To install DVTB to the target file system, do the following on the host machine where DVSDK is installed:

DM6446(Linux)

- `$ cd $(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx`
- `$ make clean`
- `$ make dm6446`

  Release (dvtb-r) and Debug (dvtb-d) binaries will be created under the following folder

  `$(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm6446/bin`
- Copy the binaries dvtb-r & dvtb-d to your DVSDK execution environment.

DM355(Linux)

- `$ cd $(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx`
- `$ make clean`
- `$ make dm355`

  Release (dvtb-r) and Debug (dvtb-d) binaries will be created under the following folder

  `$(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm355/bin`
- Copy the binaries dvtb-r & dvtb-d to your DVSDK execution environment.

DM6467(Linux)

- `$ cd $(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx`
- To build for TSPA codec server with h264 decoder (Ensure that the CODEC_INSTALL_DIR is pointing to TSPA codec server)
- `$ make clean ; make dm6467 CODECS=TSPA  H264CodecPackage=h264dec`
- To build for TSPA codec server with h264 1080p decoder (Ensure that the CODEC_INSTALL_DIR is pointing to TSPA codec server)

- $ make clean ; make dm6467 CODECS=TSPA H264CodecPackage=h2641080p60vdec
- To build for NON TSPA codec server with h264 decoder (Ensure that the CODEC_INSTALL_DIR is pointing to NON TSPA codec server)
- $ make clean ; make dm6467 CODECS=NONTSPA H264CodecPackage=h264dec
- To build for NON TSPA codec server with h264 1080p decoder (Ensure that the CODEC_INSTALL_DIR is pointing to NON TSPA codec server)
- $ make clean ; make dm6467 CODECS=NONTSPA H264CodecPackage=h2641080p60vdec

  Release (dvtb-r) and Debug (dvtb-d) binaries will be created under the following folder

  $(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm6467/bin
- Copy the binaries dvtb-r & dvtb-d to your DVSDK execution environment.

DM365(Linux)

- $ cd $(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx
- To build for TSPA codec server (Ensure that the CODEC_INSTALL_DIR is pointing to TSPA codec server)
- $ make clean ; make dm365 CODECS=TSPA
- To build for NON TSPA codec server (Ensure that the CODEC_INSTALL_DIR is pointing to NON TSPA codec server)
- $ make clean ; make dm365 CODECS=NONTSPA

  Release (dvtb-r) and Debug (dvtb-d) binaries will be created under the following folder

  $(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm365/bin
- Copy the binaries dvtb-r & dvtb-d to your DVSDK execution environment.

OMAP3530(Linux)

- $ cd $(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx
- $ make clean
- $ make omap3530

  Release (dvtb-r) and Debug (dvtb-d) binaries will be created under the following folder

  $(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx/packages/ti/sdo/dvtb/omap3530/linux/bin
- Copy the binaries dvtb-r & dvtb-d to your DVSDK execution environment.

DM3730(Linux)

- $ cd $(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx
- $ make clean
- $ make dm3730

  Release (dvtb-r) and Debug (dvtb-d) binaries will be created under the following folder

  $(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm3730/linux/bin
- Copy the binaries dvtb-r & dvtb-d to your DVSDK execution environment.

OMAP3530(WinCE)

- **Create DVTB subproject**
  - Open up your OS design in Visual Studio
  - Create a new directory wince_project/wince600/dvtb at {DVSDK_INSTALL_DIR}/dvtb_x_xx_xx/packages/ti/sdo/dvtb/omap3530/ wince, where {DVSDK_INSTALL_DIR} is the DVSDK installation directory.
  - Select Project-> Add New Subproject from the menu. Select "WCE Console Application" under "Available templates". Set "Subproject name" as "dvtb" and "Location" as {DVSDK_INSTALL_DIR}/dvtb_x_xx_xx/packages/ti/sdo/dvtb/omap3530/ wince_project/wince600/dvtb
  - In Next Window select "An empty subproject".
  - Go to folder {DVSDK_INSTALL_DIR}/dvtb_x_xx_xx/packages/ti/sdo/dvtb/omap3530/ wince_project/wince600/dvtb
  - Delete postlink.bat
  - Append prelink.bat with following

```
@echo off
rem ============================================================================
rem @file   prelink.bat
rem
rem @path   $(DVTB_INSTALL_DIR)/wince_project/wince600/dvtb
rem
rem @desc   Sets the DVTB environment and builds DVTB lib. This
rem         file is used by WinCE platform builder to build the DVTB subproject.
rem
rem @ver
rem ============================================================================
rem Copyright (C) 2010 Texas Instruments Incorporated - http://www.ti.com/
rem ============================================================================

@REM Add pre-link commands below.
@set VAR_NAMES_SET=_WINCEROOT _winceroot _PROJECTROOT _projectroot _TGTCPU _tgtcpu
@set  VAR_NAMES_SET=%VAR_NAMES_SET%  _TGTCPUFAMILY  _tgtcpufamily  _TARGETPLATROOT
_targetplatroot
@set  VAR_NAMES_SET=%VAR_NAMES_SET%  _FLATRELEASEDIR  _flatreleasedir  WINCEDEBUG
wincedebug
@set VAR_NAMES_SET=%VAR_NAMES_SET% PATH path BIOS_INSTALL_DIR bios_install_dir
@call :Loop4 %VAR_NAMES_SET%
@goto start_build

:Loop4
```

```
@if [%1]==[] goto:EOF
@call set my_temp=%%%2%%
@set %2=
@set %1=%my_temp%
@shift
@shift
@goto Loop4

:start_build
@set VAR_NAMES_SET=
@set my_temp=

echo ============================================================
echo Starting DMAI DVTB builds
echo ============================================================
echo Current dir
cd %CD%\..\..\..\..\..\..\..\..\..
gmake clean
gmake omap3530_w
gmake omap3530_w_install
echo ============================================================
echo DVTB lib done
echo ============================================================

cd %_WINCEROOT%
```

- Remove the contains of sources and add following

```
_COMMONPUBROOT=$(_PROJECTROOT)\cesysgen
__PROJROOT=$(_PROJECTROOT)
RELEASETYPE=LOCAL
_ISVINCPATH=$(_WINCEROOT)\public\common\sdk\inc;
_OEMINCPATH=$(_WINCEROOT)\public\common\oak\inc;$(_WINCEROOT)\public\common\sdk\in
c;
TARGETNAME=dvtb
FILE_VIEW_ROOT_FOLDER= \
    ReadMe.txt \
    prelink.bat \

FILE_VIEW_RESOURCE_FOLDER= \

FILE_VIEW_INCLUDES_FOLDER= \
```

```
SOURCES= \

EXEENTRY=mainWCRTStartup
TARGETLIBS= \
    $(_PROJECTROOT)\cesysgen\sdk\lib\$(_CPUINDPATH)\coredll.lib \

TARGETTYPE=NOTARGET
PRELINK_PASS_CMD=prelink.bat
FILE_VIEW_PARAMETER_FOLDER= \
   ProjSysgen.bat \
```

- Close your OS Design

- **Build DVTB**

  There are two methods for building DVTB. Either as a Wince platform builder subproject or from WinCE platform builder command line.

  1) Build DVTB Using Platform builder subproject

  - In Windows, add the installation directory of the XDCTOOLS to your PATH environment variable if not already done.
  - Include following in {DVSDK_INSTALL_DIR}/dvtb_x_xx_xx/Makefile

    -include Rules.make

    If it is already included in Makefile then this step is not required.
  - Open up your OS design in Visual Studio
  - Select Project->Add existing subproject from the menu. Open {DVSDK_INSTALL_DIR}/dvtb_x_xx_xx/packages/ti/sdo/dvtb/omap3530/ wince/wince_project/wince600/dvtb/dvtb.pbpxml, where {DVSDK_INSTALL_DIR} is the DVSDK installation directory.
  - Select Project->EVM_3530 Properties. Under Configuration Properties->Environment, set the following environment variables (if not set already):

    i) XDC_INSTALL_DIR: The installation directory of the XDCTOOLS from TI

    ii) XDCPATH: The xdcpath used by the configuro command invocation in the makefile. This path needs to point to the xdc packages of TI software(Codec Engine, Codec Engine cetools, DSPLINK, the codec server, xdctools) and cesysgen include files. Eg. XDCPATH= C:/ti_tools/xdctools_3_10_05_61/packages;D:/DVTB_ON_WINCE/dvsdk/DVSDK_WINCE_01_00_00_04/codec_engine_2_24/packages;D:/DVTB_ON_WINCE/dvsdk/DVSDK_WINCE_01_00_00_04/codec_engine_2_24/cetools/packages;D:/DVTB_ON_WINCE/dvsdk/DVSDK_WINCE_01_00_00_04/dsplink_sla_1_61_04;D:/DVTB_ON_WINCE/dvsdk/DVSDK_WINCE_01_00_00_04/cs1omap3530_1_00_00/packages;C:\WINCE600\OSDesigns\EVM_3530\EVM_3530\Wince600\TI_EVM_3530_ARMV4I\cesysgen\sdk\inc;
  - Select Build->Advanced Build Commands->Build Current BSP and Subprojects. This will rebuild DVTB. Note that any errors when building the DVTB will not appear in the Build window. They will however be

logged in the build.log file under the {DVSDK_INSTALL_DIR}/dvtb_x_xx_xx/packages/ti/sdo/dvtb/omap3530/wince/wince_project/wince600/dvtb/ directory. This is because the build is done using a batch file in the background

- Binaries are generated at

  {DVSDK_INSTALL_DIR}/dvtb_X_XX_XX/packages/ti/sdo/dvtb/omap3530/wince/bin and copied to $(_FLATRELEASEDIR) as well.

- Any future rebuild of DVTB can be done either using the GUI by right-clicking on the dvtb subproject in the Solution Explorer or selecting "Build".

- load OS image onto the EVM. dvtb.exe will be loaded with OS image. Alternatively DVTB executables (dvtb.exe & dvtb-d.exe) can be copied to target.

- From WinCE command window dvtb app can be run.

2) Build DVTB Using Platform builder command Line

- In Windows, add the installation directory of the XDCTOOLS to your PATH environment variable if not already done.

- Include following in {DVSDK_INSTALL_DIR}/dvtb_x_xx_xx/Makefile

  -include Rules.make

  If it is already included in Makefile then this step is not required.

- Open up your OS design in Visual Studio

- Select Build->Rebuild EVM_3530. If BSP is already build then this is not required.

- Select Project->Add existing subproject from the menu. Open {DVSDK_INSTALL_DIR}/dvtb_x_xx_xx/packages/ti/sdo/dvtb/omap3530/wince/wince_project/wince600/dvtb/dvtb.pbpxml, where {DVSDK_INSTALL_DIR} is the DVSDK installation directory.

- Select Project->EVM_3530 Properties. Under Configuration Properties->Environment, set the following environment variables (if not set already):

  iii) XDC_INSTALL_DIR: The installation directory of the XDCTOOLS from TI

  iv) XDCPATH: The xdcpath used by the configuro command invocation in the makefile. This path needs to point to the xdc packages of TI software (Codec Engine, Codec Engine cetools, DSPLINK, the codec server, xdctools) and cesysgen include files. E.g. XDCPATH= C:/ti_tools/xdctools_3_10_05_61/packages;D:/DVTB_ON_WINCE/dvsdk/DVSDK_WINCE_01_00_00_04/codec_engine_2_24/packages;D:/DVTB_ON_WINCE/dvsdk/DVSDK_WINCE_01_00_00_04/codec_engine_2_24/cetools/packages;D:/DVTB_ON_WINCE/dvsdk/DVSDK_WINCE_01_00_00_04/dsplink_sla_1_61_04;D:/DVTB_ON_WINCE/dvsdk/DVSDK_WINCE_01_00_00_04/cs1omap3530_1_00_00/packages;C:\WINCE600\OSDesigns\EVM_3530\EVM_3530\Wince600\TI_EVM_3530_ARMV4I\cesysgen\sdk\inc;

- Select "Build->Open Release Directory in Build window" and open up a build window.

- To build DVTB, change directory into the dvtb installation directory {DVSDK_INSTALL_DIR}/dvtb_x_xx_xx/. Then run the following commands in the build window

  - gmake clean

- gmake omap3530_w

- Binaries are generated at {DVSDK_INSTALL_DIR}/dvtb_X_XX_XX/packages/ti/sdo/dvtb/omap3530/wince/bin.
- Run following command to copy dvtb binaries at $(_FLATRELEASEDIR)
  - gmake omap3530_w_install
- Copy the executables (dvtb.exe & dvtb-d.exe) to your target execution environment.
- From WinCE command window dvtb app can be run.

2.3 Updating DVTB Configuration to support Evaluation Combos

The current DVTB supports production combos and does not support evaluation combos. However use can modify the DVTB configuration file to support the evaluation combos.

Following example explains how to modify DVTB configuration file to support decode evaluation combo:

Do the following on the host machine:

DM6446(Linux)

```
$(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm6446/

$ vi dvtb_dm6446.cfg

Add following in the configuration file

“
var decode_e = Engine.createFromServer(
           "decode_e",
           "./decodeCombo_e.x64P",
           "ti.sdo.servers.decode"
           );
“
Close the configuration file.
On the target, engine name should be set to decode_e(setp engine
name decode_e)before executing func command at DVTB prompt.
```

DM6467(Linux)

```
$(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm6467/

$ vi dvtb_dm6467.cfg

Add following in the configuration file

“
var decode_e = Engine.createFromServer(
           "decode_e",
```

```
                ",./decodeCombo_e.x64P",
                "ti.sdo.servers.decode"
                );
```

"

```
Close the configuration file.
On the target, engine name should be set to decode_e(setp engine
name decode_e)before executing func command at DVTB prompt.
```

Note: As explained in above example user can also modify the DVTB configuration file to support other evaluation combos (encode & loopback).

2.4 Using DVTB

Please read the section "Using the Digital Video Test Bench (DVTB)" in the *DVSDK Getting Started Guide* to get a quick insight into using DVTB.

Linux

In order to launch DVTB the following command must be executed:

<target prompt># ./dvtb-r

Ex:

```
root@158.218.33.135:/opt/dvsdk# ./dvtb-r
```

The result is a command prompt for the user to type in commands:

```
------------------------------------------------------------------

==================**************************=======================

                 Digital Video Test Bench (DVTB)

==================**************************=======================

------------------------------------------------------------------

<DVTB> $
```

Note: To exit DVTB operating in command line mode, type 'exit' or 'z' on the command line.

WinCE

In order to launch DVTB the following command must be executed:

<target prompt># dvtb.exe

Ex:

```
\Temp> dvtb.exe
```

The result is a command prompt for the user to type in commands:

```
------------------------------------------------------------

==========**************************==========================

                Digital Video Test Bench (DVTB)

==========**************************==========================

------------------------------------------------------------

<DVTB> $
```

Note:

1. To exit DVTB operating in command line mode, type 'exit' or 'z' on the command line.

2. DVTB expects codec server to be available at root of file system represented by a backslash (\).

3. WinCe does not support the current directory logic.So, User has to provide the full path of all the files. If there are spaces in the path then provide them in inverted comma like

```
    <DVTB> $ func viddec2 –s "\Storage Card\test.264" –t "\Storage
Card\test.yuv"
```

2.4.1 Commands

All commands in DVTB have the following syntax:

```
<Command> <Class> [Options]
```

The following commands are supported:

- func: Functionality - Execute data flows
- setp: Set parameter
- getp: Get parameter

The entity on which the commands operate is called <Class>. The classes can be grouped in peripheral classes and codec classes.

The peripheral classes include:

- audio: Audio device specific parameters.
- vpbe: Video Processing Back End (VPBE) specific parameters
- vpfe: Video Processing Front End (VPFE) specific parameters

The codec classes include:

DM6446(Linux)

- viddec2: XDM 1.2 Video Decoder specific parameters
- videnc1: XDM 1.0 Video Encoder specific parameters
- imgenc1: XDM 1.0 Image Encoder specific parameters
- imgdec1: XDM 1.0 Image Decoder specific parameters
- auddec1: XDM 1.0 Audio Decoder specific parameters
- audenc1: XDM 1.0 Audio Encoder specific parameters
- sphdec1: XDM 1.0 Speech Decoder specific parameters
- sphenc1: XDM 1.0 Speech Encoder specific parameters
- mp3basedec1: XDM 1.0 MP3 Audio Decoder specific parameters
- aacheenc1: XDM 1.0 AACHE Encoder Extended Parameters
- aaclcenc1: XDM 1.0 AACLC Encoder Extended Parameters
- vidloopback1: XDM 1.0 Video loopback

Each <class> has a set of specific parameters. These parameters must be provided in the <Option> field. The list of list of parameters is provided in the next section.

Note: vidloopback1 class does not have any specific parameters. It uses parameters of class videnc1 and viddec2.

DM355(Linux)

- engine: Codec Engine parameters
- viddec2: XDM 1.0 Video Decoder specific parameters
- videnc1: XDM 1.0 Video Encoder specific parameters
- sphdec1: XDM 1.0 Speech Decoder specific parameters
- sphenc1: XDM 1.0 Speech Encoder specific parameters
- imgenc1: XDM 1.0 Image Encoder specific parameters
- imgdec1: XDM 1.0 Image Decoder specific parameters
- jpegenc1: XDM 1.0 Image Encoder – Extended Parameters

- jpegdec1: XDM 1.0 Image Decoder – Extended Parameters
- mpeg4spenc1 : XDM 1.0 Video Encoder – Extended Parameters
- mpeg4spdec2 : XDM 1.0 Video Decoder - Extended Parameters
- IPNCUC0: IPNC Use-case 0
    - `MPEG4 encode (720P 30 fps 6 Mbps + SIF 30 fps 512 Kbps)`
- IPNCUC1: IPNC Use-case 1
    - `MPEG4, D1, 30fps, 2Mbps + MPEG4, SIF, 30fps, 0.5Mbps + JPEG, D1, 15 fps, QP=75`
- IPNCUC2: IPNC Use-case 2
    - `MPEG4, 720p, 30fps, 5-6Mbps + MPEG4, SIF, 30fps, 0.5Mbps + JPEG, D1, 15 fps, QP=75`
- DVRUC0: DVR Use-case 0
    - `2channels of D1 MPEG4, 30fps, 2Mbps each channel`
- DVREnc: DVR Use-case 1
    - `8 channels of CIF, MPEG4, 30fps, bitrate appropriately scaled down, using 2Mbps at D1 as reference.`
- DVREncDec: DVR Use-case 2
    - `4 channels of SIF (enc + dec)`

Each <class> has a set of specific parameters. These parameters must be provided in the <Option> field. The list of list of parameters is provided in the next section.

DM6467(Linux)

- engine: Codec Engine parameters
- viddec2: XDM1.2 Video Decoder specific parameters
- videnc1: XDM1.0 Video Encoder specific parameters
- audenc1: XDM1.0 Audio Encoder specific parameters
- auddec1: XDM1.0 Audio Decoder specific parameters
- sphdec1: XDM1.0 Speech Decoder specific parameters
- sphenc1: XDM1.0 Speech Encoder specific parameters
- imgenc1: XDM 1.0 Image Encoder specific parameters

- imgdec1: XDM 1.0 Image Decoder specific parameters
- jpegenc1: XDM 1.0 Image Encoder – Extended Parameters
- jpegdec1: XDM 1.0 Image Decoder – Extended Parameters
- h264enc1: XDM1.0 Video Encoder – Extended parameters
- h264fhdenc1: XDM1.0 Video Encoder – Extended parameters
- mpeg4enc1: XDM1.0 Video Encoder – Extended parameters
- mpeg4dec2 : XDM1.2 Video Decoder – Extended parameters
- mpeg2dec2 : XDM1.2 Video Decoder – Extended parameters
- h264dec2 : XDM1.2 Video Decoder – Extended parameters
- h2641080pdec2: XDM1.2 Video Decoder – Extended parameters
- aachedec1: XDM1.0 Audio Decoder – Extended parameters
- ac3dec1: XDM1.0 Audio Decoder – Extended parameters
- aacheenc1: XDM1.0 Audio Decoder – Extended parameters
- vidloopback1: XDM 1.0 Video loopback

Each <class> has a set of specific parameters. These parameters must be provided in the <Option> field. The list of parameters is provided in the next section.

Note: vidloopback1 class does not have any specific parameters. It uses parameters of class videnc1 and viddec2.

DM365(Linux)

- engine: Codec Engine parameters
- viddec2: XDM 1.0 Video Decoder specific parameters
- videnc1: XDM 1.0 Video Encoder specific parameters
- sphdec1: XDM 1.0 Speech Decoder specific parameters
- sphenc1: XDM 1.0 Speech Encoder specific parameters
- imgenc1: XDM 1.0 Image Encoder specific parameters
- imgdec1: XDM 1.0 Image Decoder specific parameters
- audenc1: XDM 1.0 Audio Encoder specific parameters

- auddec1: XDM 1.0 Audio Decoder specific parameters
- jpegenc1: XDM 1.0 Image Encoder – Extended Parameters
- jpegdec1: XDM 1.0 Image Decoder – Extended Parameters
- mpeg4enc1 : XDM 1.0 Video Encoder – Extended Parameters
- mpeg4dec2 : XDM 1.0 Video Decoder - Extended Parameters
- mpeg4hdvicpenc1 : XDM 1.0 Video Encoder – Extended Parameters
- mpeg4hdvicpdec2 : XDM 1.0 Video Decoder - Extended Parameters
- h264enc1 : XDM 1.0 Video Encoder – Extended Parameters
- h264dec2 : XDM 1.0 Video Decoder - Extended Parameters
- mpeg2enc1 : XDM 1.0 Video Encoder – Extended Parameters
- mpeg2dec2 : XDM 1.0 Video Decoder - Extended Parameters
- vc1dec2 : XDM 1.0 Video Decoder - Extended Parameters
- aaclcenc1: XDM 1.0 Audio Encoder specific parameters
- aachedec1: XDM 1.0 Audio Decoder specific parameters

Each <class> has a set of specific parameters. These parameters must be provided in the <Option> field. The list of list of parameters is provided in the next section.

OMAP3530(Linux)

- engine: Codec Engine parameters
- viddec2:XDM 1.2 Video Decoder specific parameters
- auddec1: XDM 1.0 Audio Decoder specific parameters
- mp3basedec1: XDM 1.0 Audio Decoder specific parameters
- imgdec1: XDM 1.0 Image Decoder specific parameters
- jpegdec1: XDM 1.0 Jpeg Decoder specific extended parameters
- imgenc1: XDM 1.0 Image Encoder specific parameters
- videnc1: XDM 1.0 Video Encoder Specific parameters
- sphdec1: XDM1.0 Speech Decoder specific parameters
- sphenc1: XDM1.0 Speech Encoder specific parameters
- h264enc1: XDM 1.0 H264 Encoder Specific Extended parameters
- mpeg4spenc1: XDM 1.0 Mpeg4 Encoder Specific Extended parameters

- mpeg2dec2: XDM 1.2 Mpeg2 Decoder specific Extended parameters
- mpeg4spdec2: XDM 1.2 Mpeg4 Decoder specific Extended parameters
- h264dec2: XDM 1.2 H264 Decoder specific Extended parameters
- aachedec1: XDM 1.0 AAC HE Decoder specific Extended parameters
- pwrmanager : Power Manager specific parameters
- vidloopback1: XDM 1.0 Video loopback

Each <class> has a set of specific parameters. These parameters must be provided in the <Option> field. The list of parameters is provided in the next section.

Note: vidloopback1 class does not have any specific parameters. It uses parameters of class videnc1 and viddec2.

DM3730(Linux)

- engine: Codec Engine parameters
- viddec2:XDM 1.2 Video Decoder specific parameters
- auddec1: XDM 1.0 Audio Decoder specific parameters
- mp3basedec1: XDM 1.0 Audio Decoder specific parameters
- imgdec1: XDM 1.0 Image Decoder specific parameters
- jpegdec1: XDM 1.0 Jpeg Decoder specific extended parameters
- imgenc1: XDM 1.0 Image Encoder specific parameters
- videnc1: XDM 1.0 Video Encoder Specific parameters
- sphdec1: XDM1.0 Speech Decoder specific parameters
- sphenc1: XDM1.0 Speech Encoder specific parameters
- h264enc1: XDM 1.0 H264 Encoder Specific Extended parameters
- mpeg4spenc1: XDM 1.0 Mpeg4 Encoder Specific Extended parameters
- mpeg2dec2: XDM 1.2 Mpeg2 Decoder specific Extended parameters
- mpeg4spdec2: XDM 1.2 Mpeg4 Decoder specific Extended parameters
- h264dec2: XDM 1.2 H264 Decoder specific Extended parameters
- aachedec1: XDM 1.0 AAC HE Decoder specific Extended parameters
- pwrmanager : Power Manager specific parameters

- vidloopback1: XDM 1.0 Video loopback

Each <class> has a set of specific parameters. These parameters must be provided in the <Option> field. The list of parameters is provided in the next section.

Note: vidloopback1 class does not have any specific parameters. It uses parameters of class videnc1 and viddec2.

OMAP3530(WinCE)

- engine: Codec Engine parameters
- viddec2: XDM 1.2 Video Decoder specific parameters
- videnc1: XDM 1.0 Video Encoder Specific parameters

Each <class> has a set of specific parameters. These parameters must be provided in the <Option> field. The list of parameters is provided in the next section.

2.4.2 DVTB configuration parameters

Each DVTB <class> has a specific set of paramters. The DVTB is initialized with default values for each parameter.

In order to display the parameters of a <class> the following command should be used:

```
getp <class>
```

For example, to get the values for video decoder class that will be used by the next video decode data flow, do the following on the DVTB command prompt:

```
<DVTB> $ getp viddec2
```

The following table presents the configuration parameters for all peripheral classes and codec classes supported by DVTB.

2.4.2.1 Peripheral Classes configuration parameters

The following table presents the configuration parameters for the peripheral classes. The description of each parameter and its corresponding values are listed in below tables. The default value for each parameter is indicated below in this ***font***

In case the parameter can be configured for any user input, the set of commonly used values for the parameter is given in the table. In addition, the last value for such parameters is given as … indicating that other values can be provided by the user.

2.4.2.1.1 DM6446(Linux)

Table 1. Peripheral Classes Configuration Parameters

Class	Parameter	Description	Values
Audio	Device	Name of the audio device	***plughw:0,0***
	Format	PCM Format *For more information look in snd_pcm_format_t*	***2***
	samplerate	Sample rate of audio in Hz	***8000*** 44100 48000 …
	Channels	Number of audio channels *Only 2 channels are supported by audio driver*	1 ***2***
	Type	Blocking Mode *Supported value of type is only 0*	***0*** : Blocking 1 : Non Blocking
Vpbe	Device	Name of the VPBE device	***/dev/video2***
	Width	Width of Image	***720*** 352 640 …
	Height	Height of Image	***480*** 576 288 …
	std	Video standard	***1***(NTSC) 2(1080i_30) 3(720p_60)

			4(PAL) 5(720p_50) 6(1080i_25) 7(480p_60) 8(576p_50)
	output	Video Output	***1***(Composite) 3(S-Video)
	bpp	Bite per pixel *Supported value of bpp is only 16*	***16***
Vpfe	device	Name of the VPFE device	***/dev/video0***
	standard	Standard of video capture device connected to EVM	***0*** : Auto detect 1 : NTSC 2 : PAL 3 : SECAM
	format	Format of video data to be captured *Only YUYV and UYVY supported by the VPFE driver currently*	0 : YUYV ***1*** : UYVY 2 : YUV420 3 : YUV422P 4 : YUV410 6 : RGB565
	input	Type of physical video input connected to the EVM	***0*** : Composite 1 : S-Video
	width	Width of capture window	***720*** 352 640 …
	height	Height of capture window	***480*** 576

			288 …

2.4.2.2 DM355(Linux)

Table 2. Peripheral Classes Configuration Parameters

Class	Parameter	Description	Values
Audio	device	Name of the audio device	***plughw:0,0***
	format	PCM Format *For more information look in snd_pcm_format_t*	***2***
	samplerate	Sample rate of audio in Hz	***8000*** 44100 48000 …
	channels	Number of audio channels *Only 2 channels are supported by audio driver*	1 ***2***
	type	Blocking Mode *Supported value of type is only 0*	***0*** : Blocking 1 : Non Blocking
vpbe	device	Name of the VPBE device	***/dev/video2***
	width	Width of Image	***720*** 352 640 …
	height	Height of Image	***480*** 576 288

			…
	std	Video standard	***1***(NTSC) 2(1080i_30) 3(720p_60) 4(PAL) 5(720p_50) 6(1080i_25) 7(480p_60) 8(576p_50)
	output	Video Output	***1***(Composite)
Vpfe	device	Name of the VPFE device	***/dev/video0***
	standard	Standard of video capture device connected to EVM	***0*** : Auto detect 1 : NTSC 2 : PAL 3 : SECAM
	format	Format of video data to be captured *Only YUYV and UYVY supported by the VPFE driver currently*	0 : YUYV ***1*** : UYVY 2 : YUV420 3 : YUV422P 4 : YUV410 6 : RGB565
	input	Type of physical video input connected to the EVM	***0*** : Composite 1 : S-Video
	width	Width of capture window	***720*** 352 640 …
	height	Height of capture window	***480***

			576 288 …

2.4.2.3 DM6467(Linux)

Table 3. Peripheral Classes Configuration Parameters

Class	Parameter	Description	Values
Audio	Device	Name of the audio device	***plughw:0,0***
	Format	PCM Format *For more information look in snd_pcm_format_t*	***2***
	samplerate	Sample rate of audio in Hz	***8000*** 44100 48000 …
	Channels	Number of audio channels *Only 2 channels are supported by audio driver*	1 ***2***
	Type	Blocking Mode *Supported value of type is only 0*	***0*** : Blocking 1 : Non Blocking
vpbe	device	Name of the VPBE device	***/dev/video2***
	width	Width of display window	***1920***
	height	Height of display window *Restricted by whether NTSC/PAL display is connected*	***1080***
	std	Video standard	1(NTSC) ***2***(1080i_30) 3(720p_60)

			4(PAL) 5(720p_50) 6(1080i_25) 7(480p_60) 8(576p_50)
	output	Video Output	1(Composite) ***2***(Component) 3(S-Video)
vpfe	device	Name of the VPFE device	***/dev/video0***
	standard	Standard of video capture device connected to EVM *Only NA is supported*	0 : Auto detect 1 : NTSC 2 : PAL 3 : SECAM ***4*** : NA
	format	Format of video data to be captured *Only YUV422UVP is supported*	0 : YUYV 1 : UYVY 2 : YUV420 3 : YUV422P 4 : YUV410 6 : RGB565 ***7*** : YUV422UVP
	input	Type of physical video input connected to the EVM	0 : Composite ***1*** : component
	width	Width of capture window	1280 ***720***
	height	Height of capture window	720 ***480*** 576

2.4.2.4 DM365(Linux)

Table 4. Peripheral Classes Configuration Parameters

Class	Parameter	Description	Values
audio	device	Name of the audio device	***plughw:0,0***
	format	PCM Format *For more information look in snd_pcm_format_t*	***2***
	samplerate	Sample rate of audio in Hz	***8000*** 44100 48000 …
	channels	Number of audio channels *Only 2 channels are supported by audio driver*	1 ***2***
	type	Blocking Mode *Supported value of type is only 0*	***0*** : Blocking 1 : Non Blocking
vpbe	device	Name of the VPBE device	***/dev/video2***
	opMode	Driver operation mode	0 : MMAP mode ***2*** : User-buffer mode
	standard	Video standard	1(NTSC) 2(1080i_30) ***3***(720p_60) 4(PAL) 5(720p_50) 6(1080i_25) 7(480p_60) 8(576p_50)

	output	Video Output	***3***(Component) 1(Composite)
	cropEnable	Enables crop feature	***0***(Disabled) 1(Enabled)
	cropCapLeft	Crop left	***0*** *<Set as per the driver support features>*
	cropCapTop	Crop top	***0*** *<Set as per the driver support features>*
	cropCapWidth	Crop width	***720*** *<Set as per the driver support features>*
	cropCapHeight	Crop height	***480*** *<Set as per the driver support features>*
vpfe	device	Name of the VPFE device	***/dev/video0***
	standard	Video standard	***5***(720p) 1(NTSC) …
	format	Format of video data to be captured	**8(YUV420SP)** 0 : YUYV ***1*** : UYVY 2 : YUV420 3 : YUV422P 4 : YUV410 6 : RGB565 7 : YUV422UVP
	input	Type of physical video input connected to the EVM	*2* : Composite ***4*** : Component

	width	Width of capture window < *driver and codec limitations apply*>	***1280*** 736 640 …
	height	Height of capture window < *driver and codec limitations apply* >	***720*** 576 480 …
	cropEnable	Enables crop feature	***0***(Disabled) 1(Enabled)
	cropCapLeft	Crop left	***0*** <*Set as per the driver support features*>
	cropCapTop	Crop top	***0*** <*Set as per the driver support features*>
	cropCapWidth	Crop width	***720*** <*Set as per the driver support features*>
	cropCapHeight	Crop height	***480*** <*Set as per the driver support features*>
	hdCaptureFactor	FPS controlling factor for 720P standard <em.m Setting to '2' would have the capture at (60/2)FPS, 3 to (60/3) FPS etc> <Should be non multiple of 0>	***1****(Driver captures at 60 FPS) FPS* **…**

2.4.2.5 OMAP3530(Linux)

Table 1. Peripheral Classes Configuration Parameters

Class	Parameter	Description	Values
audio	device	Name of the audio device	***plughw:0,0***
	format	PCM Format *For more information look in snd_pcm_format_t*	0 1 ***2*** …..
	samplerate	Sample rate of audio in Hz	48000 44100 …….. ***8000***
	channels	Number of audio channels	1 ***2***
	type	Blocking mode *Only blocking mode 0 is supported by audio driver*	***0*** 1
vpbe	device	Name of the VPBE device	***/dev/video1*** /dev/video2
	width	Width of display window *For LCD out max width is 640*	…….. ***720*** 640 352 ……..
	height	Height of display window *For LCD out max height is 480*	…….. ***576*** 480 …….

	maxBuffers	maxBuffers	***3***
	output	Video output device	1(tv) ***2*** (lcd) 3(DVI-D)
	scaling	Upscaling/downscaling	***1***(on) 0(off)
	ch_mode	Use to set mode on channel0/channel1	720P (only for DVI-D) 480P (only for DVI-D) ***VGA*** (only for LCD) NTSC (only for TV) PAL (only for TV) Note: Presently this field supports only 480P, VGA and NTSC
	rotate	Set rotation	***0*** 90 180 270
vpfe	device	Name of the VPFE device	***/dev/video0***
	standard	Standard of video capture device connected to EVM	***1*** : NTSC 2 : PAL 4 : NA
	format	Format of video data to be captured	0 : YUYV ***1*** : UYVY
	input	Type of physical video input connected to the EVM	***0*** : Composite 1 : S-video
	width	Width of capture window	***720***
	height	Height of capture window	***480*** 576

vprsz	device	Name of the Memory to memory resizer device	***/dev/omap-resizer***
	rszEnable	Enable/Disable Resizer *Only supported value is 0*	***0*** : Resizer not in use 1 : Resizer in use

Note:

1) ***VPBE:***

When scaling is off -> Displays original size image on screen

When scaling is on -> image displays on full screen (TV(720x576) , LCD (640x480),DVI-D(*1280x720))*

Scaling is done in combination with output parameter irrespective of the interface use for display. For example

Scaling	Output	Resolution
1	1	720x576
1	2	640x480
1	3	1280x720

2) ***vprsz:*** It is not supported in DVTB.

2.4.2.6 DM3730(Linux)

Table 2. Peripheral Classes Configuration Parameters

Class	Parameter	Description	Values
audio	device	Name of the audio device	***plughw:0,0***
	format	PCM Format *For more information look in snd_pcm_format_t*	0 1 ***2*** …..
	samplerate	Sample rate of audio in Hz	48000 44100 …….. ***8000***
	channels	Number of audio channels	1 ***2***

	type	Blocking mode *Only blocking mode 0 is supported by audio driver*	***0*** 1
vpbe	device	Name of the VPBE device	***/dev/video1*** /dev/video2
	width	Width of display window *For LCD out max width is 640*	…….. ***720*** 640 352 ……..
	height	Height of display window *For LCD out max height is 480*	…….. ***576*** 480 …….
	maxBuffers	maxBuffers	***3***
	output	Video output device	1(tv) ***2*** (lcd) 3(DVI-D)
	scaling	Upscaling/downscaling	***1***(on) 0(off)
	ch_mode	Use to set mode on channel0/channel1	720P (only for DVI-D) 480P (only for DVI-D) ***VGA*** (only for LCD) NTSC (only for TV) PAL (only for TV) Note: Presently this field supports only 480P, VGA and NTSC
	rotate	Set rotation	***0*** 90 180

			270
vpfe	device	Name of the VPFE device	***/dev/video0***
	standard	Standard of video capture device connected to EVM	***1*** : NTSC 2 : PAL 4 : NA
	format	Format of video data to be captured	0 : YUYV ***1*** : UYVY
	input	Type of physical video input connected to the EVM	***0*** : Composite 1 : S-video
	width	Width of capture window	***720***
	height	Height of capture window	***480*** 576
vprsz	device	Name of the Memory to memory resizer device	***/dev/omap-resizer***
	rszEnable	Enable/Disable Resizer *Only supported value is 0*	***0*** : Resizer not in use 1 : Resizer in use

Note:

3) ***VPBE:***

When scaling is off -> Displays original size image on screen

When scaling is on -> image displays on full screen (TV(720x576) , LCD (640x480),DVI-D(*1280x720))*

Scaling is done in combination with output parameter irrespective of the interface use for display. For example

Scaling	Output	Resolution
1	1	720x576
1	2	640x480
1	3	1280x720

4) ***vprsz:*** It is not supported in DVTB.

2.4.2.7 OMAP3530(WinCE)

N/A

2.4.2.8 Codec Classes configuration parameters

The following platform-wise tables present the configuration parameters for the codec classes.

2.4.2.9 DM6446(Linux)

Table 3. Engine Configuration Parameters

Class	Parameter	Description	Values
engine	name	Name of codec server to be used	***decode*** encode loopback

Table 4. viddec2 Class Configuration Parameters

Class	Parameter	Description	Values
viddec2	codec	Name of video decoder	***mpeg2dec*** mpeg4dec h264dec
	maxHeight	Maximum video height in pixels	***480*** …
	maxWidth	Maximum video width in pixels	***720*** …
	maxFrameRate	Maximum frame rate in fps*1000	***30000*** 25000 …
	maxBitRate	Maximum bit rate in bits per second	***10000000*** 1500000 …
	dataEndianness	Endianness of data input	***1*** : XDM_BYTE

			Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	forceChromaFormat	Force decode in given chroma format *Other YUV/RGB formats are not supported by video decoders*	***1*** : XDM_YUV_420P YUV 4:2:0 planar 4 : XDM_YUV_422ILE YUV 4:2:2 interleaved
	decodeHeader	Number of access units to decode Dynamic Parameter	***0*** : XDM_DECODE_AU Decode access unit 1 : XDM_PARSE_HEADER Decode only header
	displayWidth	Pitch Dynamic Parameter	***0*** …
	frameSkipMode	Frame Skip Mode Dynamic Parameter *Only IVIDEO_NO_SKIP supported by video decoders*	***0*** : IVIDEO_NO_SKIP Don't skip current frame
	frameOrder	Frame Order	***0***
	newFrameFlag	New Frame Flag	***0***
	mbDataFlag	MB Data Flag	***0***
	numFrames	Number of frames to decode	***100***

Table 5. videnc1 Class Configuration Parameters

Class	Parameter	Description	Value
videnc1	Codec	Name of video encoder	h264enc1

			mpeg4enc
	encodingPreset	Encoding Preset	***0***
	rateControlPreset	Rate Control Presets	***1*** : IVIDEO_LOW_DELAY Stringent CBR 2 : IVIDEO_STORAGE Constrained VBR 3 : IVIDEO_TWOPASS Two pass rate control 4 : IVIDEO_NONE Unconstrained VBR
	maxHeight	Maximum Height	576 ***480*** …
	maxWidth	Maximum Width	***720*** …
	maxFrameRate	Maximum Frame Rate in fps * 1000	***30000*** 25000 …
	maxBitRate	Maximum Bit Rate	***2000000*** …
	dataEndianness	Endianness of data output	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	maxInterFrameInterval	I to P frame distance	***0*** …

	inputChromaFormat	Input Chroma Format	***1*** : XDM_YUV_420P YUV 4:2:0 planar 4 : XDM_YUV_422ILE YUV 4:2:2 interleaved
	inputContentType	Type of input video content *Only progressive video is supported in H.264 BP and MPEG4 SP*	***0*** : IVIDEO_PROGRESSIVE Progressive video content 1 : IVIDEO_INTERLACED Interlaced video content
	reconChromaFormat	Recon Chroma Format	***1*** : XDM_YUV_420P YUV 4:2:0 planar 4 : XDM_YUV_422ILE YUV 4:2:2 interleaved
	inputHeight	Input Frame Height Dynamic Parameter	***480*** …
	inputWidth	Input Frame Width Dynamic Parameter	***720*** …
	refFrameRate	Reference or input frame rate in fps * 1000 Dynamic Parameter	***30000*** 25000 …
	targetFrameRate	Target Frame Rate in fps * 1000 Dynamic Parameter	***30000*** 25000 …
	targetBitRate	Target Bit Rate Dynamic Parameter	***2000000*** …
	intraFrameInterval	I frame interval Dynamic Parameter	***30*** 15 …
	generateHeader	Mode of encode	***0*** : XDM_ENCODE_AU

		Dynamic Parameter	Encode access unit 1 : XDM_GENERATE_HEADER Generate only header
	captureWidth	Pitch Dynamic Parameter	***720*** …
	forceFrame	Force the encoded frames to be I frames Dynamic Parameter	***0*** …
	interFrameInterval	Inter Frame Interval	***0***
	mbDataFlag	MB Data Flag	***0***
	framePitch	Frame Pitch	***720***
	numframes	Number of frames to capture/encode DVTB parameter	***30*** 300 1800 …

Table 6. auddec1 Class Configuration Parameters

Class	Parameter	Description	Values
auddec1	Codec	Name of audio decoder	***aachedec*** mp3dec
	outputPCMWidth	Output PCM width	***16*** …
	pcmFormat	PCM Format	***1*** …
	dataEndianness	Endianness of input data	***1*** : XDM_BYTE Big Endian stream

			2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	downSampleSbrFlag	Down Sample the output	***0*** 1
	inbufsize	Size of the input buffer	***2048***
	outbufsize	Size of the output buffer	***8192***

Table 7. mp3basedec1 Class Configuration Parameters

Class	Parameter	Description	Values
mp3basedec1	codec	Name of audio decoder	***mp3dec***
	outputPCMWidth	Output PCM width	***16*** …
	pcmFormat	PCM Format	***1*** …
	dataEndianness	Endianness of input data	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	downSampleSbrFlag	Down Sample the output	***0*** 1
	inbufsize	Size of the input buffer	***2880***
	outbufsize	Size of the output buffer	***4608***

Table 8. sphdec1 Class Configuration Parameters

Class	Parameter	Description	Values
sphdec1	codec	Name of speech decoder	***g711dec***
	compandingLaw	Companding Law	***1*** : A-Law 2 : U-Law
	packingType	Packing Format Option	0 ***1*** 2
	codecSelection	Codec Selection	***0***
	bitRate	Bit Rate	***12000***
	inbufsize	Size of the input buffer	***80***
	outbufsize	Size of the output buffer	***320***

Table 9. sphenc1 Class Configuration Parameters

Class	Parameter	Description	Values
sphenc1	Codec	Name of speech encoder	***g711enc***
	Seconds	Number of seconds to capture DVTB parameter	***10*** …
	frameSize	Input Frame Size in bytes	***80***
	compandingLaw	Companding Law	***1*** : A-Law 2 : U-Law
	packingType	Packing Format Option	***0*** 1 2
	vadSelection	VAD selection	***0***

	codecSelection	Codec Selection	***0***
	bitRate	Bit Rate	***12000***
	inbufsize	Size of the input buffer	***160***
	outbufsize	Size of the output buffer	***80***

Table 10. audenc1 Class Configuration Parameters

Class	Parameter	Description	Values
audenc1	codec	Name of audio encoder	***aacheenc***
	seconds	Number of seconds to capture DVTB parameter	***10*** …
	sampleRate	Sample Rate	***44100***
	bitRate	Average Bit Rate	***64000***
	channelMode	Input channel Mode	0 : IAUDIO_MONO ***1*** : IAUDIO_STEREO
	dataEndianness	Endianness of input data	1 : XDM_BYTE Big Endian stream ***2*** : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	encMode	Encoding Mode	***0***: IAUDIO_CBR Constant bit-rate 1: IAUDIO_VBR Variable bit-rate
	inputFormat	Input PCM format	0:IAUDIO_BLOCK Left channel data followed by right channel data

			1: IAUDIO_INTERLEAVED Left and Right channel data interleaved
	inputBitsPerSample	Number of bits per input PCM sample	***16***
	maxBitRate	Maximum bit rate in case of VBR	***192000***
	dualMonoMode	Mode to indicate type of dual mono	***0***: IAUDIO_DUALMONO_LR Play/Encode both left and right channel 1: IAUDIO_DUALMONO_LEFT Play/Encode only left channel 2: IAUDIO_DUALMONO_RIGHT Play/Encode only right channel 3: IAUDIO_DUALMONO_LR_MIX MIX and Play
	crcFlag	Flag indicates whether the encoder should insert CRC bit into the bitstream or not	***0***: XDAS_FALSE 1: XDAS_TRUE
	ancFlag	Ancillary Data Flag	0: XDAS_FALSE ***1***: XDAS_TRUE
	lfeFlag	Flag Indicates whether LFE channel data is present or not in the input	***0***: XDAS_FALSE 1: XDAS_TRUE
	dynamicparams.sampleRate	Sampling Frequency in Hertz	***44100***
	dynamicparams.bitRate	Average bitrate in bits per second	***64000***
	dynamicparams.channelMode	Number of channels	0 : IAUDIO_MONO ***1*** : IAUDIO_STEREO
	dynamicparams.lfeFlag	Flag Indicates whether LFE channel data is	***0***: XDAS_FALSE

		present or not in the input	1: XDAS_TRUE
	dynamicparams.dualMonoMode	Mode to indicate type of dual mono	***0***: IAUDIO_DUALMONO_LR Play/Encode both left and right channel 1: IAUDIO_DUALMONO_LEFT Play/Encode only left channel 2: IAUDIO_DUALMONO_RIGHT Play/Encode only right channel 3: IAUDIO_DUALMONO_LR_MIX MIX and Play
	dynamicparams.inputBitsPerSample	Number of bits per input PCM sample	***16*** 32

Table 11. aacheenc1 Class Configuration Parameters

Class	Parameter	Description	Values
aacheenc1	codec	Name of audio encoder	***aacheenc***
	seconds	Number of seconds to capture DVTB parameter	***10*** …
	sampleRate	Sample Rate	***44100***
	bitRate	Average Bit Rate	***64000***
	channelMode	Input channel Mode	0 : IAUDIO_MONO ***1*** : IAUDIO_STEREO
	dataEndianness	Endianness of input data	1 : XDM_BYTE Big Endian stream ***2*** : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32

			32 Little Endian stream
	encMode	Encoding Mode	***0***: IAUDIO_CBR Constant bit-rate 1: IAUDIO_VBR Variable bit-rate
	inputFormat	Input PCM format	0:IAUDIO_BLOCK Left channel data followed by right channel data ***1***: IAUDIO_INTERLEAVED Left and Right channel data interleaved
	inputBitsPerSample	Number of bits per input PCM sample	***16***
	maxBitRate	Maximum bit rate in case of VBR	***192000***
	dualMonoMode	Mode to indicate type of dual mono	***0***: IAUDIO_DUALMONO_LR Play/Encode both left and right channel 1: IAUDIO_DUALMONO_LEFT Play/Encode only left channel 2: IAUDIO_DUALMONO_RIGHT Play/Encode only right channel 3: IAUDIO_DUALMONO_LR_MIX MIX and Play
	crcFlag	Flag indicates whether the encoder should insert CRC bit into the bitstream or not	***0***: AACENC_FALSE 1: AACENC_TRUE
	ancFlag	Ancillary Data Flag	0: AACENC_FALSE ***1***: AACENC_TRUE
	lfeFlag	Flag Indicates whether LFE channel data is present or not in the	***0***: AACENC_FALSE 1: AACENC_TRUE

		input	
	outObjectType	Output Object Type	2: AACENC_OBJ_TYP_LC AAC Low complexity ***5***: AACENC_OBJ_TYP_HEAAC AAC Encoder with SBR capability 29: AACENC_OBJ_TYP_PS AAC Encoder with SBR and PS
	outFileFormat	Output File Format	0: AACENC_TT_RAW RAW output format 1: AACENC_TT_ADIF ADIF file format ***2***: AACENC_TT_ADTS ADTS file format
	useTns	Flag to activate TNS	0: AACENC_FALSE ***1***: AACENC_TRUE
	usePns	Flag to activate PNS	0: AACENC_FALSE ***1***: AACENC_TRUE
	downMixFlag	Flag to enable down mixing	***0***: AACENC_FALSE 1: AACENC_TRUE
	bitRateMode	Type of VBR mode	***1***: AACENC_BR_MODE_VBR_1 2: AACENC_BR_MODE_VBR_2 3: AACENC_BR_MODE_VBR_3 4: AACENC_BR_MODE_VBR_4 5: AACENC_BR_MODE_VBR_5
	ancRate	Ancillary data rate	***-1***
	dynamicparams.sampleRate	Sampling Frequency in Hertz	***44100***
	dynamicparams.bitRate	Average bitrate in bits per second	***64000***

	dynamicparams.channelMode	Number of channels	0 : IAUDIO_MONO ***1*** : IAUDIO_STEREO
	dynamicparams.lfeFlag	Flag Indicates whether LFE channel data is present or not in the input	***0***: AACENC_FALSE 1: AACENC_TRUE
	dynamicparams.dualMonoMode	Mode to indicate type of dual mono	***0***: IAUDIO_DUALMONO_LR Play/Encode both left and right channel 1: IAUDIO_DUALMONO_LEFT Play/Encode only left channel 2: IAUDIO_DUALMONO_RIGHT Play/Encode only right channel 3: IAUDIO_DUALMONO_LR_MIX MIX and Play
	dynamicparams.inputBitsPerSample	Number of bits per input PCM sample	***16*** 32
	dynamicparams.useTns	Flag to activate TNS	0: AACENC_FALSE ***1***: AACENC_TRUE
	dynamicparams.usePns	Flag to activate PNS	0: AACENC_FALSE ***1***: AACENC_TRUE
	dynamicparams.downMixFlag	Flag to enable down mixing	***0***: AACENC_FALSE 1: AACENC_TRUE
	dynamicparams.ancFlag	Ancillary data flag	0: AACENC_FALSE ***1***: AACENC_TRUE
	dynamicparams.ancRate	Ancillary data rate	***-1***

Table 12. aaclcenc1 Class Configuration Parameters

Class	Parameter	Description	Values

aaclcenc1	codec	Name of audio encoder	***aaclcenc***
	sampleRate	Sample Rate	***44100***
	bitRate	Average Bit Rate	***48000***
	channelMode	Input channel Mode	***0*** : IAUDIO_MONO 1 : IAUDIO_STEREO
	dataEndianness	Endianness of input data	1 : XDM_BYTE Big Endian stream ***2*** : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	encMode	Encoding Mode	***0***: IAUDIO_CBR Constant bit-rate 1: IAUDIO_VBR Variable bit-rate
	inputFormat	Input PCM format	0:IAUDIO_BLOCK Left channel data followed by right channel data ***1***: IAUDIO_INTERLEAVED Left and Right channel data interleaved
	inputBitsPerSample	Number of bits per input PCM sample	***16***
	maxBitRate	Maximum bit rate in case of VBR	***192000***
	dualMonoMode	Mode to indicate type of dual mono	***0***: IAUDIO_DUALMONO_LR Play/Encode both left and right channel 1: IAUDIO_DUALMONO_LEFT

			Play/Encode only left channel 2: IAUDIO_DUALMONO_RIGHT Play/Encode only right channel 3: IAUDIO_DUALMONO_LR_MIX MIX and Play
	crcFlag	Flag indicates whether the encoder should insert CRC bit into the bitstream or not	***0***: AACENC_FALSE 1: AACENC_TRUE
	ancFlag	Ancillary Data Flag	0: AACENC_FALSE ***1***: AACENC_TRUE
	lfeFlag	Flag Indicates whether LFE channel data is present or not in the input	***0***: AACENC_FALSE 1: AACENC_TRUE
	outObjectType	Output Object Type	***2***: AACENC_OBJ_TYP_LC AAC Low complexity 5: AACENC_OBJ_TYP_HEAAC AAC Encoder with SBR capability 29: AACENC_OBJ_TYP_PS AAC Encoder with SBR and PS
	outFileFormat	Output File Format	0: AACENC_TT_RAW RAW output format 1: AACENC_TT_ADIF ADIF file format ***2***: AACENC_TT_ADTS ADTS file format
	useTns	Flag to activate TNS	0: AACENC_FALSE ***1***: AACENC_TRUE
	usePns	Flag to activate PNS	0: AACENC_FALSE ***1***: AACENC_TRUE

	downMixFlag	Flag to enable down mixing	***0***: AACENC_FALSE 1: AACENC_TRUE
	bitRateMode	Type of VBR mode	***1***: AACENC_BR_MODE_VBR_1 2: AACENC_BR_MODE_VBR_2 3: AACENC_BR_MODE_VBR_3 4: AACENC_BR_MODE_VBR_4 5: AACENC_BR_MODE_VBR_5
	ancRate	Ancillary data rate	***-1***
	dynamicparams.sampleRate	Sampling Frequency in Hertz	***44100***
	dynamicparams.bitRate	Average bitrate in bits per second	***48000***
	dynamicparams.channelMode	Number of channels	0 : IAUDIO_MONO ***1*** : IAUDIO_STEREO
	dynamicparams.lfeFlag	Flag Indicates whether LFE channel data is present or not in the input	***0***: AACENC_FALSE 1: AACENC_TRUE
	dynamicparams.dualMonoMode	Mode to indicate type of dual mono	***0***: IAUDIO_DUALMONO_LR Play/Encode both left and right channel 1: IAUDIO_DUALMONO_LEFT Play/Encode only left channel 2: IAUDIO_DUALMONO_RIGHT Play/Encode only right channel 3: IAUDIO_DUALMONO_LR_MIX MIX and Play
	dynamicparams.inputBitsPerSample	Number of bits per input PCM sample	***16*** 32
	dynamicparams.useTns	Flag to activate TNS	0: AACENC_FALSE

			1: AACENC_TRUE
	dynamicparams.usePns	Flag to activate PNS	0: AACENC_FALSE ***1***: AACENC_TRUE
	dynamicparams.downMixFlag	Flag to enable down mixing	***0***: AACENC_FALSE 1: AACENC_TRUE
	dynamicparams.outObjectType	Output Object Type	***2***: AACENC_OBJ_TYP_LC AAC Low complexity 5: AACENC_OBJ_TYP_HEAAC AAC Encoder with SBR capability 29: AACENC_OBJ_TYP_PS AAC Encoder with SBR and PS
	dynamicparams.outFileFormat	Output File Format	0: AACENC_TT_RAW RAW output format 1: AACENC_TT_ADIF ADIF file format ***2***: AACENC_TT_ADTS ADTS file format
	dynamicparams.ancFlag	Ancillary data flag	0: AACENC_FALSE ***1***: AACENC_TRUE
	dynamicparams.ancRate	Ancillary data rate	***-1***

Table 13. imgenc1 Class Configuration Parameters

Class	Parameter	Description	Values
imgenc1	codec	Name of image encoder	***Jpegenc***
	maxHeight	Maximum video height in pixels	***480***
	maxWidth	Maximum video width in pixels	***720***
	maxScans	Maximum number of scans	***0***

		for progressive mode	
	dataEndianness	Endianness of input data	***1***
	forceChromaFormat	Force decoding in the given Chroma Format	***2*** : XDM_YUV_422P YUV 4:2:2 planar
	numAU	Number of Access unit	***0***
	inputChromaFormat	Color format of the input data	***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved
	inputHeight	Input Height of the image	***480***
	inputWidth	Input Width of the image	***720***
	captureWidth	Capture width in pels	***720***
	generateHeader	Set to 0 to encode complete AU	***0***
	qValue	Q value compression factor for encoder	***73***

Table 14. imgdec1 Class Configuration Parameters

Class	Parameter	Description	Values
Imgdec1	codec	Name of image decoder	***jpegdec***
	maxHeight	Maximum video height in pixels	***480***
	maxWidth	Maximum video width in pixels	***720***
	maxScans	Maximum number of scans for progressive mode	***1***
	dataEndianness	Endianness of input data	***1***
	forceChromaFormat	Force decoding in the given Chroma Format	***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved
	numAU	Number of access unit	***0***
	decodeHeader	Set to 0 to decode complete AU	***0***

	displayWidth	Set to 0 decode entire width of the image	***0***

The codec parameters and the codec dynamic parameters are defined in the following structures defined by the VISA APIs:

- VIDENC1_Params, VIDENC1_DynamicParams
- VIDDEC2_Params, VIDDEC2_DynamicParams
- AUDDEC1_Params, AUDDEC1_DynamicParams
- AUDENC1_Params, AUDENC1_DynamicParams
- SPHENC1_Params, SPHENC1_DynamicParams
- SPHDEC1_Params, SPHDEC1_DynamicParams
- IMGENC1_Params, IMGENC1_DynamicParams
- IMGDEC1_Params, IMDDEC1_DynamicParams

2.4.2.10 DM355(Linux)

Table 15. Engine Configuration Parameters

Class	Parameter	Description	Values
Engine	name	Name of codec server to be used	***encodedecode*** decode encode

Table 16. viddec2 Class Configuration Parameters

Class	Parameter	Description	Values
viddec2	Codec	Name of video decoder	***mpeg4dec***
	maxHeight	Maximum video height in pixels	***480*** …

	maxWidth	Maximum video width in pixels	***720*** …
	maxFrameRate	Maximum frame rate in fps*1000	***30000*** 25000 …
	maxBitRate	Maximum bit rate in bits per second	***10000000*** 1500000 …
	dataEndianness	Endianness of data input	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	forceChromaFormat	Force decode in given chroma format *Other YUV/RGB formats are not supported by video decoders*	1 : XDM_YUV_420P YUV 4:2:0 planar ***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved
	decodeHeader	Number of access units to decode Dynamic Parameter	***0*** : XDM_DECODE_AU Decode access unit 1 : XDM_PARSE_HEADER Decode only header
	displayWidth	Pitch Dynamic Parameter	***0*** …
	frameSkipMode	Frame Skip Mode Dynamic Parameter *Only IVIDEO_NO_SKIP supported by video decoders*	***0*** : IVIDEO_NO_SKIP Don't skip current frame
	frameOrder	Frame order	***0*** :

		Dynamic Parameter	IVIDDEC_DISPLAY_ORDER Video decoder output frame order.
	newFrameFlag	Indicate that the algorithm should start a new frame Dynamic Parameter	***512*** …
	mbDataFlag	Flag to indicate generate MB data	***80*** …
	numFrames	Number frame to decode	***10*** …

Table 17. videnc1 Class Configuration Parameters

Class	Parameter	Description	Value
videnc1	codec	Name of video encoder	***mpeg4enc***
	encodingPreset	Encoding Preset	***0***
	rateControlPreset	Rate Control Presets	***1*** : IVIDEO_LOW_DELAY Stringent CBR 2 : IVIDEO_STORAGE Constrained VBR 3 : IVIDEO_TWOPASS Two pass rate control 4 : IVIDEO_NONE Unconstrained VBR
	maxHeight	Maximum Height	576 ***480*** …
	maxWidth	Maximum Width	***720*** …

	maxFrameRate	Maximum Frame Rate in fps * 1000	***30000*** 25000 …
	maxBitRate	Maximum Bit Rate	***2000000*** …
	dataEndianness	Endianness of data output	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	maxInterFrameInterval	I to P frame distance	***0*** …
	inputChromaFormat	Input Chroma Format	1 : XDM_YUV_420P YUV 4:2:0 planar ***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved
	inputContentType	Type of input video content	***0*** : IVIDEO_PROGRESSIVE Progressive video content 1 : IVIDEO_INTERLACED Interlaced video content
	reconChromaFormat	Chroma format reconstructed for frames	***1*** : XDM_YUV_420P
	topFieldFirstFlag	Flag	***0***
	inputHeight	Input Frame Height Dynamic Parameter	***480*** …
	inputWidth	Input Frame Width Dynamic Parameter	***720*** …

	refFrameRate	Reference or input frame rate in fps * 1000 Dynamic Parameter	***30000*** 25000 …
	targetFrameRate	Target Frame Rate in fps * 1000 Dynamic Parameter	***30000*** 25000 …
	targetBitRate	Target Bit Rate Dynamic Parameter	***2000000*** …
	intraFrameInterval	I frame interval Dynamic Parameter	***30*** 15 …
	generateHeader	Mode of encode Dynamic Parameter	***0*** : XDM_ENCODE_AU Encode access unit 1 : XDM_GENERATE_HEADER Generate only header
	captureWidth	Pitch Dynamic Parameter	***0*** …
	forceFrame	Force the encoded frames to be I frames Dynamic Parameter	***0*** …
	interFrameInterval	Number consecutive B frames	***0***
	mbDataFlag	MB flag	***0***
	numFrames	Number of frames to capture/encode DVTB parameter	***30*** 300 1800 …

Table 18. sphdec1 Class Configuration Parameters

Class	Parameter	Description	Values
sphdec1	codec	Name of speech decoder	***g711dec***
	compandingLaw	Companding Law	***1*** : A-Law 2 : U-Law
	packingType	Packing Format Option	0 ***1*** 2
	codecSelection	Codec Selection	***0***
	bitRate	Bit Rate	***12000***
	inbufsize	Size of the input buffer	***80***
	outbufsize	Size of the output buffer	***320***

Table 19. sphenc1 Class Configuration Parameters

Class	Parameter	Description	Values
sphenc1	Codec	Name of speech encoder	***g711enc***
	Seconds	Number of seconds to capture DVTB parameter	***10*** …
	frameSize	Input Frame Size in bytes	***80***
	compandingLaw	Companding Law	***1*** : A-Law 2 : U-Law
	packingType	Packing Format Option	***0*** 1 2
	vadSelection	VAD selection	***0***

	codecSelection	Codec Selection	***0***
	bitRate	Bit Rate	***12000***
	inbufsize	Size of the input buffer	***160***
	outbufsize	Size of the output buffer	***80***

Table 20. imgenc1 Class Configuration Parameters

Class	Parameter	Description	Values
imgenc1	codec	Name of image encoder	***Jpegenc***
	maxHeight	Maximum video height in pixels	***480***
	maxWidth	Maximum video width in pixels	***720***
	maxScans	Maximum number of scans for progressive mode	***0***
	dataEndianness	Endianness of input data	***1***
	forceChromaFormat	Force decoding in the given Chroma Format	***2*** : XDM_YUV_422P YUV 4:2:2 planar
	numAU	Number of Access unit	***0***
	inputChromaFormat	Color format of the input data	***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved
	inputHeight	Input Height of the image	***480***
	inputWidth	Input Width of the image	***720***
	captureWidth	Capture width in pels	***720***
	generateHeader	Set to 0 to encode complete AU	***0***
	qValue	Q value compression factor for encoder	***73***

Table 21. imgdec1 Class Configuration Parameters

Class	Parameter	Description	Values
Imgdec1	codec	Name of image decoder	***jpegdec***
	maxHeight	Maximum video height in pixels	***480***
	maxWidth	Maximum video width in pixels	***720***
	maxScans	Maximum number of scans for progressive mode	***1***
	dataEndianness	Endianness of input data	***1***
	forceChromaFormat	Force decoding in the given Chroma Format	***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved
	numAU	Number of access unit	***0***
	decodeHeader	Set to 0 to decode complete AU	***0***
	displayWidth	Set to 0 decode entire width of the image	***0***

DVTB classes supporting extended parameters are listed below:

“jpegenc1” – JPEG Encoder Extended parameters

“jpegdec1” – JPEG Decoder Extended parameters

“mpeg4spenc1” - MPEG4 Encoder Extended parameters

“mpeg4spdec2” – MPEG4 Decoder Extended parameters

NOTE:

1) Refer respective Codec’s User Guide for the list of parameters and their supported values.
2) In Video Decode and dump use case, if forceChromaFormat is set to XDM_YUV_420SP then output is generated in XDM_YUV_420P format.

The codec parameters and the codec dynamic parameters are defined in the following structures defined by the VISA APIs:

- VIDENC1_Params, VIDENC1_DynamicParams
- VIDDEC2_Params, VIDDEC2_DynamicParams
- SPHENC1_Params, SPHENC1_DynamicParams
- SPHDEC1_Params, SPHDEC1_DynamicParams
- IMGENC1_Params, IMGENC1_DynamicParams
- IMGDEC1_Params, IMDDEC1_DynamicParams

2.4.2.11 DM6467(Linux)

Table 22. Engine Configuration Parameters

Class	Parameter	Description	Values
Engine	name	Name of codec server to be used	***encodedecode***

Table 23. viddec2 Class Configuration Parameters

Class	Parameter	Description	Values
viddec2	codec	Name of video decoder	mpeg2dec h2641080p60vdec mpeg4dec ***h264dec***
	maxHeight	Maximum video height in pixels	***1088*** 576 …
	maxWidth	Maximum video width in pixels	***1920*** 720 …
	maxFrameRate	Maximum frame rate in	***30000***

		fps*1000	25000 …
	maxBitRate	Maximum bit rate in bits per second	***10000000*** 2000000 …
	dataEndianness	Endianness of data input	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	forceChromaFormat	Force decode in given chroma format *Other YUV/RGB formats are not supported by video decoders* *In Decode and dump use case, if forceChromaFormat is set to XDM_YUV_420SP then output is generated in XDM_YUV_420P format*	***9*** : XDM_YUV_420SP YUV 4:2:0 semi planer
	decodeHeader	Number of access units to decode Dynamic Parameter	***0*** : XDM_DECODE_AU Decode access unit 1 : XDM_PARSE_HEADER Decode only header
	displayWidth	Pitch Dynamic Parameter	***0*** …
	frameSkipMode	Frame Skip Mode Dynamic Parameter *Only IVIDEO_NO_SKIP supported by video decoders*	***0*** : IVIDEO_NO_SKIP Don’t skip current frame
	frameOrder	Frame order	***0***
	newFrameFlag	New frame flag	***0***

	mbDataFlag	Macro block data flag	***0***
	numFrames	Number of frames to decode	***100*** …

Table 24. videnc1 Class Configuration Parameters

Class	Parameter	Description	Value
videnc1	codec	Name of video encoder	***h264enc*** h264fhdvenc mpeg4enc
	encodingPreset	Encoding Preset	***0*** : XDM_DEFAULT
	rateControlPreset	Rate Control Presets	***1*** : IVIDEO_LOW_DELAY Stringent CBR 2 : IVIDEO_STORAGE Constrained VBR 3 : IVIDEO_TWOPASS Two pass rate control 4 : IVIDEO_NONE Unconstrained VBR
	maxHeight	Maximum Height	720 ***576*** 480 …
	maxWidth	Maximum Width	1280 ***720*** …
	maxFrameRate	Maximum Frame Rate in fps * 1000	***30000*** 25000 …

	maxBitRate	Maximum Bit Rate	***2000000*** …
	dataEndianness	Endianness of data output	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	maxInterFrameInterval	I to P frame distance	***0*** …
	inputChromaFormat	Input Chroma Format *In Read and Encode use case, if inputChromaFormat is set to XDM_YUV_420SP then input file should be in XDM_YUV_420P format. DVTB internally converts 420P to 420SP format before passing it to codec*	***9*** : XDM_YUV_420SP YUV 4:2:0 SP
	inputContentType	Type of input video content	***0*** : IVIDEO_PROGRESSIVE Progressive video content 1 : IVIDEO_INTERLACED Interlaced video content
	reconChromaFormat	Recon Chroma Format	***1*** : XDM_YUV_420P YUV 4:2:0 planar 9 : XDM_YUV_420SP YUV 4:2:0 Semi planar
	inputHeight	Input Frame Height Dynamic Parameter	***576*** …
	inputWidth	Input Frame Width	***720***

		Dynamic Parameter	…
	refFrameRate	Reference or input frame rate in fps * 1000 Dynamic Parameter	***30000*** 25000 …
	targetFrameRate	Target Frame Rate in fps * 1000 Dynamic Parameter	***30000*** 25000 …
	targetBitRate	Target Bit Rate Dynamic Parameter	***2000000*** …
	intraFrameInterval	I frame interval Dynamic Parameter	***30*** 15 …
	generateHeader	Mode of encode Dynamic Parameter	***0*** : XDM_ENCODE_AU Encode access unit 1 : XDM_GENERATE_HEADER Generate only header
	captureWidth	Pitch Dynamic Parameter	***0*** …
	forceFrame	Force the encoded frames to be I frames Dynamic Parameter	***-1*** 0 …
	interFrameInterval	Number of B frames between to reference frames Dynamic Parameter	***0*** : use maxInterFrameInterval …
	mbDataFlag	Flag to indicate that the algorithm should use the MB data provided in the inBuf	***0***
	framePitch	Frame Pitch	***1280***

	numFrames	Number of frames to capture/encode DVTB parameter	***30*** …

Table 25. auddec1 Class Configuration Parameters

Class	Parameter	Description	Values
auddec1	Codec	Name of audio decoder	***aachedec*** mp3dec dolbyac3dec
	outputPCMWidth	Output PCM width	***16*** …
	pcmFormat	PCM Format	***1*** …
	dataEndianness	Endianness of input data	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	desiredChannelMode	Desired channel configuration	0: IAUDIO_1_0 MONO ***1***: IAUDIO_2_0 Stereo 2: IAUDIO_11_0 Dual Mono …….
	downSampleSbrFlag	Down Sample the output	***0*** 1
	inbufsize	Size of the input buffer	***10000***

	outbufsize	Size of the output buffer	***16384***

Table 26. audenc1 Class Configuration Parameters

Class	Parameter	Description	Values
audenc1	Codec	Name of audio encoder	***aacheenc***
	Seconds	Number of seconds to capture DVTB parameter	***10*** …
	sampleRate	Sample Rate	***44100***
	bitRate	Average Bit Rate	***64000***
	channelMode	Input channel Mode	0 : IAUDIO_MONO ***1*** : IAUDIO_STEREO
	dataEndianness	Endianness of input data	1 : XDM_BYTE Big Endian stream ***2*** : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	encMode	Encoding Mode	***0***: IAUDIO_CBR Constant bit-rate 1: IAUDIO_VBR Variable bit-rate
	inputFormat	Input PCM format	0:IAUDIO_BLOCK Left channel data followed by right channel data ***1***: IAUDIO_INTERLEAVED Left and Right channel data interleaved

	inputBitsPerSample	Number of bits per input PCM sample	***16***
	maxBitRate	Maximum bit rate in case of VBR	***192000***
	dualMonoMode	Mode to indicate type of dual mono	***0***: IAUDIO_DUALMONO_LR Play/Encode both left and right channel 1: IAUDIO_DUALMONO_LEFT Play/Encode only left channel 2: IAUDIO_DUALMONO_RIGHT Play/Encode only right channel 3: IAUDIO_DUALMONO_LR_MIX MIX and Play
	crcFlag	Flag indicates whether the encoder should insert CRC bit into the bitstream or not	***0***: XDAS_FALSE 1: XDAS_TRUE
	ancFlag	Ancillary Data Flag	0: XDAS_FALSE ***1***: XDAS_TRUE
	lfeFlag	Flag Indicates whether LFE channel data is present or not in the input	***0***: XDAS_FALSE 1: XDAS_TRUE
	dynamicparams.sampleRate	Sampling Frequency in Hertz	***44100***
	dynamicparams.bitRate	Average bitrate in bits per second	***64000***
	dynamicparams.channelMode	Number of channels	0 : IAUDIO_MONO ***1*** : IAUDIO_STEREO
	dynamicparams.lfeFlag	Flag Indicates whether LFE channel data is present or not in the input	***0***: XDAS_FALSE 1: XDAS_TRUE
	dynamicparams.du	Mode to indicate type	***0***: IAUDIO_DUALMONO_LR

	alMonoMode	of dual mono	Play/Encode both left and right channel 1: IAUDIO_DUALMONO_LEFT Play/Encode only left channel 2: IAUDIO_DUALMONO_RIGHT Play/Encode only right channel 3: IAUDIO_DUALMONO_LR_MIX MIX and Play
	dynamicparams.inputBitsPerSample	Number of bits per input PCM sample	***16*** 32

Table 27. sphdec1 Class Configuration Parameters

Class	Parameter	Description	Values
sphdec1	codec	Name of speech decoder	***g711dec***
	compandingLaw	Companding Law	***1*** : A-Law 2 : U-Law
	packingType	Packing Format Option	0 ***1*** 2
	codecSelection	Codec Selection	***0***
	bitRate	Bit Rate	***12000***
	inbufsize	Size of the input buffer	***80***
	outbufsize	Size of the output buffer	***320***

Table 28. sphenc1 Class Configuration Parameters

Class	Parameter	Description	Values
sphenc1	Codec	Name of speech encoder	***g711enc***
	Seconds	Number of seconds to capture DVTB parameter	***10*** …
	frameSize	Input Frame Size in bytes	***80***
	compandingLaw	Companding Law	***1*** : A-Law 2 : U-Law
	packingType	Packing Format Option	***0*** 1 2
	vadSelection	VAD selection	***0***
	codecSelection	Codec Selection	***0***

	bitRate	Bit Rate	***0***
	vadFlag	VAD Flag	***0***
	inbufsize	Size of the input buffer	***160***
	outbufsize	Size of the output buffer	***80***

Table 29. imgenc1 Class Configuration Parameters

Class	Parameter	Description	Values
imgenc1	codec	Name of image encoder	***jpegenc***
	maxHeight	Maximum video height in pixels	720 ***576***
	maxWidth	Maximum video width in pixels	1280 ***720***
	maxScans	Maximum number of scans for progressive mode	***0***
	dataEndianness	Endianness of input data	***1***
	forceChromaFormat	Force decoding in the given Chroma Format	***2*** : XDM_YUV_422P YUV 4:2:2 planar
	numAU	Number of Access unit	***0***
	inputChromaFormat	Color format of the input data	9 : XDM_YUV_420SP YUV 4:2:0 semi_planar ***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved
	inputHeight	Input Height of the image	***480***
	inputWidth	Input Width of the image	***720***
	captureWidth	Capture width in pels	***720***
	generateHeader	Set to 0 to encode complete AU	***0***
	qValue	Q value compression factor for encoder	***73***

Table 30. imgdec1 Class Configuration Parameters

Class	Parameter	Description	Values
Imgdec1	codec	Name of image decoder	***jpegdec***
	maxHeight	Maximum video height in pixels	***576*** 480
	maxWidth	Maximum video width in pixels	1280 ***720***
	maxScans	Maximum number of scans for progressive mode	***1***
	dataEndianness	Endianness of input data	***1***
	forceChromaFormat	Force decoding in the given Chroma Format	9 : XDM_YUV_420SP YUV 4:2:0 semi_planar ***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved
	numAU	Number of access unit	***0***
	decodeHeader	Set to 0 to decode complete AU	***0***
	displayWidth	Set to 0 decode entire width of the image	***0***

DVTB classes supporting extended parameters are listed below:

“jpegenc1” – JPEG Encoder Extended parameters

“jpegdec1” – JPEG Decoder Extended parameters

“mpeg4enc1” – MPEG4 Encoder Extended parameters

“mpeg4dec2” – MPEG4 Decoder Extended parameters

“h264enc1” – H264 Encoder Extended parameters

“h264fhdenc1” – H264FHD Encoder Extended parameters

“h264dec2” – H264 Decoder Extended parameters

"h2641080pdec2" – H264 1080p Decoder Extended parameters

"mpeg2dec2" – MPEG4 Decoder Extended parameters

"ac3dec1" – AC3 Decoder Extended parameters

"aacheenc1" – AACLC Encoder Extended Parameters

"aachedec1" – AACHE Decoder Extended Parameters

NOTE:

1) Refer respective Codec's User Guide for the list of parameters and their supported values.

2) In Video Decode and dump use case, if forceChromaFormat is set to XDM_YUV_420SP then output is generated in XDM_YUV_420P format.

3) In Video Encode use case, if inputChromaFormat is set to XDM_YUV_420SP then it needs input file to be in XDM_YUV_420P format. It internally converts 420P to 420SP.

The codec parameters and the codec dynamic parameters are defined in the following structures defined by the VISA APIs:

- VIDENC1_Params, VIDENC1_DynamicParams
- VIDDEC2_Params, VIDDEC2_DynamicParams
- AUDENC1_Params, AUDENC1_DynamicParams
- AUDDEC1_Params, AUDDEC1_DynamicParams
- SPHENC1_Params, SPHENC1_DynamicParams
- SPHDEC1_Params, SPHDEC1_DynamicParams
- IMGENC1_Params, IMGENC1_DynamicParams
- IMGDEC1_Params, IMGDEC1_DynamicParams

2.4.2.12 DM365(Linux)

Table 31. Engine Configuration Parameters

Class	Parameter	Description	Values
Engine	Name	Name of codec server to be used	***encodedecode*** decode

			encode

Table 32. viddec2 Class Configuration Parameters

Class	Parameter	Description	Values
viddec2	Codec	Name of video decoder	mpeg4dec2 mpeg4hdvicpdec2 vc1dec2 mpeg2dec2 ***h264dec2***
	maxHeight	Maximum video height in pixels	***480*** …
	maxWidth	Maximum video width in pixels	***720*** …
	maxFrameRate	Maximum frame rate in fps*1000	***30000*** …
	maxBitRate	Maximum bit rate in bits per second	***10000000*** …
	dataEndianness	Endianness of data input	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	forceChromaFormat	Force decode in given chroma format *In Decode and dump use case, if forceChromaFormat is set to XDM_YUV_420SP then output is generated in XDM_YUV_420P format*	***9*** : XDM_YUV_420SP YUV 4:2:0 semi_planar *4* : XDM_YUV_422ILE YUV 4:2:2 interleaved

	decodeHeader	Number of access units to decode Dynamic Parameter	***0*** : XDM_DECODE_AU Decode access unit 1 : XDM_PARSE_HEADER Decode only header
	displayWidth	Pitch Dynamic Parameter	***0*** …
	frameSkipMode	Frame Skip Mode Dynamic Parameter *Only IVIDEO_NO_SKIP supported by video decoders*	***0*** : IVIDEO_NO_SKIP Don't skip current frame
	frameOrder	Frame order Dynamic Parameter	***0*** : IVIDDEC_DISPLAY_ORDER Video decoder output frame order.
	newFrameFlag	Indicate that the algorithm should start a new frame Dynamic Parameter	***0*** …
	mbDataFlag	Flag to indicate generate MB data	***0*** …
	numFrames	Number frame to decode	***30*** …

Table 33. videnc1 Class Configuration Parameters

Class	Parameter	Description	Value
videnc1	codec	Name of video encoder	mpeg4enc1 mpeg4hdvicpenc1 mpeg2enc1 ***h264enc1***
	encodingPreset	Encoding Preset	***0***

	rateControlPreset	Rate Control Presets	***1*** : IVIDEO_LOW_DELAY Stringent CBR 2 : IVIDEO_STORAGE Constrained VBR 3 : IVIDEO_TWOPASS Two pass rate control 4 : IVIDEO_NONE Unconstrained VBR 5 : IVIDEO_USER_DEFINED User defined configuration
	maxHeight	Maximum Height	720 576 ***480*** …
	maxWidth	Maximum Width	1280 ***720*** …
	maxFrameRate	Maximum Frame Rate in fps * 1000	***30000*** 25000 …
	maxBitRate	Maximum Bit Rate	***2000000*** …
	dataEndianness	Endianness of data output	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream

	maxInterFrameInterval	I to P frame distance	***0*** …
	inputChromaFormat	Input Chroma Format	***9*** : XDM_YUV_420SP YUV 4:2:0 semi_planar *4* : XDM_YUV_422ILE YUV 4:2:2 interleaved
	inputContentType	Type of input video content	***0*** : IVIDEO_PROGRESSIVE Progressive video content 1 : IVIDEO_INTERLACED Interlaced video content
	reconChromaFormat	Chroma format reconstructed for frames	***9*** : XDM_YUV_420SP YUV 4:2:0 semi_planar
	topFieldFirstFlag	Flag	***0***
	inputHeight	Input Frame Height Dynamic Parameter	***480*** …
	inputWidth	Input Frame Width Dynamic Parameter	***720*** …
	refFrameRate	Reference or input frame rate in fps * 1000 Dynamic Parameter	***30000*** 25000 …
	targetFrameRate	Target Frame Rate in fps * 1000 Dynamic Parameter	***30000*** 25000 …
	targetBitRate	Target Bit Rate Dynamic Parameter	***2000000*** …
	intraFrameInterval	I frame interval Dynamic Parameter	***30*** 15

			…
	generateHeader	Mode of encode Dynamic Parameter	***0*** : XDM_ENCODE_AU Encode access unit 1 : XDM_GENERATE_HEADER Generate only header
	captureWidth	Pitch Dynamic Parameter	***0*** …
	forceFrame	Force the encoded frames to be I frames Dynamic Parameter	***0*** …
	interFrameInterval	Number consecutive B frames	***0***
	mbDataFlag	MB flag	***0***
	numFrames	Number of frames to capture/encode DVTB parameter	***30*** 300 1800 …

Table 34. imgenc1 Class Configuration Parameters

Class	Parameter	Description	Values
imgenc1	codec	Name of image encoder	***Jpegenc1***
	maxHeight	Maximum video height in pixels	720 ***480***
	maxWidth	Maximum video width in pixels	1280 ***720***
	maxScans	Maximum number of scans for progressive mode	***0***

	dataEndianness	Endianness of input data	***1***
	forceChromaFormat	Force decoding in the given Chroma Format	***2*** : XDM_YUV_422P YUV 4:2:2 planar
	numAU	Number of Access unit	***0***
	inputChromaFormat	Color format of the input data	9 : XDM_YUV_420SP YUV 4:2:0 semi_planar ***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved
	inputHeight	Input Height of the image	***480***
	inputWidth	Input Width of the image	***720***
	captureWidth	Capture width in pels	***720***
	generateHeader	Set to 0 to encode complete AU	***0***
	qValue	Q value compression factor for encoder	***73***

Table 35. imgdec1 Class Configuration Parameters

Class	Parameter	Description	Values
Imgdec1	codec	Name of image decoder	***Jpegdec1***
	maxHeight	Maximum video height in pixels	***720*** 480
	maxWidth	Maximum video width in pixels	***1280*** 720
	maxScans	Maximum number of scans for progressive mode	***1***
	dataEndianness	Endianness of input data	***1***
	forceChromaFormat	Force decoding in the given Chroma Format	9 : XDM_YUV_420SP YUV 4:2:0 semi_planar

			4 : XDM_YUV_422ILE YUV 4:2:2 interleaved
	numAU	Number of access unit	***0***
	decodeHeader	Set to 0 to decode complete AU	***0***
	displayWidth	Set to 0 decode entire width of the image	***0***

Table 36. sphdec1 Class Configuration Parameters

Class	Parameter	Description	Values
sphdec1	codec	Name of speech decoder	***g711dec1***
	compandingLaw	Companding Law	***1*** : A-Law 2 : U-Law
	packingType	Packing Format Option	0 ***1*** 2
	codecSelection	Codec Selection	***0***
	bitRate	Bit Rate	***12000***
	inbufsize	Size of the input buffer	***80***
	outbufsize	Size of the output buffer	***320***

Table 37. sphenc1 Class Configuration Parameters

Class	Parameter	Description	Values
sphenc1	Codec	Name of speech encoder	***g711enc1***
	Seconds	Number of seconds to capture DVTB parameter	***10*** …
	frameSize	Input Frame Size in bytes	***80***

	compandingLaw	Companding Law	***1*** : A-Law 2 : U-Law
	packingType	Packing Format Option	***0*** 1 2
	vadSelection	VAD selection	***0***
	codecSelection	Codec Selection	***0***
	bitRate	Bit Rate	***12000***
	inbufsize	Size of the input buffer	***160***
	outbufsize	Size of the output buffer	***80***

Table 38. audenc1 Class Configuration Parameters

Class	Parameter	Description	Values
audenc1	codec	Name of audio encoder	***aaclcenc1***
	seconds	Number of seconds to capture DVTB parameter	***10*** …
	sampleRate	Sample Rate	***44100***
	bitRate	Average Bit Rate	***64000***
	channelMode	Input channel Mode	0 : IAUDIO_MONO ***1*** : IAUDIO_STEREO
	dataEndianness	Endianness of input data	1 : XDM_BYTE Big Endian stream ***2*** : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream

	encMode	Encoding Mode	***0***: IAUDIO_CBR Constant bit-rate 1: IAUDIO_VBR Variable bit-rate
	inputFormat	Input PCM format	0:IAUDIO_BLOCK Left channel data followed by right channel data ***1***: IAUDIO_INTERLEAVED Left and Right channel data interleaved
	inputBitsPerSample	Number of bits per input PCM sample	***16***
	maxBitRate	Maximum bit rate in case of VBR	***144000***
	dualMonoMode	Mode to indicate type of dual mono	***0***: IAUDIO_DUALMONO_LR Play/Encode both left and right channel 1: IAUDIO_DUALMONO_LEFT Play/Encode only left channel 2: IAUDIO_DUALMONO_RIGHT Play/Encode only right channel 3: IAUDIO_DUALMONO_LR_MIX MIX and Play
	crcFlag	Flag indicates whether the encoder should insert CRC bit into the bitstream or not	***0***: XDAS_FALSE 1: XDAS_TRUE
	ancFlag	Ancillary Data Flag	***0***: XDAS_FALSE 1: XDAS_TRUE
	lfeFlag	Flag Indicates whether LFE channel data is present or not in the input	***0***: XDAS_FALSE 1: XDAS_TRUE

	dynamicparams.sampleRate	Sampling Frequency in Hertz	***44100***
	dynamicparams.bitRate	Average bitrate in bits per second	***64000***
	dynamicparams.channelMode	Number of channels	0 : IAUDIO_MONO ***1*** : IAUDIO_STEREO
	dynamicparams.lfeFlag	Flag Indicates whether LFE channel data is present or not in the input	***0***: XDAS_FALSE 1: XDAS_TRUE
	dynamicparams.dualMonoMode	Mode to indicate type of dual mono	***0***: IAUDIO_DUALMONO_LR Play/Encode both left and right channel 1: IAUDIO_DUALMONO_LEFT Play/Encode only left channel 2: IAUDIO_DUALMONO_RIGHT Play/Encode only right channel 3: IAUDIO_DUALMONO_LR_MIX MIX and Play
	dynamicparams.inputBitsPerSample	Number of bits per input PCM sample	***16*** 32
	InSamplesPerChannel	Per channel sample size for one frame	***1024*** : for AACLC 2048 : for AACHE 1152: for MP3

Table 39. auddec1 Class Configuration Parameters

Class	Parameter	Description	Values
auddec1	Codec	Name of audio decoder	***aacdec1***
	outputPCMWidth	Output PCM width	***16*** …
	pcmFormat	PCM Format	***1***

			…
	dataEndianness	Endianness of input data	1 : XDM_BYTE Big Endian stream ***2*** : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	desiredChannelMode	Desired channel configuration	0: IAUDIO_1_0 MONO ***1***: IAUDIO_2_0 Stereo 2: IAUDIO_11_0 Dual Mono …….
	downSampleSbrFlag	Down Sample the output	***0*** 1
	inbufsize	Size of the input buffer	***10000***
	outbufsize	Size of the output buffer	***16384***

DVTB classes supporting extended parameters are listed below:

“jpegenc1” – JPEG Encoder Extended parameters

“jpegdec1” – JPEG Decoder Extended parameters

“mpeg4enc1” – MPEG4 Encoder Extended parameters

“mpeg4dec2” – MPEG4 Decoder Extended parameters

“mpeg4hdvicpenc1” – MPEG4 HDVICP Encoder Extended parameters

“mpeg4hdvicpdec2” – MPEG4 HDVICP Decoder Extended parameters

“h264enc1” – H264 Encoder Extended parameters

“h264dec2” – H264 Decoder Extended parameters

"mpeg2enc1" – MPEG4 Encoder Extended parameters

"mpeg2dec2" – MPEG4 Decoder Extended parameters

"vc1dec2" – MPEG4 Decoder Extended parameters

"aaclcenc1" – AACLC Encoder Extended Parameters

"aacdec1" – AACHE Decoder Extended Parameters

NOTE: Refer respective Codec's User Guide for the list of parameters and their supported values.

The codec parameters and the codec dynamic parameters are defined in the following structures defined by the VISA APIs:

- VIDENC1_Params, VIDENC1_DynamicParams
- VIDDEC2_Params, VIDDEC2_DynamicParams
- AUDENC1_Params, AUDENC1_DynamicParams
- AUDDEC1_Params, AUDDEC1_DynamicParams
- SPHENC1_Params, SPHENC1_DynamicParams
- SPHDEC1_Params, SPHDEC1_DynamicParams
- IMGENC1_Params, IMGENC1_DynamicParams
- IMGDEC1_Params, IMGDEC1_DynamicParams

2.4.2.13 OMA3530(Linux)

Table 40. viddec2 Class Configuration Parameters

Class	Parameter	Description	Values
viddec2	Codec	Name of video decoder	***mpeg2dec*** h264dec mpeg4dec
	maxHeight	Maximum video height in pixels	***576***
	maxWidth	Maximum video width in pixels	***720***
	maxFrameRate	Maximum frame rate in fps*1000	***30000***

	maxBitRate	Maximum bit rate in bits per second	***10000000***
	dataEndianness	Endianness of data input	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	forceChromaFormat	Force decode in given chroma format *Other YUV/RGB formats are not supported by video decoders*	1 : XDM_YUV_420P YUV 4:2:0 planar ***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved
	decodeHeader	Number of access units to decode Dynamic Parameter	***0*** : XDM_DECODE_AU Decode access unit 1 : XDM_PARSE_HEADER Decode only header
	displayWidth	Pitch Dynamic Parameter	***0***
	frameSkipMode	Frame Skip Mode Dynamic Parameter *Only IVIDEO_NO_SKIP supported by video decoders*	***0*** : IVIDEO_NO_SKIP Don't skip current frame
	frameOrder	Frame display order	***0***
	newFrameFlag	Flag to indicate that algorithm should start a new frame.	***0***
	mbDataFlag	Generate MB Data	***0***
	numFrames	Number of frames to decode	***100***
	pwrManagerEnable	Power Manager On/Off	***0***: Off 1 : ON

Table 41. videnc1 Class Configuration Parameters

Class	Parameter	Description	Values
videnc1	codec	Name of video encoder	***h264enc*** mpeg4enc
	encodingPreset	Encoding Present	***0***
	rateControlPreset	Rate control preset	***1*** : IVIDEO_LOW_DELAY Stringent CBR 2 : IVIDEO_STORAGE Constrained VBR 3 : IVIDEO_TWOPASS Two pass rate control 4 : IVIDEO_NONE Unconstrained VBR
	maxHeight	Maximum Height	***576*** 480 …
	maxWidth	Maximum Width	***720*** …
	maxFrameRate	Maximum Frame Rate if fps *1000	***30000***
	maxBitRate	Maximum Bit Rate	***2000000***
	dataEndianness	Endianness of data output	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	maxInterFrameInterval	I to Pframe distance	***1***

	inputChromaFormat	Input Chroma Format	***1*** : XDM_YUV_420P YUV 4:2:0 planar 4 : XDM_YUV_422ILE YUV 4:2:2 interleaved
	inputContentType	Type of input video content	***0*** : IVIDEO_PROGRESSIVE Progressive video content 1 : IVIDEO_INTERLACED Interlaced video content
	reconChromaFormat	Chroma format for the reconstruction buffers	***-1***:XDM_CHROMA_NA
	inputHeight	Input Frame Height Dynamic Parameter	576 ***480*** …
	inputWidth	Input Frame Width Dynamic Parameter	***720*** …
	refFrameRate	Reference or input frame rate in fps * 1000 Dynamic Parameter	***30000*** …
	targetFrameRate	Target Frame Rate in fps * 1000 Dynamic Parameter	***30000*** …
	targetBitRate	Target Bit Rate Dynamic Parameter	***2000000***
	intraFrameInterval	I frame interval Dynamic Parameter	***30***
	generateHeader	Mode of encode Dynamic Parameter	***0*** : XDM_ENCODE_AU Encode access unit 1:XDM_GENERATE_HEADER Generate only header
	captureWidth	Pitch	***0***

		Dynamic Parameter	
	forceFrame	Force the encoded frames to be I frames Dynamic Parameter	***0***
	interFrameInterval	Number of B frames between two reference frames	***1***
	mbDataFlag	Flag to indicate that the algorithm should use MB data provided in additional buffer with inBufs	***0***
	framePitch	Frame pitch to store the frames	***720***
	numFrames	Number of frames to capture/encode DVTB parameter	***30***

Table 42. auddec1 Class Configuration Parameters

Class	Parameter	Description	Values
auddec1	codec	Name of audio decoder	***aachedec*** wmadec mp3dec
	outputPCMWidth	Number of bits per output PCM Sample	***16***
	pcmFormat	Output PCM format	0 : IAUDIO_BLOCK Left channel data is followed by right channel ***1*** : IAUDIO_INTERLEAVED Left and right channel data interleaved
	dataEndianness	Endianness of input data	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16

			16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	desiredChannelMode	Desired channel configuration	0: IAUDIO_1_0 MONO ***1***: IAUDIO_2_0 Stereo 2: IAUDIO_11_0 Dual Mono …….
	downSampleSbrFlag	Flag to indicate down sampling for SBR	***0***
	inbufsize	Inbuf size	***25600***
	outbufsize	Outbuf size	***16384***
	pwrManagerEnable	Power Manager On/Off	***0***: Off 1 : ON

Table 43. mp3basedec1 Class Configuration Parameters

Class	Parameter	Description	Values
mp3basedec1	codec	Name of audio decoder	***mp3dec***
	outputPCMWidth	Number of bits per output PCM Sample	***16***
	pcmFormat	Output PCM format	0 : IAUDIO_BLOCK Left channel data is followed by right channel ***1*** : IAUDIO_INTERLEAVED Left and right channel data interleaved
	dataEndianness	Endianness of input data	***1*** : XDM_BYTE

			Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	downSampleSbrFlag	Flag to indicate down sampling for SBR	***0***
	Inbufsize	Inbuf size	***25600***
	Outbufsize	Outbuf size	***16384***
	pwrManagerEnable	Power Manager On/Off	***0***: Off 1 : ON

Table 44. jpegdec1 Class Configuration Parameters

Class	Parameter	Description	Values
jpegdec1	codec	Name of image decoder	***jpegdec***
	maxHeight	Maximum height in pixels	***576***
	maxWidth	Maximum width in pixels	***720***
	maxScans	Maximum number of scans for progressive mode	***15***
	dataEndianness	Endianness of input data	***1***
	forceChromaFormat	Force decoding in the given chroma format	***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved 0: default
	progressiveDecFlag	Flag for progressive Decoding	***1***: Progressive decoding ***0***: sequential mode
	outImgRes	Output image resolution	***0***: Even Image resolution 1: Actual Image resolution
	numAU	Number of access unit	***0***
	decodeHeader	Set to 0 to decode complete AU	***0***

	displayWidth	Set to 0 decode entire width of the image	***0***
	progDisplay	Dispplay option for progressive mode	***0***: output buffer contains the decoded image only after all the scans are decoded 1: output buffer contains the partially(progressive) decoded image after each scan is decoded
	resizeOption	Resize output image	***0***: No resizing 1: Resize output image by 1/2 2: Resize output image by 1/4 3: Resize output image by 1/8 4: Up-scale output image by 2 5: Up-scale output image by 4 6: Up-scale output image by 8
	RGB_Format	Output RGB Format	***0***: BGR24 1: BGR32 2: RGB16
	numMCU_row	Number of Rows of access units to decode	***0***
	x_org	Start point of the X-axis of the subregion	***0***
	y_org	Start point of the Y-axis of the subregion	***0***
	x_length	X-length of subregion	***0***
	y_length	Y-length of subregion	***0***
	alpha_rgb	Alpha value to fill rgb32	***0***

Table 45. imgdec1 Class Configuration Parameters

Class	Parameter	Description	Values
imgdec1	codec	Name of image decoder	***jpegdec***
	maxHeight	Maximum height in pixels	***576***
	maxWidth	Maximum width in pixels	***720***
	maxScans	Maximum number of scans for progressive mode	***1***
	dataEndianness	Endianness of input data	***1***
	forceChromaFormat	Force decoding in the given chroma format	***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved 0 : default
	numAU	Number of access unit	***0***
	decodeHeader	Set to 0 to decode complete AU	***0***
	displayWidth	Set to 0 decode entire width of the image	***0***

Table 46. imgenc1 Class Configuration Parameters

Class	Parameter	Description	Values
imgenc1	codec	Name of image encoder	***jpegenc***
	maxHeight	Maximum height in pixels	***576***
	maxWidth	Maximum width in pixels	***720***
	maxScans	Maximum number of scans for progressive mode	***0***: XDM_DEFAULT
	dataEndianness	Endianness of input data	***1***: XDM_BYTE
	forceChromaFormat	Output color format for encoder	1 : XDM_YUV_420P YUV 4:2:0 Planer ***2***: XDM_YUV_422P YUV 4:2:2 Planer

			4 : XDM_YUV_422ILE YUV 4:2:2 interleaved …….
	numAU	Number of access unit	***0***
	inputChromaFormat	Color format of the input data	1 : XDM_YUV_420P YUV 4:2:0 Planer 2: XDM_YUV_422P YUV 4:2:2 Planer ***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved …….
	inputHeight	Input height of the image	***480***
	inputWidth	Input width of the image	***720***
	captureWidth	Total buffer width of the input data	***720***
	generateHeader		***0*** : XDM_ENCODE_AU
	qValue	Q value of encoder	***73***

Table 47. sphdec1 Class Configuration Parameters

Class	Parameter	Description	Values
sphdec1	codec	Name of speech decoder	***g711dec***
	compandingLaw	Companding Law	0 : Linear ***1*** : A-Law 2 : U-Law
	packingType	Packing Format Option	0 ***1*** 2
	codecSelection	Codec Selection	***0***
	bitRate	Bit Rate	***12000***
	inbufsize	Size of the input buffer	***80***
	outbufsize	Size of the output buffer	***320***

Table 48. sphenc1 Class Configuration Parameters

Class	Parameter	Description	Values
sphenc1	codec	Name of speech encoder	***g711enc***
	seconds	Number of seconds to capture DVTB parameter	***10*** …
	frameSize	Input Frame Size in bytes	***80***
	compandingLaw	Companding Law	0 : Linear ***1*** : A-Law 2 : U-Law
	packingType	Packing Format Option	***0*** 1 2
	vadSelection	VAD selection	***0***
	codecSelection	Codec Selection	***0***
	bitRate	Bit Rate	***12000***
	inbufsize	Size of the input buffer	***160***
	outbufsize	Size of the output buffer	***80***

Table 49. De Interlacer Class Configuration Parameters

Class	Parameter	Description	Values
deinter	codec	Name of deinterlacer	***deinterlacer***
	deinterEnable	Deinterlacer State	***0:*** Not in Use 1: In Use

Table 50. pwrmanager Class Configuration Parameters

Class	Parameter	Description	Values
pwrmanager	scalingGovernor	Decides the target C-State of the system	***performance***: ondemand:
	vdd1PState	VDD1 P-States	1: OPP1 2: OPP2 3: OPP3 ***4***: OPP4 5: OPP5
	vdd2PState	VDD2 P-States	2: OPP2 ***3***: OPP3
	cpuIdleState	C-states	0: C0 ***1***: C1 2: C2 3: C3 4: C4 5: C5 6: C6

Note: Refer to PSP User Guide for details on Governors, P-States and C-states

DVTB classes supporting extended parameters are listed below:

“jpegdec1” – JPEG Decoder Extended parameters

“h264enc1” – H264 Encoder Extended parameters

“h264dec2” – H264 Decoder Extended parameters

“mpeg4spenc1” – MPEG4 Encoder Extended parameters

“mpeg4spdec2” – MPEG4 Decoder Extended parameters

“mpeg2dec2” – MPEG4 Decoder Extended parameters

“aachedec1” – AAC Decoder Decoder Extended parameters

For a list of the parameters and their description refer the respective Codec’s User Guide.

Note:

1) DVTB does not support following parameters

Handle	Parameters
videnc1	forceFrame
viddec2	newFrameFlag, mbDataFlag
jpegdec1	progDisplay
h264enc1	forceFrame, mvDataEnable, streamFormat
mpeg4spenc1	forceFrame, mvDataEnable, ResyncDataEnable
mpeg2dec2	dyna_chroma_format, getDisplayHdrInfo, reverse_play
mpeg4spdec2	newFrameFlag, mbDataFlag
h264dec2	Sei_Vui_parse_flag, numNALunits

2) Mpeg4spenc1 handler: Name of "numFrames" parameter required by VM4 rate control algorithm is VM4RCnumFrames.

3) nodsp option is not supported with Extended parameter handlers

4) DeInterlacer can be used only with Video Encoder Handlers.

The codec parameters and the codec dynamic parameters are defined in the following structures defined by the VISA APIs:

- *VIDDEC2_Params, VIDDEC2_DynamicParams*
- *VIDENC1_Params, VIDENC1_DynamicParams*
- *IMGDEC1_Params, IMGDEC1_DynamicParams*
- *IMGENC1_Params, IMGENC1_DynamicParams*
- *AUDDEC1_Params, AUDDEC1_DynamicParams*
- SPHENC1_Params, SPHENC1_DynamicParams
- SPHDEC1_Params, SPHDEC1_DynamicParams

For more information about the parameters supported by Codec Engine VISA API, the reader is referred to the *Codec Engine API Reference* included in $(DVSDK_INSTALL_DIR)`/codec_engine_x_xx_xx_xx/docs/html/index.html`

For more information on the supported values for individual codecs, the reader is referred to the Codec data sheets and user guides included in
$(CODEC_INSTALL_DIR)`/packages/ti/sdo/codecs/<codec>/docs`

For more information on the supported values for audio and video peripherals, please refer to the LSP documentation included in
<LSP Installation Dir>`/ti-davinci/Documentation`

2.4.2.14 DM3730(Linux)

Table 51. viddec2 Class Configuration Parameters

Class	Parameter	Description	Values
viddec2	Codec	Name of video decoder	***mpeg2dec*** h264dec mpeg4dec
	maxHeight	Maximum video height in pixels	***576***
	maxWidth	Maximum video width in pixels	***720***
	maxFrameRate	Maximum frame rate in fps*1000	***30000***
	maxBitRate	Maximum bit rate in bits per second	***10000000***
	dataEndianness	Endianness of data input	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	forceChromaFormat	Force decode in given chroma format *Other YUV/RGB formats are not supported by video decoders*	1 : XDM_YUV_420P YUV 4:2:0 planar ***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved
	decodeHeader	Number of access units to decode Dynamic Parameter	***0*** : XDM_DECODE_AU Decode access unit 1 : XDM_PARSE_HEADER Decode only header
	displayWidth	Pitch Dynamic Parameter	***0***
	frameSkipMode	Frame Skip Mode	***0*** : IVIDEO_NO_SKIP

		Dynamic Parameter *Only IVIDEO_NO_SKIP supported by video decoders*	Don't skip current frame
	frameOrder	Frame display order	***0***
	newFrameFlag	Flag to indicate that algorithm should start a new frame.	***0***
	mbDataFlag	Generate MB Data	***0***
	numFrames	Number of frames to decode	***100***
	pwrManagerEnable	Power Manager On/Off	***0***: Off 1 : ON

Table 52. videnc1 Class Configuration Parameters

Class	Parameter	Description	Values
videnc1	codec	Name of video encoder	***h264enc*** mpeg4enc
	encodingPreset	Encoding Present	***0***
	rateControlPreset	Rate control preset	***1*** : IVIDEO_LOW_DELAY Stringent CBR 2 : IVIDEO_STORAGE Constrained VBR 3 : IVIDEO_TWOPASS Two pass rate control 4 : IVIDEO_NONE Unconstrained VBR
	maxHeight	Maximum Height	***576*** 480 …
	maxWidth	Maximum Width	***720***

			…
	maxFrameRate	Maximum Frame Rate if fps *1000	***30000***
	maxBitRate	Maximum Bit Rate	***2000000***
	dataEndianness	Endianness of data output	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	maxInterFrameInterval	I to Pframe distance	***1***
	inputChromaFormat	Input Chroma Format	***1*** : XDM_YUV_420P YUV 4:2:0 planar 4 : XDM_YUV_422ILE YUV 4:2:2 interleaved
	inputContentType	Type of input video content	***0*** : IVIDEO_PROGRESSIVE Progressive video content 1 : IVIDEO_INTERLACED Interlaced video content
	reconChromaFormat	Chroma format for the reconstruction buffers	***-1***:XDM_CHROMA_NA
	inputHeight	Input Frame Height Dynamic Parameter	576 ***480*** …
	inputWidth	Input Frame Width Dynamic Parameter	***720*** …
	refFrameRate	Reference or input frame rate in fps * 1000 Dynamic Parameter	***30000*** …

	targetFrameRate	Target Frame Rate in fps * 1000 Dynamic Parameter	***30000*** …
	targetBitRate	Target Bit Rate Dynamic Parameter	***2000000***
	intraFrameInterval	I frame interval Dynamic Parameter	***30***
	generateHeader	Mode of encode Dynamic Parameter	***0*** : XDM_ENCODE_AU Encode access unit 1:XDM_GENERATE_HEADER Generate only header
	captureWidth	Pitch Dynamic Parameter	***0***
	forceFrame	Force the encoded frames to be I frames Dynamic Parameter	***0***
	interFrameInterval	Number of B frames between two reference frames	***1***
	mbDataFlag	Flag to indicate that the algorithm should use MB data provided in additional buffer with inBufs	***0***
	framePitch	Frame pitch to store the frames	***720***
	numFrames	Number of frames to capture/encode DVTB parameter	***30***

Table 53. auddec1 Class Configuration Parameters

Class	Parameter	Description	Values
auddec1	codec	Name of audio decoder	***aachedec*** wmadec

			mp3dec
	outputPCMWidth	Number of bits per output PCM Sample	***16***
	pcmFormat	Output PCM format	0 : IAUDIO_BLOCK Left channel data is followed by right channel ***1*** : IAUDIO_INTERLEAVED Left and right channel data interleaved
	dataEndianness	Endianness of input data	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	desiredChannelMode	Desired channel configuration	0: IAUDIO_1_0 MONO ***1***: IAUDIO_2_0 Stereo 2: IAUDIO_11_0 Dual Mono …….
	downSampleSbrFlag	Flag to indicate down sampling for SBR	***0***
	inbufsize	Inbuf size	***25600***
	outbufsize	Outbuf size	***16384***
	pwrManagerEnable	Power Manager On/Off	***0***: Off 1 : ON

Table 54. mp3basedec1 Class Configuration Parameters

Class	Parameter	Description	Values
mp3basedec1	codec	Name of audio decoder	***mp3dec***
	outputPCMWidth	Number of bits per output PCM Sample	***16***
	pcmFormat	Output PCM format	0 : IAUDIO_BLOCK Left channel data is followed by right channel ***1*** : IAUDIO_INTERLEAVED Left and right channel data interleaved
	dataEndianness	Endianness of input data	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	downSampleSbrFlag	Flag to indicate down sampling for SBR	***0***
	Inbufsize	Inbuf size	***25600***
	Outbufsize	Outbuf size	***16384***
	pwrManagerEnable	Power Manager On/Off	***0***: Off 1 : ON

Table 55. jpegdec1 Class Configuration Parameters

Class	Parameter	Description	Values
jpegdec1	codec	Name of image decoder	***jpegdec***
	maxHeight	Maximum height in pixels	***576***
	maxWidth	Maximum width in pixels	***720***

	maxScans	Maximum number of scans for progressive mode	***15***
	dataEndianness	Endianness of input data	***1***
	forceChromaFormat	Force decoding in the given chroma format	***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved 0: default
	progressiveDecFlag	Flag for progressive Decoding	***1***: Progressive decoding ***0***: sequential mode
	outImgRes	Output image resolution	***0***: Even Image resolution 1: Actual Image resolution
	numAU	Number of access unit	***0***
	decodeHeader	Set to 0 to decode complete AU	***0***
	displayWidth	Set to 0 decode entire width of the image	***0***
	progDisplay	Dispplay option for progressive mode	***0***: output buffer contains the decoded image only after all the scans are decoded 1: output buffer contains the partially(progressive) decoded image after each scan is decoded
	resizeOption	Resize output image	***0***: No resizing 1: Resize output image by 1/2 2: Resize output image by 1/4 3: Resize output image by 1/8 4: Up-scale output image by 2 5: Up-scale output image by 4 6: Up-scale output image by 8
	RGB_Format	Output RGB Format	***0***: BGR24 1: BGR32 2: RGB16

	numMCU_row	Number of Rows of access units to decode	***0***
	x_org	Start point of the X-axis of the subregion	***0***
	y_org	Start point of the Y-axis of the subregion	***0***
	x_length	X-length of subregion	***0***
	y_length	Y-length of subregion	***0***
	alpha_rgb	Alpha value to fill rgb32	***0***

Table 56. imgdec1 Class Configuration Parameters

Class	Parameter	Description	Values
imgdec1	codec	Name of image decoder	***jpegdec***
	maxHeight	Maximum height in pixels	***576***
	maxWidth	Maximum width in pixels	***720***
	maxScans	Maximum number of scans for progressive mode	***1***
	dataEndianness	Endianness of input data	***1***
	forceChromaFormat	Force decoding in the given chroma format	***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved 0 : default
	numAU	Number of access unit	***0***
	decodeHeader	Set to 0 to decode complete AU	***0***
	displayWidth	Set to 0 decode entire width of the image	***0***

Table 57. imgenc1 Class Configuration Parameters

Class	Parameter	Description	Values
imgenc1	codec	Name of image encoder	***jpegenc***
	maxHeight	Maximum height in pixels	***576***
	maxWidth	Maximum width in pixels	***720***
	maxScans	Maximum number of scans for progressive mode	***0***: XDM_DEFAULT
	dataEndianness	Endianness of input data	***1***: XDM_BYTE
	forceChromaFormat	Output color format for encoder	1 : XDM_YUV_420P YUV 4:2:0 Planer ***2***: XDM_YUV_422P YUV 4:2:2 Planer 4 : XDM_YUV_422ILE YUV 4:2:2 interleaved …….
	numAU	Number of access unit	***0***
	inputChromaFormat	Color format of the input data	1 : XDM_YUV_420P YUV 4:2:0 Planer 2: XDM_YUV_422P YUV 4:2:2 Planer ***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved …….
	inputHeight	Input height of the image	***480***
	inputWidth	Input width of the image	***720***
	captureWidth	Total buffer width of the input data	***720***
	generateHeader		***0*** : XDM_ENCODE_AU
	qValue	Q value of encoder	***73***

Table 58. sphdec1 Class Configuration Parameters

Class	Parameter	Description	Values
sphdec1	codec	Name of speech decoder	***g711dec***
	compandingLaw	Companding Law	0 : Linear ***1*** : A-Law 2 : U-Law
	packingType	Packing Format Option	0 ***1*** 2
	codecSelection	Codec Selection	***0***
	bitRate	Bit Rate	***12000***
	inbufsize	Size of the input buffer	***80***
	outbufsize	Size of the output buffer	***320***

Table 59. sphenc1 Class Configuration Parameters

Class	Parameter	Description	Values
sphenc1	codec	Name of speech encoder	***g711enc***
	seconds	Number of seconds to capture DVTB parameter	***10*** …
	frameSize	Input Frame Size in bytes	***80***
	compandingLaw	Companding Law	0 : Linear ***1*** : A-Law 2 : U-Law
	packingType	Packing Format Option	***0*** 1 2
	vadSelection	VAD selection	***0***

	codecSelection	Codec Selection	***0***
	bitRate	Bit Rate	***12000***
	inbufsize	Size of the input buffer	***160***
	outbufsize	Size of the output buffer	***80***

Table 60. De Interlacer Class Configuration Parameters

Class	Parameter	Description	Values
deinter	codec	Name of deinterlacer	***deinterlacer***
	deinterEnable	Deinterlacer State	***0:*** Not in Use 1: In Use

Table 61. pwrmanager Class Configuration Parameters

Class	Parameter	Description	Values
pwrmanager	scalingGovernor	Decides the target C-State of the system	***performance***: ondemand:
	vdd1PState	VDD1 P-States	1: OPP1 2: OPP2 3: OPP3 ***4***: OPP4 5: OPP5
	vdd2PState	VDD2 P-States	2: OPP2 ***3***: OPP3
	cpuIdleState	C-states	0: C0 ***1***: C1 2: C2 3: C3

			4: C4 5: C5 6: C6

Note: Refer to PSP User Guide for details on Governors, P-States and C-states

DVTB classes supporting extended parameters are listed below:

“jpegdec1” – JPEG Decoder Extended parameters

“h264enc1” – H264 Encoder Extended parameters

“h264dec2” – H264 Decoder Extended parameters

“mpeg4spenc1” – MPEG4 Encoder Extended parameters

“mpeg4spdec2” – MPEG4 Decoder Extended parameters

“mpeg2dec2” – MPEG4 Decoder Extended parameters

“aachedec1” – AAC Decoder Decoder Extended parameters

For a list of the parameters and their description refer the respective Codec’s User Guide.

Note:

1) DVTB does not support following parameters

Handle	Parameters
videnc1	forceFrame
viddec2	newFrameFlag, mbDataFlag
jpegdec1	progDisplay
h264enc1	forceFrame, mvDataEnable, streamFormat
mpeg4spenc1	forceFrame, mvDataEnable
mpeg2dec2	dyna_chroma_format, getDisplayHdrInfo, reverse_play
mpeg4spdec2	newFrameFlag, mbDataFlag

2) nodsp option is not supported with Extended parameter handlers

3) DeInterlacer can be used only with Video Encoder Handlers.

The codec parameters and the codec dynamic parameters are defined in the following structures defined by the VISA APIs:

- *VIDDEC2_Params, VIDDEC2_DynamicParams*
- *VIDENC1_Params, VIDENC1_DynamicParams*
- *IMGDEC1_Params, IMGDEC1_DynamicParams*
- *IMGENC1_Params, IMGENC1_DynamicParams*

- *AUDDEC1_Params, AUDDEC1_DynamicParams*
- SPHENC1_Params, SPHENC1_DynamicParams
- SPHDEC1_Params, SPHDEC1_DynamicParams

For more information about the parameters supported by Codec Engine VISA API, the reader is referred to the *Codec Engine API Reference* included in $(DVSDK_INSTALL_DIR)/codec_engine_x_xx_xx_xx/docs/html/index.html

For more information on the supported values for individual codecs, the reader is referred to the Codec data sheets and user guides included in **$(CODEC_INSTALL_DIR)**/packages/ti/sdo/codecs/<codec>/docs

For more information on the supported values for audio and video peripherals, please refer to the LSP documentation included in **<LSP Installation Dir>**/ti-davinci/Documentation

2.4.2.15 OMA3530(WinCE)

Table 62. Engine Configuration Parameters

Class	Parameter	Description	Values
engine	name	Name of codec server to be used	***encodedecode***

Table 63. viddec2 Class Configuration Parameters

Class	Parameter	Description	Values
viddec2	Codec	Name of video decoder	***mpeg2dec*** h264dec mpeg4dec
	maxHeight	Maximum video height in pixels	***576***
	maxWidth	Maximum video width in pixels	***720***
	maxFrameRate	Maximum frame rate in fps*1000	***30000***
	maxBitRate	Maximum bit rate in bits per second	***10000000***

	dataEndianness	Endianness of data input	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	forceChromaFormat	Force decode in given chroma format *Other YUV/RGB formats are not supported by video decoders*	1 : XDM_YUV_420P YUV 4:2:0 planar ***4*** : XDM_YUV_422ILE YUV 4:2:2 interleaved
	decodeHeader	Number of access units to decode Dynamic Parameter	***0*** : XDM_DECODE_AU Decode access unit 1 : XDM_PARSE_HEADER Decode only header
	displayWidth	Pitch Dynamic Parameter	***0***
	frameSkipMode	Frame Skip Mode Dynamic Parameter *Only IVIDEO_NO_SKIP supported by video decoders*	***0*** : IVIDEO_NO_SKIP Don't skip current frame
	frameOrder	Frame display order	***0***
	newFrameFlag	Flag to indicate that algorithm should start a new frame.	***0***
	mbDataFlag	Generate MB Data	***0***
	numFrames	Number of frames to decode	***100***
	pwrManagerEnable	Power Manager On/Off	***0***: Off 1 : ON

Table 64. videnc1 Class Configuration Parameters

Class	Parameter	Description	Values
videnc1	codec	Name of video encoder	***h264enc*** mpeg4enc
	encodingPreset	Encoding Present	***0***
	rateControlPreset	Rate control preset	***1*** : IVIDEO_LOW_DELAY Stringent CBR 2 : IVIDEO_STORAGE Constrained VBR 3 : IVIDEO_TWOPASS Two pass rate control 4 : IVIDEO_NONE Unconstrained VBR
	maxHeight	Maximum Height	***576*** 480 …
	maxWidth	Maximum Width	***720*** …
	maxFrameRate	Maximum Frame Rate if fps *1000	***30000***
	maxBitRate	Maximum Bit Rate	***2000000***
	dataEndianness	Endianness of data output	***1*** : XDM_BYTE Big Endian stream 2 : XDM_LE_16 16 Little Endian stream 3 : XDM_LE_32 32 Little Endian stream
	maxInterFrameInterval	I to Pframe distance	***1***

	inputChromaFormat	Input Chroma Format	***1*** : XDM_YUV_420P YUV 4:2:0 planar 4 : XDM_YUV_422ILE YUV 4:2:2 interleaved
	inputContentType	Type of input video content	***0*** : IVIDEO_PROGRESSIVE Progressive video content 1 : IVIDEO_INTERLACED Interlaced video content
	reconChromaFormat	Chroma format for the reconstruction buffers	***-1***:XDM_CHROMA_NA
	inputHeight	Input Frame Height Dynamic Parameter	576 ***480*** …
	inputWidth	Input Frame Width Dynamic Parameter	***720*** …
	refFrameRate	Reference or input frame rate in fps * 1000 Dynamic Parameter	***30000*** …
	targetFrameRate	Target Frame Rate in fps * 1000 Dynamic Parameter	***30000*** …
	targetBitRate	Target Bit Rate Dynamic Parameter	***2000000***
	intraFrameInterval	I frame interval Dynamic Parameter	***30***
	generateHeader	Mode of encode Dynamic Parameter	***0*** : XDM_ENCODE_AU Encode access unit 1:XDM_GENERATE_HEADER Generate only header
	captureWidth	Pitch	***0***

		Dynamic Parameter	
	forceFrame	Force the encoded frames to be I frames Dynamic Parameter	***0***
	interFrameInterval	Number of B frames between two reference frames	***1***
	mbDataFlag	Flag to indicate that the algorithm should use MB data provided in additional buffer with inBufs	***0***
	framePitch	Frame pitch to store the frames	***720***
	numFrames	Number of frames to capture/encode DVTB parameter	***30***

The codec parameters and the codec dynamic parameters are defined in the following structures defined by the VISA APIs:

- *VIDDEC2_Params, VIDDEC2_DynamicParams*
- *VIDENC1_Params, VIDENC1_DynamicParams*

For more information about the parameters supported by Codec Engine VISA API, the reader is referred to the *Codec Engine API Reference* included in $(DVSDK_INSTALL_DIR)`/codec_engine_x_xx_xx_xx/docs/html/index.html`

For more information on the supported values for individual codecs, the reader is referred to the Codec data sheets and user guides included in $(CODEC_INSTALL_DIR)/`packages/ti/sdo/codecs/<codec>/docs`

2.4.3 DVTB scripting

The DVTB can be used in command line mode or script mode. In command line mode, all the commands are typed and executed one by one. In script mode, all the commands are included in a script file which is executed with the –s <script_file> option.

Example:

In order to execute the script “myscript.dvs” with the dvtb, the following needs to be typed.

Linux:

```
root@158.218.33.135:/opt/dvsdk#  ./dvtb-r –s myscript.dvs
```

Sample scripts are provided in the `dvtb_x_xx_xxx/dvtb-scripts` directory.

WinCE:

```
\Temp>dvtb.exe –s myscript.dvs
```

Sample scripts are provided in the `dvtb_x_xx_xxx/ packages/ti/sdo/dvtb/scripts` directory.

Note: Wince does not support current directory logic. User has to provide the full path of all the files. If there are spaces in the path then provide them in inverted comma like

```
\Temp>dvtb.exe –s "\Storage Card\myscript.dvs"
```

2.4.4 DVTB Debug Mode

Linux

Two versions of DVTB are packaged with DVSDK – “dvtb-r” and “dvtb-d” corresponding to release version and debug version of DVTB respectively.

Usage and functionality of both the versions is exactly the same. The only difference in debug version is that all debug messages related to the data flow being executed is printed on the console. This will be helpful in analyzing a failed data flow scenario. With the debug messages, the user can track down the failure to a particular component in the system [drivers, codecs, codec servers, etc] which is causing the failure.

WinCE

Two versions of DVTB are packaged with DVSDK – “dvtb.exe” and “dvtb-d.exe” corresponding to release version and debug version of DVTB respectively.

Usage and functionality of both the versions is exactly the same. The only difference in debug version is that all debug messages related to the data flow being executed is printed on the console. This will be helpful in analyzing a failed data flow scenario. With the debug messages, the user can track down the failure to a particular component in the system [drivers, codecs, codec servers, etc] which is causing the failure.

2.4.5 Rebuilding DVTB for different codec servers

The DVTB can be rebuilt to suit the needs of different codec servers. The following steps are needed to rebuild the DVTB for a different codec server that is not a part of the DVSDK codec servers:

DM6446(Linux)

- Go to $(DVSDK_INSTALL_DIR)/dvtb_x_xx_xx on host machine where DVSDK is installed
- Modify dvtb_dm6446.cfg in "dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm6446/" as per the new codec server. Please refer to Codec Engine Application Developer User's Guide on information regarding the engine configuration file.
- Rebuild dvtb

```
$ make clean ; make dm6446
```

- Release (dvtb-r) and Debug (dvtb-d) binaries will be created under the following folder

```
$(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm6446/bin
```

- Copy the binaries dvtb-r and dvtb-d to your execution environment

DM355(Linux)

- Go to $(DVSDK_INSTALL_DIR)/dvtb_x_xx_xx on host machine where DVSDK is installed
- Modify dvtb_dm355.cfg in "dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm355/" as per the new codec server. Please refer to Codec Engine Application Developer User's Guide on information regarding the engine configuration file.
- Rebuild dvtb

```
$ make clean ; make dm355
```

- Release (dvtb-r) and Debug (dvtb-d) binaries will be created under the following folder

```
$(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm355/bin
```

- Copy the binaries dvtb-r and dvtb-d to your execution environment

DM6467(Linux)

- Go to $(DVSDK_INSTALL_DIR)/dvtb_x_xx_xxx on host machine where DVSDK is installed
- Modify dm6467.cfg in “dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm6467/” as per the new codec server. Please refer to Codec Engine Application Developer User’s Guide on information regarding the engine configuration file.
- Rebuild dvtb

  For TSPA codecs with h264 decoder

  ```
  $ make clean ; make dm6467 CODECS=TSPA H264CodecPackage=h264dec
  ```

  For TSPA codecs with h264 1080p decoder

  ```
  $ make clean ; make dm6467 CODECS=TSPA H264CodecPackage=
h2641080p60vdec
  ```

  For NONTSPA codecs with h264 decoder

  ```
  $ make clean ; make dm6467 CODECS=NONTSPA H264CodecPackage=h264dec
  ```

  For NONTSPA codecs with h264 1080p decoder

  ```
  $ make clean ; make dm6467 CODECS=NONTSPA H264CodecPackage=
h2641080p60vdec
  ```

- Release (dvtb-r) and Debug (dvtb-d) binaries will be created under the following folder

  ```
  $(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm6467/bin
  ```

- Copy the binaries dvtb-r and dvtb-d to your execution environment

DM365(Linux)

- Go to $(DVSDK_INSTALL_DIR)/dvtb_x_xx_xx on host machine where DVSDK is installed
- Modify dvtb_dm365.cfg in “dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm365/” as per the new codec server. Please refer to Codec Engine Application Developer User’s Guide on information regarding the engine configuration file.
- Rebuild dvtb

  For TSPA codecs

  ```
  $ make clean ; make dm365 CODECS=TSPA
  ```

  For NONTSPA codecs

  ```
  $ make clean ; make dm365 CODECS=NONTSPA
  ```

- Release (dvtb-r) and Debug (dvtb-d) binaries will be created under the following folder

```
$(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm365/bin
```

- Copy the binaries dvtb-r and dvtb-d to your execution environment

OMAP3530(Linux)

- Go to $(DVSDK_INSTALL_DIR)/dvtb_x_xx_xx on host machine where DVSDK is installed
- Modify dvtb_omap3530.cfg in “dvtb_x_xx_xx/packages/ti/sdo/dvtb/omap3530/linux” as per the new codec server. Please refer to Codec Engine Application Developer User’s Guide on information regarding the engine configuration file.
- Rebuild dvtb
- $ make clean ; make omap3530
- Release (dvtb-r) and Debug (dvtb-d) binaries will be created under the following folder

```
$(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx/packages/ti/sdo/dvtb/omap3530/linux/bin
```

- Copy the binaries dvtb-r and dvtb-d to your execution environment

DM3730(Linux)

- Go to $(DVSDK_INSTALL_DIR)/dvtb_x_xx_xx on host machine where DVSDK is installed
- Modify dvtb_dm3730.cfg in “dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm3730/linux” as per the new codec server. Please refer to Codec Engine Application Developer User’s Guide on information regarding the engine configuration file.
- Rebuild dvtb
- $ make clean ; make dm3730
- Release (dvtb-r) and Debug (dvtb-d) binaries will be created under the following folder

```
$(DVSDK_INSTALL_ROOT)/dvtb_x_xx_xx/packages/ti/sdo/dvtb/dm3730/linux/bin
```

- Copy the binaries dvtb-r and dvtb-d to your execution environment

OMAP3530(WinCE)

- Go to $(DVSDK_INSTALL_DIR)/dvtb_x_xx_xx on host machine where DVSDK is installed

- Modify dvtb_omap3530.cfg in “dvtb_x_xx_xx/packages/ti/sdo/dvtb/omap3530/wince” as per the new codec server. Please refer to Codec Engine Application Developer User’s Guide on information regarding the engine configuration file.
- To Rebuild dvtb: Refer section 2.2 (Installing DVTB)

3 DVTB Use Cases

This section describes the most common DVTB use cases. Some of these use cases are provided in the dvtb_x_xx_xx/ packages/ti/sdo/dvtb/scripts directory.

NOTE: For DM3730 user can use the scripts available for omap3530.

3.1 DM6446(Linux) Usecases

3.1.1 Video Usecases

3.1.1.1 Video Capture and Playback

3.1.1.2 Video Capture and Encode

- *Description*

  Capture video, encode using MPEG4 encoder and store to a file.

- *Implementation*

– Input: Video stream input through composite video input.

– Output: File “mp4-encode.mpeg4” stored on the Hard Disk / NFS

– Sample script:

```
setp engine name encode
setp videnc1 codec mpeg4enc
setp videnc1 encodingPreset 0
setp videnc1 rateControlPreset 1
setp videnc1 maxHeight 576
setp videnc1 maxWidth 720
setp videnc1 maxFrameRate 30000
setp videnc1 maxBitRate 2000000
setp videnc1 dataEndianness 1
setp videnc1 maxInterFrameInterval 1
setp videnc1 inputChromaFormat 4
setp videnc1 inputContentType 0
setp videnc1 reconChromaFormat -1
setp videnc1 inputHeight 240
setp videnc1 inputWidth 320
setp videnc1 refFrameRate 30000
```

```
setp videnc1 targetFrameRate 30000
setp videnc1 targetBitRate 2000000
setp videnc1 intraFrameInterval 30
setp videnc1 generateHeader 0
setp videnc1 captureWidth 320
setp videnc1 forceFrame 0
setp videnc1 interFrameInterval 1
setp videnc1 mbDataFlag 0
setp videnc1 framePitch 720
setp videnc1 numFrames 300
setp vpfe device /dev/v4l/video0
setp vpfe standard 0
setp vpfe format 1
setp vpfe input 0
setp vpfe width 320
setp vpfe height 240
func videnc1 -t mp4-encode.mpeg4
```

- *Notes*

– Encoder: all available encoders can be selected.

– Options: all encoder options can be selected.

3.1.1.3 Video Decode and Playback

- *Description*

Read an encoded file, decode and display.

- *Implementation*

– Input: Encoded File

– Output: Video Stream displayed through VPBE

– Sample script:

```
setp engine name decode
setp viddec2 codec h264dec
setp viddec2 maxWidth 720
setp viddec2 maxHeight 480
setp viddec2 maxFrameRate 30000
setp viddec2 maxBitRate 10000000
setp viddec2 dataEndianness 1
setp viddec2 forceChromaFormat 4
setp viddec2 decodeHeader 0
setp viddec2 displayWidth 0
setp viddec2 frameSkipMode 0
```

```
setp viddec2 frameOrder 0
setp viddec2 newFrameFlag 0
setp viddec2 mbDataFlag 0
setp viddec2 numFrames 100

setp vpbe device /dev/video2
setp vpbe width 720
setp vpbe height 480
setp vpbe std 1
setp vpbe output 1
func viddec2 -s ./data/videos/davincieffect_ntsc.264
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.1.1.4 Video File Encode and Decode

3.1.1.5 Video Encode from File

- *Description*

Read a YUV 422 interleaved file, encode using a video encoder, store encoded file.

- *Implementation*

– Input: YUV 422 interleaved file.

– Output: Encoded file.

– Sample script:

```
setp engine name encode
setp videnc1 codec mpeg4enc
setp videnc1 encodingPreset 0
setp videnc1 rateControlPreset 1
setp videnc1 maxHeight 576
setp videnc1 maxWidth 720
setp videnc1 maxFrameRate 30000
setp videnc1 maxBitRate 2000000
setp videnc1 dataEndianness 1
setp videnc1 maxInterFrameInterval 1
setp videnc1 inputChromaFormat 4
setp videnc1 inputContentType 0
setp videnc1 reconChromaFormat -1
setp videnc1 inputHeight 240
```

```
setp videnc1 inputWidth 320
setp videnc1 refFrameRate 30000
setp videnc1 targetFrameRate 30000
setp videnc1 targetBitRate 2000000
setp videnc1 intraFrameInterval 30
setp videnc1 generateHeader 0
setp videnc1 captureWidth 320
setp videnc1 forceFrame 0
setp videnc1 interFrameInterval 1
setp videnc1 mbDataFlag 0
setp videnc1 framePitch 720
setp videnc1 numFrames 300
func videnc1 –s input_320x240_422ile.yuv -t mp4-encode.mpeg4
```

- *Notes*

– Encoder: all available encoders can be selected.

– Options: all encoder options can be selected.

3.1.1.6 Video Decode to File

- *Description*

Read an encoded file, decode using a video decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: YUV 422 interleaved file.

– Sample script:

```
setp engine name decode
setp viddec2 codec h264dec
setp viddec2 maxWidth 720
setp viddec2 maxHeight 480
setp viddec2 maxFrameRate 30000
setp viddec2 maxBitRate 10000000
setp viddec2 dataEndianness 1
setp viddec2 forceChromaFormat 4
setp viddec2 decodeHeader 0
setp viddec2 displayWidth 0
setp viddec2 frameSkipMode 0
setp viddec2 frameOrder 0
setp viddec2 newFrameFlag 0
setp viddec2 mbDataFlag 0
setp viddec2 numFrames 100
```

```
func viddec2 -s ./data/videos/davincieffect_ntsc.264 -t out.yuv
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.1.2 Audio Usecases

3.1.2.1 Audio Playback

3.1.2.2 Audio Decode and Playback

- *Description*

Read an encoded file and play it out through the audio driver.

- *Implementation*

– Input: Audio encoded file.

– Output: Audio stream played out on line-out/headphone

– Sample script:

```
setp engine name decode
setp auddec1 codec aachedec
setp auddec1 outputPCMWidth 16
setp auddec1 pcmFormat 1
setp auddec1 dataEndianness 1
setp auddec1 downSampleSbrFlag 0
setp auddec1 inbufsize 2880
setp auddec1 outbufsize 4608
func auddec1 -s ./data/sounds/davincieffect.aac
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.1.2.3 Audio File Decode

3.1.2.4 Audio Decode to File

- *Description*

Read an encoded audio file, decode it using an audio decoder, store the decoded PCM data to a file.

- *Implementation*

– Input: Audio encoded file.
– Output: Audio PCM file.
– Sample script:

```
setp engine name decode
setp auddec1 codec aachedec
setp auddec1 outputPCMWidth 16
setp auddec1 pcmFormat 1
setp auddec1 dataEndianness 1
setp auddec1 downSampleSbrFlag 0
setp auddec1 inbufsize 2880
setp auddec1 outbufsize 4608
func auddec1 -s ./data/sounds/davincieffect.aac –t out.pcm
```

- *Notes*

– Decoder: all available decoders can be selected.
– Options: all decoder options can be selected

3.1.3 Speech Usecases

3.1.3.1 Speech Capture and Playback

3.1.3.2 Speech Capture and Encode

- *Description*

Capture speech, encode using speech encoder and store to a file.

- *Implementation*

– Input: Speech stream.
– Output: Encoded file.
– Sample script:

```
setp engine name encode
setp sphenc1 codec g711enc
setp sphenc1 seconds 10
setp sphenc1 frameSize 80
setp sphenc1 compandingLaw 1
setp sphenc1 packingType 0
setp sphenc1 vadSelection 0
setp sphenc1 codecSelection 0
setp sphenc1 bitRate 12000
setp sphenc1 inbufsize 160
```

```
setp sphenc1 outbufsize 80
setp audio device plughw:0,0
setp audio format 2
setp audio samplerate 8000
setp audio channels 2
setp audio type 0
func sphenc1 -t sph-enc.g711
```

- *Notes*

– Encoder: all available encoders can be selected.

– Options: all encoder options can be selected.

3.1.3.3 Speech Decode and Playback

- *Description*

Read a speech encoded file, decode and play out.

- *Implementation*

– Input: Speech encoded file.

– Output: Speech.

– Sample script:

```
setp engine name decode
setp sphdec1 codec g711dec
setp sphdec1 compandingLaw 1
setp sphdec1 packingType 1
setp sphdec1 codecSelection 0
setp sphdec1 bitRate 12000
setp sphdec1 inbufsize 80
setp sphdec1 outbufsize 320
setp audio device plughw:0,0
setp audio format 2
setp audio samplerate 8000
setp audio channels 2
setp audio type 0
func sphdec1 -s ./data/sounds/davincieffect.g711
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decode options can be selected.

3.1.3.4 Speech File Encode and Decode

3.1.3.5 Speech Encode from File

- *Description*

Read PCM file, encode using speech codec and store the encoded file.

- *Implementation*

– Input: PCM file.

– Output: Encoded file.

– Sample script:

```
setp engine name encode
setp sphenc1 codec g711enc
setp sphenc1 seconds 10
setp sphenc1 frameSize 80
setp sphenc1 compandingLaw 1
setp sphenc1 packingType 0
setp sphenc1 vadSelection 0
setp sphenc1 codecSelection 0
setp sphenc1 bitRate 12000
setp sphenc1 inbufsize 160
setp sphenc1 outbufsize 80
func sphenc1 –s input.pcm -t sph-enc.g711
```

- *Notes*

– Encoder: all available encoders can be selected.

– Options: all encoder options can be selected.

3.1.3.6 Speech Decode to File

- *Description*

Read an encoded file, decode using a speech decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: PCM file.

– Sample script:

```
setp engine name decode
sphdec1 codec g711dec
```

```
setp sphdec1 compandingLaw 1
setp sphdec1 packingType 1
setp sphdec1 codecSelection 0
setp sphdec1 bitRate 12000
setp sphdec1 inbufsize 80
setp sphdec1 outbufsize 320
func sphdec1 -s ./data/sounds/davincieffect.g711 –t out.pcm
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.1.4 Simultaneous Operations - Usecases

3.1.4.1 Audio-Video Playback

- *Description*

Read encoded audio/video files and play it out through the audio driver and VPBE.

- *Implementation*

– Input: Audio encoded file, Video encoded file

– Output: Audio stream played out on line-out/headphone

Video played out on the VPBE

– Sample script:

```
setp engine name decode
setp viddec2 codec mpeg2dec
setp viddec2 maxWidth 720
setp viddec2 maxHeight 480
setp viddec2 maxFrameRate 30000
setp viddec2 maxBitRate 10000000
setp viddec2 dataEndianness 1
setp viddec2 forceChromaFormat 4
setp viddec2 decodeHeader 0
setp viddec2 displayWidth 0
setp viddec2 frameSkipMode 0
setp viddec2 frameOrder 0
setp viddec2 newFrameFlag 0
setp viddec2 mbDataFlag 0
setp viddec2 numFrames 100
```

```
setp vpbe device /dev/video2
setp vpbe width 720
setp vpbe height 480
setp vpbe std 1
setp vpbe output 1
setp auddec1 codec aachedec
setp auddec1 outputPCMWidth 16
setp auddec1 pcmFormat 1
setp auddec1 dataEndianness 1
setp auddec1 downSampleSbrFlag 0
setp auddec1 inbufsize 2880
setp auddec1 outbufsize 4608
func auddec1 -s ./data/sounds/davincieffect.aac
func viddec2 -s ./data/videos/davincieffect_ntsc.m2v
```

- Ex for running the script:

```
root@158.218.33.135:/opt/dvsdk# ./dvtb-r –s <sample script file>
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.2 DM355(Linux) Usecases

3.2.1 Video Usecases

3.2.1.1 Video File Encode and Decode

3.2.1.1.1 Video Encode from File

- *Description*

  Read a YUV 422 interleaved file, encode using a video encoder, store encoded file.

- *Implementation*

– Input: YUV 422 interleaved file.

– Output: Encoded file.

– Sample script:

```
setp engine name encodedecode
setp videnc1 codec mpeg4enc
setp videnc1 encodingPreset 0
setp videnc1 rateControlPreset 1
setp videnc1 maxHeight 240
setp videnc1 maxWidth 320
setp videnc1 maxFrameRate 30000
setp videnc1 maxBitRate 2000000
setp videnc1 dataEndianness 1
setp videnc1 maxInterFrameInterval 1
setp videnc1 inputChromaFormat 4
setp videnc1 inputContentType 0
setp videnc1 reconChromaFormat 1
setp videnc1 inputHeight 240
setp videnc1 inputWidth 320
setp videnc1 refFrameRate 30000
setp videnc1 targetFrameRate 30000
setp videnc1 targetBitRate 2000000
setp videnc1 intraFrameInterval 30
setp videnc1 generateHeader 0
setp videnc1 captureWidth 320
setp videnc1 forceFrame 0
setp videnc1 interFrameInterval 1
setp videnc1 mbDataFlag 0
setp videnc1 framePitch 720
setp videnc1 numFrames 300
```

```
func videnc1 -s test_320x240_422i.yuv -t mp4-encode.mpeg4
```

- *Notes*

– Encoder: all available encoders can be selected.

– Options: all encoder options can be selected.

3.2.1.1.2 Video Decode to File

- *Description*

Read an encoded file, decode using a video decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: YUV 422 interleaved file.

– Sample script:

```
setp engine name decode
setp viddec2 codec mpeg4dec
setp viddec2 maxHeight 480
setp viddec2 maxWidth 720
setp viddec2 maxFrameRate 30000
setp viddec2 maxBitRate 6000000
setp viddec2 dataEndianness 1
setp viddec2 forceChromaFormat 4
setp viddec2 decodeHeader 0
setp viddec2 displayWidth 0
setp viddec2 frameSkipMode 0
setp viddec2 frameOrder 0
setp viddec2 newFrameFlag 0
setp viddec2 mbDataFlag 0
setp viddec2 numFrames 100
func viddec2 -s ./data/videos/davincieffect_ntsc.mpeg4 -t out_720x480_422i.yuv
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.2.2 Speech Usecases

3.2.2.1 Speech File Encode and Decode

3.2.2.1.1 Speech Encode from File

- *Description*

Read PCM file, encode using speech codec and store the encoded file.

- *Implementation*

– Input: PCM file.

– Output: Encoded file.

– Sample script:

```
setp engine name encode
setp sphenc1 codec g711enc
setp sphenc1 seconds 10
setp sphenc1 frameSize 80
setp sphenc1 compandingLaw 1
setp sphenc1 packingType 0
setp sphenc1 vadSelection 0
setp sphenc1 codecSelection 0
setp sphenc1 bitRate 12000
setp sphenc1 inbufsize 160
setp sphenc1 outbufsize 80
func sphenc1 -s input.pcm -t sph-enc.g711
```

- *Notes*

– Encoder: all available encoders can be selected.

– Options: all encoder options can be selected.

3.2.2.1.2 Speech Decode to File

- *Description*

Read an encoded file, decode using a speech decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: PCM file.

– Sample script:

```
setp engine name decode
```

```
setp sphdec1 codec g711dec
setp sphdec1 compandingLaw 1
setp sphdec1 packingType 1
setp sphdec1 codecSelection 0
setp sphdec1 bitRate 12000
setp sphdec1 inbufsize 80
setp sphdec1 outbufsize 320
func sphdec1 -s ./data/sounds/davincieffect.g711 -t out_8k.pcm
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.2.3 Simultaneous Operations - Usecases

3.2.3.1 Audio-Video Decode

- *Description*

Read encoded audio/video files and play it out through the audio driver and VPBE.

- *Implementation*

– Input: Audio encoded file, Video encoded file

– Output: Audio stream played out on line-out/headphone
Video played out on the VPBE

– Sample script:

```
setp engine name decode
setp viddec2 codec mpeg4dec
setp viddec2 maxHeight 480
setp viddec2 maxWidth 720
setp viddec2 maxFrameRate 30000
setp viddec2 maxBitRate 6000000
setp viddec2 dataEndianness 1
setp viddec2 forceChromaFormat 4
setp viddec2 decodeHeader 0
setp viddec2 displayWidth 0
setp viddec2 frameSkipMode 0
setp viddec2 frameOrder 0
setp viddec2 newFrameFlag 0
setp viddec2 mbDataFlag 0
setp viddec2 numFrames 100
```

```
setp sphdec1 codec g711dec
setp sphdec1 compandingLaw 1
setp sphdec1 packingType 1
setp sphdec1 codecSelection 0
setp sphdec1 bitRate 12000
setp sphdec1 inbufsize 80
setp sphdec1 outbufsize 320
func viddec2 -s ./data/videos/davincieffect_ntsc.mpeg4 -t out_720x480_422i.yuv
func sphdec1 -s ./data/sounds/davincieffect.g711 -t out_8k.pcm
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.2.4 Image Usecases

3.2.4.1 Image Encode from File

- *Description*

Encode a 420P YUV file and store.

- *Implementation*

– Input: YUV file

– Output: Encoded JPEG file in the current directory.

– Sample script:

```
setp engine name encodedecode
setp imgenc1 codec jpegenc
setp imgenc1 maxHeight 480
setp imgenc1 maxWidth 720
setp imgenc1 maxScans 1
setp imgenc1 dataEndianness 1
setp imgenc1 forceChromaFormat 2
setp imgenc1 numAU 0
setp imgenc1 inputChromaFormat 4
setp imgenc1 inputHeight 480
setp imgenc1 inputWidth 720
setp imgenc1 captureWidth 720
setp imgenc1 generateHeader 0
setp imgenc1 qValue 73
func imgenc1 -s input_720x480_422i.yuv -t out.jpg
```

3.2.4.2 Image Decode to File

- *Description*

  Decode JPEG image and store

- *Implementation*

– Input: JPEG file

– Output: output yuv

– Sample script:

```
setp engine name encodedecode
setp imgdec1 codec jpegdec
setp imgdec1 maxHeight 480
setp imgdec1 maxWidth 720
setp imgdec1 maxScans 1
setp imgdec1 dataEndianness 1
setp imgdec1 forceChromaFormat 4
setp imgdec1 numAU 0
setp imgdec1 decodeHeader 0
setp imgdec1 displayWidth 0
func imgdec1 -s shrek_720x480.jpg -t out_720x480_422i.yuv
```

3.2.5 JPEG Usecases

3.2.5.1 Image Encode from File

- *Description*

  Encode a 420P YUV file and store.

- *Implementation*

– Input: YUV file

– Output: Encoded JPEG file in the current directory.

– Sample script:

```
setp engine name encodedecode
setp jpegenc1 codec jpegenc
setp jpegenc1 maxHeight 480
setp jpegenc1 maxWidth 720
setp jpegenc1 maxScans 1
setp jpegenc1 dataEndianness 1
setp jpegenc1 forceChromaFormat 2
setp jpegenc1 numAU 0
setp jpegenc1 inputChromaFormat 4
```

```
setp jpegenc1 inputHeight 480
setp jpegenc1 inputWidth 720
setp jpegenc1 captureWidth 720
setp jpegenc1 generateHeader 0
setp jpegenc1 qValue 73
setp jpegenc1 extDynParamsRstInterval 4
setp jpegenc1 extDynParamsDisableEOI 0
setp jpegenc1 extDynParamsRotation 0
func jpegenc1 -s input_720x480_422i.yuv -t out.jpg
```

3.2.5.2 Image Decode to File

- *Description*

  Decode JPEG image and store

- *Implementation*

– Input: JPEG file

– Output: output yuv

– Sample script:

```
setp engine name encodedecode
setp jpegdec1 codec jpegdec
setp jpegdec1 maxHeight 480
setp jpegdec1 maxWidth 720
setp jpegdec1 maxScans 1
setp jpegdec1 dataEndianness 1
setp jpegdec1 forceChromaFormat 4
setp jpegdec1 numAU 0
setp jpegdec1 decodeHeader 0
setp jpegdec1 displayWidth 0
setp jpegdec1 extDynParamsDisableEOI 0
setp jpegdec1 extDynParamsResizeOption 0
setp jpegdec1 extDynParamsSubRegUpLeftX 0
setp jpegdec1 extDynParamsSubRegUpLeftY 0
setp jpegdec1 extDynParamsSubRegDownRightX 0
setp jpegdec1 extDynParamssubRegDownRightY 0
setp jpegdec1 extDynParamsRotation 0
setp jpegdec1 extDynParamsDecodingPreset 1
func jpegdec1 -s shrek_720x480.jpg -t out_720x480_422i.yuv
```

3.2.6 MPEG4 Usecases

3.2.6.1 MPEG4 Encode from File

- *Description*

Read a YUV 422 interleaved file, encode using a video encoder, store encoded file.

- *Implementation*

– Input: YUV 422 interleaved file.

– Output: Encoded file.

– Sample script:

```
setp engine name encodedecode
setp mpeg4spenc1 codec mpeg4enc
setp mpeg4spenc1 encodingPreset 3
setp mpeg4spenc1 rateControlPreset 1
setp mpeg4spenc1 maxHeight 240
setp mpeg4spenc1 maxWidth 320
setp mpeg4spenc1 maxFrameRate 30000
setp mpeg4spenc1 maxBitRate 2000000
setp mpeg4spenc1 dataEndianness 1
setp mpeg4spenc1 maxInterFrameInterval 1
setp mpeg4spenc1 inputChromaFormat 4
setp mpeg4spenc1 inputContentType 0
setp mpeg4spenc1 reconChromaFormat 1
setp mpeg4spenc1 inputHeight 240
setp mpeg4spenc1 inputWidth 320
setp mpeg4spenc1 refFrameRate 30000
setp mpeg4spenc1 targetFrameRate 30000
setp mpeg4spenc1 targetBitRate 2000000
setp mpeg4spenc1 intraFrameInterval 30
setp mpeg4spenc1 generateHeader 0
setp mpeg4spenc1 captureWidth 320
setp mpeg4spenc1 forceFrame 0
setp mpeg4spenc1 interFrameInterval 1
setp mpeg4spenc1 mbDataFlag 0
setp mpeg4spenc1 framePitch 720
setp mpeg4spenc1 extParamsSubWindowHeight 240
setp mpeg4spenc1 extParamsSubWindowWidth 320
setp mpeg4spenc1 extParamsRotation 0
setp mpeg4spenc1 extParamsVBVsize 0
setp mpeg4spenc1 extParamsIFrameBitRateBiasFactor 256
setp mpeg4spenc1 extParamsPFrameBitRateBiasFactor 256
```

```
setp mpeg4spenc1 extParamsPeakBufWindow 2
setp mpeg4spenc1 extParamsMinBitRate 258
setp mpeg4spenc1 extParamsSVHMode 0
setp mpeg4spenc1 extDynParamsIntraAlgo 1
setp mpeg4spenc1 extDynParamsNumMBRows 1
setp mpeg4spenc1 extDynParamsIntraFrameQP 0
setp mpeg4spenc1 extDynParamsInterFrameQP 0
setp mpeg4spenc1 extDynParamsInitQ 3
setp mpeg4spenc1 extDynParamsRcQMAX 31
setp mpeg4spenc1 extDynParamsRcQMIN 2
setp mpeg4spenc1 extDynParamsRateFix 0
setp mpeg4spenc1 extDynParamsRateFixRange 2
setp mpeg4spenc1 extDynParamsMeAlgo 1
setp mpeg4spenc1 extDynParamsSkipMBAlgo 0
setp mpeg4spenc1 extDynParamsUnrestrictedMV 0
setp mpeg4spenc1 extDynParamsMVDataEnable 0
setp mpeg4spenc1 numFrames 300
func mpeg4spenc1 -s test_320x240_422i.yuv -t mp4-encode.mpeg4
```

- *Notes*

– Encoder: all available encoders can be selected.

– Options: all encoder options can be selected.

3.2.6.2 MPEG4 Decode to File

- *Description*

Read an encoded file, decode using a video decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: YUV 422 interleaved file.

– Sample script:

```
setp engine name encodedecode
setp mpeg4spdec2 codec mpeg4dec
setp mpeg4spdec2 maxHeight 480
setp mpeg4spdec2 maxWidth 720
setp mpeg4spdec2 maxFrameRate 30000
setp mpeg4spdec2 maxBitRate 6000000
setp mpeg4spdec2 dataEndianness 1
setp mpeg4spdec2 forceChromaFormat 4
```

```
setp mpeg4spdec2 decodeHeader 0
setp mpeg4spdec2 displayWidth 0
setp mpeg4spdec2 frameSkipMode 0
setp mpeg4spdec2 frameOrder 0
setp mpeg4spdec2 newFrameFlag 0
setp mpeg4spdec2 mbDataFlag 0
setp mpeg4spdec2 extParamsMeRange 31
setp mpeg4spdec2 extParamsDisplayWidth 0
setp mpeg4spdec2 extParamsRotation 0
setp mpeg4spdec2 extParamsUnrestrictedMV 0
setp mpeg4spdec2 numFrames 100
func mpeg4spdec2 -s ./data/videos/davincieffect_ntsc.mpeg4 -t
out_720x480_422i.yuv
```

- *Notes*

– Decoder: all available decoders can be selected.

Options: all decoder options can be selected.

3.2.7 IPNC Usecases

3.2.7.1 IPNCUC0

- *Description*

MPEG4 encode (720P 30 fps 6 Mbps + SIF 30 fps 512 Kbps)

- *Implementation*

```
DVTB$> func IPNCUC0 -s1 <1280x720 422i YUV> -s2 <352x240 422i YUV>
s1 and s2 are input to the MPEG4 Encoder instances
Output files generated will be enc720pOutput.m4v and encSIFOutput.m4v
```

3.2.7.2 IPNCUC1

- *Description*

MPEG4, D1, 30fps, 2Mbps + MPEG4, SIF, 30fps, 0.5Mbps + JPEG, D1, 15 fps, QP=75

- *Implementation*

```
DVTB$> func IPNCUC1 -s1 <720x480 422i YUV> -s2 <352x240 422i YUV> -s3
<720x480 422i YUV>

s1 and s2 are input to the MPEG4 Encoder instances
s3 is input to the JPEG instance
```

Output files generated for MPEG4 instances will be encD1Output.m4v and encSIFOutput.m4v

Output files generated for JPEG instance will be encJPEGD1Output%d.jpg (%d will be 1,2,... etc based on the # of the frame.)

3.2.7.3 IPNCUC2

- *Description*

MPEG4, 720p, 30fps, 5-6Mbps + MPEG4, SIF, 30fps, 0.5Mbps + JPEG, D1, 15 fps, QP=75

- *Implementation*

```
DVTB$> func IPNCUC2 -s1 <1280x720 422i YUV> -s2 <352x240 422i YUV> -s3
<720x480 422i YUV>
```

s1 and s2 are input to the MPEG4 Encoder instances

s3 is input to the JPEG instance

Output files generated for MPEG4 instances will be enc720pOutput.m4v and encSIFOutput.m4v

Output files generated for JPEG instance will be encJPEGD1Output%d.jpg (%d will be 1,2,... etc based on the # of the frame.)

3.2.8 DVR Usecases

3.2.8.1 DVRUC0

- *Description*

2channels of D1 MPEG4, 30fps, 2Mbps each channel

- *Implementation*

```
DVTB$> func DVRUC0 -s1 <720x480 422i YUV> -s2 <720x480 422i YUV>
```

s1 and s2 are input to the MPEG4 Encoder instances

Output files generated for MPEG4 instances will be enc1D1Output.m4v and enc2D1Output.m4v

3.2.8.2 DVRUC1

- *Description*

8 channels of CIF, MPEG4, 30fps, bitrate appropriately scaled down, using 2Mbps at D1 as reference

- *Implementation*

```
DVTB$> func DVREnc -s <Input File>
```

Input file should specify the source-target files for how many MPEG4 instances to be created. (Maximum 8 MPEG4 Encoder instances.)

3.2.8.3 DVRUC2

- *Description*

4 channels of SIF (enc + dec)

- *Implementation*

```
DVTB$> func DVREncDec -s <Input File>
```

Input file should specify the source-target files for how many MPEG4 Encoder and Decoder instances are to be created. (Maximum 4 MPEG4 Encoder instances + 4 MPEG4 Decoder Instances.)

3.3 DM6467(Linux) Usecases

3.3.1 Video Usecases

3.3.1.1 Video File Encode and Decode

3.3.1.2 Video Encode from File

- *Description*

  Read a YUV 420 planar file, encode using a video encoder, store encoded file.

- *Implementation*

– Input: YUV 420 planar file

– Output: Encoded file

– Sample script:

```
setp engine name encodedecode
setp videnc1 codec h264enc
setp videnc1 encodingPreset 0
setp videnc1 rateControlPreset 1
setp videnc1 maxHeight 720
setp videnc1 maxWidth 1280
setp videnc1 maxFrameRate 30000
setp videnc1 maxBitRate 10000000
setp videnc1 dataEndianness 1
setp videnc1 maxInterFrameInterval 30
setp videnc1 inputChromaFormat 1
setp videnc1 inputContentType 0
setp videnc1 reconChromaFormat 1
setp videnc1 inputHeight 720
setp videnc1 inputWidth 1280
setp videnc1 refFrameRate 30000
setp videnc1 targetFrameRate 30000
setp videnc1 targetBitRate 10000000
setp videnc1 intraFrameInterval 30
setp videnc1 generateHeader 0
setp videnc1 captureWidth 0
setp videnc1 forceFrame -1
setp videnc1 interFrameInterval 0
setp videnc1 mbDataFlag 0
setp videnc1 framePitch 1280
setp videnc1 numFrames 30
```

```
func videnc1 –s input_1280x720_yuv420.yuv -t h264-encode.264
```

- *Notes*

– Encoder: all available encoders can be selected.

– Options: all encoder options can be selected.

3.3.1.3 Video Decode to File

- *Description*

Read an encoded file, decode using a video decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: YUV 420P file.

– Sample script:

```
setp engine name encodedecode
setp viddec2 codec h264dec
setp viddec2 maxWidth 1920
setp viddec2 maxHeight 1088
setp viddec2 maxFrameRate 30000
setp viddec2 maxBitRate 10000000
setp viddec2 dataEndianness 1
setp viddec2 forceChromaFormat 9
setp viddec2 decodeHeader 0
setp viddec2 displayWidth 0
setp viddec2 frameSkipMode 0
setp viddec2 frameOrder 0
setp viddec2 newFrameFlag 0
setp viddec2 mbDataFlag 0
setp viddec2 numFrames 100
func viddec2 –s ./data/videos/davincieffect_1080i_30fps.264 –t out.yuv
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.3.2 Audio Usecases

3.3.2.1 Audio Decode

3.3.2.1.1 Audio Decode to File

- *Description*

Read an encoded file and play it out through the audio driver.

- *Implementation*

– Input: Audio encoded file.

– Output: Decoded PCM file

– Sample script:

```
setp engine name encodedecode
setp auddec1 codec aachedec
setp auddec1 outputPCMWidth 16
setp auddec1 pcmFormat 1
setp auddec1 dataEndianness 1
setp auddec1 downSampleSbrFlag 0
setp auddec1 inbufsize 2880
setp auddec1 outbufsize 4608
func auddec1 -s ./data/sounds/davincieffect.aac –t out.pcm
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.3.3 Speech Usecases

3.3.3.1 Speech File Encode and Decode

3.3.3.1.1 Speech Encode from File

- *Description*

Read PCM file, encode using speech codec and store the encoded file.

- *Implementation*

– Input: PCM file.

– Output: Encoded file.

– Sample script:

```
setp engine name encodedecode
```

```
setp sphenc1 codec g711enc
setp sphenc1 seconds 10
setp sphenc1 frameSize 80
setp sphenc1 compandingLaw 1
setp sphenc1 packingType 0
setp sphenc1 vadSelection 0
setp sphenc1 codecSelection 0
setp sphenc1 bitRate 12000
setp sphenc1 inbufsize 160
setp sphenc1 outbufsize 80
func sphenc1 –s input.pcm -t sph-enc.g711
```

- *Notes*

– Encoder: all available encoders can be selected.

– Options: all encoder options can be selected.

3.3.3.1.2 Speech Decode to File

- *Description*

Read an encoded file, decode using a speech decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: PCM file.

– Sample script:

```
setp engine name encodedecode
setp sphdec1 codec g711dec
setp sphdec1 compandingLaw 1
setp sphdec1 packingType 1
setp sphdec1 codecSelection 0
setp sphdec1 bitRate 12000
setp sphdec1 inbufsize 80
setp sphdec1 outbufsize 320
func sphdec1 -s ./data/sounds/davincieffect.g711 –t out.pcm
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.3.4 Image Usecases

3.3.4.1 Image Encode from File

- *Description*

  Encode a 422ILE YUV file and store.

- *Implementation*

– Input: YUV file

– Output: Encoded JPEG file in the current directory.

– Sample script:

```
setp engine name encodedecode
setp imgenc1 codec jpegenc
setp imgenc1 maxHeight 480
setp imgenc1 maxWidth 720
setp imgenc1 maxScans 0
setp imgenc1 dataEndianness 1
setp imgenc1 forceChromaFormat 2
setp imgenc1 numAU 0
setp imgenc1 inputChromaFormat 4
setp imgenc1 inputHeight 480
setp imgenc1 inputWidth 720
setp imgenc1 captureWidth 720
setp imgenc1 generateHeader 0
setp imgenc1 qValue 73
func imgenc1 -s input_720x480_422i.yuv -t test.jpg
```

3.3.4.2 Image Decode to File

- *Description*

  Decode JPEG image and store

- *Implementation*

– Input: JPEG file

– Output: output yuv

– Sample script:

```
setp engine name encodedecode
setp imgdec1 codec jpegdec
setp imgdec1 maxHeight 480
setp imgdec1 maxWidth 720
setp imgdec1 maxScans 1
```

```
setp imgdec1 dataEndianness 1
setp imgdec1 forceChromaFormat 4
setp imgdec1 numAU 0
setp imgdec1 decodeHeader 0
setp imgdec1 displayWidth 0
getp imgdec1
func imgdec1 -s shrek_720x480.jpg -t test_720x480.yuv
```

3.3.5 Extended parameters AACHE Usecases

3.3.5.1 Audio Decode

3.3.5.1.1 Audio Decode to File

- *Description*

  Read an encoded file and decode dump in a file.

- *Implementation*

– Input: Audio encoded file.

– Output: Decoded PCM file

– Sample script:

```
setp engine name encodedecode
setp aachedec1 codec aachedec
setp aachedec1 outputPCMWidth 16
setp aachedec1 pcmFormat 1
setp aachedec1 dataEndianness 1
setp aachedec1 desiredChannelMode 1
setp aachedec1 downSampleSbrFlag 0
setp aachedec1 sixChannelMode 1
setp aachedec1 enablePS 0
setp aachedec1 ulSamplingRateIdx 3
setp aachedec1 nProfile 0
setp aachedec1 bRawFormat 0
setp aachedec1 pseudoSurroundEnableFlag 1
setp aachedec1 enableARIBDownmix 0
setp aachedec1 inbufsize 2880
setp aachedec1 outbufsize 4608
func aachedec1 -s ./data/sounds/davincieffect.aac -t test.pcm
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.3.6 Extended parameters H264 Video Usecases

3.3.6.1 H264 File Encode and Decode Decode

3.3.6.1.1 H264 Encode from File

- *Description*

Read a YUV 420 planar file, encode using a video encoder, store encoded file.

- *Implementation*

– Input: YUV 420 planar file

– Output: Encoded file

– Sample script:

```
setp engine name encodedecode
setp h264enc1 codec h264enc
setp h264enc1 encodingPreset 3
setp h264enc1 rateControlPreset 1
setp h264enc1 maxHeight 720
setp h264enc1 maxWidth 1280
setp h264enc1 maxFrameRate 30000
setp h264enc1 maxBitRate 10000000
setp h264enc1 dataEndianness 3
setp h264enc1 maxInterFrameInterval 30
setp h264enc1 inputChromaFormat 9
setp h264enc1 inputContentType 0
setp h264enc1 reconChromaFormat 1
setp h264enc1 profileIdc 66
setp h264enc1 METype 1
setp h264enc1 ScalingMatType 0
setp h264enc1 ScalingFactor 0
setp h264enc1 levelIdc 31
setp h264enc1 EntropyCodingMode 0
setp h264enc1 inputHeight 720
setp h264enc1 inputWidth 1280
setp h264enc1 refFrameRate 30000
setp h264enc1 targetFrameRate 30000
setp h264enc1 targetBitRate 10000000
setp h264enc1 intraFrameInterval 30
setp h264enc1 generateHeader 0
```

```
setp h264enc1 captureWidth 0
setp h264enc1 forceFrame -1
setp h264enc1 interFrameInterval 0
setp h264enc1 mbDataFlag 0
setp h264enc1 OutBufSize -1
setp h264enc1 QPISlice 28
setp h264enc1 QPSlice 28
setp h264enc1 RateCtrlQpMax 51
setp h264enc1 RateCtrlQpMin 0
setp h264enc1 NumRowsInSlice 3
setp h264enc1 LfDisableIdc 0
setp h264enc1 LFAlphaC0Offset 6
setp h264enc1 LFBetaOffset 6
setp h264enc1 ChromaQPOffset 0
setp h264enc1 SecChromaQPOffset 0
setp h264enc1 framePitch 1280
setp h264enc1 numFrames 30
func h264enc1 -s input_1280x720_yuv420.yuv -t h264ext-encode.264
```

- *Notes*

– Encoder: all available encoders can be selected.

– Options: all encoder options can be selected.

3.3.6.1.2 H264FHD Encode from File

- *Description*

Read a YUV 420 planar file, encode using a video encoder, store encoded file.

- *Implementation*

– Input: YUV 420 planar file

– Output: Encoded file

– Sample script:

```
setp engine name encodedecode
setp h264fhdenc1 codec h264fhdvenc
setp h264fhdenc1 encodingPreset 0
setp h264fhdenc1 rateControlPreset 2
setp h264fhdenc1 maxHeight 1088
setp h264fhdenc1 maxWidth 1920
setp h264fhdenc1 maxFrameRate 30000
```

```
setp h264fhdenc1 maxBitRate 10000000
setp h264fhdenc1 dataEndianness 3
setp h264fhdenc1 maxInterFrameInterval 0
setp h264fhdenc1 inputChromaFormat 9
setp h264fhdenc1 inputContentType 0
setp h264fhdenc1 reconChromaFormat -1
setp h264fhdenc1 profileIdc 66
setp h264fhdenc1 levelIdc 40
setp h264fhdenc1 EntropyCodingMode 0
setp h264fhdenc1 inputHeight 1088
setp h264fhdenc1 inputWidth 1920
setp h264fhdenc1 refFrameRate 30000
setp h264fhdenc1 targetFrameRate 30000
setp h264fhdenc1 targetBitRate 10000000
setp h264fhdenc1 intraFrameInterval 30
setp h264fhdenc1 generateHeader 0
setp h264fhdenc1 captureWidth 0
setp h264fhdenc1 forceFrame -1
setp h264fhdenc1 interFrameInterval 0
setp h264fhdenc1 mbDataFlag 0
setp h264fhdenc1 sliceCodingPreset 0
setp h264fhdenc1 sliceMode  0
setp h264fhdenc1 sliceUnitSize  0
setp h264fhdenc1 sliceStartOffset[0]  0
setp h264fhdenc1 sliceStartOffset[1]  0
setp h264fhdenc1 sliceStartOffset[2]  0
setp h264fhdenc1 sliceStartOffset[3]  0
setp h264fhdenc1 streamFormat  0
setp h264fhdenc1 OutBufSize -1
setp h264fhdenc1 QPISlice 28
setp h264fhdenc1 QPSlice 28
setp h264fhdenc1 RateCtrlQpMax 51
setp h264fhdenc1 RateCtrlQpMin 0
setp h264fhdenc1 NumRowsInSlice 0
setp h264fhdenc1 LfDisableIdc 0
setp h264fhdenc1 LFAlphaC0Offset 0
setp h264fhdenc1 LFBetaOffset 0
setp h264fhdenc1 ChromaQPOffset 0
setp h264fhdenc1 SecChromaQPOffset 0
setp h264fhdenc1 PicAFFFlag 0
setp h264fhdenc1 PicOrderCountType 0
setp h264fhdenc1 AdaptiveMBs 0
```

```
setp h264fhdenc1 SEIParametersFlag 0
setp h264fhdenc1 VUIParametersFlag 0
setp h264fhdenc1 SkipStartCodesInCallback 1
setp h264fhdenc1 Intra4x4EnableFlag 0
setp h264fhdenc1 BlockingCallFlag 0
setp h264fhdenc1 MESelect 1
setp h264fhdenc1 ME1080iMode 0
setp h264fhdenc1 MVDataFlag 0
setp h264fhdenc1 Transform8x8DisableFlag 1
setp h264fhdenc1 Intra8x8EnableFlag 0
setp h264fhdenc1 InterlaceReferenceMode 0
setp h264fhdenc1 ChromaConversionMode 0
setp h264fhdenc1 maxDelay 1000
setp h264fhdenc1 MaxSlicesSupported 90
setp h264fhdenc1 framePitch 1920
setp h264fhdenc1 numFrames 30
func h264fhdenc1 -s input_1920x1088_yuv420.yuv -t h264fhdext-encode.264
```

- *Notes*

– Encoder: all available encoders can be selected.

– Options: all encoder options can be selected.

3.3.6.1.3 H264 File Decode

- *Description*

Read an encoded file, decode using a video decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: YUV 420P file.

Sample script:

```
setp engine name encodedecode
setp h264dec2 codec h264dec
setp h264dec2 maxWidth 1920
setp h264dec2 maxHeight 1088
setp h264dec2 maxFrameRate 30000
setp h264dec2 maxBitRate 10000000
setp h264dec2 dataEndianness 1
setp h264dec2 forceChromaFormat 9
setp h264dec2 displayDelay 16
setp h264dec2 presetLevelIdc 11
setp h264dec2 presetProfileIdc 3
```

```
setp h264dec2 temporalDirModePred 1
setp h264dec2 decodeHeader 0
setp h264dec2 displayWidth 0
setp h264dec2 frameSkipMode 0
setp h264dec2 frameOrder 0
setp h264dec2 newFrameFlag 0
setp h264dec2 mbDataFlag 0
setp h264dec2 numFrames 30
func h264dec2 -s ./data/videos/davincieffect_1080i_30fps.264 -t test.yuv
```

3.3.6.1.4 H2641080p File Decode

- *Description*

  Read an encoded file, decode using a video decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: YUV 420P file.

Sample script:

```
setp engine name encodedecode
setp h2641080pdec2 codec h2641080p60vdec
setp h2641080pdec2 maxWidth 1920
setp h2641080pdec2 maxHeight 1088
setp h2641080pdec2 maxFrameRate 30000
setp h2641080pdec2 maxBitRate 10000000
setp h2641080pdec2 dataEndianness 1
setp h2641080pdec2 forceChromaFormat 9
setp h2641080pdec2 displayDelay 16
setp h2641080pdec2 presetLevelIdc 11
setp h2641080pdec2 presetProfileIdc 3
setp h2641080pdec2 temporalDirModePred 1
setp h2641080pdec2 decodeHeader 0
setp h2641080pdec2 displayWidth 0
setp h2641080pdec2 frameSkipMode 0
setp h2641080pdec2 frameOrder 0
setp h2641080pdec2 newFrameFlag 0
setp h2641080pdec2 mbDataFlag 0
setp h2641080pdec2 numFrames 30
func h2641080pdec2 -s ./data/videos/davincieffect_1080i_30fps.264 -t test.yuv
```

3.3.7 Extended parameters MPEG2 Video Usecases

3.3.7.1 MPEG2 File Encode and Decode Decode

3.3.7.1.1 MPEG2 File Decode

- *Description*

  Read an encoded file, decode using a video decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: YUV 420P file.

Sample script:

```
setp engine name encodedecode
setp mpeg2dec2 codec mpeg2dec
setp mpeg2dec2 maxWidth 1920
setp mpeg2dec2 maxHeight 1088
setp mpeg2dec2 maxFrameRate 30000
setp mpeg2dec2 maxBitRate 10000000
setp mpeg2dec2 dataEndianness 1
setp mpeg2dec2 forceChromaFormat 9
setp mpeg2dec2 DeBlocking 0
setp mpeg2dec2 decodeHeader 0
setp mpeg2dec2 displayWidth 0
setp mpeg2dec2 frameSkipMode 0
setp mpeg2dec2 frameOrder 0
setp mpeg2dec2 newFrameFlag 0
setp mpeg2dec2 mbDataFlag 0
setp mpeg2dec2 bottom_fld_DDR_Opt 0
setp mpeg2dec2 mb_error_reporting 1
setp mpeg2dec2 errorConceal 1
setp mpeg2dec2 numFrames 30
func mpeg2dec2 -s ./data/videos/davincieffect_1080i_30fps.m2v -t test.yuv
```

3.3.8 Extended parameters MPEG4 Video Usecases

3.3.8.1 Mpeg4 File Encode and Decode Decode

3.3.8.1.1 Mpeg4 Encode from File

- *Description*

Read a YUV 420 planar file, encode using a video encoder, store encoded file.

- *Implementation*

– Input: YUV 420 planar file

– Output: Encoded file

– Sample script:

```
setp engine name encodedecode
setp mpeg4enc1 codec mpeg4enc
setp mpeg4enc1 encodingPreset 0
setp mpeg4enc1 rateControlPreset 1
setp mpeg4enc1 maxHeight 576
setp mpeg4enc1 maxWidth 720
setp mpeg4enc1 maxFrameRate 30000
setp mpeg4enc1 maxBitRate 2000000
setp mpeg4enc1 dataEndianness 1
setp mpeg4enc1 maxInterFrameInterval 30
setp mpeg4enc1 inputChromaFormat 9
setp mpeg4enc1 inputContentType 0
setp mpeg4enc1 reconChromaFormat -1
setp mpeg4enc1 inputHeight 480
setp mpeg4enc1 inputWidth 720
setp mpeg4enc1 refFrameRate 30000
setp mpeg4enc1 targetFrameRate 30000
setp mpeg4enc1 targetBitRate 10000000
setp mpeg4enc1 intraFrameInterval 30
setp mpeg4enc1 generateHeader 0
setp mpeg4enc1 captureWidth 0
setp mpeg4enc1 forceFrame -1
setp mpeg4enc1 interFrameInterval 1
setp mpeg4enc1 mbDataFlag 0
setp mpeg4enc1 MPEG4_mode 1
setp mpeg4enc1 levelIdc 5
setp mpeg4enc1 profileIdc 0
setp mpeg4enc1 useVOS 1
setp mpeg4enc1 useGOV 0
setp mpeg4enc1 useVOLatGOV 0
setp mpeg4enc1 useQpel 0
setp mpeg4enc1 useInterlace 0
setp mpeg4enc1 aspectRatio 1
setp mpeg4enc1 pixelRange 1
setp mpeg4enc1 timerResolution 30000
```

```
setp mpeg4enc1 reset_vIMCOP_every_frame 0
setp mpeg4enc1 UMV 0
setp mpeg4enc1 Four_MV_mode 0
setp mpeg4enc1 PacketSize 0
setp mpeg4enc1 qpIntra 8
setp mpeg4enc1 qpInter 8
setp mpeg4enc1 useHEC 0
setp mpeg4enc1 useGOBSync 0
setp mpeg4enc1 RcAlgo 8
setp mpeg4enc1 QPMax 31
setp mpeg4enc1 QPMin 2
setp mpeg4enc1 maxDelay 1000
setp mpeg4enc1 qpInit 0
setp mpeg4enc1 MVaccessFlag 0
setp mpeg4enc1 ME_Type 1
setp mpeg4enc1 PerceptualRC 0
setp mpeg4enc1 Insert_End_Seq_code 0
setp mpeg4enc1 framePitch 1280
setp mpeg4enc1 numFrames 30
func mpeg4enc1 -s input_720x480_420p.yuv -t mpeg4ext-encode.264
```

3.3.8.1.2 MPEG4 File Decode

- *Description*

  Read an encoded file, decode using a video decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: YUV 420P file.

Sample script:

```
setp engine name encodedecode
setp mpeg4dec2 codec mpeg4dec
setp mpeg4dec2 maxWidth 720
setp mpeg4dec2 maxHeight 576
setp mpeg4dec2 maxFrameRate 30000
setp mpeg4dec2 maxBitRate 10000000
setp mpeg4dec2 dataEndianness 1
setp mpeg4dec2 forceChromaFormat 9
setp mpeg4dec2 display_delay 0
setp mpeg4dec2 reset_vIMCOP_every_frame 1
setp mpeg4dec2 decodeHeader 0
```

```
setp mpeg4dec2 displayWidth 0
setp mpeg4dec2 frameSkipMode 0
setp mpeg4dec2 frameOrder 0
setp mpeg4dec2 newFrameFlag 0
setp mpeg4dec2 mbDataFlag 0
setp mpeg4dec2 outloopDeblocking 0
setp mpeg4dec2 outloopDeRinging 0
setp mpeg4dec2 numFrames 100
func mpeg4dec2 -s
../../input/mpeg4SP/amelie_4_720x480_420p_732frames_128000bps.mpeg4 -t
test_720x480.yuv
```

3.3.9 Extended parameters Image Usecases

3.3.9.1 Image Encode from File

- *Description*

Encode a 422ILE YUV file and store.

- *Implementation*

– Input: YUV file

– Output: Encoded JPEG file in the current directory.

– Sample script:

```
setp engine name encodedecode
setp jpegenc1 codec jpegenc
setp jpegenc1 maxHeight 480
setp jpegenc1 maxWidth 720
setp jpegenc1 maxScans 0
setp jpegenc1 dataEndianness 1
setp jpegenc1 forceChromaFormat 2
setp jpegenc1 numAU 0
setp jpegenc1 inputChromaFormat 4
setp jpegenc1 inputHeight 480
setp jpegenc1 inputWidth 720
setp jpegenc1 captureWidth 720
setp jpegenc1 generateHeader 0
setp jpegenc1 qValue 73
func jpegenc1 -s input_720x480_422i.yuv -t test.jpg
```

3.3.9.2 Image Decode to File

- *Description*

Decode JPEG image and store

- *Implementation*

– Input: JPEG file

– Output: output yuv

– Sample script:

```
setp engine name encodedecode
setp jpegdec1 codec jpegdec
setp jpegdec1 maxHeight 480
setp jpegdec1 maxWidth 720
setp jpegdec1 maxScans 1
setp jpegdec1 dataEndianness 1
setp jpegdec1 forceChromaFormat 4
setp jpegdec1 progressiveDecFlag 1
setp jpegdec1 outImgRes 0
setp jpegdec1 numAU 0
setp jpegdec1 decodeHeader 0
setp jpegdec1 displayWidth 0
setp jpegdec1 progDisplay 0
setp jpegdec1 resizeOption 0
setp jpegdec1 RGB_Format 0
setp jpegdec1 numMCU_row 0
setp jpegdec1 x_org 0
setp jpegdec1 y_org 0
setp jpegdec1 x_length 0
setp jpegdec1 y_length 0
setp jpegdec1 alpha_rgb 0
func jpegdec1 -s ./shrek_720x480.jpg -t test_720x480.yuv
```

3.4 DM365(Linux) Use-cases

3.4.1 Video Usecases

3.4.1.1 Video File Encode and Decode

3.4.1.1.1 Video Encode from File

- *Description*

Read a YUV 422 SP file, encode using a video encoder, store encoded file.

- *Implementation*

– Input: YUV 422 SP file.

– Output: Encoded file.

– Sample script:

```
setp engine name encodedecode
setp videnc1 codec h264enc1
setp videnc1 encodingPreset          0
setp videnc1 rateControlPreset       5
setp videnc1 maxHeight               1088
setp videnc1 maxWidth                1920
setp videnc1 maxFrameRate            30000
setp videnc1 maxBitRate              4000000
setp videnc1 dataEndianness          1
setp videnc1 maxInterFrameInterval   0
setp videnc1 inputChromaFormat       9
setp videnc1 inputContentType        0
setp videnc1 reconChromaFormat       9
setp videnc1 topFieldFirstFlag       0
setp videnc1 inputHeight             1088
setp videnc1 inputWidth              1920
setp videnc1 refFrameRate            30000
setp videnc1 targetFrameRate         30000
setp videnc1 targetBitRate           4000000
setp videnc1 intraFrameInterval      30
setp videnc1 generateHeader          0
setp videnc1 captureWidth            0
setp videnc1 forceFrame              0
setp videnc1 interFrameInterval      0
setp videnc1 mbDataFlag              0
setp videnc1 numFrames               30
func videnc1 -s input_1920x1088_420SP.yuv -t test.264
```

- *Notes*

– Encoder: all available encoders can be selected.

– Options: all encoder options can be selected.

3.4.1.2 Video Decode to File

- *Description*

Read an encoded file, decode using a video decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: YUV 422 SP file.

– Sample script:

```
setp engine name encodedecode
setp viddec2 codec h264dec2
setp viddec2 maxHeight             1088
setp viddec2 maxWidth              1920
setp viddec2 maxFrameRate          30000
setp viddec2 maxBitRate            10485760
setp viddec2 dataEndianness        1
setp viddec2 forceChromaFormat     9
setp viddec2 decodeHeader          0
setp viddec2 displayWidth          0
setp viddec2 frameSkipMode         0
setp viddec2 frameOrder            0
setp viddec2 newFrameFlag          0
setp viddec2 mbDataFlag            0
setp viddec2 numFrames             30
func viddec2 -s data/videos/davincieffect.264 -t test.yuv
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.4.2 Audio Usecases

3.4.2.1 Audio Decode to File

- *Description*

Read an encoded file and play it out through the audio driver.

- *Implementation*

– Input: Audio encoded file.

– Output: Decoded PCM file

– Sample script:

```
setp engine name encodedecode
setp auddec1 codec aacdec1
setp auddec1 outputPCMWidth 16
setp auddec1 pcmFormat 1
setp auddec1 dataEndianness 2
```

```
setp auddec1 desiredChannelMode 1
setp auddec1 lfeFlag 0
setp auddec1 downSampleSbrFlag 0
setp auddec1 inbufsize 10000
setp auddec1 outbufsize 16384
func auddec1 -s davincieffect.aac -t test.pcm
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.4.2.2 Audio Encode from File

- *Description*

Read an encoded file and play it out through the audio driver.

- *Implementation*

– Input: PCM file.

– Output: Audio Encoded file

– Sample script:

```
setp engine name encodedecode
setp audenc1 codec aaclcenc1
setp audenc1 seconds 10
setp audenc1 sampleRate 44100
setp audenc1 bitRate 64000
setp audenc1 channelMode 1
setp audenc1 dataEndianness 2
setp audenc1 encMode 0
setp audenc1 inputFormat 1
setp audenc1 inputBitsPerSample 16
setp audenc1 maxBitRate 192000
setp audenc1 dualMonoMode 0
setp audenc1 crcFlag 0
setp audenc1 ancFlag 1
setp audenc1 lfeFlag 0
setp audenc1 dynamicparams.sampleRate 44100
setp audenc1 dynamicparams.bitRate 64000
setp audenc1 dynamicparams.channelMode 1
setp audenc1 dynamicparams.lfeFlag 0
setp audenc1 dynamicparams.dualMonoMode 0
setp audenc1 dynamicparams.inputBitsPerSample 16
func audenc1 -s input_44k.pcm -t test.aac
```

- *Notes*

– Decoder: all available encoders can be selected.

Options: all encoder options can be selected.

3.4.3 Speech Usecases

3.4.3.1 Speech File Encode and Decode

3.4.3.1.1 Speech Encode from File

- *Description*

Read PCM file, encode using speech codec and store the encoded file.

- *Implementation*

– Input: PCM file.

– Output: Encoded file.

– Sample script:

```
setp engine name encodedecode
setp sphenc1 codec g711enc1
setp sphenc1 seconds 10
setp sphenc1 frameSize 80
setp sphenc1 compandingLaw 1
setp sphenc1 packingType 0
setp sphenc1 vadSelection 0
setp sphenc1 codecSelection 0
setp sphenc1 bitRate 12000
setp sphenc1 inbufsize 160
setp sphenc1 outbufsize 80
func sphenc1 –s input.pcm -t sph-enc.g711
```

- *Notes*

– Encoder: all available encoders can be selected.

– Options: all encoder options can be selected.

3.4.3.1.2 Speech Decode to File

- *Description*

Read an encoded file, decode using a speech decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: PCM file.

– Sample script:

```
setp engine name encodedecode
setp sphdec1 codec g711dec1
setp sphdec1 compandingLaw 1
setp sphdec1 packingType 1
setp sphdec1 codecSelection 0
setp sphdec1 bitRate 12000
setp sphdec1 inbufsize 80
setp sphdec1 outbufsize 320
func sphdec1 -s ./data/sounds/davincieffect.g711 –t out.pcm
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.4.4 Image Usecases

3.4.4.1 Image Encode from File

- *Description*

Encode a 422ILE YUV file and store.

- *Implementation*

– Input: YUV file

– Output: Encoded JPEG file in the current directory.

– Sample script:

```
setp engine name encode
setp imgenc1 codec jpegenc1
setp imgenc1 maxHeight                    720
setp imgenc1 maxWidth                     1280
setp imgenc1 maxScans                     0
setp imgenc1 dataEndianness               1
setp imgenc1 forceChromaFormat            2
setp imgenc1 numAU                        0
```

```
setp imgenc1 inputChromaFormat                4
setp imgenc1 inputHeight                      720
setp imgenc1 inputWidth                       1280
setp imgenc1 captureWidth                     1280
setp imgenc1 generateHeader                   0
setp imgenc1 qValue                           73
func imgenc1 -s input-720p_422i.yuv -t out.jpg
```

3.4.4.2 Image Decode to File

- *Description*

  Decode JPEG image and store

- *Implementation*

– Input: JPEG file
– Output: output yuv
– Sample script:

```
setp engine name encodedecode
setp imgdec1 codec jpegdec1
setp imgdec1 maxHeight 720
setp imgdec1 maxWidth 1280
setp imgdec1 maxScans 1
setp imgdec1 dataEndianness 1
setp imgdec1 forceChromaFormat 0
setp imgdec1 numAU 0
setp imgdec1 decodeHeader 0
func imgdec1 -s input-720p.jpg -t output-720p.yuv
```

3.4.5 Audio Extended parameter Usecases

3.4.5.1 Audio Playback

3.4.5.1.1 Audio Decode to File

- *Description*

  Read an encoded file and play it out through the audio driver.

- *Implementation*

– Input: Audio encoded file.
– Output: Decoded PCM file
– Sample script:

```
setp engine name encodedecode
setp aachedec1 codec aacdec1
setp aachedec1 outputPCMWidth 16
setp aachedec1 pcmFormat 1
setp aachedec1 dataEndianness 2
setp aachedec1 desiredChannelMode 1
setp aachedec1 downSampleSbrFlag 0
setp aachedec1 i_max_channels 2
setp aachedec1 i_flag_08khz_out 0
setp aachedec1 i_flag_16khz_out 0
setp aachedec1 i_interleave 1
setp aachedec1 i_mp4_flag 0
setp aachedec1 i_disable_sync 0
setp aachedec1 i_auto_sbr_upsample 1
setp aachedec1 i_sampfreq 0
setp aachedec1 i_coupling_channel 0
setp aachedec1 inbufsize 10000
setp aachedec1 outbufsize 16384
func aachedec1 -s davincieffect.aac -t test.pcm
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.4.5.1.2 Audio Encode from File

- *Description*

Read an encoded file and play it out through the audio driver.

- *Implementation*

– Input: PCM file.

– Output: Audio encoded file.

– Sample script:

```
setp engine name encodedecode
setp aaclcenc1 codec aaclcenc1
setp aaclcenc1 seconds 10
setp aaclcenc1 sampleRate 44100
setp aaclcenc1 bitRate 64000
setp aaclcenc1 channelMode 1
setp aaclcenc1 dataEndianness 2
setp aaclcenc1 encMode 0
setp aaclcenc1 inputFormat 1
```

```
setp aaclcenc1 inputBitsPerSample 16
setp aaclcenc1 maxBitRate 144000
setp aaclcenc1 dualMonoMode 0
setp aaclcenc1 crcFlag 0
setp aaclcenc1 ancFlag 1
setp aaclcenc1 lfeFlag 0
setp aaclcenc1 noChannels 2
setp aaclcenc1 aacClassic 0
setp aaclcenc1 psEnable 0
setp aaclcenc1 dualMono 0
setp aaclcenc1 downmix 0
setp aaclcenc1 useSpeechConfig 0
setp aaclcenc1 fNoStereoPreprocessing 0
setp aaclcenc1 invQuant 2
setp aaclcenc1 useTns 1
setp aaclcenc1 use_ADTS 1
setp aaclcenc1 use_ADIF 0
setp aaclcenc1 full_bandwidth 0
setp aaclcenc1 i_channels_mask 0
setp aaclcenc1 i_num_coupling_chan 0
setp aaclcenc1 write_program_config_element 0
setp aaclcenc1 dynamicparams.sampleRate 44100
setp aaclcenc1 dynamicparams.bitRate 64000
setp aaclcenc1 dynamicparams.channelMode 1
setp aaclcenc1 dynamicparams.lfeFlag 0
setp aaclcenc1 dynamicparams.dualMonoMode 0
setp aaclcenc1 dynamicparams.inputBitsPerSample 16
func aaclcenc1 -s input_44k.pcm -t test.aac
```

- *Notes*

– Decoder: all available encoders can be selected.

– Options: all encoder options can be selected.

3.4.6 Extended parameters MPEG4 Video Usecases

3.4.6.1 Extended parameters MPEG4 Video File Encode and Decode

3.4.6.2 Video Encode from File

- *Description*

Read a YUV 420SP, encode MPEG4 using a video encoder, store encoded file.

- *Implementation*

– Input: YUV 420SP file.

– Output: Encoded file.

– Sample script:

```
setp engine name encodedecode
setp mpeg4enc1 codec mpeg4enc1
setp mpeg4enc1 encodingPreset             2
setp mpeg4enc1 rateControlPreset          1
setp mpeg4enc1 maxHeight                  1088
setp mpeg4enc1 maxWidth                   1920
setp mpeg4enc1 maxFrameRate               30000
setp mpeg4enc1 maxBitRate                 4000000
setp mpeg4enc1 dataEndianness             1
setp mpeg4enc1 maxInterFrameInterval      0
setp mpeg4enc1 inputChromaFormat          9
setp mpeg4enc1 inputContentType           0
setp mpeg4enc1 reconChromaFormat          9
setp mpeg4enc1 inputHeight                1088
setp mpeg4enc1 inputWidth                 1920
setp mpeg4enc1 refFrameRate               30000
setp mpeg4enc1 targetFrameRate            30000
setp mpeg4enc1 targetBitRate              4000000
setp mpeg4enc1 intraFrameInterval         30
setp mpeg4enc1 generateHeader             0
setp mpeg4enc1 captureWidth               1280
setp mpeg4enc1 forceFrame                 0
setp mpeg4enc1 interFrameInterval         0
setp mpeg4enc1 mbDataFlag                 0
setp mpeg4enc1 extsubWindowHeight         1088
setp mpeg4enc1 extsubWindowWidth          1920
setp mpeg4enc1 extrotation                0
setp mpeg4enc1 extvbvSize                 22
setp mpeg4enc1 extsvhMode                 0
setp mpeg4enc1 extIFrameBitRateBiasFactor 256
setp mpeg4enc1 extPFrameBitRateBiasFactor 256
setp mpeg4enc1 extpeakBufWindow           2
setp mpeg4enc1 extminBitRate              2000000
setp mpeg4enc1 extintraAlgo               0
setp mpeg4enc1 extnumMBRows               1
setp mpeg4enc1 extinitQ                   0
setp mpeg4enc1 extrcQMax                  31
```

```
setp mpeg4enc1 extrcQMin                1
setp mpeg4enc1 extintraFrameQP          0
setp mpeg4enc1 extinterFrameQP          0
setp mpeg4enc1 extrateFix               0
setp mpeg4enc1 extrateFixRange          0
setp mpeg4enc1 extmeAlgo                0
setp mpeg4enc1 extskipMBAlgo            0
setp mpeg4enc1 extunrestrictedMV        0
setp mpeg4enc1 extmvDataEnable          0
setp mpeg4enc1 numFrames                30
func mpeg4enc1 -s input_1920x1088_420SP.yuv -t test.m4v
```

- *Notes*

– Options: all encoder options can be selected.

3.4.6.3 Video Decode to File

- *Description*

Read an encoded file, decode using a MPEG4 video decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: YUV 420P file.

– Sample script:

```
setp engine name encodedecode
setp mpeg4dec2 codec                mpeg4dec2
setp mpeg4dec2 maxHeight            720
setp mpeg4dec2 maxWidth             1280
setp mpeg4dec2 maxFrameRate         30000
setp mpeg4dec2 maxBitRate           10485760
setp mpeg4dec2 dataEndianness       1
setp mpeg4dec2 forceChromaFormat    9
setp mpeg4dec2 decodeHeader         0
setp mpeg4dec2 displayWidth         0
setp mpeg4dec2 frameSkipMode        0
setp mpeg4dec2 frameOrder           0
setp mpeg4dec2 newFrameFlag         0
setp mpeg4dec2 mbDataFlag           80
setp mpeg4dec2 numFrames            30
setp mpeg4dec2 extrotation          0
setp mpeg4dec2 extdisplayWidth      1280
setp mpeg4dec2 extmeRange           31
```

```
setp mpeg4dec2 extunrestrictedMV     1
func mpeg4dec2 -s data/videos/davincieffect.mpeg4 -t out.yuv
```

- *Notes*

Options: all decoder options can be selected.

3.4.7 Extended parameters MPEG4 HDVICP Video Usecases

3.4.7.1 Extended parameters MPEG4 HDVICP Video File Encode and Decode

3.4.7.2 Video Encode from File

- *Description*

  Read a YUV 420SP, encode MPEG4 using a video encoder, store encoded file.

- *Implementation*

– Input: YUV 420SP file.

– Output: Encoded file.

– Sample script:

```
setp engine name encodedecode
setp mpeg4hdvicpenc1 codec mpeg4hdvicpenc1
setp mpeg4hdvicpenc1 encodingPreset             3
setp mpeg4hdvicpenc1 rateControlPreset          0
setp mpeg4hdvicpenc1 maxHeight                  1088
setp mpeg4hdvicpenc1 maxWidth                   1920
setp mpeg4hdvicpenc1 maxFrameRate               30000
setp mpeg4hdvicpenc1 maxBitRate                 4000000
setp mpeg4hdvicpenc1 dataEndianness             1
setp mpeg4hdvicpenc1 maxInterFrameInterval      0
setp mpeg4hdvicpenc1 inputChromaFormat          9
setp mpeg4hdvicpenc1 inputContentType           0
setp mpeg4hdvicpenc1 reconChromaFormat          9
setp mpeg4hdvicpenc1 inputHeight                1088
setp mpeg4hdvicpenc1 inputWidth                 1920
setp mpeg4hdvicpenc1 refFrameRate               30000
setp mpeg4hdvicpenc1 targetFrameRate            30000
setp mpeg4hdvicpenc1 targetBitRate              4000000
setp mpeg4hdvicpenc1 intraFrameInterval         30
setp mpeg4hdvicpenc1 generateHeader             0
setp mpeg4hdvicpenc1 captureWidth               0
setp mpeg4hdvicpenc1 forceFrame                 0
setp mpeg4hdvicpenc1 interFrameInterval         0
setp mpeg4hdvicpenc1 mbDataFlag                 0
setp mpeg4hdvicpenc1 extMPEG4_mode              1
setp mpeg4hdvicpenc1 extlevelIdc                5
setp mpeg4hdvicpenc1 extuseVOS                  1
setp mpeg4hdvicpenc1 extuseGOV                  0
```

```
setp mpeg4hdvicpenc1 extuseDataPartition        0
setp mpeg4hdvicpenc1 extuseRVLC                 0
setp mpeg4hdvicpenc1 extaspectRatio             1
setp mpeg4hdvicpenc1 extpixelRange              1
setp mpeg4hdvicpenc1 exttimerResolution         30000
setp mpeg4hdvicpenc1 extME_Type                 0
setp mpeg4hdvicpenc1 extUMV                     1
setp mpeg4hdvicpenc1 extEncQuality_mode         0
setp mpeg4hdvicpenc1 extFour_MV_mode            0
setp mpeg4hdvicpenc1 extPacketSize              0
setp mpeg4hdvicpenc1 extqpIntra                 8
setp mpeg4hdvicpenc1 extqpInter                 8
setp mpeg4hdvicpenc1 extairRate                 0
setp mpeg4hdvicpenc1 extuseHEC                  0
setp mpeg4hdvicpenc1 extuseGOBSync              0
setp mpeg4hdvicpenc1 extRcAlgo                  8
setp mpeg4hdvicpenc1 extQPMax                   31
setp mpeg4hdvicpenc1 extQPMin                   2
setp mpeg4hdvicpenc1 extmaxDelay                1000
setp mpeg4hdvicpenc1 extqpInit                  8
setp mpeg4hdvicpenc1 extPerceptualRC            0
setp mpeg4hdvicpenc1 extreset_vIMCOP_every_frame 1
setp mpeg4hdvicpenc1 extmvSADoutFlag            0
setp mpeg4hdvicpenc1 numFrames                  30
func mpeg4hdvicpenc1 -s input_1920x1088_420SP.yuv -t test.m4v
```

- *Notes*

– Options: all encoder options can be selected.

3.4.7.3 Video Decode to File

- *Description*

Read an encoded file, decode using a MPEG4 video decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: YUV 420P file.

– Sample script:

```
setp engine name encodedecode
setp mpeg4hdvicpdec2 codec                mpeg4hdvicpdec2
setp mpeg4hdvicpdec2 maxHeight            1088
setp mpeg4hdvicpdec2 maxWidth             1920
```

```
setp mpeg4hdvicpdec2 maxFrameRate          30000
setp mpeg4hdvicpdec2 maxBitRate            10485760
setp mpeg4hdvicpdec2 dataEndianness        1
setp mpeg4hdvicpdec2 forceChromaFormat     9
setp mpeg4hdvicpdec2 decodeHeader          0
setp mpeg4hdvicpdec2 displayWidth          0
setp mpeg4hdvicpdec2 frameSkipMode         0
setp mpeg4hdvicpdec2 frameOrder            0
setp mpeg4hdvicpdec2 newFrameFlag          0
setp mpeg4hdvicpdec2 mbDataFlag            0
setp mpeg4hdvicpdec2 numFrames             30
setp mpeg4hdvicpdec2 extdisplayDelay            1
setp mpeg4hdvicpdec2 extdisableHDVICPeveryFrame 0
setp mpeg4hdvicpdec2 extoutloopDeblocking       0
setp mpeg4hdvicpdec2 extoutloopDeRinging        0
setp mpeg4hdvicpdec2 extresetHDVICPeveryFrame   1
func mpeg4hdvicpdec2 -s data/videos/davincieffect.mpeg4 -t out.yuv
```

- *Notes*

Options: all decoder options can be selected.

3.4.8 Extended parameters Image Usecases

3.4.8.1 Image Encode from File

- *Description*

Encode a 422ILE YUV file and store.

- *Implementation*

– Input: YUV file

– Output: Encoded JPEG file in the current directory.

– Sample script:

```
setp engine name encodedecode
setp jpegenc1 codec                jpegenc1
setp jpegenc1 maxHeight            720
setp jpegenc1 maxWidth             1280
setp jpegenc1 maxScans             1
setp jpegenc1 dataEndianness       1
setp jpegenc1 forceChromaFormat    2
setp jpegenc1 numAU                0
setp jpegenc1 inputChromaFormat    4
setp jpegenc1 inputHeight          720
```

```
setp jpegenc1 inputWidth             1280
setp jpegenc1 captureWidth           0
setp jpegenc1 generateHeader         0
setp jpegenc1 qValue                 73
setp jpegenc1 halfBufCB              0
setp jpegenc1 halfBufCBarg           0
setp jpegenc1 extDynParamsRstInterval            84
setp jpegenc1 extDynParamsRstInterval 4
setp jpegenc1 extDynParamsDisableEOI 0
setp jpegenc1 extDynParamsRotation   0
setp jpegenc1 customQ                0
func jpegenc1 -s input-720p_422i.yuv -t out.jpg
```

3.4.8.2 Image Decode to File

- *Description*

  Decode JPEG image and store

- *Implementation*

– Input: JPEG file

– Output: output yuv

– Sample script:

```
setp engine name encodedecode
setp jpegdec1 codec jpegdec1
setp jpegdec1 maxHeight                  720
setp jpegdec1 maxWidth                   1280
setp jpegdec1 maxScans                   1
setp jpegdec1 dataEndianness             1
setp jpegdec1 forceChromaFormat          9
setp jpegdec1 numAU                      0
setp jpegdec1 decodeHeader               0
setp jpegdec1 displayWidth               1280
setp jpegdec1 halfBufCB                  0
setp jpegdec1 halfBufCBarg               0
setp jpegdec1 disableEOI                 0
setp jpegdec1 resizeOption               0
setp jpegdec1 subRegionUpLeftX           0
setp jpegdec1 subRegionUpLeftY           0
setp jpegdec1 subRegionDownRightX        0
setp jpegdec1 subRegionDownRightY        0
setp jpegdec1 rotation                   0
func jpegdec1 -s input-720p.jpg -t output-720p.yuv
```

3.4.9 Extended parameters H264 Video Usecases

3.4.9.1 Extended parameters H264 Video File Encode and Decode

3.4.9.1.1 Video Encode from File

- *Description*

  Read a YUV 420SP file, encode using a H264 video encoder, store encoded file.

- *Implementation*

– Input: YUV 420 SP file.

– Output: Encoded file.

– Sample script:

```
setp engine name encodedecode
setp h264enc1 codec h264enc1
setp h264enc1 encodingPreset                0
setp h264enc1 rateControlPreset             5
setp h264enc1 maxHeight                     1088
setp h264enc1 maxWidth                      1920
setp h264enc1 maxFrameRate                  30000
setp h264enc1 maxBitRate                    4000000
setp h264enc1 dataEndianness                1
setp h264enc1 maxInterFrameInterval         0
setp h264enc1 inputChromaFormat             9
setp h264enc1 inputContentType              0
setp h264enc1 reconChromaFormat             9
setp h264enc1 topFieldFirstFlag             0
setp h264enc1 inputHeight                   1088
setp h264enc1 inputWidth                    1920
setp h264enc1 refFrameRate                  30000
setp h264enc1 targetFrameRate               30000
setp h264enc1 targetBitRate                 4000000
setp h264enc1 intraFrameInterval            30
setp h264enc1 generateHeader                0
setp h264enc1 captureWidth                  0
setp h264enc1 forceFrame                    0
setp h264enc1 interFrameInterval            0
setp h264enc1 mbDataFlag                    0
setp h264enc1 profileIdc                    100
setp h264enc1 levelIdc                      31
setp h264enc1 meAlgo                        0
setp h264enc1 enableVUIparams               0
```

```
setp h264enc1 EntropyMode                       1
setp h264enc1 Transform8x8FlagIntraFrame        1
setp h264enc1 Transform8x8FlagInterFrame        0
setp h264enc1 EnableLongTermFrame               0
setp h264enc1 ConstraintSetFlag                 0
setp h264enc1 Log2MaxFrameNumMinus4             0
setp h264enc1 svcSyntaxEnable                   0
setp h264enc1 numTemporalLayers                 0
setp h264enc1 resetHDVICPeveryFrame             1
setp h264enc1 SeqScalingFlag                    1
setp h264enc1 EncQuality                        0
setp h264enc1 enableARM926Tcm                   0
setp h264enc1 enableDDRbuff                     0
setp h264enc1 sliceMode                         0
setp h264enc1 outputDataMode                    1
setp h264enc1 sliceFormat                       1
setp h264enc1 sliceSize                         0
setp h264enc1 airRate                           0
setp h264enc1 initQ                             28
setp h264enc1 intraFrameQP                      28
setp h264enc1 interPFrameQP                     28
setp h264enc1 rcQMax                            51
setp h264enc1 rcQMin                            0
setp h264enc1 rcAlgo                            1
setp h264enc1 maxDelay                          2000
setp h264enc1 lfDisableIdc                      0
setp h264enc1 aspectRatioX                      1
setp h264enc1 aspectRatioY                      1
setp h264enc1 enableBufSEI                      0
setp h264enc1 enablePicTimSEI                   0
setp h264enc1 perceptualRC                      1
setp h264enc1 idrFrameInterval                  0
setp h264enc1 mvSADoutFlag                      0
setp h264enc1 enableROI                         0
setp h264enc1 metaDataGenerateConsume           0
setp h264enc1 maxBitrateCVBR                    2000000
setp h264enc1 maxHighCmpxIntCVBR                0
setp h264enc1 CVBRsensitivity                   0
setp h264enc1 LBRmaxpicsize                     0
setp h264enc1 LBRminpicsize                     0
setp h264enc1 LBRskipcontrol                    327684
setp h264enc1 interlaceRefMode                  0
```

```
setp h264enc1 LongTermRefreshInterval          0
setp h264enc1 UseLongTermFrame                 0
setp h264enc1 SetLongTermFrame                 0
setp h264enc1 enableGDR                        0
setp h264enc1 GDRduration                      5
setp h264enc1 GDRinterval                      30
setp h264enc1 numOfROI                         0
setp h264enc1 numFrames                        30
func h264enc1 -s input_1920x1088_420SP.yuv  -t test.264
```

- *Notes*

– Options: all encoder options can be selected.

3.4.9.2 Video Decode to File

- *Description*

Read an encoded file, decode using a H264 video decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: YUV 420P file.

– Sample script:

```
setp engine name encodedecode
setp h264dec2 codec                      h264dec2
setp h264dec2 maxHeight                  720
setp h264dec2 maxWidth                   1280
setp h264dec2 maxFrameRate               30000
setp h264dec2 maxBitRate                 10485760
setp h264dec2 dataEndianness             1
setp h264dec2 forceChromaFormat          9
setp h264dec2 decodeHeader               0
setp h264dec2 displayWidth               0
setp h264dec2 frameSkipMode              0
setp h264dec2 frameOrder                 0
setp h264dec2 newFrameFlag               0
setp h264dec2 mbDataFlag                 0
setp h264dec2 displayDelay               0
setp h264dec2 hdvicpHandle               0
setp h264dec2 resetHDVICPeveryFrame      0
setp h264dec2 disableHDVICPeveryFrame    1
setp h264dec2 numFrames                  30
func h264dec2 -s data/videos/davincieffect.264 -t test.yuv
```

- *Notes*

Options: all decoder options can be selected.

3.4.9.3 Extended parameters MPEG2 Video File Encode and Decode

3.4.9.4 Video Encode from File

- *Description*

Read a YUV 420SP file, encode using a H264 video encoder, store encoded file.

- *Implementation*

– Input: YUV 420 SP file.

– Output: Encoded file.

– Sample script:

```
setp engine name encode
setp mpeg2enc1 codec mpeg2enc1
setp mpeg2enc1 encodingPreset          0
setp mpeg2enc1 rateControlPreset       1
setp mpeg2enc1 maxHeight               1088
setp mpeg2enc1 maxWidth                1920
setp mpeg2enc1 maxFrameRate            30000
setp mpeg2enc1 maxBitRate              4000000
setp mpeg2enc1 dataEndianness          1
setp mpeg2enc1 maxInterFrameInterval   0
setp mpeg2enc1 inputChromaFormat       9
setp mpeg2enc1 inputContentType        0
setp mpeg2enc1 reconChromaFormat       9
setp mpeg2enc1 topFieldFirstFlag       0
setp mpeg2enc1 inputHeight             1088
setp mpeg2enc1 inputWidth              1920
setp mpeg2enc1 refFrameRate            30000
setp mpeg2enc1 targetFrameRate         30000
setp mpeg2enc1 targetBitRate           4000000
setp mpeg2enc1 intraFrameInterval      30
setp mpeg2enc1 generateHeader          0
setp mpeg2enc1 captureWidth            0
setp mpeg2enc1 forceFrame              0
setp mpeg2enc1 interFrameInterval      0
setp mpeg2enc1 mbDataFlag              0
setp mpeg2enc1 aspectRatio             1
setp mpeg2enc1 pixelRange              1
```

```
setp mpeg2enc1 timerResolution         60
setp mpeg2enc1 ME_Type                 1
setp mpeg2enc1 QscaleType              0
setp mpeg2enc1 IntraDCPrec             0
setp mpeg2enc1 qpIntra                 8
setp mpeg2enc1 qpInter                 8
setp mpeg2enc1 RcAlgo                  8
setp mpeg2enc1 QPMax                   31
setp mpeg2enc1 QPMin                   1
setp mpeg2enc1 maxDelay                1000
setp mpeg2enc1 qpInit                  8
setp mpeg2enc1 PerceptualRC            0
setp mpeg2enc1 reset_vIMCOP_every_frame 1
setp mpeg2enc1 mvSADoutFlag            0
setp mpeg2enc1 numFrames               30
func mpeg2enc1 -s input_1920x1088_420SP.yuv  -t test.m2v
```

3.4.9.5 Video Decode to File

- *Description*

Read an encoded file, decode using a H264 video decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: YUV 420P file.

– Sample script:

```
setp engine name encodedecode
setp mpeg2dec2 codec mpeg2dec2
setp mpeg2dec2 maxWidth 1920
setp mpeg2dec2 maxHeight 1088
setp mpeg2dec2 maxFrameRate 30000
setp mpeg2dec2 maxBitRate 10000000
setp mpeg2dec2 dataEndianness 1
setp mpeg2dec2 forceChromaFormat 9
setp mpeg2dec2 display_delay 1
setp mpeg2dec2 decodeHeader 0
setp mpeg2dec2 displayWidth 0
setp mpeg2dec2 frameSkipMode 0
setp mpeg2dec2 frameOrder 0
setp mpeg2dec2 newFrameFlag 0
setp mpeg2dec2 mbDataFlag 0
setp mpeg2dec2 bottom_fld_DDR_Opt 0
```

```
setp mpeg2dec2 mb_error_reporting 0
setp mpeg2dec2 errorConceal 1
setp mpeg2dec2 reset_HDVICP_every_frame 1
setp mpeg2dec2 numFrames 30
func mpeg2dec2 -s davincieffect_1080i_30fps.m2v -t test.yuv
```

- *Notes*

Options: all decoder options can be selected.

3.4.9.6 Extended parameters VC1 Video File Decode

3.4.9.6.1 Video Decode to File

- *Description*

Read an encoded file, decode using a H264 video decoder, store decoded file.

- *Implementation*

– Input: Encoded file.

– Output: YUV 420P file.

– Sample script:

```
setp engine name encodedecode
setp vc1dec2 codec                      vc1dec2
setp vc1dec2 maxHeight              1088
setp vc1dec2 maxWidth               1920
setp vc1dec2 maxFrameRate           30000
setp vc1dec2 maxBitRate             10485760
setp vc1dec2 dataEndianness         1
setp vc1dec2 forceChromaFormat      9
setp vc1dec2 decodeHeader           0
setp vc1dec2 displayWidth           0
setp vc1dec2 frameSkipMode          0
setp vc1dec2 frameOrder             0
setp vc1dec2 newFrameFlag           0
setp vc1dec2 mbDataFlag             0
setp vc1dec2 display_delay          1
setp vc1dec2 rcvParseInLib          1
setp vc1dec2 streamType             0
setp vc1dec2 outloopDeblocking      0
setp vc1dec2 reset_HDVICP_every_frame 1
setp vc1dec2 numFrames              30
func vc1dec2 -s The_Rules_of_Attraction_1080.wmv.rcv -t test.yuv
```

- *Notes*

Options: all decoder options can be selected.

3.5 OMAP3530(Linux) Usecases

3.5.1 Video Use Cases

3.5.1.1 Video Decode

3.5.1.2 Video Playback Elementary Stream

- *Description*

  Read 422 interleaved file, display

- *Implementation*
  - Input: 422 interleaved file
  - Output: video stream display through VPBE on LCD
  - Sample script:

```
setp vpbe width 720
setp vpbe height 480
setp vpbe rotate 90
func viddec2 –s source.yuv –-nodsp
```

- *Notes*
  - Decoder: all available decoders can be selected.
  - Options: all decoder options can be selected.

3.5.1.3 Video File Decode

- *Description*

  Read an encoded file, decode using a video decoder, store decoded file.

- *Implementation*
  - Input: Encoded file.
  - Output: YUV 420P planar file.
  - Sample script:

```
setp engine name encodedecode
setp viddec2 codec mpeg4dec
setp viddec2 maxHeight 480
```

```
setp viddec2 maxWidth 720
setp viddec2 maxFrameRate 30000
setp viddec2 maxBitRate 6000000
setp viddec2 dataEndianness 1
setp viddec2 forceChromaFormat 4
setp viddec2 decodeHeader 0
setp viddec2 displayWidth 0
setp viddec2 frameSkipMode 0
setp viddec2 frameOrder 0
setp viddec2 newFrameFlag 0
setp viddec2 mbDataFlag 0
setp viddec2 numFrames 100
func viddec2 –s source.m4v –t source-dec-420.yuv
```

- *Notes*
  - Decoder: all available decoders can be selected.
  - Options: all decoder options can be selected.

3.5.1.4 Video File Decode and display

- *Description*

  Read an encoded file, decode using a video decoder, and display on LCD.

- *Implementation*

– Input: Encoded file.

– Output: display on LCD

– Sample script:

```
setp engine name encodedecode
setp viddec2 codec mpeg4dec
setp viddec2 maxHeight 480
setp viddec2 maxWidth 720
setp viddec2 maxFrameRate 30000
setp viddec2 maxBitRate 6000000
setp viddec2 dataEndianness 1
setp viddec2 forceChromaFormat 4
setp viddec2 decodeHeader 0
setp viddec2 displayWidth 0
setp viddec2 frameSkipMode 0
```

```
setp viddec2 frameOrder 0
setp viddec2 newFrameFlag 0
setp viddec2 mbDataFlag 0
setp viddec2 numFrames 100
setp vpbe device /dev/video2
setp vpbe width 720
setp vpbe height 480
setp vpbe scaling 1
setp vpbe ch_mode VGA
setp vpbe output 2
func viddec2 –s source.m4v
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.5.1.5 H264 File Decode and display

- *Description*

  Read an h264 encoded file, decode using a video decoder, and display on LCD.

- *Implementation*

– Input: H264 Encoded file.

– Output: display on LCD

– Sample script:

```
setp engine name encodedecode
setp h264dec2 codec h264dec
setp h264dec2 maxHeight 480
setp h264dec2 maxWidth 720
setp h264dec2 maxFrameRate 30000
setp h264dec2 maxBitRate 10000000
setp h264dec2 dataEndianness 1
setp h264dec2 forceChromaFormat 4
setp h264dec2 inputStreamFormat 0
setp h264dec2 maxDisplayDelay 0
setp h264dec2 decodeHeader 0
setp h264dec2 displayWidth 0
setp h264dec2 frameSkipMode 0
setp h264dec2 frameOrder 0
```

```
setp h264dec2 newFrameFlag 0
setp h264dec2 mbDataFlag 0
setp h264dec2 mbErrorBufFlag 0
setp h264dec2 mbErrorBufSize 0
setp h264dec2 Sei_Vui_parse_flag 0
setp h264dec2 numNALunits 0
setp h264dec2 numFrames 100
setp vpbe device /dev/video2
setp vpbe width 720
setp vpbe height 480
setp vpbe scaling 1
setp vpbe ch_mode VGA
setp vpbe output 2
setp vpbe rotate 90
func h264dec2 -s ./data/videos/davincieffect_ntsc_1_50s.264
```

- *Notes*

– Options: all decoder options can be selected.

3.5.1.6 Mpeg4 File Decode and display

- *Description*

  Read an mpeg4 encoded file, decode using a video decoder, and display on LCD.

- *Implementation*

– Input: mpeg4 Encoded file.

– Output: display on LCD

– Sample script:

```
setp engine name encodedecode
setp mpeg4spdec2 codec mpeg4dec
setp mpeg4spdec2 maxHeight 480
setp mpeg4spdec2 maxWidth 720
setp mpeg4spdec2 maxFrameRate 30000
setp mpeg4spdec2 maxBitRate 6000000
setp mpeg4spdec2 dataEndianness 1
setp mpeg4spdec2 forceChromaFormat 4
setp mpeg4spdec2 decodeHeader 0
setp mpeg4spdec2 displayWidth 0
setp mpeg4spdec2 frameSkipMode 0
```

```
setp mpeg4spdec2 frameOrder 0
setp mpeg4spdec2 newFrameFlag 0
setp mpeg4spdec2 mbDataFlag 0
setp mpeg4spdec2 postDeblock 0
setp mpeg4spdec2 postDering 0
setp mpeg4spdec2 errorConceal 0
setp mpeg4spdec2 FrameLevelByteSwap 1
setp mpeg4spdec2 numFrames 100
setp vpbe device /dev/video2
setp vpbe width 720
setp vpbe height 480
setp vpbe scaling 1
setp vpbe ch_mode VGA
setp vpbe output 2
setp vpbe rotate 90
func mpeg4spdec2 -s ./data/videos/davincieffect_ntsc_1.m4v
```

- *Notes*

– Options: all decoder options can be selected.

3.5.1.7 Mpeg2 File Decode and display

- *Description*

  Read an mpeg2 encoded file, decode using a video decoder, and display on LCD.

- *Implementation*

– Input: mpeg2 Encoded file.

– Output: display on LCD

– Sample script:

```
setp engine name encodedecode
setp mpeg2dec2 codec mpeg2dec
setp mpeg2dec2 maxHeight 480
setp mpeg2dec2 maxWidth 720
setp mpeg2dec2 maxFrameRate 30000
setp mpeg2dec2 maxBitRate 6000000
setp mpeg2dec2 dataEndianness 1
setp mpeg2dec2 forceChromaFormat 4
setp mpeg2dec2 decodeHeader 0
setp mpeg2dec2 displayWidth 0
```

```
setp mpeg2dec2 frameSkipMode 0
setp mpeg2dec2 frameOrder 0
setp mpeg2dec2 newFrameFlag 0
setp mpeg2dec2 mbDataFlag 0
setp mpeg2dec2 ppNone 0
setp mpeg2dec2 dyna_chroma_format 4
setp mpeg2dec2 displayFieldReorder 0
setp mpeg2dec2 frameLevelByteSwap 0
setp mpeg2dec2 skip_B_frame 0
setp mpeg2dec2 goto_next_I_frame 0
setp mpeg2dec2 skip_curr_frame 0
setp mpeg2dec2 seek_frame_end 0
setp mpeg2dec2 getDisplayHdrInfo 0
setp mpeg2dec2 reverse_play 0
setp mpeg2dec2 robustness_level 1
setp mpeg2dec2 no_delay_display 0
setp mpeg2dec2 numFrames 100
setp vpbe device /dev/video2
setp vpbe width 720
setp vpbe height 480
setp vpbe scaling 1
setp vpbe ch_mode VGA
setp vpbe output 2
setp vpbe rotate 90
func mpeg2dec2 -s ./data/videos/davincieffect_ntsc_1.m2v
```

- *Notes*

– Options: all decoder options can be selected.

3.5.2 Video Encode

3.5.2.1 Video capture Encode

- *Description*

  Capture video, encode using H264 encoder and store to a file.

- *Implementation*

– Input: NTSC Video stream input through component video input.

– Output :File “h264-encode.264” stored on the NFS

– Sample script:

```
setp engine name encodedecode
setp videnc1 codec h264enc
setp videnc1 encodingPreset 0
setp videnc1 rateControlPreset 1
setp videnc1 maxHeight 576
setp videnc1 maxWidth 720
setp videnc1 maxFrameRate 30000
setp videnc1 maxBitRate 2000000
setp videnc1 dataEndianness 1
setp videnc1 maxInterFrameInterval 1
setp videnc1 inputChromaFormat 4
setp videnc1 inputContentType 0
setp videnc1 reconChromaFormat -1
setp videnc1 inputHeight 480
setp videnc1 inputWidth 720
setp videnc1 refFrameRate 30000
setp videnc1 targetFrameRate 30000
setp videnc1 targetBitRate 2000000
setp videnc1 intraFrameInterval 30
setp videnc1 generateHeader 0
setp videnc1 captureWidth 0
setp videnc1 forceFrame 0
setp videnc1 interFrameInterval 1
setp videnc1 mbDataFlag 0
setp videnc1 framePitch 720
setp videnc1 numFrames 300
setp vpfe device /dev/video0
setp vpfe standard 1
setp vpfe format 1
setp vpfe input 0
setp vpfe width 720
setp vpfe height 480
func videnc1 -t h264-encode.264
```

3.5.2.2 H264 capture Encode

- *Description*

  Capture video, encode using H264 encoder and store to a file.

- *Implementation*

– Input: NTSC Video stream input through component video input.

– Output :File “h264-encode.264” stored on the NFS

– Sample script:

```
setp engine name encodedecode
setp h264enc1 codec h264enc
setp h264enc1 encodingPreset 0
setp h264enc1 rateControlPreset 1
setp h264enc1 maxHeight 576
setp h264enc1 maxWidth 720
setp h264enc1 maxFrameRate 30000
setp h264enc1 maxBitRate 2000000
setp h264enc1 dataEndianness 1
setp h264enc1 maxInterFrameInterval 1
setp h264enc1 inputChromaFormat 4
setp h264enc1 inputContentType 0
setp h264enc1 reconChromaFormat -1
setp h264enc1 profileIdc 66
setp h264enc1 levelIdc 30
setp h264enc1 rcAlgo 0
setp h264enc1 searchRange 64
setp h264enc1 inputHeight 480
setp h264enc1 inputWidth 720
setp h264enc1 refFrameRate 30000
setp h264enc1 targetFrameRate 30000
setp h264enc1 targetBitRate 2000000
setp h264enc1 intraFrameInterval 30
setp h264enc1 generateHeader 0
setp h264enc1 captureWidth 0
setp h264enc1 forceFrame 0
setp h264enc1 interFrameInterval 1
setp h264enc1 mbDataFlag 0
```

```
setp h264enc1 qpIntra 24
setp h264enc1 qpInter 24
setp h264enc1 qpMax 51
setp h264enc1 qpMin 0
setp h264enc1 lfDisableIdc 0
setp h264enc1 quartPelDisable 1
setp h264enc1 airMbPeriod 0
setp h264enc1 maxMBsPerSlice 0
setp h264enc1 maxBytesPerSlice 0
setp h264enc1 sliceRefreshRowStartNumber 0
setp h264enc1 sliceRefreshRowNumber 0
setp h264enc1 filterOffsetA 0
setp h264enc1 filterOffsetB 0
setp h264enc1 log2MaxFNumMinus4 0
setp h264enc1 chromaQPIndexOffset 0
setp h264enc1 constrainedIntraPredEnable 0
setp h264enc1 picOrderCountType 0
setp h264enc1 maxMVperMB 1
setp h264enc1 intra4x4EnableIdc 0
setp h264enc1 mvDataEnable 0
setp h264enc1 hierCodingEnable 0
setp h264enc1 streamFormat 0
setp h264enc1 intraRefreshMethod 0
setp h264enc1 perceptualQuant 0
setp h264enc1 sceneChangeDet 0
setp h264enc1 numSliceASO 0
setp h264enc1 asoSliceOrder[0] 0
setp h264enc1 asoSliceOrder[1] 0
setp h264enc1 asoSliceOrder[2] 0
setp h264enc1 asoSliceOrder[3] 0
setp h264enc1 asoSliceOrder[4] 0
setp h264enc1 asoSliceOrder[5] 0
setp h264enc1 asoSliceOrder[6] 0
setp h264enc1 asoSliceOrder[7] 0
setp h264enc1 numSliceGroups 0
setp h264enc1 sliceGroupMapType 0
```

```
setp h264enc1 sliceGroupChangeDirectionFlag 0
setp h264enc1 sliceGroupChangeRate 0
setp h264enc1 sliceGroupChangeCycle 0
setp h264enc1 sliceGroupParams[0] 0
setp h264enc1 sliceGroupParams[1] 0
setp h264enc1 sliceGroupParams[2] 0
setp h264enc1 sliceGroupParams[3] 0
setp h264enc1 sliceGroupParams[4] 0
setp h264enc1 sliceGroupParams[5] 0
setp h264enc1 sliceGroupParams[6] 0
setp h264enc1 sliceGroupParams[7] 0
setp h264enc1 framePitch 720
setp h264enc1 numFrames 300
setp vpfe device /dev/video0
setp vpfe standard 1
setp vpfe format 1
setp vpfe input 0
setp vpfe width 720
setp vpfe height 480
func h264enc1 -t h264-encode.264
```

3.5.2.3 Mpeg4 capture Encode

- *Description*

  Capture video, encode using mpeg4 encoder and store to a file.

- *Implementation*

– Input: NTSC Video stream input through component video input.

– Output :File “mp4-encode.mpeg4” stored on the NFS

– Sample script:

```
setp engine name encodedecode
setp mpeg4spenc1 codec mpeg4enc
setp mpeg4spenc1 encodingPreset 0
setp mpeg4spenc1 rateControlPreset 1
setp mpeg4spenc1 maxHeight 576
setp mpeg4spenc1 maxWidth 720
setp mpeg4spenc1 maxFrameRate 30000
setp mpeg4spenc1 maxBitRate 2000000
```

```
setp mpeg4spenc1 dataEndianness 1
setp mpeg4spenc1 maxInterFrameInterval 1
setp mpeg4spenc1 inputChromaFormat 4
setp mpeg4spenc1 inputContentType 0
setp mpeg4spenc1 reconChromaFormat -1
setp mpeg4spenc1 inputHeight 480
setp mpeg4spenc1 inputWidth 720
setp mpeg4spenc1 refFrameRate 30000
setp mpeg4spenc1 targetFrameRate 30000
setp mpeg4spenc1 targetBitRate 2000000
setp mpeg4spenc1 intraFrameInterval 30
setp mpeg4spenc1 generateHeader 0
setp mpeg4spenc1 captureWidth 0
setp mpeg4spenc1 forceFrame 0
setp mpeg4spenc1 interFrameInterval 1
setp mpeg4spenc1 mbDataFlag 0
setp mpeg4spenc1 encodeMode 1
setp mpeg4spenc1 levelIdc 5
setp mpeg4spenc1 VM4RCnumFrames 2147483647
setp mpeg4spenc1 rcAlgo 8
setp mpeg4spenc1 vbvBufferSize 112
setp mpeg4spenc1 vbvParamEnable 0
setp mpeg4spenc1 useVOS 1
setp mpeg4spenc1 useGOV 0
setp mpeg4spenc1 useDataPartition 0
setp mpeg4spenc1 useRVLC 0
setp mpeg4spenc1 maxDelay 100
setp mpeg4spenc1 h263SliceMode 0
setp mpeg4spenc1 resyncInterval 0
setp mpeg4spenc1 hecInterval 0
setp mpeg4spenc1 airRate 0
setp mpeg4spenc1 mirRate 0
setp mpeg4spenc1 qpIntra 8
setp mpeg4spenc1 qpInter 3
setp mpeg4spenc1 fCode 3
setp mpeg4spenc1 useHpi 1
```

```
setp mpeg4spenc1 useAcPred 0
setp mpeg4spenc1 lastFrame 0
setp mpeg4spenc1 MVDataEnable 0
setp mpeg4spenc1 useUMV 1
setp mpeg4spenc1 use4MV 0
setp mpeg4spenc1 ResyncDataEnable 0
setp mpeg4spenc1 framePitch 720
setp mpeg4spenc1 numFrames 300
setp vpfe device /dev/video0
setp vpfe standard 1
setp vpfe format 1
setp vpfe input 0
setp vpfe width 720
setp vpfe height 480
func mpeg4spenc1 -t mp4-encode.mpeg4
```

3.5.2.4 Video File Encode

- *Description*

  Read a 422 interleaved file, encode using a video encoder, store encoded file.

- *Implementation*

– Input: YUV 422 interleaved file.

– Output: Encoded file.

– Sample script:

```
setp engine name encodedecode
setp videnc1 codec h264enc
setp videnc1 encodingPreset 0
setp videnc1 rateControlPreset 1
setp videnc1 maxHeight 576
setp videnc1 maxWidth 720
setp videnc1 maxFrameRate 30000
setp videnc1 maxBitRate 2000000
setp videnc1 dataEndianness 1
setp videnc1 maxInterFrameInterval 1
setp videnc1 inputChromaFormat 4
setp videnc1 inputContentType 0
setp videnc1 reconChromaFormat -1
```

```
setp videnc1 inputHeight 480
setp videnc1 inputWidth 720
setp videnc1 refFrameRate 30000
setp videnc1 targetFrameRate 30000
setp videnc1 targetBitRate 2000000
setp videnc1 intraFrameInterval 30
setp videnc1 generateHeader 0
setp videnc1 captureWidth 0
setp videnc1 forceFrame 0
setp videnc1 interFrameInterval 1
setp videnc1 mbDataFlag 0
setp videnc1 framePitch 720
setp videnc1 numFrames 300
setp vpfe device /dev/video0
setp vpfe standard 1
setp vpfe format 1
setp vpfe input 0
setp vpfe width 720
setp vpfe height 480
func videnc1 -s source-422.yuv -t source-enc-422.264
```

- *Notes*

– Encoder: all available encoders can be selected.

Options: all encoder options can be selected.

3.5.2.5 Video Loopback

3.5.2.6 Video File Encode and Decode

- *Description*

Read an yuv file, encode it, decode if and dump the output in a file.

- *Implementation*

– Input: YUV file.

– Output: YUV file

– Sample script:

```
setp engine name encodedecode
setp videnc1 codec h264enc
setp videnc1 inputChromaFormat 4
```

```
setp videnc1 encodingPreset 0
setp videnc1 rateControlPreset 1
setp videnc1 maxHeight 480
setp videnc1 maxWidth 720
setp videnc1 inputHeight 480
setp videnc1 inputWidth 720
setp videnc1 maxBitRate 4000000
setp videnc1 targetBitRate 4000000
setp videnc1 numFrames 200
setp viddec2 codec h264dec
setp viddec2 forceChromaFormat 4
func vidloopback1 –s source.yuv –t destination.yuv
```

- *Notes*

– Encoder: only H264 encoder can be selected

– Decoder: only H264 decoder can be selected

Options: all decoder and encoder options can be selected

3.5.3 Audio Use Cases

3.5.3.1 Audio Playback

3.5.3.2 Audio Playback Elementary Stream

- *Description*

  Read a PCM file, Play.

- *Implementation*

– Input: PCM file.

– Output: Play

– Sample script:

```
setp engine name encodedecode
setp audio samplerate 32000
setp audio channels 2
setp auddec1 codec aachedec
func auddec1 –s source.pcm --nodsp
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.5.3.3 Audio File Decode

- *Description*

  Read an encoded file, decode using a audio decoder, store decoded file.

- *Implementation*

– Input: Audio encoded file.

– Output: Decoded PCM file

– Sample script:

```
setp engine name encodedecode
setp auddec1 codec aachedec
func auddec1 –s source.aac –t audDecOutput.pcm
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.5.3.4 Audio File Decode and Play

- *Description*

  Read an encoded file, decode using a audio decoder, play it.

- *Implementation*

– Input: Audio encoded file.

– Output: Play

– Sample script:

```
setp engine name encodedecode
setp auddec1 codec aachedec
func auddec1 –s source.aac
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.5.4 Speech Use Cases

3.5.4.1 Speech Playback and Capture

3.5.4.2 Speech Capture and Encode

- *Description*

Capture speech, encode using speech codec and store to a file.

- *Implementation*

– Input: Speech stream.
– Output: Encoded file.
– Sample script:

```
setp engine name encodedecode
setp sphenc1 seconds 300
func sphenc1 –t capture.g711
```

- *Notes*

– Encoder: all available encoders can be selected.
– Options: all encoder options can be selected.

3.5.4.3 Speech Playback Encoded Streams

- *Description*

Read a speech encoded file, decode and play out.

- *Implementation*

– Input: Speech encoded file.
– Output: Speech.
– Sample script:

```
setp engine name encodedecode
func sphdec1 –s source.g711
```

- *Notes*

– Decoder: all available decoders can be selected.
– Options: all decode options can be selected.

3.5.4.4 Speech capture and dump PCM Streams

- *Description*

Read a speech pcm file, and play out.

- *Implementation*

– Input: capture from driver
– Output: dump in a file
– Sample script:

```
func sphenc1 –t source.pcm –-nodsp
```

- *Notes*

– Decoder:
– Options:

3.5.4.5 Speech File Encode and Decode

3.5.4.6 Speech File Encode

- *Description*

Read PCM file, encode using speech codec and store the encoded file.

- *Implementation*

– Input: PCM file.
– Output: Encoded file.
– Sample script:

```
setp engine name encodedecode
func sphenc1 –s source.pcm –t source-enc.g711
```

- *Notes*

– Encoder: all available encoders can be selected.
– Options: all encoder options can be selected.

3.5.4.7 Speech File Decode

- *Description*

Read an encoded file, decode using a speech decoder, store decoded file.

- *Implementation*

– Input: Encoded file.
– Output: PCM file.
– Sample script:

```
setp engine name encodedecode
func sphdec1 –s source.g711 –t source-dec-sph.pcm
```

- *Notes*

– Decoder: all available decoders can be selected.
– Options: all decoder options can be selected.

3.5.5 Image Use Cases

3.5.5.1 Image Decode

3.5.5.2 Image File Decode and Store

- *Description*

  Decode JPEG image and store

- *Implementation*
  – Input: JPEG file
  – Output: output.yuv
  – Sample script:

```
setp engine name encodedecode
setp imgdec1 codec jpegdec
setp imgdec1 maxHeight 480
setp imgdec1 maxWidth 720
setp imgdec1 forceChromaFormat 4
func imgdec1 -s source_720x480.jpg -t output.yuv
```

- *Notes*
  – Decoder: all available decoders can be selected.
  – Options: all decoder options can be selected.

3.5.5.3 Image File Decode and Display

- *Description*

  Decode JPEG image and store

- *Implementation*
  – Input: JPEG file

- Output: display
- Sample script:

```
setp engine name encodedecode
setp imgdec1 codec jpegdec
setp imgdec1 maxHeight 480
setp imgdec1 maxWidth 720
setp imgdec1 forceChromaFormat 4
func imgdec1 -s source_720x480.jpg
```

- *Notes*
  - Decoder: all available decoders can be selected.
  - Options: all decoder options can be selected.

3.5.5.4 Image Encode

3.5.5.5 Image File Encode

- *Description*

  Read YUV, encode it using image encoder and store the encoded file

- *Implementation*
  - Input: YUV
  - Output: JPEG file
  - Sample script:

```
setp engine name encodedecode
setp imgenc1 codec jpegenc
setp imgenc1 maxHeight 480
setp imgenc1 maxWidth 720
setp imgenc1 forceChromaFormat 2
setp imgenc1  inputChromaFormat 4
func imgenc1 -s source_720x480.yuv –t output.jpg
```

- *Notes*
  - Decoder: all available encoders can be selected.
  - Options: all encoder options can be selected.

3.5.6 Simultaneous Operations - Use Cases

3.5.6.1 Audio-Video Playback

- *Description*

Read encoded audio/video files and play it out through the audio driver and VPBE.

- *Implementation*
  - Input: Audio encoded file, Video encoded file
  - Output: Audio stream played out on line-out/headphone
    Video played out on the VPBE

– Sample script:

```
setp engine name encodedecode
setp viddec2 codec mpeg4dec
setp viddec2 forceChromaFormat 4
setp vpbe rotate 90
func viddec2 –s source.m4v
setp auddec1 codec aachedec1
func auddec1 –s source.aac
```

- *Notes*
  - Decoder: all available decoders can be selected.
  - Options: all decoder options can be selected.

3.5.6.2 Speech-Video Playback

- *Description*

Read encoded speech/video files and play it out through the audio driver and VPBE.

- *Implementation*
  - Input: speech encoded file, Video encoded file
  - Output: speech played out on line-out/headphone
    Video played out on the VPBE

– Sample script:

```
setp engine name encodedecode
setp viddec2 codec mpeg4dec
setp viddec2 forceChromaFormat 4
setp vpbe rotate 90
func viddec2 –s source.m4v
setp sphdec1 codec g711dec
func shpdec1 –s source.g711
```

- *Notes*
  - Decoder: all available decoders can be selected.
  - Options: all decoder options can be selected.

3.5.6.3 Audio-JPEG Playback

- *Description*

  Read encoded audio/jpeg files and play it out through the audio driver and VPBE.

- *Implementation*
  - Input: Audio encoded file, Jpeg encoded file
  - Output: Audio stream played out on line-out/headphone
    Jpeg played out on the VPBE

– Sample script:

```
setp engine name encodedecode
setp imgdec1 codec jpegdec
setp imgdec1 maxHeight 480
setp imgdec1 maxWidth 720
setp imgdec1 forceChromaFormat 4
func imgdec1 -s source_720x480.jpg
setp auddec1 codec aachedec1
func auddec1 –s source.aac
```

- *Notes*
  - Decoder: all available decoders can be selected.
  - Options: all decoder options can be selected.

3.5.6.4 Speech-Video Encode

- *Description*

  Read RAW PCM and YUV files then encode them using speech and video encoder respectively.

- *Implementation*
  - Input: PCM file, YUV file
  - Output: Dump in files
  - Sample script:

```
setp engine name encodedecode
setp videnc1 codec mpeg4enc
setp videnc1 inputChromaFormat 4
setp videnc1 maxHeight 480
setp videnc1 maxWidth 720
setp videnc1 inputHeight 480
setp videnc1 inputWidth 720
```

```
setp videnc1 numFrames 100
func videnc1 -s source-422.yuv -t source-enc-422.mpeg4
func sphenc1 –s source.pcm –t source-enc.pcm
```

- *Notes*
  - Decoder: all available decoders can be selected.
  - Options: all decoder options can be selected.

3.5.7 Power Manager- Use Cases

- *Description*

  Change the governor, P-State and C-State of power manager

- *Implementation*
  - Input:
  - Output:

  - Sample script:

```
setp pwrmanager scalingGovernor performance
setp pwrmanager vdd1PState 4
setp pwrmanager vdd2PState 3
setp pwrmanager cpuIdleState 1
func pwrmanager
```

3.6 DM3730(Linux) Usecases

3.6.1 Video Use Cases

3.6.1.1 Video Decode

3.6.1.2 Video Playback Elementary Stream

- *Description*

  Read 422 interleaved file, display

- *Implementation*
  - Input: 422 interleaved file
  - Output: video stream display through VPBE on LCD

– Sample script:

```
setp vpbe width 720
setp vpbe height 480
setp vpbe rotate 90
func viddec2 –s source.yuv –-nodsp
```

- *Notes*
  - Decoder: all available decoders can be selected.
  - Options: all decoder options can be selected.

3.6.1.3 Video File Decode

- *Description*

  Read an encoded file, decode using a video decoder, store decoded file.

- *Implementation*
  - Input: Encoded file.
  - Output: YUV 420P planar file.
  - Sample script:

```
setp engine name encodedecode
setp viddec2 codec mpeg4dec
setp viddec2 maxHeight 480
setp viddec2 maxWidth 720
setp viddec2 maxFrameRate 30000
setp viddec2 maxBitRate 6000000
setp viddec2 dataEndianness 1
setp viddec2 forceChromaFormat 4
setp viddec2 decodeHeader 0
setp viddec2 displayWidth 0
setp viddec2 frameSkipMode 0
setp viddec2 frameOrder 0
setp viddec2 newFrameFlag 0
setp viddec2 mbDataFlag 0
setp viddec2 numFrames 100
func viddec2 –s source.m4v –t source-dec-420.yuv
```

- *Notes*
  - Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.6.1.4 Video File Decode and display

- *Description*

  Read an encoded file, decode using a video decoder, and display on LCD.

- *Implementation*

– Input: Encoded file.

– Output: display on LCD

– Sample script:

```
setp engine name encodedecode
setp viddec2 codec mpeg4dec
setp viddec2 maxHeight 480
setp viddec2 maxWidth 720
setp viddec2 maxFrameRate 30000
setp viddec2 maxBitRate 6000000
setp viddec2 dataEndianness 1
setp viddec2 forceChromaFormat 4
setp viddec2 decodeHeader 0
setp viddec2 displayWidth 0
setp viddec2 frameSkipMode 0
setp viddec2 frameOrder 0
setp viddec2 newFrameFlag 0
setp viddec2 mbDataFlag 0
setp viddec2 numFrames 100
setp vpbe device /dev/video2
setp vpbe width 720
setp vpbe height 480
setp vpbe scaling 1
setp vpbe ch_mode VGA
setp vpbe output 2
func viddec2 –s source.m4v
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.6.1.5 H264 File Decode and display

- *Description*

  Read an h264 encoded file, decode using a video decoder, and display on LCD.

- *Implementation*

– Input: H264 Encoded file.

– Output: display on LCD

– Sample script:

```
setp engine name encodedecode
setp h264dec2 codec h264dec
setp h264dec2 maxHeight 480
setp h264dec2 maxWidth 720
setp h264dec2 maxFrameRate 30000
setp h264dec2 maxBitRate 10000000
setp h264dec2 dataEndianness 1
setp h264dec2 forceChromaFormat 4
setp h264dec2 i4_num_ref_frames 16
setp h264dec2 i4_num_reorder_frames 16
setp h264dec2 decodeHeader 0
setp h264dec2 displayWidth 0
setp h264dec2 frameSkipMode 0
setp h264dec2 frameOrder 0
setp h264dec2 newFrameFlag 0
setp h264dec2 mbDataFlag 0
setp h264dec2 numFrames 100
setp vpbe device /dev/video2
setp vpbe width 720
setp vpbe height 480
setp vpbe scaling 1
setp vpbe ch_mode VGA
setp vpbe output 2
setp vpbe rotate 90
func h264dec2 -s ./data/videos/davincieffect_480p30.264
```

- *Notes*

– Options: all decoder options can be selected.

3.6.1.6 Mpeg4 File Decode and display

- *Description*

  Read an mpeg4 encoded file, decode using a video decoder, and display on LCD.

- *Implementation*

– Input: mpeg4 Encoded file.

– Output: display on LCD

– Sample script:

```
setp engine name encodedecode
setp mpeg4spdec2 codec mpeg4dec
setp mpeg4spdec2 maxHeight 480
setp mpeg4spdec2 maxWidth 720
setp mpeg4spdec2 maxFrameRate 30000
setp mpeg4spdec2 maxBitRate 6000000
setp mpeg4spdec2 dataEndianness 1
setp mpeg4spdec2 forceChromaFormat 4
setp mpeg4spdec2 post_proc_deblocking 0
setp mpeg4spdec2 decodeHeader 0
setp mpeg4spdec2 displayWidth 0
setp mpeg4spdec2 frameSkipMode 0
setp mpeg4spdec2 frameOrder 0
setp mpeg4spdec2 newFrameFlag 0
setp mpeg4spdec2 mbDataFlag 0
setp mpeg4spdec2 numFrames 100
setp vpbe device /dev/video2
setp vpbe width 720
setp vpbe height 480
setp vpbe scaling 1
setp vpbe ch_mode VGA
setp vpbe output 2
setp vpbe rotate 90
func mpeg4spdec2 -s ./data/videos/davincieffect_480p30.m4v
```

- *Notes*

– Options: all decoder options can be selected.

3.6.1.7 Mpeg2 File Decode and display

- *Description*

  Read an mpeg2 encoded file, decode using a video decoder, and display on LCD.

- *Implementation*

– Input: mpeg2 Encoded file.

– Output: display on LCD

– Sample script:

```
setp engine name encodedecode
setp mpeg2dec2 codec mpeg2dec
setp mpeg2dec2 maxHeight 480
setp mpeg2dec2 maxWidth 720
setp mpeg2dec2 maxFrameRate 30000
setp mpeg2dec2 maxBitRate 6000000
setp mpeg2dec2 dataEndianness 1
setp mpeg2dec2 forceChromaFormat 4
setp mpeg2dec2 decodeHeader 0
setp mpeg2dec2 displayWidth 0
setp mpeg2dec2 frameSkipMode 0
setp mpeg2dec2 frameOrder 0
setp mpeg2dec2 newFrameFlag 0
setp mpeg2dec2 mbDataFlag 0
setp mpeg2dec2 numFrames 100
setp vpbe device /dev/video2
setp vpbe width 720
setp vpbe height 480
setp vpbe scaling 1
setp vpbe ch_mode VGA
setp vpbe output 2
setp vpbe rotate 90
func mpeg2dec2 -s ./data/videos/davincieffect_ntsc_1.m2v
```

- *Notes*

– Options: all decoder options can be selected.

3.6.2 Video Encode

3.6.2.1 Video capture Encode

- *Description*

  Capture video, encode using H264 encoder and store to a file.

- *Implementation*

– Input: NTSC Video stream input through component video input.

– Output :File “h264-encode.264” stored on the NFS

– Sample script:

```
setp engine name encodedecode
setp videnc1 codec h264enc
setp videnc1 encodingPreset 0
setp videnc1 rateControlPreset 2
setp videnc1 maxHeight 576
setp videnc1 maxWidth 720
setp videnc1 maxFrameRate 30000
setp videnc1 maxBitRate 2000000
setp videnc1 dataEndianness 1
setp videnc1 maxInterFrameInterval 1
setp videnc1 inputChromaFormat 4
setp videnc1 inputContentType 0
setp videnc1 reconChromaFormat -1
setp videnc1 inputHeight 480
setp videnc1 inputWidth 720
setp videnc1 refFrameRate 30000
setp videnc1 targetFrameRate 30000
setp videnc1 targetBitRate 2000000
setp videnc1 intraFrameInterval 30
setp videnc1 generateHeader 0
setp videnc1 captureWidth 0
setp videnc1 forceFrame 0
setp videnc1 interFrameInterval 1
setp videnc1 mbDataFlag 0
setp videnc1 framePitch 720
setp videnc1 numFrames 300
setp vpfe device /dev/video0
```

```
setp vpfe standard 1
setp vpfe format 1
setp vpfe input 0
setp vpfe width 720
setp vpfe height 480
func videnc1 -t h264-encode.264
```

3.6.2.2 H264 capture Encode

- *Description*

  Capture video, encode using H264 encoder and store to a file.

- *Implementation*

– Input: NTSC Video stream input through component video input.

– Output :File “h264-encode.264” stored on the NFS

– Sample script:

```
setp engine name encodedecode
setp h264enc1 codec h264enc
setp h264enc1 encodingPreset 0
setp h264enc1 rateControlPreset 2
setp h264enc1 maxHeight 576
setp h264enc1 maxWidth 720
setp h264enc1 maxFrameRate 30000
setp h264enc1 maxBitRate 2000000
setp h264enc1 dataEndianness 1
setp h264enc1 maxInterFrameInterval 1
setp h264enc1 inputChromaFormat 4
setp h264enc1 inputContentType 0
setp h264enc1 reconChromaFormat -1
setp h264enc1 inputHeight 480
setp h264enc1 inputWidth 720
setp h264enc1 refFrameRate 30000
setp h264enc1 targetFrameRate 30000
setp h264enc1 targetBitRate 2000000
setp h264enc1 intraFrameInterval 30
setp h264enc1 generateHeader 0
setp h264enc1 captureWidth 0
setp h264enc1 forceFrame 0
```

```
setp h264enc1 interFrameInterval 1
setp h264enc1 mbDataFlag 0
setp h264enc1 i4_dsp_processing1 0
setp h264enc1 i4_dsp_processing2 0
setp h264enc1 i4_coding_tools1 3
setp h264enc1 i4_coding_tools2 0
setp h264enc1 i4_min_tgt_dimensions_xy 0
setp h264enc1 i4_max_search_range_xy 4194320
setp h264enc1 rcOpt_i4_init_i_qp 7
setp h264enc1 rcOpt_i4_init_p_qp 7
setp h264enc1 rcOpt_i4_init_b_qp 7
setp h264enc1 rcOpt_i4_stuffing_disabled 0
setp h264enc1 rcOpt_i4_buffer_delay 500
setp h264enc1 rcOpt_i4_limit_vbv_to_std_def_buf_size 0
setp h264enc1 vui_i4_aspect_ratio_x 1
setp h264enc1 vui_i4_aspect_ratio_y 1
setp h264enc1 vui_i4_pixel_range 0
setp h264enc1 cbr_i4_min_I_qp 2
setp h264enc1 cbr_i4_max_I_qp 31
setp h264enc1 cbr_i4_min_P_qp 2
setp h264enc1 cbr_i4_max_P_qp 31
setp h264enc1 cbr_i4_min_B_qp 2
setp h264enc1 cbr_i4_max_B_qp 31
setp h264enc1 vbr_i4_min_I_qp 2
setp h264enc1 vbr_i4_max_I_qp 31
setp h264enc1 vbr_i4_min_P_qp 2
setp h264enc1 vbr_i4_max_P_qp 31
setp h264enc1 vbr_i4_min_B_qp 2
setp h264enc1 vbr_i4_max_B_qp 31
setp h264enc1 vbr_i4_avg_channel_bandwidth 2000000
setp h264enc1 vbr_i4_peak_channel_bandwidth 2000000
setp h264enc1 iir_i4_luma_noise_var 20
setp h264enc1 iir_i4_chrm_noise_var_cb 6
setp h264enc1 iir_i4_chrm_noise_var_cr 6
setp h264enc1 iir_i4_alpha_0 50
setp h264enc1 iir_i4_alpha_1 16
```

```
setp h264enc1 iir_i4_alpha_2 8
setp h264enc1 i4_delta_time_stamp 1
setp h264enc1 framePitch 720
setp h264enc1 numFrames 300
setp vpfe device /dev/video0
setp vpfe standard 1
setp vpfe format 1
setp vpfe input 0
setp vpfe width 720
setp vpfe height 480
func h264enc1 -t h264-encode.264
```

3.6.2.3 Mpeg4 capture Encode

- *Description*

  Capture video, encode using mpeg4 encoder and store to a file.

- *Implementation*

– Input: NTSC Video stream input through component video input.

– Output :File “mp4-encode.mpeg4” stored on the NFS

– Sample script:

```
setp engine name encodedecode
setp mpeg4spenc1 codec mpeg4enc
setp mpeg4spenc1 encodingPreset 0
setp mpeg4spenc1 rateControlPreset 2
setp mpeg4spenc1 maxHeight 576
setp mpeg4spenc1 maxWidth 720
setp mpeg4spenc1 maxFrameRate 30000
setp mpeg4spenc1 maxBitRate 2000000
setp mpeg4spenc1 dataEndianness 1
setp mpeg4spenc1 maxInterFrameInterval 1
setp mpeg4spenc1 inputChromaFormat 4
setp mpeg4spenc1 inputContentType 0
setp mpeg4spenc1 reconChromaFormat -1
setp mpeg4spenc1 inputHeight 480
setp mpeg4spenc1 inputWidth 720
setp mpeg4spenc1 refFrameRate 30000
setp mpeg4spenc1 targetFrameRate 30000
```

```
setp mpeg4spenc1 targetBitRate 2000000
setp mpeg4spenc1 intraFrameInterval 30
setp mpeg4spenc1 generateHeader 0
setp mpeg4spenc1 captureWidth 0
setp mpeg4spenc1 forceFrame -1
setp mpeg4spenc1 interFrameInterval 1
setp mpeg4spenc1 mbDataFlag 0
setp mpeg4spenc1 i4_dsp_processing1 0
setp mpeg4spenc1 i4_dsp_processing2 0
setp mpeg4spenc1 i4_coding_tools1 7
setp mpeg4spenc1 i4_coding_tools2 0
setp mpeg4spenc1 i4_min_tgt_dimensions_xy 0
setp mpeg4spenc1 i4_max_search_range_xy 4194336
setp mpeg4spenc1 rcOpt_i4_init_i_qp 7
setp mpeg4spenc1 rcOpt_i4_init_p_qp 7
setp mpeg4spenc1 rcOpt_i4_init_b_qp 7
setp mpeg4spenc1 rcOpt_i4_stuffing_disabled 0
setp mpeg4spenc1 rcOpt_i4_buffer_delay 500
setp mpeg4spenc1 rcOpt_i4_limit_vbv_to_std_def_buf_size 0
setp mpeg4spenc1 cbr_i4_min_I_qp 2
setp mpeg4spenc1 cbr_i4_max_I_qp 31
setp mpeg4spenc1 cbr_i4_min_P_qp 2
setp mpeg4spenc1 cbr_i4_max_P_qp 31
setp mpeg4spenc1 cbr_i4_min_B_qp 2
setp mpeg4spenc1 cbr_i4_max_B_qp 31
setp mpeg4spenc1 vbr_i4_min_I_qp 2
setp mpeg4spenc1 vbr_i4_max_I_qp 31
setp mpeg4spenc1 vbr_i4_min_P_qp 2
setp mpeg4spenc1 vbr_i4_max_P_qp 31
setp mpeg4spenc1 vbr_i4_min_B_qp 2
setp mpeg4spenc1 vbr_i4_max_B_qp 31
setp mpeg4spenc1 vbr_i4_avg_channel_bandwidth 2000000
setp mpeg4spenc1 vbr_i4_peak_channel_bandwidth 2000000
setp mpeg4spenc1 iir_i4_luma_noise_var 20
setp mpeg4spenc1 iir_i4_chrm_noise_var_cb 6
setp mpeg4spenc1 iir_i4_chrm_noise_var_cr 6
```

```
setp mpeg4spenc1 iir_i4_alpha_0 8192
setp mpeg4spenc1 iir_i4_alpha_1 4195
setp mpeg4spenc1 iir_i4_alpha_2 4195
setp mpeg4spenc1 i4_delta_time_stamp 1
setp mpeg4spenc1 framePitch 720
setp mpeg4spenc1 numFrames 300
setp vpfe device /dev/video0
setp vpfe standard 1
setp vpfe format 1
setp vpfe input 0
setp vpfe width 720
setp vpfe height 480
func mpeg4spenc1 -t mp4-encode.mpeg4
```

3.6.2.4 Video File Encode

- *Description*

  Read a 422 interleaved file, encode using a video encoder, store encoded file.

- *Implementation*

– Input: YUV 422 interleaved file.

– Output: Encoded file.

– Sample script:

```
setp engine name encodedecode
setp videnc1 codec h264enc
setp videnc1 encodingPreset 0
setp videnc1 rateControlPreset 2
setp videnc1 maxHeight 576
setp videnc1 maxWidth 720
setp videnc1 maxFrameRate 30000
setp videnc1 maxBitRate 2000000
setp videnc1 dataEndianness 1
setp videnc1 maxInterFrameInterval 1
setp videnc1 inputChromaFormat 4
setp videnc1 inputContentType 0
setp videnc1 reconChromaFormat -1
setp videnc1 inputHeight 480
setp videnc1 inputWidth 720
```

```
setp videnc1 refFrameRate 30000
setp videnc1 targetFrameRate 30000
setp videnc1 targetBitRate 2000000
setp videnc1 intraFrameInterval 30
setp videnc1 generateHeader 0
setp videnc1 captureWidth 0
setp videnc1 forceFrame 0
setp videnc1 interFrameInterval 1
setp videnc1 mbDataFlag 0
setp videnc1 framePitch 720
setp videnc1 numFrames 300
setp vpfe device /dev/video0
setp vpfe standard 1
setp vpfe format 1
setp vpfe input 0
setp vpfe width 720
setp vpfe height 480
func videnc1 -s source-422.yuv -t source-enc-422.264
```

- *Notes*

– Encoder: all available encoders can be selected.

Options: all encoder options can be selected.

3.6.2.5 Video Loopback

3.6.2.6 Video File Encode and Decode

- *Description*

Read an yuv file, encode it, decode if and dump the output in a file.

- *Implementation*

– Input: YUV file.

– Output: YUV file

– Sample script:

```
setp engine name encodedecode
setp videnc1 codec h264enc
setp videnc1 inputChromaFormat 4
setp videnc1 encodingPreset 0
setp videnc1 rateControlPreset 2
```

```
setp videnc1 maxHeight 480
setp videnc1 maxWidth 720
setp videnc1 inputHeight 480
setp videnc1 inputWidth 720
setp videnc1 maxBitRate 4000000
setp videnc1 targetBitRate 4000000
setp videnc1 numFrames 200
setp viddec2 codec h264dec
setp viddec2 forceChromaFormat 4
func vidloopback1 –s source.yuv –t destination.yuv
```

- *Notes*

– Encoder: only H264 encoder can be selected

– Decoder: only H264 decoder can be selected

Options: all decoder and encoder options can be selected

3.6.3 Audio Use Cases

3.6.3.1 Audio Playback

3.6.3.2 Audio Playback Elementary Stream

- *Description*

  Read a PCM file, Play.

- *Implementation*

– Input: PCM file.

– Output: Play

– Sample script:

```
setp engine name encodedecode
setp audio samplerate 32000
setp audio channels 2
setp auddec1 codec aachedec
func auddec1 –s source.pcm --nodsp
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.6.3.3 Audio File Decode

- *Description*

  Read an encoded file, decode using a audio decoder, store decoded file.

- *Implementation*

– Input: Audio encoded file.

– Output: Decoded PCM file

– Sample script:

```
setp engine name encodedecode
setp auddec1 codec aachedec
func auddec1 –s source.aac –t audDecOutput.pcm
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.6.3.4 Audio File Decode and Play

- *Description*

  Read an encoded file, decode using a audio decoder, play it.

- *Implementation*

– Input: Audio encoded file.

– Output: Play

– Sample script:

```
setp engine name encodedecode
setp auddec1 codec aachedec
func auddec1 –s source.aac
```

- *Notes*

– Decoder: all available decoders can be selected.

– Options: all decoder options can be selected.

3.6.4 Speech Use Cases

3.6.4.1 Speech Playback and Capture

3.6.4.2 Speech Capture and Encode

- *Description*

Capture speech, encode using speech codec and store to a file.

- *Implementation*

– Input: Speech stream.
– Output: Encoded file.
– Sample script:

```
setp engine name encodedecode
setp sphenc1 seconds 300
func sphenc1 –t capture.g711
```

- *Notes*

– Encoder: all available encoders can be selected.
– Options: all encoder options can be selected.

3.6.4.3 Speech Playback Encoded Streams

- *Description*

Read a speech encoded file, decode and play out.

- *Implementation*

– Input: Speech encoded file.
– Output: Speech.
– Sample script:

```
setp engine name encodedecode
func sphdec1 –s source.g711
```

- *Notes*

– Decoder: all available decoders can be selected.
– Options: all decode options can be selected.

3.6.4.4 Speech capture and dump PCM Streams

- *Description*

Read a speech pcm file, and play out.

- *Implementation*

– Input: capture from driver
– Output: dump in a file
– Sample script:

```
func sphenc1 –t source.pcm –-nodsp
```

- *Notes*

– Decoder:
– Options:

3.6.4.5 Speech File Encode and Decode

3.6.4.6 Speech File Encode

- *Description*

Read PCM file, encode using speech codec and store the encoded file.

- *Implementation*

– Input: PCM file.
– Output: Encoded file.
– Sample script:

```
setp engine name encodedecode
func sphenc1 –s source.pcm –t source-enc.g711
```

- *Notes*

– Encoder: all available encoders can be selected.
– Options: all encoder options can be selected.

3.6.4.7 Speech File Decode

- *Description*

Read an encoded file, decode using a speech decoder, store decoded file.

- *Implementation*

– Input: Encoded file.
– Output: PCM file.
– Sample script:

```
setp engine name encodedecode
func sphdec1 –s source.g711 –t source-dec-sph.pcm
```

- *Notes*

– Decoder: all available decoders can be selected.
– Options: all decoder options can be selected.

3.6.5 Image Use Cases

3.6.5.1 Image Decode

3.6.5.2 Image File Decode and Store

- *Description*

  Decode JPEG image and store

- *Implementation*
  – Input: JPEG file
  – Output: output.yuv
  – Sample script:

```
setp engine name encodedecode
setp imgdec1 codec jpegdec
setp imgdec1 maxHeight 480
setp imgdec1 maxWidth 720
setp imgdec1 forceChromaFormat 4
func imgdec1 -s source_720x480.jpg -t output.yuv
```

- *Notes*
  – Decoder: all available decoders can be selected.
  – Options: all decoder options can be selected.

3.6.5.3 Image File Decode and Display

- *Description*

  Decode JPEG image and store

- *Implementation*
  – Input: JPEG file

– Output: display

– Sample script:

```
setp engine name encodedecode
setp imgdec1 codec jpegdec
setp imgdec1 maxHeight 480
setp imgdec1 maxWidth 720
setp imgdec1 forceChromaFormat 4
func imgdec1 -s source_720x480.jpg
```

- *Notes*
  – Decoder: all available decoders can be selected.
  – Options: all decoder options can be selected.

3.6.5.4 Image Encode

3.6.5.5 Image File Encode

- *Description*

  Read YUV, encode it using image encoder and store the encoded file

- *Implementation*
  – Input: YUV
  – Output: JPEG file
  – Sample script:

```
setp engine name encodedecode
setp imgenc1 codec jpegenc
setp imgenc1 maxHeight 480
setp imgenc1 maxWidth 720
setp imgenc1 forceChromaFormat 2
setp imgenc1  inputChromaFormat 4
func imgenc1 -s source_720x480.yuv –t output.jpg
```

- *Notes*
  – Decoder: all available encoders can be selected.
  – Options: all encoder options can be selected.

3.6.6 Simultaneous Operations - Use Cases

3.6.6.1 Audio-Video Playback

- *Description*

Read encoded audio/video files and play it out through the audio driver and VPBE.

- *Implementation*
  - Input: Audio encoded file, Video encoded file
  - Output: Audio stream played out on line-out/headphone
    Video played out on the VPBE

– Sample script:

```
setp engine name encodedecode
setp viddec2 codec mpeg4dec
setp viddec2 forceChromaFormat 4
setp vpbe rotate 90
func viddec2 –s source.m4v
setp auddec1 codec aachedec1
func auddec1 –s source.aac
```

- *Notes*
  - Decoder: all available decoders can be selected.
  - Options: all decoder options can be selected.

3.6.6.2 Speech-Video Playback

- *Description*

  Read encoded speech/video files and play it out through the audio driver and VPBE.

- *Implementation*
  - Input: speech encoded file, Video encoded file
  - Output: speech played out on line-out/headphone
    Video played out on the VPBE

– Sample script:

```
setp engine name encodedecode
setp viddec2 codec mpeg4dec
setp viddec2 forceChromaFormat 4
setp vpbe rotate 90
func viddec2 –s source.m4v
setp sphdec1 codec g711dec
func shpdec1 –s source.g711
```

- *Notes*
  - Decoder: all available decoders can be selected.
  - Options: all decoder options can be selected.

3.6.6.3 Audio-JPEG Playback

- *Description*

  Read encoded audio/jpeg files and play it out through the audio driver and VPBE.

- *Implementation*
  - Input: Audio encoded file, Jpeg encoded file
  - Output: Audio stream played out on line-out/headphone
    Jpeg played out on the VPBE

– Sample script:

```
setp engine name encodedecode
setp imgdec1 codec jpegdec
setp imgdec1 maxHeight 480
setp imgdec1 maxWidth 720
setp imgdec1 forceChromaFormat 4
func imgdec1 -s source_720x480.jpg
setp auddec1 codec aachedec1
func auddec1 –s source.aac
```

- *Notes*
  - Decoder: all available decoders can be selected.
  - Options: all decoder options can be selected.

3.6.6.4 Speech-Video Encode

- *Description*

  Read RAW PCM and YUV files then encode them using speech and video encoder respectively.

- *Implementation*
  - Input: PCM file, YUV file
  - Output: Dump in files
  - Sample script:

```
setp engine name encodedecode
setp videnc1 codec mpeg4enc
setp videnc1 inputChromaFormat 4
setp videnc1 maxHeight 480
setp videnc1 maxWidth 720
setp videnc1 inputHeight 480
setp videnc1 inputWidth 720
```

```
setp videnc1 numFrames 100
func videnc1 -s source-422.yuv -t source-enc-422.mpeg4
func sphenc1 –s source.pcm –t source-enc.pcm
```

- *Notes*
  - Decoder: all available decoders can be selected.
  - Options: all decoder options can be selected.

3.7 OMAP3530(WinCE) Usecases

3.7.1 Video Use Cases

3.7.1.1 Video Decode

3.7.1.2 Video File Decode

- *Description*

  Read an encoded file, decode using a video decoder, store decoded file.

- *Implementation*
  - Input: Encoded file.
  - Output: YUV 422i planar file.
  - Sample script:

```
setp engine name encodedecode
setp viddec2 codec mpeg4dec
setp viddec2 maxHeight 480
setp viddec2 maxWidth 720
setp viddec2 maxFrameRate 30000
setp viddec2 maxBitRate 6000000
setp viddec2 dataEndianness 1
setp viddec2 forceChromaFormat 4
setp viddec2 decodeHeader 0
setp viddec2 displayWidth 0
setp viddec2 frameSkipMode 0
setp viddec2 frameOrder 0
setp viddec2 newFrameFlag 0
setp viddec2 mbDataFlag 0
setp viddec2 numFrames 100
```

```
func viddec2 -s \data\videos\davincieffect_ntsc_1.m4v
-t \output\out_720x480_422i.yuv
```

- *Notes*
  - Decoder: all available decoders can be selected.
  - Options: all decoder options can be selected.

3.7.1.3 Video Encode

3.7.1.4 Video File Encode

- *Description*

  Read a 422 interleaved file, encode using a video encoder, store encoded file.

- *Implementation*

– Input: YUV 422 interleaved file.

– Output: Encoded file.

– Sample script:

```
setp engine name encodedecode
setp videnc1 codec h264enc
setp videnc1 encodingPreset 0
setp videnc1 rateControlPreset 1
setp videnc1 maxHeight 240
setp videnc1 maxWidth 320
setp videnc1 maxFrameRate 30000
setp videnc1 maxBitRate 2000000
setp videnc1 dataEndianness 1
setp videnc1 maxInterFrameInterval 1
setp videnc1 inputChromaFormat 4
setp videnc1 inputContentType 0
setp videnc1 reconChromaFormat -1
setp videnc1 inputHeight 240
setp videnc1 inputWidth 320
setp videnc1 refFrameRate 30000
setp videnc1 targetFrameRate 30000
setp videnc1 targetBitRate 2000000
setp videnc1 intraFrameInterval 30
setp videnc1 generateHeader 0
```

```
setp videnc1 captureWidth 0
setp videnc1 forceFrame 0
setp videnc1 interFrameInterval 1
setp videnc1 mbDataFlag 0
setp videnc1 framePitch 720
setp videnc1 numFrames 300
func videnc1 -s \input\input_320x240_422i.yuv –t
\output\mp4-encode.mpeg4
```

- *Notes*

– Encoder: all available encoders can be selected.

Options: all encoder options can be selected.

3.8 Performance measurement for the use-cases

All the DVTB use-cases generate a “perf-data.csv” file which will contain frame-wise codec performance numbers measured from the system. Constraint is that, to capture the performance numbers for a use-case, user should prepare a DVTB script for the same and run DVTB in script mode with this script. (For Script mode, refer Sec 2.3.3)