# 建立DVSDK

## setup.sh

该脚本在host端。

|  |
| --- |
| host $ ${DVSDK}/setup.sh |
| dm368@dm368-desktop:~/ti-dvsdk\_dm368-evm\_4\_02\_00\_06$ source setup.sh  --------------------------------------------------------------------------------  TISDK setup script  This script will set up your development host for dvsdk development.  Parts of this script require administrator priviliges (sudo access).  --------------------------------------------------------------------------------  --------------------------------------------------------------------------------  Verifying Linux host distribution  Ubuntu 10.04 LTS found successfully, continuing..  --------------------------------------------------------------------------------  --------------------------------------------------------------------------------  This step will make sure you have the proper host support packages installed  using the following command: sudo apt-get install xinetd tftpd nfs-kernel-server minicom build-essential libncurses5-dev uboot-mkimage autoconf automake  Note! This command requires you to have administrator priviliges (sudo access)  on your host.  Press return to continue  正在读取软件包列表... 完成  正在分析软件包的依赖关系树  正在读取状态信息... 完成  xinetd 已经是最新的版本了。  tftpd 已经是最新的版本了。  nfs-kernel-server 已经是最新的版本了。  minicom 已经是最新的版本了。  build-essential 已经是最新的版本了。  libncurses5-dev 已经是最新的版本了。  uboot-mkimage 已经是最新的版本了。  autoconf 已经是最新的版本了。  automake 已经是最新的版本了。  下列软件包是自动安装的并且现在不需要了：  linux-headers-2.6.32-38 linux-headers-2.6.32-38-generic  使用'apt-get autoremove'来删除它们  升级了 0 个软件包，新安装了 0 个软件包，要卸载 0 个软件包，有 0 个软件包未被升级。  Package verification and installation successfully completed  --------------------------------------------------------------------------------  --------------------------------------------------------------------------------  In which directory do you want to install the target filesystem?(if this directory does not exist it will be created)  [ /home/dm368/targetfs ]“直接回车”  This step will extract the target filesystem to /home/dm368/targetfs  Note! This command requires you to have administrator priviliges (sudo access)  on your host.  Press return to continue  Note! This command requires you to have administrator priviliges (sudo access)  on your host.  Press return to continue  /home/dm368/targetfs already exists  (r) rename existing filesystem (o) overwrite existing filesystem (s) skip filesystem extraction  [r]  [r]  Path for old filesystem:  [ /home/dm368/targetfs.06102015\_21.25 ]  Successfully moved old /home/dm368/targetfs to /home/dm368/targetfs.06102015\_21.25  Successfully extracted dvsdk-dm368-evm-rootfs.tar.gz to /home/dm368/targetfs  --------------------------------------------------------------------------------  --------------------------------------------------------------------------------  This step will update the EXEC\_DIR variables in the Rules.make file  This will facilitate the SDK to install (with make install) rebuilt binaries in  /home/dm368/targetfs  The files will be available from / on the target.  This setting can be changed later by editing Rules.make and changing the  EXEC\_DIR variable.  Press return to continueRules.make edited successfully..  --------------------------------------------------------------------------------  --------------------------------------------------------------------------------  This step will export your target filesystem for NFS access.  Note! This command requires you to have administrator priviliges (sudo access)  on your host.  Press return to continue/home/dm368/targetfs already NFS exported, skipping..  \* Stopping NFS kernel daemon [ OK ]  \* Unexporting directories for NFS kernel daemon... [ OK ]  \* Exporting directories for NFS kernel daemon... [ OK ]  \* Starting NFS kernel daemon [ OK ]  --------------------------------------------------------------------------------  --------------------------------------------------------------------------------  Which directory do you want to be your tftp root directory?(if this directory does not exist it will be created for you)  [ /tftpboot ]  --------------------------------------------------------------------------------  --------------------------------------------------------------------------------  This step will set up the tftp server in the /tftpboot directory.  Note! This command requires you to have administrator priviliges (sudo access)  on your host.  Press return to continue  /tftpboot already exists, not creating..  /tftpboot/uImage-dm368-evm.bin already exists. The new installed file can be renamed and saved under the new name.  (r) rename (o) overwrite (s) skip copy  [r] s  Skipping copy of uImage-dm368-evm.bin, existing version will be used  /etc/xinetd.d/tftp already exists..  /tftpboot already exported for TFTP, skipping..  Restarting tftp server  \* Stopping internet superserver xinetd [ OK ]  \* Starting internet superserver xinetd [ OK ]  --------------------------------------------------------------------------------  --------------------------------------------------------------------------------  This step will set up minicom (serial communication application) for  DVSDK development  Which serial port do you want to use with minicom?  [ /dev/ttyS0 ]  Copied existing /home/dm368/.minirc.dfl to /home/dm368/.minirc.dfl.old  Configuration saved to /home/dm368/.minirc.dfl. You can change it further from inside  minicom, see the Software Development Guide for more information.  --------------------------------------------------------------------------------  --------------------------------------------------------------------------------  This step will set up the u-boot variables for booting the EVM.  Autodetected the following ip address of your host, correct it if necessary  [ ]  Select Capture input type :  1: Camera  2: Composite/Component  [ 1 ] 2  Select Linux kernel location:  1: TFTP  2: SD card  3: flash (refer to SDG on how to flash kernel)  [ 1 ] 2  Select root file system location:  1: NFS  2: SD card  3: flash (refer to SDG on how to flash file system)  [ 1 ] 2  Resulting u-boot variable settings:  setenv bootdelay 4  setenv baudrate 115200  setenv bootargs console=ttyS0,115200n8 rw dm365\_imp.oper\_mode=0 video=davincifb:vid0=OFF:vid1=OFF:osd0=480x272x16,4050K mem=48MB davinci\_capture.device\_type=4 vpfe\_capture.cont\_bufsize=12582912 davinci\_enc\_mngr.ch0\_output=LCD davinci\_enc\_mngr.ch0\_mode=480x272 root=/dev/mmcblk0p2 rootwait ip=off  setenv bootcmd 'mmc rescan 0; fatload mmc 0 0x80700000 uImage; bootm 0x80700000'  --------------------------------------------------------------------------------  --------------------------------------------------------------------------------  Would you like to create a minicom script with the above parameters (y/n)?  [ y ] y  Moving existing setup\_uimage-sd\_fs-sd.minicom to setup\_uimage-sd\_fs-sd.minicom.old  Successfully wrote setup\_uimage-sd\_fs-sd.minicom  Would you like to run the setup script now (y/n)? This requires you to connect  the RS-232 cable between your host and EVM as well as your ethernet cable as  described in the Quick Start Guide. Once answering 'y' on the prompt below  you will have 300 seconds to connect the board and power cycle it  before the setup times out  After successfully executing this script, your EVM will be set up. You will be  able to connect to it by executing 'minicom -w' or if you prefer a windows host  you can set up Tera Term as explained in the Software Developer's Guide.  If you connect minicom or Tera Term and power cycle the board Linux will boot.  [ y ]  ~/ti-dvsdk\_dm368-evm\_4\_02\_00\_06 ~/ti-dvsdk\_dm368-evm\_4\_02\_00\_06  Welcome to minicom 2.4  OPTIONS: I18n  Compiled on Jan 25 2010, 06:49:09.  Port /dev/ttyS0  Press CTRL-A Z for help on special keys |

# 命令行运行DVSDK demo

## loadmodule-rc

该脚本描述了CMEM2.0的特征。

|  |
| --- |
| target # vi /etc/init.d/loadmodule-rc |
| #!/bin/sh  # Default DM368 EVM Memory Map  #  # Start Addr Size Description  # -------------------------------------------  # 0x00001000 32K ARM TCM memory  # 0x80000000 48 MB Linux  # 0x83000000 12 MB Video driver memory (Linux)  # 0x83C00000 68 MB CMEM  # 0x88000000 BOTTOM ADDRESS  load () {  modprobe cmemk phys\_start=0x83C00000 phys\_end=0x88000000 allowOverlap=1 phys  modprobe irqk  modprobe edmak  modprobe dm365mmap  mknod /dev/dm365mmap c `awk "\\$2==\"dm365mmap\" {print \\$1}" /proc/devices  }  unload () {  rmmod cmemk 2>/dev/null  rmmod irqk 2>/dev/null |

## encode.txt

|  |
| --- |
| cat encode.txt  /\*  \* encode.txt  \*  \* This readme file explains the options and describes how to use 'encode demo'  \* on DM365 platform.  \*  \* Copyright (C) 2011 Texas Instruments Incorporated - http://www.ti.com/  \*  \*  \* Redistribution and use in source and binary forms, with or without  \* modification, are permitted provided that the following conditions  \* are met:  \*  \* Redistributions of source code must retain the above copyright  \* notice, this list of conditions and the following disclaimer.  \*  \* Redistributions in binary form must reproduce the above copyright  \* notice, this list of conditions and the following disclaimer in the  \* documentation and/or other materials provided with the  \* distribution.  \*  \* Neither the name of Texas Instruments Incorporated nor the names of  \* its contributors may be used to endorse or promote products derived  \* from this software without specific prior written permission.  \*  \* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS  \* "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT  \* LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR  \* A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT  \* OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,  \* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT  \* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,  \* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY  \* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT  \* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE  \* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.  \*  \*/  NAME  encode - encode video and/or audio or speech files  SYNOPSIS  encode [options...]  DESCRIPTION  This demo uses the Codec Engine to encode data from peripheral device  drivers to files. Video, audio and speech files are supported. The files  created will be encoded elementary streams of video/audio/speech.  You must supply at least one file for the demo to run.  The DM365MM and CMEM kernel modules need to be inserted for this demo  to run. Use the script 'loadmodule-rc' in the DVSDK to make sure both  kernel modules are loaded with adequate parameters.  OPTIONS  -v <video file>, --videofile <video file>  Encodes video data to the given file. The file will be created if  it doesn't exist, and truncated if it does exist. The demo  detects which type of video file is supplied using the file  extension. Supported video algorithm is MPEG4 SP, H.264 MP, MPEG2  (.mpeg4 or .m4v extension, .264, .m2v).  -s <speech file>, --speechfile <speech file>  Encodes speech data to the given file. The file will be created  if it doesn't exist, and truncated if it does exist. The demo  detects which type of speech file is supplied using the file  extension. The only supported speech algorithm as of now is  G.711 (.g711 extension).  -a <audio file>, --audiofile <audio file>  Encodes audio data to the given file. The file will be created  if it doesn't exist, and truncated if it does exist. The demo  detects which type of audio file is supplied using the file  extension. The only supported speech algorithm as of now is  AAC (.aac extension).  -y <1-5>, --display\_standard <1-5>  Sets the resolution of the display. If the captured resolution  is larger than the display it will be center clamped, and if it  is smaller the image will be centered.  1 D1 @ 30 fps (NTSC)  2 D1 @ 25 fps (PAL)  3 720P @ 60 fps [Default]  5 1080I @ 30 fps - for DM368  -r <resolution>, --resolution <resolution>  The resolution of video to encode in the format 'width'x'height'.  Default is the size of the video standard (720x480 for NTSC,  720x576 for PAL, 1280x720 for 720P).  -b <bit rate>, --videobitrate <bit rate>  This option sets the bit rate with which the video will be  encoded. Use a negative value for variable bit rate. Default is  variable bit rate.  -p <bit rate>, --soundbitrate <bit rate>  This option sets the bit rate with which the audio will be  encoded. Use a negative value for variable bit rate. Default is  96000.  -u <sample rate>, --samplerate <sample rate>  This option sets the sample rate with which the video will be  encoded. Default is 44100 Hz.  -w, --preview\_disable  Disable preview of captured video frames.  -f, --write\_disable  Disable recording of encoded file. This helps to validate  performance without file I/O.  -I, --video\_input  Video input source to use.  1 Composite  2 S-video  3 Component  4 Imager/Camera - for DM368  When not specified, the video input is chosen based on the display  video standard selected. NTSC/PAL use Composite, 720P uses  Component, and 1080I uses the Imager/Camera.  -l, --linein  This option makes the input device for sound recording be the  'line in' as opposed to the 'mic in' default.  -k, --keyboard  Enables the keyboard input mode which lets the user input  commands using the keyboard in addition to the QT-based OSD  interface. At the prompt type 'help' for a list of available  commands.  -t <seconds>, --time <seconds>  The number of seconds to run the demo. Defaults to infinite time.  -o, --osd  Enables the On Screen Display for configuration and data  visualization using a QT-based UI. If this option is not passed,  the data will be output to stdout instead.  -h, --help  This will print the usage of the demo.  EXAMPLE USAGE  First execute this script to load kernel modules required:  ./loadmodules.sh  General usage:  ./encode -h    H264 HP video encode only @720p resolution with OSD:  ./encode -v test.264 -y 3 -o    H264 HP video encode from s-video and G.711 speech encode:  ./encode -v test.264 -s test.g711 -I 2  MPEG4 SP video encode only in CIF NTSC resolution with OSD:  ./encode -v test.mpeg4 -r 352x240 -o  MPEG4 SP video encode at 1Mbps with keyboard interface on D1 PAL display:  ./encode -v test.mpeg4 -b 1000000 -k -y 2  COPYRIGHT  Copyright (c) Texas Instruments Inc 2011  Use of this software is controlled by the terms and conditions found in  the license agreement under which this software has been supplied or  provided.  KNOWN ISSUES  VERSION  4.02  CHANGELOG  from 4.01:  Modified data flow to eliminate frame copies to maximize  performance.  Modified thread priorities to enhance performance.  from 4.0:  Added support of 1080I display video standard for DM368  Added support of Camera input for DM368  Removed s-video (-x) option.  Added -I option to select video input.  Added options to disable preview and recording of encoded output.  from 3.10:  Replaced old remote-driven interface with QT-based interface. This  can be turned on using the '-o' flag.  Added audio thread to allow audio encode.  Added options for audio support ('-a', '-p', '-u').  Added MPEG2 support.  SEE ALSO  For documentation and release notes on the individual components see  the html files in the host installation directory. |
|  |
|  |

## encode -h

|  |
| --- |
| ./encode -h  Usage: encode [options]  Options:  -s | --speechfile Speech file to record to  -a | --audiofile Audio file to play  -v | --videofile Video file to record to  -y | --display\_standard Video standard to use for display (see below).  Same video standard is used at input.  -r | --resolution Video resolution ('width'x'height')  [video standard default]  -b | --videobitrate Bit rate to encode video at [variable]  -p | --soundbitrate Bit rate to encode audio at [96000]  -u | --samplerate Sample rate to encode audio at [16000]  -w | --preview\_disable Disable preview [preview enabled]  -f | --write\_disable Disable recording of encoded file [enabled]  -I | --video\_input Video input source [video standard default]  -l | --linein Use linein as sound input instead of mic  [off]  -k | --keyboard Enable keyboard interface [off]  -t | --time Number of seconds to run the demo [infinite]  -o | --osd Show demo data on an OSD [off]  -h | --help Print this message  Video standards available  1 D1 @ 30 fps (NTSC)  2 D1 @ 25 fps (PAL)  3 720P @ 60 fps [Default]  5 1080I @ 30 fps - for DM368  Video inputs available:  1 Composite  2 S-video  3 Component  4 Imager/Camera - for DM368  You must supply at least a video or a speech file or both  with appropriate extensions for the file formats. |

## decode.txt

|  |
| --- |
| cat decode.txt  /\*  \* decode.txt  \*  \* This readme file provides the necessary information to the user for  \* using the decode demo for DM365 platform  \*  \* Copyright (C) 2011 Texas Instruments Incorporated - http://www.ti.com/  \*  \*  \* Redistribution and use in source and binary forms, with or without  \* modification, are permitted provided that the following conditions  \* are met:  \*  \* Redistributions of source code must retain the above copyright  \* notice, this list of conditions and the following disclaimer.  \*  \* Redistributions in binary form must reproduce the above copyright  \* notice, this list of conditions and the following disclaimer in the  \* documentation and/or other materials provided with the  \* distribution.  \*  \* Neither the name of Texas Instruments Incorporated nor the names of  \* its contributors may be used to endorse or promote products derived  \* from this software without specific prior written permission.  \*  \* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS  \* "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT  \* LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR  \* A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT  \* OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,  \* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT  \* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,  \* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY  \* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT  \* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE  \* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.  \*  \*/  NAME  decode - decode video and/or audio or speech files  SYNOPSIS  decode [options...]  DESCRIPTION  This demo uses the Codec Engine to decode data from files and output  the uncompressed data using peripheral device drivers. Video, audio and  speech files are supported. All files must consist of raw encoded  frames of data (elementary streams). For MPEG4 and H264 video, we use  the restricted version of the decoder, hence only videos encoded with  TI encoder can be decoded.  You must supply at least one file for the demo to run.  The DM365MM and CMEM kernel modules need to be inserted for this demo  to run. Use the script 'loadmodule-rc' in the DVSDK to make sure both  kernel modules are loaded with adequate parameters.  OPTIONS  -v <video file>, --videofile <video file>  Decodes a video file. The demo detects which type of video file  is supplied using the file extension. Supported video algorithms  are:  MPEG4 SP (.mpeg4 or .m4v extension)  H.264 HP (.264 extension)  MPEG2 (.m2v extension)  -s <speech file>, --speechfile <speech file>  Decodes the speech file. The demo detects which type of speech  file is supplied using the file extension. The only supported  speech algorithm as of now is G.711 (.g711 extension).  -a <audio file>, --audiofile <audio file>  Decodes the audio file. The demo detects which type of audio  file is supplied using the file extension. The only supported  audio algorithm as of now is AAC (.aac extension).  -y <1-3>, --display\_standard <1-3>  Sets the resolution of the display. Clips larger than the display  will be center clamped, and clips smaller than the display will  be centered.  1 D1 @ 30 fps (NTSC)  2 D1 @ 25 fps (PAL)  3 720P @ 60 fps [Default]  5 1080I @ 30 fps - for DM368  -O <display output>, --display\_output  Selects the video output to display, valid outputs are  'composite', or 'component'.  -l, --loop  If this option is selected, the demo will restart at the  beginning of a file (or files) when the end is reached. If this  option is not selected (which is the default) the demo will quit  once the end of the file (or files) has been reached.  -o, --osd  Enables the On Screen Display for configuration and data  visualization using a QT-based UI. If this option is not passed,  the data will be output to stdout instead.  -k, --keyboard  Enables the keyboard input mode which lets the user input  commands using the keyboard in addition to the QT-based OSD  interface. At the prompt type 'help' for a list of available  commands.  -t <seconds>, --time <seconds>  The number of seconds to run the demo. Defaults to infinite time.  -h, --help  This will print the usage of the demo.  EXAMPLE USAGE  First execute this script to load the kernel modules required:  ./loadmodules.sh  General usage:  ./decode -h  H264 HP video on a 720p display and G.711 speech decode with OSD:  ./decode -s test.g711 -v test.264 -o -y 3  MPEG4 NTSC video decode only with keyboard interface on component output:  ./decode -v test.mpeg4 -k -y 1  COPYRIGHT  Copyright (c) Texas Instruments Inc 2011  Use of this software is controlled by the terms and conditions found in  the license agreement under which this software has been supplied or  provided.  KNOWN ISSUES  VERSION  4.02  CHANGELOG  from 4.01:  Modified data flow to eliminate frame copies to maximize  performance.  To improve performance further, we have also moved to using the  restricted (closed-loop) version of the H264 and MPEG4 decoder.  The result is that only clips encoded with TI encoders can be  played back using this version of the demo.  from 4.0:  No change.  from 3.10:  Replaced old remote-driven interface with QT-based interface. This  can be turned on using the '-o' flag.  Added audio thread to allow audio playback.  Added option for audio support ('-a').  Added MPEG2 support.  SEE ALSO  For documentation and release notes on the individual components see  the html files in the host installation directory. |

## decode -h

|  |
| --- |
| ./decode -h  Usage: decode [options]  Options:  -a | --audiofile Audio file to play  -s | --speechfile Speech file to play  -v | --videofile Video file to play  -y | --display\_standard Video standard to use for display (see below).  -O | --display\_output Video output to use (see below).  -k | --keyboard Enable keyboard interface [off]  -t | --time Number of seconds to run the demo [infinite]  -l | --loop Loop to beginning of files when done [off]  -o | --osd Show demo data on an OSD [off]  -h | --help Print this message  Video standards available:  1 D1 @ 30 fps (NTSC)  2 D1 @ 25 fps (PAL)  3 720P @ 60 fps [Default]  5 1080I @ 30 fps - for DM368  Video outputs available:  composite  component (Only 720P and 1080I available) [Default]  You must supply at least a video or a speech or an audio file  with appropriate extensions for the file formats.  You must NOT supply BOTH an audio and a speech file. |

## encodedecode.txt

|  |
| --- |
| cat encodedecode.txt  /\*  \* encodedecode.txt  \*  \* This readme file explains the options and describes how to use  \* 'encodedecode demo' on DM365 platform.  \*  \* Copyright (C) 2011 Texas Instruments Incorporated - http://www.ti.com/  \*  \*  \* Redistribution and use in source and binary forms, with or without  \* modification, are permitted provided that the following conditions  \* are met:  \*  \* Redistributions of source code must retain the above copyright  \* notice, this list of conditions and the following disclaimer.  \*  \* Redistributions in binary form must reproduce the above copyright  \* notice, this list of conditions and the following disclaimer in the  \* documentation and/or other materials provided with the  \* distribution.  \*  \* Neither the name of Texas Instruments Incorporated nor the names of  \* its contributors may be used to endorse or promote products derived  \* from this software without specific prior written permission.  \*  \* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS  \* "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT  \* LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR  \* A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT  \* OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,  \* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT  \* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,  \* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY  \* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT  \* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE  \* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.  \*  \*/  NAME  encodedecode - encode and decode video  SYNOPSIS  encodedecode [options...]  DESCRIPTION  This demo uses the Codec Engine to encode data from the capture device  into an intermediate buffer before the data is decoded to the display  framebuffer.  The DM365MM and CMEM kernel modules need to be inserted for this demo  to run. Use the script 'loadmodule-rc' in the DVSDK to make sure both  kernel modules are loaded with adequate parameters.  OPTIONS  -y <1-7>, --display\_standard <1-7>  Sets the resolution of the display. If the captured resolution  is larger than the display it will be center clamped, and if it  is smaller the image will be centered.  1 D1 @ 30 fps (NTSC)  2 D1 @ 25 fps (PAL)  3 720P @ 60 fps [Default]  7 480P @ 60 fps  -v <videocodec>, --videocodec <h264 or mpeg4 or mpeg2>  The video codec to be used for encode and decode  -r <resolution>, --resolution <resolution>  The resolution of video to encode and decode in the format  'width'x'height'. Default is the resolution of the input video  standard detected.  -b <bit rate>, --bitrate <bit rate>  This option sets the bit rate with which the video will be  encoded. Use a negative value for variable bit rate. Default is  variable bit rate.  -p, --passthrough  Pass the video through from capture device to display device  without encoding or decoding the data.  -I, --video\_input  Video input source to use.  1 Composite  2 S-video  3 Component  4 Imager/Camera - for DM368  When not specified, the video input is chosen based on the display  video standard selected. NTSC/PAL use Composite, and 480P/720P use  Component.  -k, --keyboard  Enables the keyboard input mode which lets the user input  commands using the keyboard in addition to the QT-based OSD  interface. At the prompt type 'help' for a list of available  commands.  -t <seconds>, --time <seconds>  The number of seconds to run the demo. Defaults to infinite time.  -o, --osd  Enables the On Screen Display for configuration and data  visualization using a QT-based UI. If this option is not passed,  the data will be output to stdout instead.  -h, --help  This will print the usage of the demo.  EXAMPLE USAGE  First execute this script to load kernel modules required:  ./loadmodules.sh  General usage:  ./encodedecode -h  Use NTSC CIF resolution with keyboard interface and quit after 20 sec:  ./encodedecode -r 352x240 -v h264 -t 20 -k  Use a PAL display with OSD display and mpeg4 codec:  ./encodedecode -y 2 -v mpeg4 -o  Use default resolution (720x480) at 2Mbit bit rate with mpeg4 codec and s-video input:  ./encodedecode -b 2000000 -I 2  COPYRIGHT  Copyright (c) Texas Instruments Inc 2011  Use of this software is controlled by the terms and conditions found in  the license agreement under which this software has been supplied or  provided.  KNOWN ISSUES  VERSION  4.02  CHANGELOG  from 4.01:  Modified data flow to eliminate frame copies to maximize  performance. The application is no longer centering the display as  a result.  To improve performance further, we have also moved to using the  restricted (closed-loop) version of the H264 and MPEG4 decoder.  The application now defaults to 720P resolution instead of D1 when  display standard -y3 is selected. Use the -r flag if D1 resolution  is needed.  from 4.0:  Added support of Camera input for DM368.  Removed s-video (-x) option.  Added option to select video input source (-I)  from 3.10:  Replaced old remote-driven interface with QT-based interface. This  can be turned on using the '-o' flag.  Added MPEG2 support.  SEE ALSO  For documentation and release notes on the individual components see  the html files in the host installation directory. |

# 运行DMAI应用

DMAI(Davinci Multimedia Application Interface)

## 视频

从H.264编码视频解码30帧为YUV文件：

|  |
| --- |
| target # ./video\_decode\_io2\_dm368.x470MV -c h264dec -e decode -i /usr/share\  /ti/data/videos/davincieffect.264 -n 30 -o output.yuv |

从分辨率1280\*720的YUV编码为25帧H.264编码文件。

|  |
| --- |
| target # ./video\_encode\_io1\_dm368.x470MV -c h264enc -i output.yuv -o output.264 \  -n 25 -e encode -r 1280x720 |

1080p component输入到1080icomponent输出执行：

|  |
| --- |
| target # ./video\_loopback\_dm368.x470MV -O component -I camera -y5 |

## 执行DVTB

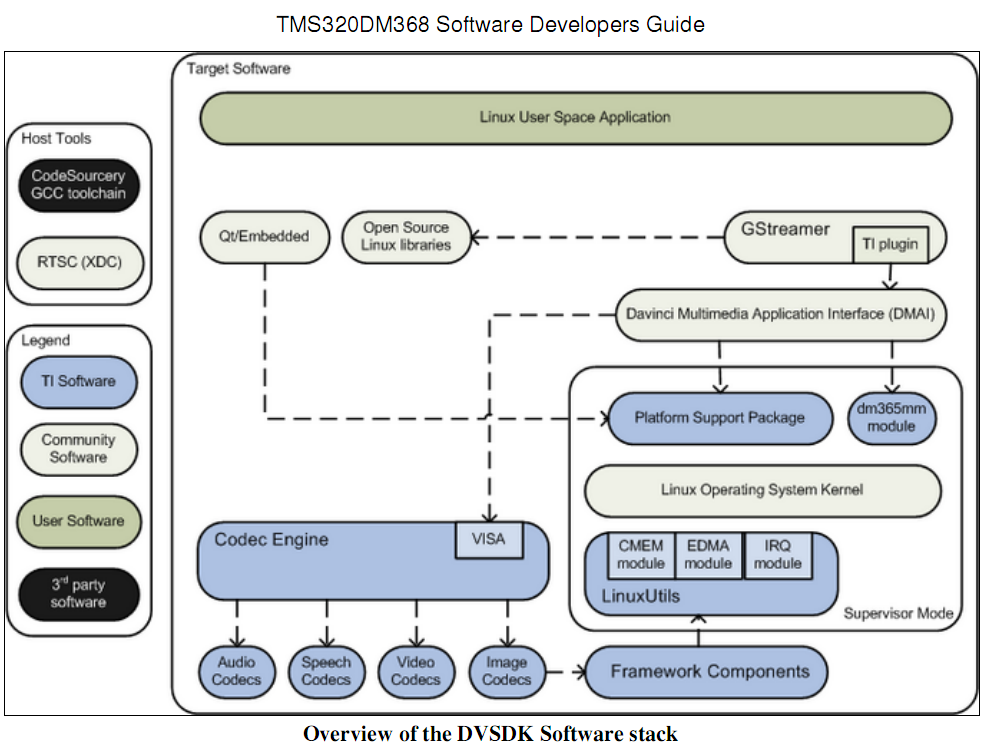
DVTB：Digital Video Test Bench

|  |
| --- |
| getp viddec2  codec => h264dec2  maxHeight => 720  maxWidth => 1280  maxFrameRate => 30000  maxBitRate => 10000000  dataEndianness => 1  forceChromaFormat => 9  decodeHeader => 0  displayWidth => 0  frameSkipMode => 0  frameOrder => 0  newFrameFlag => 0  mbDataFlag => 0  numFrames => 30  PASS: getp |

## 执行Gstreamer流水线

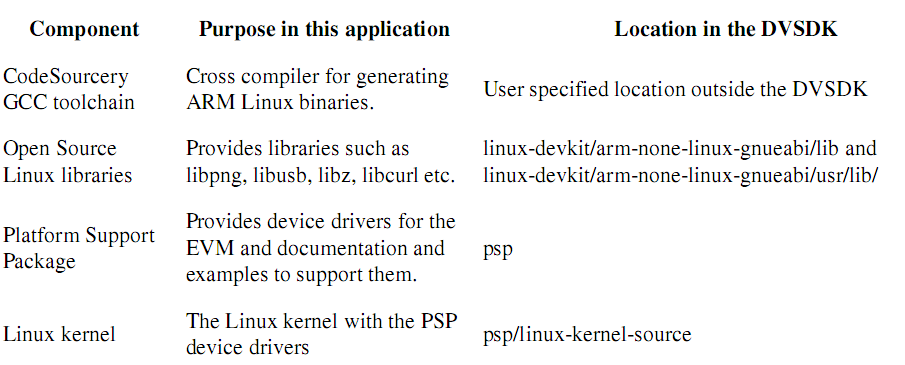
|  |
| --- |
| target # /etc/init.d/loadmodule-rc restart  target # cat /dev/zero > /dev/fb2 2> /dev/null |

# TMS320DM368 软件结构



蓝色标的部分是TI开发的。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 分类 | 名称 | | 作用 | 备注 | |
| HostTool  （主机工具） | Codesourcery gcc toochain | | 编译工具链 |  | |
| RTSC（XDC） | | 实时工具包 | xdctools\_xx\_xx\_xx\_xx | |
| Legend  （第三方） | TI software | | TI软件 |  | |
| Community Software | | 社区软件 |  | |
| User software | | 用户软件 |  | |
| 3rd party software | | 第三方软件 |  | |
| Target software  目标软件 | Linux user space application | | Linux用户态应用 |  | |
| Qt/embedded | | QT/嵌入式 | linux-devkit/arm-none-linux-gnueabi/usr/lib/libQt\*  提供GUI工具包 | |
| Opensource Linux libraries | | 开源linux库 | linux-devkit/arm-none-linux-gnueabi/lib and  linux-devkit/arm-none-linux-gnueabi/usr/lib/ | |
| Gstreamer（TI plugin） | | （TI插件） | 多媒体框架  linux-devkit/arm-none-linux-gnueabi/usr/lib | |
| Davinci Multimedia application interface | | 达芬奇多媒体应用接口 | dmai\_xx\_xx\_xx\_xx | |
| Platform support package | | 平台支持包  设备驱动、文档、实例 | psp | 超级用户模式 |
| Dm365 mm module | |  |  |
| Linux operating system kernel | | linuxOS内核 | psp/linux-kernel-source |
| Codec engine（VISA） | | 编解码引擎  VISA接口 | codec\_engine\_xx\_xx\_xx\_xx |
| Linuxutils | CMEM module | 连续内存 | linuxutils\_xx\_xx\_xx\_xx |
| EDMA module | DMA |  |
| IRQ module | 多个协处理器资源的中断处理 |  |
| Audio codecs | | 音频编解码 |  | |
| Speech codecs | | 语音编解码 |  | |
| Video codecs | | 视频编解码 | codecs\_<platform>\_xx\_xx\_xx\_xx | |
| Image codec | | 图像编解码 |  | |
| Framework component | | 框架组件 | framework\_components\_xx\_xx\_xx\_xx  算法资源的交叉平台框架 | |



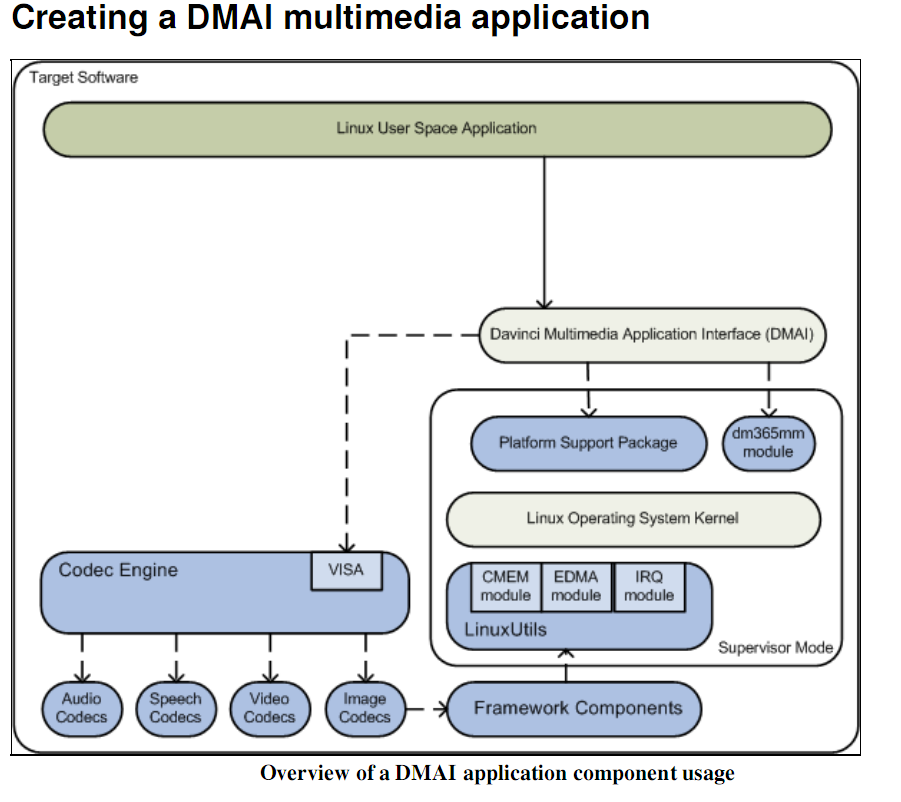
# 创建一个DMAI多媒体应用

DMAI是一个在codec引擎与linux内核上上的薄应用层。使用DMAI的优点：

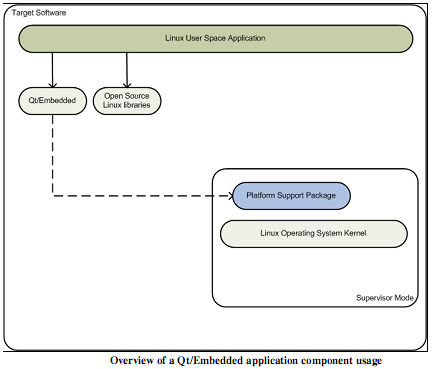
* DMAI及其实例应用采用XDM1.X语法实现多媒体编解码。编解码引擎方便编解码的开发，但是DMAI提供了使用编解码插件及运行的语法。
* DMAI将linux设备驱动封装在关注API的多媒体函数中，能让用户远离linux内核，增加了编码的方便性。
* DVSDK demo与gst-ti插件基于DMAI编写。如果可以基于DMAI样例应用编写codec，其就像这些应用一样的工作。

从下面开始一个应用：

* DMAI样例应用提供了小而简单的实例（dmai\_xx\_xx\_xx\_xx/packages/ti/sdo/dmai/apps），说明了如何使用DMAI创建一个多媒体应用。
* 如果有专门的设备，DVSDK demo位于dvsdk\_demos\_xx\_xx\_xx\_xx。这些使用DMAI提供完整的多媒体应用。但是，应用不支持A/V同步，这对于GStreamer是一个更好的选项。



# 创建QT/嵌入式应用



Qt/Embedded is a Graphical User Interface toolkit for rendering graphics to the Linux framebuffer device, and

is included in this kit. The base Qt toolkit on the other hand renders the graphics to the X11 graphical user

interface instead of to the basic framebuffer.

In addition to the components used for the basic Linux app, these are used (and the rest is greyed out in the

diagram above):

Component Purpose in this application Location in the DVSDK

Qt/Embedded

Provides a Graphical User Interface

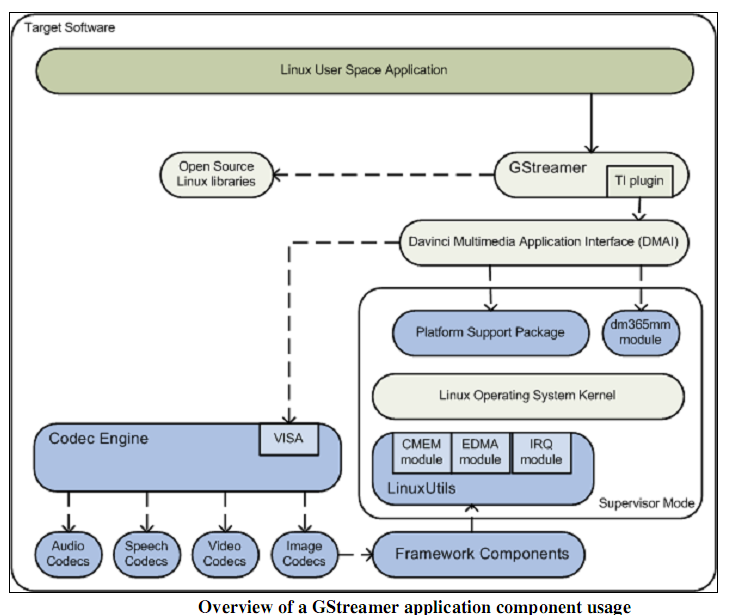
toolkit

linux-devkit/arm-none-linux-gnueabi/usr/lib/libQt\*

See the Qt Reference Documentation on various API's and its usages. You can also download some Qt/e

example applications from Qt Examples web page.

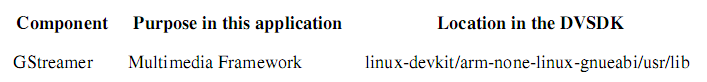
# 创建一个Gstreamer应用



Gstreamer是一个开源的多媒体框架，帮助来构建连接插件来处理多媒体内容流水线。通过使用DMAI和CODEC引擎来加速多媒体。

与直接基于DMAI创建应用相比，可以获得A/V同步，访问很多有用的开源插件。例如可以demux avi文件与mp4文件。缺点是增加了复杂性和开销。

另外，DMAI应用，使用以下组件：



来构建自己的流水线，这里有很多实例来说明如何使用开源插件。参考资料：

* GStreamer Application Development Manual
* GStreamer 0.10 Core Reference Manual

## 编译应用

DVSDK Linux开发包include了Gstreamer开发头文件、库和包配置。

# 其他程序

## 建立交叉编译环境

为了使能应用开发，DVSDK与包含开发中需要的包header、库和其他包相关信息的Linux-devkit一起提供。执行以下命令可配置交叉编译环境：

|  |
| --- |
| host $ source ${DVSDK}/linux-devkit/environment-setup |

以上命令会输出交叉编译特定环境变量。会注意到命令会增加linux-devkit到bash提示符来指示已经输出需要的交叉编译变量。



## 重新编译DVSDK组件

DVSDK提供了顶层的makefile来重新编译DVSDK中的多个组件。

注意：DVSDK组件编译环境是自约束的，并且不要求设置交叉编译环境，因此应该避免可能的编译失败。

重新编译DVSDK组件：

|  |
| --- |
| host $ cd ${DVSDK} |

DVSDK有一大堆编译目标容许重新编译DVSDK组件。

|  |
| --- |
| host $ make help |
| [linux-devkit]:~/ti-dvsdk\_dm368-evm\_4\_02\_00\_06> make help  Available build targets are:  components : Rebuild all base components (See Note below)  all : Builds all components and applications/examples  clean : Remove files generated by the 'all' target  install : Install all the targets in /home/dm368/targetfs  demos : Build the DVSDK demos for dm368  demos\_clean : Remove generated DVSDK demo files.  demos\_install : Install DVSDK demos  dvtb : Build DVTB for dm368  dvtb\_clean : Remove generated DVTB files  dvtb\_install : Install dvtb  cmem : Build the CMEM kernel module  cmem\_clean : Remove generated cmem files.  cmem\_install : Install cmemk module  irq : Build the irq kernel module  irq\_clean : Remove generated irq files.  irq\_install : Install irqk module  edma : Build the edma kernel module  edma\_clean : Remove generated edma files.  edma\_install : Install edmak module  dm365mm : Build the dm365mm kernel module  dm365mm\_clean : Remove generated dm365mm files  dm365mm\_install : Install dm365mm module  dmai : Build DMAI  dmai\_clean : Remove generated DMAI files.  dmai\_install : Install dmai app  ceexamples : Build CE codecs, extensions, servers and apps  ceexamples\_clean : Remove generated CE example files.  linux : Build Linux kernel uImage and module  linux\_config : Launch Linux kernel menuconfig  linux\_clean : Remove generated Linux kernel files  linux\_install : Install kernel binary and modules  u-boot : Build the u-boot boot loader  u-boot\_clean : Remove generated u-boot files  u-boot\_install : Install the u-boot image  psp\_examples : Build Linux examples  psp\_examples\_clean : Remove all generated PSP examples  psp\_examples\_install : Install psp examples  matrix : Build matrix application launcher  matrix\_clean : Remove all matrix files  matrix\_install : Install matrix  gstreamer\_ti : Build gstreamer-ti plugins  gstreamer\_ti\_install : Install gstreamer-ti plugins  ===============================================================================  NOTE: The 'make components' target must be issued once upon installation for  all other targets to build properly. A 'make clean' target should be  issued before this target |

DVSDK中发布的一些组件是没有预编译的。提供make clean与make component的编译目标是用来清除和编译组件的，这些是开始一个应用开发的必须编译。这些组件必须首先cleaned，然后重新编译，这样才能再重新编译。如下：

|  |
| --- |
| host $ make clean  host $ make components |
| 同时输出了以下信息：  cat: dvtb\_dm365/compiler.opt: 没有那个文件或目录  cat: dvtb\_dm355/compiler.opt: 没有那个文件或目录  cat: dvtb\_dm6446/compiler.opt: 没有那个文件或目录  cat: dvtb\_dm6467/compiler.opt: 没有那个文件或目录  cat: dvtb\_omap3530/compiler.opt: 没有那个文件或目录  cat: dvtb\_dm3730/compiler.opt: 没有那个文件或目录  cat: dvtb\_dm357/compiler.opt: 没有那个文件或目录 |
| make components |
|  |
| make all |
|  |
| display.c: In function 'main':  display.c:407: warning: assignment makes integer from pointer without a cast  此为编译告警  configure.ac:31: installing `./config.guess'  configure.ac:31: installing `./config.sub'  configure.ac:22: installing `./install-sh'  configure.ac:22: installing `./missing'  src/Makefile.am:19: shell cat $(XDC\_CONFIG\_BASENAME: non-POSIX variable name  src/Makefile.am:19: (probably a GNU make extension)  src/Makefile.am: installing `./depcomp'  Makefile.am: installing `./INSTALL'  configure: WARNING: If you wanted to set the --build type, don't use --host.  If a cross compiler is detected then cross compile mode will be used.  configure: WARNING: using cross tools not prefixed with host triplet  ../pal/linux/alsa/dvtbALSAAudio.c: In function 'dvtb\_audioSetSwParam':  ../pal/linux/alsa/dvtbALSAAudio.c:241: warning: 'snd\_pcm\_sw\_params\_get\_xfer\_align' is deprecated (declared at /home/dm368/ti-dvsdk\_dm368-evm\_4\_02\_00\_06/linux-devkit/arm-none-linux-gnueabi/usr/include/alsa/pcm.h:1115)  ../pal/linux/alsa/dvtbALSAAudio.c:248: warning: 'snd\_pcm\_sw\_params\_set\_sleep\_min' is deprecated (declared at /home/dm368/ti-dvsdk\_dm368-evm\_4\_02\_00\_06/linux-devkit/arm-none-linux-gnueabi/usr/include/alsa/pcm.h:1116)  ../pal/linux/alsa/dvtbALSAAudio.c:291: warning: 'snd\_pcm\_sw\_params\_set\_xfer\_align' is deprecated (declared at /home/dm368/ti-dvsdk\_dm368-evm\_4\_02\_00\_06/linux-devkit/arm-none-linux-gnueabi/usr/include/alsa/pcm.h:1114)  linux/dvtbH264Capture1.c: In function 'dvtb\_H264Enc1Capture':  linux/dvtbH264Capture1.c:413: warning: assignment from incompatible pointer type  /opt/arm-2009q1-203//bin/arm-none-linux-gnueabi-ar: creating ./bin/dvtb-r.a  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(readline.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(chared.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(common.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(el.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(hist.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(key.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(map.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(parse.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(prompt.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(read.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(refresh.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(search.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(sig.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(term.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(tty.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(vi.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(fgetln.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(strlcat.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(strlcpy.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(vis.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(tokenizer.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(history.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(filecomplete.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(fcns.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(help.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(emacs.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(unvis.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  ../pal/linux/alsa/dvtbALSAAudio.c: In function 'dvtb\_audioSetSwParam':  ../pal/linux/alsa/dvtbALSAAudio.c:241: warning: 'snd\_pcm\_sw\_params\_get\_xfer\_align' is deprecated (declared at /home/dm368/ti-dvsdk\_dm368-evm\_4\_02\_00\_06/linux-devkit/arm-none-linux-gnueabi/usr/include/alsa/pcm.h:1115)  ../pal/linux/alsa/dvtbALSAAudio.c:248: warning: 'snd\_pcm\_sw\_params\_set\_sleep\_min' is deprecated (declared at /home/dm368/ti-dvsdk\_dm368-evm\_4\_02\_00\_06/linux-devkit/arm-none-linux-gnueabi/usr/include/alsa/pcm.h:1116)  ../pal/linux/alsa/dvtbALSAAudio.c:291: warning: 'snd\_pcm\_sw\_params\_set\_xfer\_align' is deprecated (declared at /home/dm368/ti-dvsdk\_dm368-evm\_4\_02\_00\_06/linux-devkit/arm-none-linux-gnueabi/usr/include/alsa/pcm.h:1114)  linux/dvtbH264Capture1.c: In function 'dvtb\_H264Enc1Capture':  linux/dvtbH264Capture1.c:413: warning: assignment from incompatible pointer type  /opt/arm-2009q1-203//bin/arm-none-linux-gnueabi-ar: creating ./bin/dvtb-d.a  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(readline.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(chared.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(common.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(el.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(hist.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(key.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(map.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(parse.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(prompt.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(read.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(refresh.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(search.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(sig.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(term.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(tty.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(vi.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(fgetln.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(strlcat.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(strlcpy.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(vis.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(tokenizer.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(history.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(filecomplete.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(fcns.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(help.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(emacs.o): warning: duplicate section `.note.gnu.arm.ident' has different contents  /opt/arm-2009q1-203/bin/../lib/gcc/arm-none-linux-gnueabi/4.3.3/../../../../arm-none-linux-gnueabi/bin/ld: ./linux/libs/libedit.a(unvis.o): warning: duplicate section `.note.gnu.arm.ident' has different contents |

运行以上命令后，make help的每个编译目标都可以使用以下指令了：

|  |
| --- |
| host $ make <target>\_clean  host $ make <target>  host $ sudo make <target>\_install |

为了安装目标板的结果二进制，以超级用户权限执行一个install目标。这样，二进制文件就复制到${DVSDK}/Rules.make的$EXEC\_DIR变量。该变量指向了NFS挂载目标文件系统，这样可以通过执行DVSDK setup.sh脚本，但是也可以按照自己的需要手动修改。

当然，也可以用makeclean移除所有组件（包括demo和例子）。

当然，也可以make all来重新编译组件和demo/例子。

也可以使用sudo make install来安装结果目标文件。

注意：默认的，make install会覆盖已存文件，这个可以通过修改rules.make的EXEC\_DIR变量来控制。

注意：在目标板启动新编译内核需要运行的depmod -a，来为modprobe正常工作重新生成一个模块依赖。

## 创建自己的linux内核镜像

预先编译的linux内核镜像是按照默认配置编译的。如果想要按照自己的应用改变配置，或者是内核源代码。该节将展示如何重新compile DVSDK的linux内核，并且展示如何用其替代默认的linux内核镜像。

* 建立DVSDK。如果没有，那么就执行脚本。
* 重新compile内核：

|  |
| --- |
| host $ cd ${DVSDK}  host $ make linux\_clean  host $ make linux  host $ sudo make linux\_install |
|  |

* 你需要一个boot-loader(u-boot)的方法来获得新镜像。可以建立TFTP服务器。
* 复制新镜像到${DVSDK}/rules.make中EXEC\_DIR指定的tftpserver中，

|  |
| --- |
| host $ cp ${HOME}/targetfs/boot/uImage /tftpboot/new\_uImage |

* 运行u-boot脚本，并且跟着指令。选择TFTP作为你linux内核位置，选择新镜像文件为待烧写镜像。

|  |
| --- |
| host $ ${DVSDK}/bin/setup-uboot-env.sh |

* 注意，当时修改内核时，重新编译DVSDK子组件提供的所有内核模块是非常重要的。可以在以下路径找到这些模块：

|  |
| --- |
| /lib/modules/2.6.32-rc2-davinci1/kernel/drivers/dsp/ |

用平台提供的可用内核来替换2.6.32-rc2-davincil。

|  |
| --- |
| host $ ls ${HOME}/targetfs/lib/modules/2.6.32-rc2-davinci1/kernel/drivers/dsp/ |

对于每一个能看到的模块，可以返回到主机，重新编译它，并且从ECEC\_DIR下替换。例如：

|  |
| --- |
| host $ cd ${DVSDK}  host $ make cmem\_clean  host $ make cmem  host $ sudo make cmem\_install |

* 也可以选择用make all重新编译TI提供的内核模块，以及包含linux内核模块的示例。然后运行make install就可以了。
* 在更新所有模块后，启动minicom或Tera Term，启动单板。新内核可以通过TFTP来加载。
* 用depmod -a来重新生成内核模块依赖文件。

|  |
| --- |
| target $ depmod -a |

## 创建Tera Term

# 创建SD卡

该节描述为文件系统创建一个SD卡，已经通过了4GB/8B的SDHC卡测试。

* 将SD卡插入到linux 主机。
* 使用dmesg命令来检查设备节点。三次检查来保证你没有损坏你的HDD内容。

|  |
| --- |
| host $ dmesg  [14365.272631] sd 6:0:0:1: [sdc] 3862528 512-byte logical blocks: (1.97 GB/1.84 GiB)  [14365.310602] sd 6:0:0:1: [sdc] Assuming drive cache: write through  [14365.325542] sd 6:0:0:1: [sdc] Assuming drive cache: write through  [14365.325571] sdc: sdc1 sdc2 |
|  |

该例中，SD卡检测位置在/dev/sdc。

* 执行在DVSDK中安装的mksdboot脚本，使用检测到正确的SD卡设备。

|  |
| --- |
| host $ sudo ${DVSDK}/bin/mksdboot.sh --device /dev/sdc --sdk ${DVSDK} |

注意，当创建一个能启动的SD卡，mksdboot脚本使用在DVSDK安装中预编译的root文件系统，内核和bootloader镜像。

* 设置启动管脚。
* DVSDK用脚本来设置u-boot来从SD卡boot文件系统、linux内核。

|  |
| --- |
| host $ ${DVSDK}/bin/setup-uboot-env.sh |

如果想要重新为DVSDK安装生成一个独立的全部SD卡，那么：

|  |
| --- |
| host $ sudo ${DVSDK}/bin/mksdboot.sh --device /dev/sdc --sdk ${DVSDK} \  /path/to/dvsdk\_dm368-evm\_4\_xx\_xx\_xx\_setuplinux |

# makefile

以下是备份的原始makefile。



编译内核：

|  |
| --- |
| make\_mylinux\_config |

## rules.make

|  |  |
| --- | --- |
| # Define target platform.  PLATFORM=dm368 | 定义目标平台 |
| # The installation directory of the DVSDK.  DVSDK\_INSTALL\_DIR=/home/dm368/ti-dvsdk\_dm368-evm\_4\_02\_00\_06 | DVSDK的安装目录 |
| # For backwards compatibility  DVEVM\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR) | 后向兼容 |
| # Where the Codec Engine package is installed.  CE\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR)/codec-engine\_2\_26\_02\_11 | Codec引擎安装包 |
| # Where the codecs are installed.  CODEC\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR)/codecs-dm365\_4\_02\_00\_00 | 编解码安装路径 |
| # Where DMAI package is installed.  DMAI\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR)/dmai\_2\_20\_00\_15 | DMAI安装包 |
| # Where the SDK demos are installed  DEMO\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR)/dvsdk-demos\_4\_02\_00\_01 | SDK模板 |
| # Where the DVTB package is installed.  DVTB\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR)/dvtb\_4\_20\_18 | DVTB安装包 |
| # Where the Framework Components package is installed.  FC\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR)/framework-components\_2\_26\_00\_01 | Framework组件安装包 |
| # Where the DM365mm module is installed.  DM365MM\_MODULE\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR)/dm365mm-module\_01\_00\_03 | Dm365模块 |
| # Where the PSP is installed.  PSP\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR)/psp | PSP |
| # Where the MFC Linux Utils package is installed.  LINUXUTILS\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR)/linuxutils\_2\_26\_01\_02  CMEM\_INSTALL\_DIR=$(LINUXUTILS\_INSTALL\_DIR) | Linux Utils安装包 |
| # Where the XDAIS package is installed.  XDAIS\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR)/xdais\_6\_26\_01\_03 | XDAIS安装包 |
| # Where the RTSC tools package is installed.  XDC\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR)/xdctools\_3\_16\_03\_36 | RTSC工具包 |
| # The directory that points to your kernel source directory.  LINUXKERNEL\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR)/psp/linux-2.6.32.17-psp03.01.01.39 | 内核 |
| # Where temporary Linux headers and libs are installed.  LINUXLIBS\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR)/linux-devkit/arm-none-linux-gnueabi/usr | 临时的linux头文件和lib |
| # The prefix to be added before the GNU compiler tools (optionally including # path), i.e. "arm\_v5t\_le-" or "/opt/bin/arm\_v5t\_le-".  CSTOOL\_DIR=/opt/arm-2009q1-203/  CSTOOL\_PREFIX=$(CSTOOL\_DIR)/bin/arm-none-linux-gnueabi- | 编译工具链 |
| MVTOOL\_DIR=$(CSTOOL\_DIR)  MVTOOL\_PREFIX=$(CSTOOL\_PREFIX) |  |
| # Where to copy the resulting executables  EXEC\_DIR=/home/dm368/targetfs | 目标可执行复制目录 |

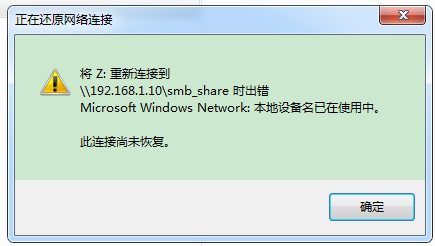
## makefile

|  |
| --- |
| include Rules.make  DEFAULT\_LINUXKERNEL\_CONFIG=davinci\_dm365\_defconfig  DEFAULT\_UBOOT\_CONFIG=davinci\_dm365evm\_config  DVTB\_PLATFORM=dm365  KERNEL\_VERSION= $(shell cat $(LINUXKERNEL\_INSTALL\_DIR)/include/config/kernel.release)  .PHONY: all clean components help cmem cmem\_clean cmem\_install dmai dmai\_clean dmai\_install dm365mm dm365mm\_clean dm365mm\_install linux linux\_clean linux\_install psp\_examples psp\_example\_clean psp\_example\_install irq irq\_clean irq\_install edma edma\_clean edma\_install install demos demos\_clean demos\_install u-boot u-boot\_clean u-boot\_install dvtb\_install matrix matrix\_clean matrix\_install gstreamer\_ti gstreamer\_ti\_install loadmodule\_install  #==============================================================================  # Default build target  #==============================================================================  default: all  #==============================================================================  # Build target build necessary components to enable all other build targets.  #==============================================================================  components: linux cmem dm365mm irq edma ceexamples dmai  #==============================================================================  # Clean up the targets built by 'make all'.  #==============================================================================  clean: linux\_clean cmem\_clean dm365mm\_clean irq\_clean edma\_clean dmai\_clean psp\_examples\_clean ceexamples\_clean demos\_clean u-boot\_clean matrix\_clean dvtb\_clean  #==============================================================================  # Build everything rebuildable.  #==============================================================================  all: components psp\_examples ceexamples demos u-boot matrix gstreamer\_ti dvtb  #==============================================================================  # Install everything  #==============================================================================  install: linux\_install cmem\_install dm365mm\_install dmai\_install psp\_examples\_install ceexamples\_install irq\_install edma\_install demos\_install u-boot\_install matrix\_install gstreamer\_ti\_install loadmodule\_install dvtb\_install  #==============================================================================  # A help message target.  #==============================================================================  help:  @echo  @echo "Available build targets are:"  @echo  @echo " components : Rebuild all base components (See Note below)"  @echo  @echo " all : Builds all components and applications/examples"  @echo " clean : Remove files generated by the 'all' target"  @echo  @echo " install : Install all the targets in "  @echo " $(EXEC\_DIR)"  @echo  @echo " demos : Build the DVSDK demos for $(PLATFORM)"  @echo " demos\_clean : Remove generated DVSDK demo files."  @echo " demos\_install : Install DVSDK demos"  @echo  @echo " dvtb : Build DVTB for $(PLATFORM)"  @echo " dvtb\_clean : Remove generated DVTB files"  @echo " dvtb\_install : Install dvtb"  @echo  @echo " cmem : Build the CMEM kernel module"  @echo " cmem\_clean : Remove generated cmem files."  @echo " cmem\_install : Install cmemk module"  @echo  @echo " irq : Build the irq kernel module"  @echo " irq\_clean : Remove generated irq files."  @echo " irq\_install : Install irqk module"  @echo  @echo " edma : Build the edma kernel module"  @echo " edma\_clean : Remove generated edma files."  @echo " edma\_install : Install edmak module"  @echo  @echo " dm365mm : Build the dm365mm kernel module"  @echo " dm365mm\_clean : Remove generated dm365mm files"  @echo " dm365mm\_install : Install dm365mm module"  @echo  @echo " dmai : Build DMAI"  @echo " dmai\_clean : Remove generated DMAI files."  @echo " dmai\_install : Install dmai app"  @echo  @echo " ceexamples : Build CE codecs, extensions, servers and apps"  @echo " ceexamples\_clean : Remove generated CE example files."  @echo  @echo " linux : Build Linux kernel uImage and module"  @echo " linux\_config : Launch Linux kernel menuconfig"  @echo " linux\_clean : Remove generated Linux kernel files"  @echo " linux\_install : Install kernel binary and modules"  @echo  @echo " u-boot : Build the u-boot boot loader"  @echo " u-boot\_clean : Remove generated u-boot files"  @echo " u-boot\_install : Install the u-boot image"  @echo  @echo " psp\_examples : Build Linux examples"  @echo " psp\_examples\_clean : Remove all generated PSP examples"  @echo " psp\_examples\_install : Install psp examples"  @echo  @echo " matrix : Build matrix application launcher"  @echo " matrix\_clean : Remove all matrix files"  @echo " matrix\_install : Install matrix"  @echo  @echo " gstreamer\_ti : Build gstreamer-ti plugins"  @echo " gstreamer\_ti\_install : Install gstreamer-ti plugins"  @echo  @echo "==============================================================================="  @echo "NOTE: The 'make components' target must be issued once upon installation for"  @echo " all other targets to build properly. A 'make clean' target should be"  @echo " issued before this target"  @echo  #==============================================================================  # Build the dvsdk demos for the configured platform. Also, an explicit cleanup  # target is defined.  #==============================================================================  demos:  . $(DVSDK\_INSTALL\_DIR)/linux-devkit/environment-setup ; cd $(DEMO\_INSTALL\_DIR)/qtInterface ; qmake ; $(MAKE) PLATFORM=dm365  $(MAKE) -C $(DEMO\_INSTALL\_DIR) dm365 DVSDK\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR) XDC\_INSTALL\_DIR=$(XDC\_INSTALL\_DIR) CE\_INSTALL\_DIR=$(CE\_INSTALL\_DIR) FC\_INSTALL\_DIR=$(FC\_INSTALL\_DIR) CMEM\_INSTALL\_DIR=$(CMEM\_INSTALL\_DIR) CODEC\_INSTALL\_DIR=$(CODEC\_INSTALL\_DIR) XDAIS\_INSTALL\_DIR=$(XDAIS\_INSTALL\_DIR) LINK\_INSTALL\_DIR=$(LINK\_INSTALL\_DIR) DMAI\_INSTALL\_DIR=$(DMAI\_INSTALL\_DIR) MVTOOL\_DIR=$(MVTOOL\_DIR) CC=$(CSTOOL\_PREFIX)gcc AR=$(CSTOOL\_PREFIX)ar CROSS\_COMPILE=$(MVTOOL\_PREFIX) LINUXLIBS\_INSTALL\_DIR=$(LINUXLIBS\_INSTALL\_DIR) PLATFORM=dm365  demos\_clean:  . $(DVSDK\_INSTALL\_DIR)/linux-devkit/environment-setup ; cd $(DEMO\_INSTALL\_DIR)/qtInterface ; qmake ; $(MAKE) PLATFORM=dm365 clean  $(MAKE) -C $(DEMO\_INSTALL\_DIR) clean DVSDK\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR) XDC\_INSTALL\_DIR=$(XDC\_INSTALL\_DIR) CE\_INSTALL\_DIR=$(CE\_INSTALL\_DIR) FC\_INSTALL\_DIR=$(FC\_INSTALL\_DIR) CMEM\_INSTALL\_DIR=$(CMEM\_INSTALL\_DIR) CODEC\_INSTALL\_DIR=$(CODEC\_INSTALL\_DIR) XDAIS\_INSTALL\_DIR=$(XDAIS\_INSTALL\_DIR) LINK\_INSTALL\_DIR=$(LINK\_INSTALL\_DIR) DMAI\_INSTALL\_DIR=$(DMAI\_INSTALL\_DIR) MVTOOL\_DIR=$(MVTOOL\_DIR) CC=$(CSTOOL\_PREFIX)gcc CROSS\_COMPILE=$(MVTOOL\_PREFIX) LINUXLIBS\_INSTALL\_DIR=$(LINUXLIBS\_INSTALL\_DIR) PLATFORM=dm365  demos\_install:  install -d $(EXEC\_DIR)//usr/share/ti/dvsdk-demos  . $(DVSDK\_INSTALL\_DIR)/linux-devkit/environment-setup ; cd $(DEMO\_INSTALL\_DIR)/qtInterface ; qmake ; $(MAKE) PLATFORM=dm365 EXEC\_DIR=$(EXEC\_DIR)/usr/share/ti/dvsdk-demos install  $(MAKE) -C $(DEMO\_INSTALL\_DIR) install DVSDK\_INSTALL\_DIR=$(DVSDK\_INSTALL\_DIR) XDC\_INSTALL\_DIR=$(XDC\_INSTALL\_DIR) CE\_INSTALL\_DIR=$(CE\_INSTALL\_DIR) FC\_INSTALL\_DIR=$(FC\_INSTALL\_DIR) CMEM\_INSTALL\_DIR=$(CMEM\_INSTALL\_DIR) CODEC\_INSTALL\_DIR=$(CODEC\_INSTALL\_DIR) XDAIS\_INSTALL\_DIR=$(XDAIS\_INSTALL\_DIR) LINK\_INSTALL\_DIR=$(LINK\_INSTALL\_DIR) DMAI\_INSTALL\_DIR=$(DMAI\_INSTALL\_DIR) MVTOOL\_DIR=$(MVTOOL\_DIR) CC=$(CSTOOL\_PREFIX)gcc CROSS\_COMPILE=$(MVTOOL\_PREFIX) LINUXLIBS\_INSTALL\_DIR=$(LINUXLIBS\_INSTALL\_DIR) EXEC\_DIR=$(EXEC\_DIR)/usr/share/ti/dvsdk-demos PLATFORM=dm365  #==============================================================================  # Build the Digital Video Test Bench for the configured platform. Also, an  # explicit cleanup target is defined.  #==============================================================================  dvtb:  $(MAKE) -C $(DVTB\_INSTALL\_DIR) PLATFORM=$(DVTB\_PLATFORM) CODECS=TSPA  @echo  @echo "dvtb can be found under $(DVTB\_INSTALL\_DIR)/packages/ti/sdo/dvtb/$(DVTB\_PLATFORM)/bin"  dvtb\_clean:  $(MAKE) -C $(DVTB\_INSTALL\_DIR) clean  dvtb\_install:  install -d $(EXEC\_DIR)/usr/share/ti/dvtb/scripts  install $(DVTB\_INSTALL\_DIR)/packages/ti/sdo/dvtb/$(DVTB\_PLATFORM)/bin/\* $(EXEC\_DIR)/usr/share/ti/dvtb  install $(DVTB\_INSTALL\_DIR)/packages/ti/sdo/dvtb/scripts/$(DVTB\_PLATFORM)/\* $(EXEC\_DIR)/usr/share/ti/dvtb/scripts  #==============================================================================  # Build the Linux kernel. Also, an explicit cleanup target is defined.  #==============================================================================  linux:  @if ! test -e $(LINUXKERNEL\_INSTALL\_DIR)/.config; then $(MAKE) -C $(LINUXKERNEL\_INSTALL\_DIR) ARCH=arm CROSS\_COMPILE=$(CSTOOL\_PREFIX) $(DEFAULT\_LINUXKERNEL\_CONFIG) ; fi  $(MAKE) -C $(LINUXKERNEL\_INSTALL\_DIR) ARCH=arm CROSS\_COMPILE=$(CSTOOL\_PREFIX) uImage  $(MAKE) -C $(LINUXKERNEL\_INSTALL\_DIR) ARCH=arm CROSS\_COMPILE=$(CSTOOL\_PREFIX) modules  linux\_myconfig:  @if ! test -e $(LINUXKERNEL\_INSTALL\_DIR)/.config; then $(MAKE) -C $(LINUXKERNEL\_INSTALL\_DIR) ARCH=arm CROSS\_COMPILE=$(CSTOOL\_PREFIX) $(DEFAULT\_LINUXKERNEL\_CONFIG) ; fi  $(MAKE) -C $(LINUXKERNEL\_INSTALL\_DIR) ARCH=arm CROSS\_COMPILE=$(CSTOOL\_PREFIX) menuconfig  @echo  @echo "NOTE: \*\*\*\* Kernel configuration is saved and will be used during 'make linux' execution."  @echo  linux\_clean:  $(MAKE) -C $(LINUXKERNEL\_INSTALL\_DIR) mrproper  $(MAKE) -C $(LINUXKERNEL\_INSTALL\_DIR) ARCH=arm CROSS\_COMPILE=$(CSTOOL\_PREFIX) clean  linux\_install:  install -d $(EXEC\_DIR)/boot  install $(LINUXKERNEL\_INSTALL\_DIR)/arch/arm/boot/uImage $(EXEC\_DIR)/boot  install $(LINUXKERNEL\_INSTALL\_DIR)/vmlinux $(EXEC\_DIR)/boot  install $(LINUXKERNEL\_INSTALL\_DIR)/System.map $(EXEC\_DIR)/boot  $(MAKE) -C $(LINUXKERNEL\_INSTALL\_DIR) ARCH=arm CROSS\_COMPILE=$(CSTOOL\_PREFIX) INSTALL\_MOD\_PATH=$(EXEC\_DIR)/ modules\_install  #==============================================================================  # Build u-boot. Also, an explicit cleanup target is defined.  #==============================================================================  u-boot:  $(MAKE) -C $(PSP\_INSTALL\_DIR)/u-boot-\* CROSS\_COMPILE=$(CSTOOL\_PREFIX) $(DEFAULT\_UBOOT\_CONFIG)  $(MAKE) -C $(PSP\_INSTALL\_DIR)/u-boot-\* CROSS\_COMPILE=$(CSTOOL\_PREFIX)  u-boot\_clean:  $(MAKE) -C $(PSP\_INSTALL\_DIR)/u-boot-\* CROSS\_COMPILE=$(CSTOOL\_PREFIX) distclean  u-boot\_install:  install -d $(EXEC\_DIR)/boot  install $(PSP\_INSTALL\_DIR)/u-boot-\*/u-boot.bin $(EXEC\_DIR)/boot  install $(PSP\_INSTALL\_DIR)/u-boot-\*/u-boot.map $(EXEC\_DIR)/boot  #==============================================================================  # Build the PSP Linux examples. Also, an explicit cleanup target is defined.  #==============================================================================  psp\_examples:  $(MAKE) -C $(PSP\_INSTALL\_DIR)/linux-driver-examples-\* KERNEL\_DIR=$(LINUXKERNEL\_INSTALL\_DIR) CROSS\_COMPILE=$(CSTOOL\_PREFIX) PLATFORM=dm365 LINUXLIBS\_DIR=$(LINUXLIBS\_INSTALL\_DIR)  psp\_examples\_clean:  $(MAKE) -C $(PSP\_INSTALL\_DIR)/linux-driver-examples-\* KERNEL\_DIR=$(LINUXKERNEL\_INSTALL\_DIR) CROSS\_COMPILE=$(CSTOOL\_PREFIX) PLATFORM=dm365 LINUXLIBS\_DIR=$(LINUXLIBS\_INSTALL\_DIR) clean  psp\_examples\_install:  $(MAKE) -C $(PSP\_INSTALL\_DIR)/linux-driver-examples-\* KERNEL\_DIR=$(LINUXKERNEL\_INSTALL\_DIR) CROSS\_COMPILE=$(CSTOOL\_PREFIX) PLATFORM=dm365 LINUXLIBS\_DIR=$(LINUXLIBS\_INSTALL\_DIR) FINAL\_DEST=$(EXEC\_DIR)/usr/share/ti/linux-driver-examples install    #==============================================================================  # Build the CMEM kernel module for the configured platform, and make sure the  # kernel\_binaries directory is kept in sync. Also, an explicit cleanup target  # is defined.  #==============================================================================  cmem:  $(MAKE) -C $(CMEM\_INSTALL\_DIR)/packages/ti/sdo/linuxutils/cmem/src/interface ../../lib/cmem.a470MV  $(MAKE) -C $(CMEM\_INSTALL\_DIR)/packages/ti/sdo/linuxutils/cmem/src/module  cmem\_clean:  $(MAKE) -C $(CMEM\_INSTALL\_DIR)/packages/ti/sdo/linuxutils/cmem/src/module clean  $(MAKE) -C $(CMEM\_INSTALL\_DIR)/packages/ti/sdo/linuxutils/cmem/src/interface clean  cmem\_install:  install -d $(EXEC\_DIR)/lib/modules/$(KERNEL\_VERSION)/kernel/drivers/dsp  install $(CMEM\_INSTALL\_DIR)/packages/ti/sdo/linuxutils/cmem/src/module/cmemk.ko $(EXEC\_DIR)/lib/modules/$(KERNEL\_VERSION)/kernel/drivers/dsp  #==============================================================================  # Build the irq kernel module for the configured platform, and make sure the  # kernel\_binaries directory is kept in sync. Also, an explicit cleanup target  # is defined.  #==============================================================================  irq:  $(MAKE) -C $(CMEM\_INSTALL\_DIR)/packages/ti/sdo/linuxutils/irq/src/module  irq\_clean:  $(MAKE) -C $(CMEM\_INSTALL\_DIR)/packages/ti/sdo/linuxutils/cmem/src/module clean  irq\_install:  install -d $(EXEC\_DIR)/lib/modules/$(KERNEL\_VERSION)/kernel/drivers/dsp  install $(CMEM\_INSTALL\_DIR)/packages/ti/sdo/linuxutils/irq/src/module/irqk.ko $(EXEC\_DIR)/lib/modules/$(KERNEL\_VERSION)/kernel/drivers/dsp  #==============================================================================  # Build the edma kernel module for the configured platform, and make sure the  # kernel\_binaries directory is kept in sync. Also, an explicit cleanup target  # is defined.  #==============================================================================  edma:  $(MAKE) -C $(CMEM\_INSTALL\_DIR)/packages/ti/sdo/linuxutils/edma/src/interface ../../lib/edma.a470MV  $(MAKE) -C $(CMEM\_INSTALL\_DIR)/packages/ti/sdo/linuxutils/edma/src/module  edma\_clean:  $(MAKE) -C $(CMEM\_INSTALL\_DIR)/packages/ti/sdo/linuxutils/edma/src/module clean  $(MAKE) -C $(CMEM\_INSTALL\_DIR)/packages/ti/sdo/linuxutils/edma/src/interface clean  edma\_install:  install -d $(EXEC\_DIR)/lib/modules/$(KERNEL\_VERSION)/kernel/drivers/dsp  install $(CMEM\_INSTALL\_DIR)/packages/ti/sdo/linuxutils/edma/src/module/edmak.ko $(EXEC\_DIR)/lib/modules/$(KERNEL\_VERSION)/kernel/drivers/dsp  #==============================================================================  # Build the dm355mm kernel module (if the configured platform is dm355). Also,  # an explicit cleanup target is defined.  #==============================================================================  dm365mm:  $(MAKE) -C $(DM365MM\_MODULE\_INSTALL\_DIR)/module LINUXKERNEL\_INSTALL\_DIR=$(LINUXKERNEL\_INSTALL\_DIR) MVTOOL\_PREFIX=$(CSTOOL\_PREFIX)  $(MAKE) -C $(DM365MM\_MODULE\_INSTALL\_DIR)/interface LINUXKERNEL\_INSTALL\_DIR=$(LINUXKERNEL\_INSTALL\_DIR) MVTOOL\_PREFIX=$(CSTOOL\_PREFIX)  dm365mm\_clean:  $(MAKE) -C $(DM365MM\_MODULE\_INSTALL\_DIR)/module LINUXKERNEL\_INSTALL\_DIR=$(LINUXKERNEL\_INSTALL\_DIR) MVTOOL\_PREFIX=$(CSTOOL\_PREFIX) clean  $(MAKE) -C $(DM365MM\_MODULE\_INSTALL\_DIR)/interface LINUXKERNEL\_INSTALL\_DIR=$(LINUXKERNEL\_INSTALL\_DIR) MVTOOL\_PREFIX=$(CSTOOL\_PREFIX) clean  dm365mm\_install:  install -d $(EXEC\_DIR)/lib/modules/$(KERNEL\_VERSION)/kernel/drivers/dsp  install $(DM365MM\_MODULE\_INSTALL\_DIR)/module/dm365mmap.ko $(EXEC\_DIR)/lib/modules/$(KERNEL\_VERSION)/kernel/drivers/dsp  #==============================================================================  # Build the Davinci Multimedia Application Interface for the configured  # platform. Also, an explicit cleanup target is defined.  #==============================================================================  dmai:  $(MAKE) -C $(DMAI\_INSTALL\_DIR) PLATFORM=$(PLATFORM)\_al \  CE\_INSTALL\_DIR\_$(PLATFORM)\_al=$(CE\_INSTALL\_DIR) \  CODEC\_INSTALL\_DIR\_$(PLATFORM)\_al=$(CODEC\_INSTALL\_DIR) \  CMEM\_INSTALL\_DIR\_$(PLATFORM)\_al=$(CMEM\_INSTALL\_DIR) \  FC\_INSTALL\_DIR\_$(PLATFORM)\_al=$(FC\_INSTALL\_DIR) \  XDAIS\_INSTALL\_DIR\_$(PLATFORM)\_al=$(XDAIS\_INSTALL\_DIR) \  LINUXLIBS\_INSTALL\_DIR\_$(PLATFORM)\_al=$(LINUXLIBS\_INSTALL\_DIR) \  LINUXKERNEL\_INSTALL\_DIR\_$(PLATFORM)\_al=$(LINUXKERNEL\_INSTALL\_DIR) \  CROSS\_COMPILE\_$(PLATFORM)\_al=$(CSTOOL\_PREFIX) \  XDC\_INSTALL\_DIR\_$(PLATFORM)\_al=$(XDC\_INSTALL\_DIR) \  VERBOSE=false \  all    dmai\_clean:  $(MAKE) -C $(DMAI\_INSTALL\_DIR) PLATFORM=$(PLATFORM)\_al \  XDC\_INSTALL\_DIR\_$(PLATFORM)\_al=$(XDC\_INSTALL\_DIR) \  clean  dmai\_install:  $(MAKE) -C $(DMAI\_INSTALL\_DIR) PLATFORM=$(PLATFORM)\_al \  EXEC\_DIR\_$(PLATFORM)\_al=$(EXEC\_DIR)/usr/share/ti/ti-dmai-apps \  install  ###############################################################################  # Build codec engine examples  ###############################################################################  ceexamples\_apps:  make -C $(CE\_INSTALL\_DIR)/examples/ti/sdo/ce/examples/apps \  DEVICES="DM365" \  GPPOS=LINUX\_GCC \  PROGRAMS=APP\_LOCAL \  CE\_INSTALL\_DIR=$(CE\_INSTALL\_DIR) \  XDC\_INSTALL\_DIR=$(XDC\_INSTALL\_DIR) \  XDAIS\_INSTALL\_DIR=$(XDAIS\_INSTALL\_DIR) \  FC\_INSTALL\_DIR=$(FC\_INSTALL\_DIR) \  CMEM\_INSTALL\_DIR=$(CMEM\_INSTALL\_DIR) \  CGTOOLS\_V5T=$(CSTOOL\_DIR) \  all  ceexamples\_apps\_clean:  make -C $(CE\_INSTALL\_DIR)/examples/ti/sdo/ce/examples/apps \  DEVICES="DM365" \  GPPOS=LINUX\_GCC \  PROGRAMS=APP\_LOCAL \  CE\_INSTALL\_DIR=$(CE\_INSTALL\_DIR) \  XDC\_INSTALL\_DIR=$(XDC\_INSTALL\_DIR) \  XDAIS\_INSTALL\_DIR=$(XDAIS\_INSTALL\_DIR) \  FC\_INSTALL\_DIR=$(FC\_INSTALL\_DIR) \  CMEM\_INSTALL\_DIR=$(CMEM\_INSTALL\_DIR) \  CGTOOLS\_V5T=$(CSTOOL\_DIR) \  clean  ceexamples\_apps\_install:  @echo  @echo "\*\*\*\* Don't have working install target \*\*\*\*"  @echo "Copy the required application binaries and data files from $(CE\_INSTALL\_DIR)/examples/ti/sdo/ce/examples/apps to target filesystem. "  @echo  ceexamples\_extensions:  make -C $(CE\_INSTALL\_DIR)/examples/ti/sdo/ce/examples/extensions \  DEVICES="DM365" \  GPPOS=LINUX\_GCC \  PROGRAMS=APP\_LOCAL \  CE\_INSTALL\_DIR=$(CE\_INSTALL\_DIR) \  XDC\_INSTALL\_DIR=$(XDC\_INSTALL\_DIR) \  XDAIS\_INSTALL\_DIR=$(XDAIS\_INSTALL\_DIR) \  FC\_INSTALL\_DIR=$(FC\_INSTALL\_DIR) \  CMEM\_INSTALL\_DIR=$(CMEM\_INSTALL\_DIR) \  CGTOOLS\_V5T=$(CSTOOL\_DIR) \  all  ceexamples\_extensions\_clean:  make -C $(CE\_INSTALL\_DIR)/examples/ti/sdo/ce/examples/apps \  DEVICES="DM365" \  GPPOS=LINUX\_GCC \  PROGRAMS=APP\_LOCAL \  CE\_INSTALL\_DIR=$(CE\_INSTALL\_DIR) \  XDC\_INSTALL\_DIR=$(XDC\_INSTALL\_DIR) \  XDAIS\_INSTALL\_DIR=$(XDAIS\_INSTALL\_DIR) \  FC\_INSTALL\_DIR=$(FC\_INSTALL\_DIR) \  CMEM\_INSTALL\_DIR=$(CMEM\_INSTALL\_DIR) \  CGTOOLS\_V5T=$(CSTOOL\_DIR) \  clean  ceexamples\_extensions\_install:  @echo  @echo "\*\*\*\* Don't have working install target \*\*\*\*"  @echo "Copy the required application binaries and data files from $(CE\_INSTALL\_DIR)/examples/ti/sdo/ce/examples/extensions to target filesystem. "  @echo  ceexamples\_servers:  make -C $(CE\_INSTALL\_DIR)/examples/ti/sdo/ce/examples/servers \  DEVICES="DM365" \  GPPOS=LINUX\_GCC \  PROGRAMS=APP\_LOCAL \  CE\_INSTALL\_DIR=$(CE\_INSTALL\_DIR) \  XDC\_INSTALL\_DIR=$(XDC\_INSTALL\_DIR) \  XDAIS\_INSTALL\_DIR=$(XDAIS\_INSTALL\_DIR) \  FC\_INSTALL\_DIR=$(FC\_INSTALL\_DIR) \  CMEM\_INSTALL\_DIR=$(CMEM\_INSTALL\_DIR) \  CGTOOLS\_V5T=$(CSTOOL\_DIR) \  all  ceexamples\_servers\_clean:  make -C $(CE\_INSTALL\_DIR)/examples/ti/sdo/ce/examples/servers \  DEVICES="DM365" \  GPPOS=LINUX\_GCC \  PROGRAMS=APP\_LOCAL \  CE\_INSTALL\_DIR=$(CE\_INSTALL\_DIR) \  XDC\_INSTALL\_DIR=$(XDC\_INSTALL\_DIR) \  XDAIS\_INSTALL\_DIR=$(XDAIS\_INSTALL\_DIR) \  FC\_INSTALL\_DIR=$(FC\_INSTALL\_DIR) \  CMEM\_INSTALL\_DIR=$(CMEM\_INSTALL\_DIR) \  CGTOOLS\_V5T=$(CSTOOL\_DIR) \  clean  ceexamples\_servers\_install:  @echo  @echo "\*\*\*\* Don't have working install target \*\*\*\*"  @echo "Copy the required application binaries and data files from $(CE\_INSTALL\_DIR)/examples/ti/sdo/ce/examples/servers to target filesystem. "  @echo  ceexamples\_codecs:  make -C $(CE\_INSTALL\_DIR)/examples/ti/sdo/ce/examples/codecs \  DEVICES="DM365" \  GPPOS=LINUX\_GCC \  PROGRAMS=APP\_LOCAL \  CE\_INSTALL\_DIR=$(CE\_INSTALL\_DIR) \  XDC\_INSTALL\_DIR=$(XDC\_INSTALL\_DIR) \  XDAIS\_INSTALL\_DIR=$(XDAIS\_INSTALL\_DIR) \  FC\_INSTALL\_DIR=$(FC\_INSTALL\_DIR) \  CMEM\_INSTALL\_DIR=$(CMEM\_INSTALL\_DIR) \  CGTOOLS\_V5T=$(CSTOOL\_DIR) \  all  ceexamples\_codecs\_clean:  make -C $(CE\_INSTALL\_DIR)/examples/ti/sdo/ce/examples/codecs \  DEVICES="DM365" \  GPPOS=LINUX\_GCC \  PROGRAMS=APP\_LOCAL \  CE\_INSTALL\_DIR=$(CE\_INSTALL\_DIR) \  XDC\_INSTALL\_DIR=$(XDC\_INSTALL\_DIR) \  XDAIS\_INSTALL\_DIR=$(XDAIS\_INSTALL\_DIR) \  FC\_INSTALL\_DIR=$(FC\_INSTALL\_DIR) \  CMEM\_INSTALL\_DIR=$(CMEM\_INSTALL\_DIR) \  CGTOOLS\_V5T=$(CSTOOL\_DIR) \  clean  ceexamples\_codecs\_install:  @echo  @echo "\*\*\*\* Don't have working install target \*\*\*\*"  @echo "Copy the required application binaries and data files from $(CE\_INSTALL\_DIR)/examples/ti/sdo/ce/examples/codecs to target filesystem. "  @echo  ceexamples: ceexamples\_codecs ceexamples\_extensions ceexamples\_servers ceexamples\_apps    ceexamples\_clean: ceexamples\_codecs\_clean ceexamples\_extensions\_clean ceexamples\_servers\_clean ceexamples\_apps\_clean  ceexamples\_install: ceexamples\_codecs\_install ceexamples\_extensions\_install ceexamples\_servers\_install ceexamples\_apps\_install  ################################################################################  # Build Martix GUI Appication Launcher  ################################################################################  matrix:  . $(DVSDK\_INSTALL\_DIR)/linux-devkit/environment-setup ; cd $(DVSDK\_INSTALL\_DIR)/example-applications/matrix-gui-\* ; qmake QMAKE\_CXXFLAGS\_RELEASE+=-DPlatform\_$(PLATFORM); $(MAKE) PLATFORM=$(PLATFORM)  chmod 755 $(DVSDK\_INSTALL\_DIR)/example-applications/matrix-gui-\*/$(PLATFORM)/bin/\*  matrix\_clean:  . $(DVSDK\_INSTALL\_DIR)/linux-devkit/environment-setup; cd $(DVSDK\_INSTALL\_DIR)/example-applications/matrix-gui-\* ; qmake ; $(MAKE) distclean  matrix\_install:  install -d $(EXEC\_DIR)/etc/init.d  install -m 0755 $(DVSDK\_INSTALL\_DIR)/example-applications/matrix-gui-\*/$(PLATFORM)/etc/init $(EXEC\_DIR)/etc/init.d/matrix-gui-e  install -d $(EXEC\_DIR)/usr/bin  cp -ar $(DVSDK\_INSTALL\_DIR)/example-applications/matrix-gui-\*/$(PLATFORM)/bin/\* \  $(EXEC\_DIR)/usr/bin/  install -m 0755 $(DVSDK\_INSTALL\_DIR)/example-applications/matrix-gui-\*/matrix\_gui $(EXEC\_DIR)/usr/bin/matrix\_guiE  install -d $(EXEC\_DIR)/usr/share/matrix  cp -ar $(DVSDK\_INSTALL\_DIR)/example-applications/matrix-gui-\*/$(PLATFORM)/html \  $(EXEC\_DIR)/usr/share/matrix/  install -d $(EXEC\_DIR)/usr/share/matrix  cp -ar $(DVSDK\_INSTALL\_DIR)/example-applications/matrix-gui-\*/images/ \  $(EXEC\_DIR)/usr/share/matrix/  ################################################################################  # Build Martix GUI Appication Launcher  ################################################################################  gstreamer\_ti:  . $(DVSDK\_INSTALL\_DIR)/linux-devkit/environment-setup ; cd $(DVSDK\_INSTALL\_DIR)/gstreamer-ti\_svn\* ;$(MAKE) -f Makefile.external DVSDK\_PATH=$(DVSDK\_INSTALL\_DIR) $(PLATFORM)  gstreamer\_ti\_install:  install -d $(EXEC\_DIR)/usr/lib/gstreamer-0.10  cp $(DVSDK\_INSTALL\_DIR)/gstreamer-ti\_svn\*/src/.libs/\*.so $(EXEC\_DIR)/usr/lib/gstreamer-0.10  loadmodule\_install:  install -d $(EXEC\_DIR)/etc/init.d/  install -d $(EXEC\_DIR)/etc/rc3.d/  install -m 0755 $(DVSDK\_INSTALL\_DIR)/etc/init.d/loadmodule-rc $(EXEC\_DIR)/etc/init.d/  cd $(EXEC\_DIR)/etc && rm -rf rc3.d/S99loadmodule-rc && ln -s ../init.d/loadmodule-rc rc3.d/S99loadmodule-rc |

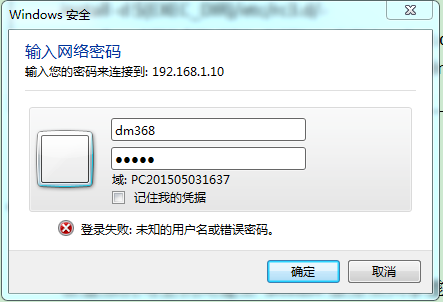
# SAMBA服务

|  |
| --- |
| 搞定静态IP和DNS，可以设置SAMBA了，使用以下命令：  #apt-get install samba  #apt-get install smbfs  #apt-get install samba-cmmon-bin  #apt-get install system-config-samba  系统自动从 Ubuntu的官方镜像网站下载稳定的最新的软件包版本，大家可以通过#gedit /etc/apt/sources.list命令看看这些链接的地址。  安装samba结束后，先使用  #touch /etc/samba/smbpasswd  #smbpasswd -a davinci （davinci普通用户）  输入samba访问密码两次。  然后编辑gedit /etc/samba/smb.conf的配置文件，在smb.conf最后增加下面一段内容：  [smb\_share]  ;comment = Shared Folder with username and password  path = /home/davinci  public = yes  writable = yes  valid users = davinci  create mask = 0700  directory mask = 0700  force user = nobody  force group = nogroup  available = yes  browseable = yes  保存退出。  重启samba，/etc/init.d/smbd restart  这个和以前的（/etc/init.d/samba restart）不一样了。  使samba生效。  最后使用：  # ufw disable  把防火墙给关闭掉，这样在WINDOWS下就可以使用\\192.168.1.252 登录samba共享目录了。 |

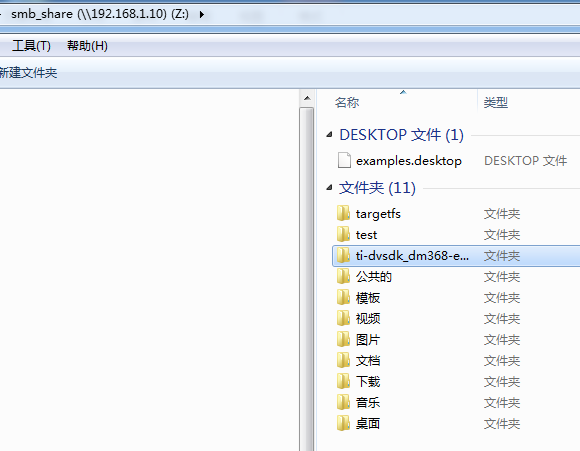
在虚拟机与主机间建立SAMBA服务来共享虚拟机上的路径，配置参照blog上的方法。需要注意的是，在我的环境上，必须将路由器启动起来，才能访问。否则会有无法连接提示。



连接时会要求输入用户名和密码（SAMBA专有的用户与密码）

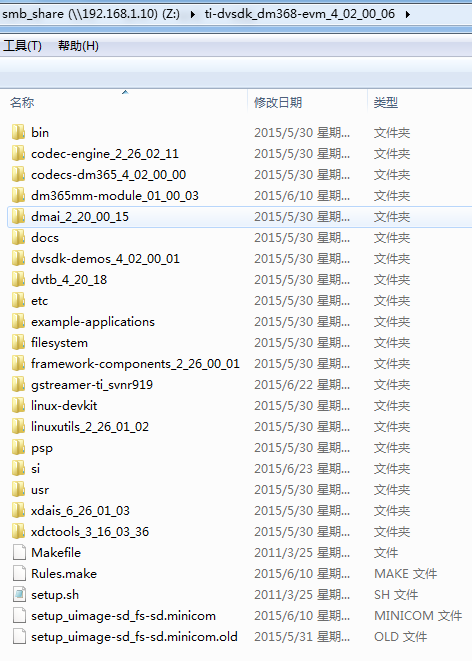


可勾住“记住我的凭据”，这样以后登录就不用重复输入了。我的环境的将虚拟机的桌面共享到了主机。



特别将Z:\ti-dvsdk\_dm368-evm\_4\_02\_00\_06共享。完成虚拟机与主机的DVSDK开发环境、代码共享。省却了频繁复制代码的过程和麻烦。

# 文件目录说明



## bin 文件夹

安装 DVSDK4\_03 的时候，用到的安装脚本，可以打开看看这些脚本理解，和编译无关。

## codec-engine\_2\_26\_02\_11

 Codecengine 是 TI DAVINCI 双核的核心思想， codec-engine\_2\_26\_02\_11\packages\ti\sdo\ce 里边有很多算法接口，比 DM6446 多了一个 vidanalytics ，设计到图像分析 LIB。

## dmai\_2\_20\_00\_15

这个是一个重要的元素，你开发 demos 的时候用到，见

TMS320DM3730\_Software\_Developers\_Guide.pd

里第 20 页。里边 packages\ti\sdo\dmai\apps 有很 多如何直接在 ARM 调用 DSP 的例子，注意如果你要编译 dmai\apps\video\_encode\_io 的例子，就必须使用 dmai\ce 目录下的 Venc\_org.c\_bk 和 Venc\_org.h\_bk （记得改回名字 Venc.c 和 Venc.h ），而当前 Venc.c 和 Venc.h 被我们修改过满足从 ARM 端传输参数给 DSP 了，和 encode 例子配合的， 更完美。

## dvsdk-demos\_4\_02\_00\_01

encode 是进行 H264 encode  的 D1 例子，可以 D1 encode 和音频输入 G711 encode ，带显示输出；

decode 可以进行 H264 DECODE 和 G711 DECODE ；

edge\_detection 是一个如何使用 c6accel 的例子，如何使用 VLIB ；

## dvtb\_4\_20\_18

这个也是一个 CODEC 应用的特例。

## example-applications

有关 matrix-gui-e-1.3 的开发包,主要是QT+html5框架的GUI图形界面开发接口，可扩展性非常强大。

## filesystem

DM3730 的文件系统，包括 NFS 文件系统和对应生成 ubifs 的工具；

## framework-components\_2\_26\_00\_01

这个也不需要我们编译， NX 的人可以去修改对应的驱动；

## gstreamer-ti\_svnr919

著名流媒体开源软件框架，编译出来为gst-launch等一系列命令行工具，可以通过Linux V4L2视频框架采集的视频及ALSA音频框架采集的音频通过 tcp/udp/rtp/rtsp等流媒体传输协议构建一个流媒体服务器，而其他终端上的流媒体体播放器则可以访问该流媒体服务器，如何利用好该开源框架，将会使 开发工作达到事半功倍的效果。

总的来说， dvsdk-demos\_4\_02\_00\_01 里边的例子就是 LINUX 的上层应用软件，而 DMAI 是中间件，也是 LINUX 深一层的应用程序，是上层应用软件 通过 DMAI 调用 codec-engine 、 dsplink 、 framework 的等元素。其实双核就是 ARM 是 HOST ，而 DSP 只是一个屏蔽的外围设备， ARM 端通过初始化 和调用一个函数就可以访问 DSP 了， DSP 处理的结果就放到共享内存里供 ARM 使用。

## linuxutils\_2\_26\_02\_05

编译这个 cmem 和 sdma 元素之前，必须先编译内核 linux-2.6.37 ；

## psp

其实个人感觉这个目录应该叫做bsp（板级支持包），不知道为什么叫psp，反正该目录的源码都是系统软件层等非常底层，如x-loader（其工作原理及流程见下篇博文）用来从不同介质引导u-boot，而u-boot的作用则调试和引导Linux内核。

* x-load-1.51 是支持 NAND BOOT 的；
* x-load-1.51-mmc 是我们自己改出来的版本，是支持 SD 卡 BOOT 的，适合生产和产品维修，使用这个编译的文件进行 SD 卡 BOOT ；
* 注意： u-boot-2010.06-psp04.02.00.07.sdk 、 linux-driver-examples-psp04.02.00.07 和 linux-2.6.37-psp04.02.00.07.sdk 是 TI DVSDK4\_03 安装的原始软件包。

## xdais\_6\_26\_01\_03

这个就是被中国一些 DSP 工程师称为万恶的 XDAIS 算法接口，把中国很多只会使用 CCS 调试程序的工程师搞得半死，很多人一直停留在 DM642 这种 单核的状态。其实多使用和研究透这个 XDAIS ，就发觉 TI 的良苦用心。本人改了 xdais\_6\_26\_01\_03\packages\ti\xdais\dm 里边 ividenc.h 进行 ARM 传参 数给 DSP 和 DSP 给 ARM 传参数，还改了 dmai\_2\_20\_00\_15\packages\ti\sdo\dmai\ce 里边的 Venc.c 和 Venc.h ，然后在 dvsdk-demos\_4\_02\_00\_01\omap3530 \encode 例子 capture.c 里举例如何在 ARM 端调用 DSP 的算法 Venc\_process() 。

## xdctools\_3\_16\_03\_36

这个是整个 DVSDK 的编译工具。

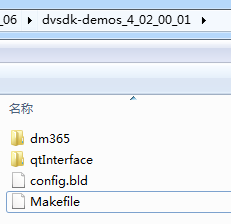
## 总结

TI 的 DSP 是个强大的好东西，特别非常适合自己开发算法的中国工程师（当然绝大部分算法都是 COPY 移植过来的），因为编程灵活，支持 C 语 言，开发成本非常低（相对 FPGA ）。在嵌入式领域，处理数字信号算法等等，目前 COTEX-A15 1.4G 都跑不赢 800MHz 的 C64+ 。当然支持浮点的 C674X 和 C66X   系列 DSP 就更恐怖了。跟我们桐烨科技合作的一些博导，对这方面感触非常深。其实，应用 TI DSP ，他们还没有完全达到骨灰级的水 平，因为很多片上资源和优化 LIB 都没用上。

## 参考

http://www.tuicool.com/articles/uUZvm2

# dvsdk-demos\_4\_02\_00\_01代码阅读



## makefile

makefile说明了可以编辑的目标。还看不懂编译依赖关系。

|  |  |
| --- | --- |
| # Makefile for making the DVSDK demos  ROOTDIR = ..  include $(ROOTDIR)/Rules.make  SUBDIRS = $(filter-out data/ packages/ patches/ qtInterface/, $(sort $(dir $(wildcard \*/))))  CLEAN\_SUBDIRS = $(addsuffix .clean, $(SUBDIRS))  INSTALL\_SUBDIRS = $(addsuffix .install, $(SUBDIRS))  .PHONY: install clean $(SUBDIRS) $(INSTALL\_SUBDIRS) $(CLEAN\_SUBDIRS) dm6467 dm6446 dm355 omap3530 dm365 dm368 dm3730  all: $(if $(wildcard dm6467), dm6467) \  $(if $(wildcard dm6446), dm6446) \  $(if $(wildcard dm365), dm365) \  $(if $(wildcard omap3530), omap3530) \  $(if $(wildcard dm355), dm355) \  $(if $(wildcard dm3730), dm3730) \  $(if $(wildcard dm368), dm368)  simplewidget:  dm6467:  @echo  @echo Building simplewidget..  $(MAKE) -C packages/ti/sdo/simplewidget dm6467  @echo Making all in subdirectory $@...  $(MAKE) -C dm6467  dm6446:  @echo  @echo Building simplewidget..  $(MAKE) -C packages/ti/sdo/simplewidget dm6446  @echo Making all in subdirectory $@...  $(MAKE) -C dm6446  dm365:  @echo  @echo Making all in subdirectory $@...  $(MAKE) -C dm365  dm355:  @echo  @echo Building simplewidget..  $(MAKE) -C packages/ti/sdo/simplewidget dm355  @echo Making all in subdirectory $@...  $(MAKE) -C dm355  omap3530:  @echo  @echo Making all in subdirectory $@...  $(MAKE) -C omap3530  dm3730:  @echo  @echo Making all in subdirectory $@...  $(MAKE) -C dm3730  dm368:  @echo  @echo Making all in subdirectory $@...  $(MAKE) -C dm368  install: $(INSTALL\_SUBDIRS)  $(INSTALL\_SUBDIRS):  @echo  @echo Executing make install in subdirectory $(basename $@)...  @cd $(basename $@) ; $(MAKE) install  clean: $(CLEAN\_SUBDIRS) $(if $(wildcard packages), simplewidget\_clean)  simplewidget\_clean:  @make -C packages/ti/sdo/simplewidget clean  $(CLEAN\_SUBDIRS):  @cd $(basename $@) ; $(MAKE) clean | 包含上一层的make规则文件  filter-out反过滤函数  排序函数——sort  Wildcard通配符  addsuffix加后缀函数  .PHONY”表示伪目标文件  -C <dir>”  “--directory=<dir>”  指定读取 makefile 的目录  “$@”表  示目标的集合，就像一个数组，“$@”依次取出目标，并执于命令 |

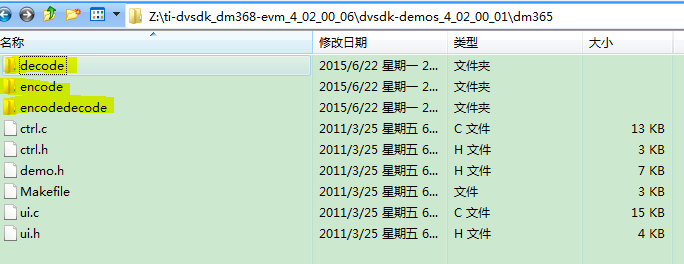
## config.bld

包含了编译工具的路径、配置等信息。

|  |
| --- |
| /\* location of your C6000 codegen tools \*/  var C64P = xdc.useModule('ti.targets.C64P');  C64P.rootDir = java.lang.System.getenv("CODEGEN\_INSTALL\_DIR");  /\* location of your Montavista Arm9 tools \*/  var MVArm9 = xdc.useModule('gnu.targets.MVArm9');  MVArm9.rootDir = java.lang.System.getenv("MVTOOL\_DIR");  MVArm9.ccOpts.prefix = "-g";  MVArm9.ccOpts.suffix = "";  /\* location of the Codec Sourcery Arm9 tools \*/  var GCArmv5T = xdc.useModule('gnu.targets.arm.GCArmv5T');  GCArmv5T.LONGNAME = 'bin/arm-none-linux-gnueabi-gcc';  GCArmv5T.platform = "ti.platforms.evm3530";  GCArmv5T.rootDir = java.lang.System.getenv("CSTOOL\_DIR");  /\*  \* ======== Build.targets ========  \* list of targets (ISAs + compilers) to build for  \*/  Build.targets = [  ]; |

## 代码组成

该文件夹下实例有三个main.c，一个是encode，一个是decode，一个是encodedecode。也就是三个实现实例，分别用来实现编码、解码、编解码。



* + decode：音频解码设备操作和调用源码。
  + encode：音频编码设备稻作和调用源码。
  + encodedecode：视频输入输出设备操作和调用源码，及UI显示操作源码。

### encodedecode

LENA系统只有视频，没有音频。直接学习这部分。

#### encodedecode说明

|  |
| --- |
| NAME  encodedecode - encode and decode video  SYNOPSIS  encodedecode [options...]  DESCRIPTION  This demo uses the Codec Engine to encode data from the capture device  into an intermediate buffer before the data is decoded to the display  framebuffer.  The DM365MM and CMEM kernel modules need to be inserted for this demo  to run. Use the script 'loadmodule-rc' in the DVSDK to make sure both  kernel modules are loaded with adequate parameters.  OPTIONS  -y <1-7>, --display\_standard <1-7>  Sets the resolution of the display. If the captured resolution  is larger than the display it will be center clamped, and if it  is smaller the image will be centered.  1 D1 @ 30 fps (NTSC)  2 D1 @ 25 fps (PAL)  3 720P @ 60 fps [Default]  7 480P @ 60 fps  -v <videocodec>, --videocodec <h264 or mpeg4 or mpeg2>  The video codec to be used for encode and decode  -r <resolution>, --resolution <resolution>  The resolution of video to encode and decode in the format  'width'x'height'. Default is the resolution of the input video  standard detected.  -b <bit rate>, --bitrate <bit rate>  This option sets the bit rate with which the video will be  encoded. Use a negative value for variable bit rate. Default is  variable bit rate.  -p, --passthrough  Pass the video through from capture device to display device  without encoding or decoding the data.  -I, --video\_input  Video input source to use.  1 Composite  2 S-video  3 Component  4 Imager/Camera - for DM368  When not specified, the video input is chosen based on the display  video standard selected. NTSC/PAL use Composite, and 480P/720P use  Component.  -k, --keyboard  Enables the keyboard input mode which lets the user input  commands using the keyboard in addition to the QT-based OSD  interface. At the prompt type 'help' for a list of available  commands.  -t <seconds>, --time <seconds>  The number of seconds to run the demo. Defaults to infinite time.  -o, --osd  Enables the On Screen Display for configuration and data  visualization using a QT-based UI. If this option is not passed,  the data will be output to stdout instead.  -h, --help  This will print the usage of the demo.  EXAMPLE USAGE  First execute this script to load kernel modules required:  ./loadmodules.sh  General usage:  ./encodedecode -h  Use NTSC CIF resolution with keyboard interface and quit after 20 sec:  ./encodedecode -r 352x240 -v h264 -t 20 -k  Use a PAL display with OSD display and mpeg4 codec:  ./encodedecode -y 2 -v mpeg4 -o  Use default resolution (720x480) at 2Mbit bit rate with mpeg4 codec and s-video input:  ./encodedecode -b 2000000 -I 2 |

#### captureThrFxn

#### videoThrFxn

#### displayThrFxn

#### passThrough

作用：不编解码视频，直接从捕获设备传输到显示设备。

结论：LENA是AIR\_STATION捕获、GROUND\_STATION显示，无法使用该功能。

思考：在DEMO代码中，基于该变量做了不少分支，有些乱。感觉设计为直接在顶层分支处理，更合理。

### decode

#### decode说明

|  |
| --- |
| NAME  decode - decode video and/or audio or speech files  SYNOPSIS  decode [options...]  DESCRIPTION  This demo uses the Codec Engine to decode data from files and output  the uncompressed data using peripheral device drivers. Video, audio and  speech files are supported. All files must consist of raw encoded  frames of data (elementary streams). For MPEG4 and H264 video, we use  the restricted version of the decoder, hence only videos encoded with  TI encoder can be decoded.  You must supply at least one file for the demo to run.  The DM365MM and CMEM kernel modules need to be inserted for this demo  to run. Use the script 'loadmodule-rc' in the DVSDK to make sure both  kernel modules are loaded with adequate parameters.  OPTIONS  -v <video file>, --videofile <video file>  Decodes a video file. The demo detects which type of video file  is supplied using the file extension. Supported video algorithms  are:  MPEG4 SP (.mpeg4 or .m4v extension)  H.264 HP (.264 extension)  MPEG2 (.m2v extension)  -s <speech file>, --speechfile <speech file>  Decodes the speech file. The demo detects which type of speech  file is supplied using the file extension. The only supported  speech algorithm as of now is G.711 (.g711 extension).  -a <audio file>, --audiofile <audio file>  Decodes the audio file. The demo detects which type of audio  file is supplied using the file extension. The only supported  audio algorithm as of now is AAC (.aac extension).  -y <1-3>, --display\_standard <1-3>  Sets the resolution of the display. Clips larger than the display  will be center clamped, and clips smaller than the display will  be centered.  1 D1 @ 30 fps (NTSC)  2 D1 @ 25 fps (PAL)  3 720P @ 60 fps [Default]  5 1080I @ 30 fps - for DM368  -O <display output>, --display\_output  Selects the video output to display, valid outputs are  'composite', or 'component'.  -l, --loop  If this option is selected, the demo will restart at the  beginning of a file (or files) when the end is reached. If this  option is not selected (which is the default) the demo will quit  once the end of the file (or files) has been reached.  -o, --osd  Enables the On Screen Display for configuration and data  visualization using a QT-based UI. If this option is not passed,  the data will be output to stdout instead.  -k, --keyboard  Enables the keyboard input mode which lets the user input  commands using the keyboard in addition to the QT-based OSD  interface. At the prompt type 'help' for a list of available  commands.  -t <seconds>, --time <seconds>  The number of seconds to run the demo. Defaults to infinite time.  -h, --help  This will print the usage of the demo.  EXAMPLE USAGE  First execute this script to load the kernel modules required:  ./loadmodules.sh  General usage:  ./decode -h  H264 HP video on a 720p display and G.711 speech decode with OSD:  ./decode -s test.g711 -v test.264 -o -y 3  MPEG4 NTSC video decode only with keyboard interface on component output:  ./decode -v test.mpeg4 -k -y 1 |

#### displayThrFxn

#### BufferGfx\_calcLineLength与ColorSpace\_getBpp

|  |
| --- |
| if (colorSpace == ColorSpace\_YUV420PSEMI) {  bufSize = gfxAttrs.dim.lineLength \* gfxAttrs.dim.height \* 3 / 2;  } else {  bufSize = gfxAttrs.dim.lineLength \* gfxAttrs.dim.height \* 2;  } |
| Int ColorSpace\_getBpp(ColorSpace\_Type colorSpace)  {  Int bpp;  switch (colorSpace) {  case ColorSpace\_RGB888:  bpp = 32;  break;  case ColorSpace\_YUV444P:  bpp = 24;  break;  case ColorSpace\_RGB565:  case ColorSpace\_UYVY:  bpp = 16;  break;  case ColorSpace\_GRAY:  case ColorSpace\_YUV422P:  case ColorSpace\_YUV420P:  case ColorSpace\_YUV420PSEMI:  case ColorSpace\_YUV422PSEMI:  bpp = 8;  break;    case ColorSpace\_2BIT:  bpp = 2;  break;  default:  Dmai\_err1("Unknown color space format (%d)\n", colorSpace);  return Dmai\_EINVAL;  }  return bpp;  } |
| /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* BufferGfx\_calcLineLength  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  Int32 BufferGfx\_calcLineLength(Int32 width, ColorSpace\_Type colorSpace)  {  Int bpp = ColorSpace\_getBpp(colorSpace);  if (bpp < 0) {  return bpp;  }  return width \* bpp / 8;  } |

### encode

#### encode说明

|  |
| --- |
| NAME  encode - encode video and/or audio or speech files  SYNOPSIS  encode [options...]  DESCRIPTION  This demo uses the Codec Engine to encode data from peripheral device  drivers to files. Video, audio and speech files are supported. The files  created will be encoded elementary streams of video/audio/speech.  You must supply at least one file for the demo to run.  The DM365MM and CMEM kernel modules need to be inserted for this demo  to run. Use the script 'loadmodule-rc' in the DVSDK to make sure both  kernel modules are loaded with adequate parameters.  OPTIONS  -v <video file>, --videofile <video file>  Encodes video data to the given file. The file will be created if  it doesn't exist, and truncated if it does exist. The demo  detects which type of video file is supplied using the file  extension. Supported video algorithm is MPEG4 SP, H.264 MP, MPEG2  (.mpeg4 or .m4v extension, .264, .m2v).  -s <speech file>, --speechfile <speech file>  Encodes speech data to the given file. The file will be created  if it doesn't exist, and truncated if it does exist. The demo  detects which type of speech file is supplied using the file  extension. The only supported speech algorithm as of now is  G.711 (.g711 extension).  -a <audio file>, --audiofile <audio file>  Encodes audio data to the given file. The file will be created  if it doesn't exist, and truncated if it does exist. The demo  detects which type of audio file is supplied using the file  extension. The only supported speech algorithm as of now is  AAC (.aac extension).  -y <1-5>, --display\_standard <1-5>  Sets the resolution of the display. If the captured resolution  is larger than the display it will be center clamped, and if it  is smaller the image will be centered.  1 D1 @ 30 fps (NTSC)  2 D1 @ 25 fps (PAL)  3 720P @ 60 fps [Default]  5 1080I @ 30 fps - for DM368  -r <resolution>, --resolution <resolution>  The resolution of video to encode in the format 'width'x'height'.  Default is the size of the video standard (720x480 for NTSC,  720x576 for PAL, 1280x720 for 720P).  -b <bit rate>, --videobitrate <bit rate>  This option sets the bit rate with which the video will be  encoded. Use a negative value for variable bit rate. Default is  variable bit rate.  -p <bit rate>, --soundbitrate <bit rate>  This option sets the bit rate with which the audio will be  encoded. Use a negative value for variable bit rate. Default is  96000.  -u <sample rate>, --samplerate <sample rate>  This option sets the sample rate with which the video will be  encoded. Default is 44100 Hz.  -w, --preview\_disable  Disable preview of captured video frames.  -f, --write\_disable  Disable recording of encoded file. This helps to validate  performance without file I/O.  -I, --video\_input  Video input source to use.  1 Composite  2 S-video  3 Component  4 Imager/Camera - for DM368  When not specified, the video input is chosen based on the display  video standard selected. NTSC/PAL use Composite, 720P uses  Component, and 1080I uses the Imager/Camera.  -l, --linein  This option makes the input device for sound recording be the  'line in' as opposed to the 'mic in' default.  -k, --keyboard  Enables the keyboard input mode which lets the user input  commands using the keyboard in addition to the QT-based OSD  interface. At the prompt type 'help' for a list of available  commands.  -t <seconds>, --time <seconds>  The number of seconds to run the demo. Defaults to infinite time.  -o, --osd  Enables the On Screen Display for configuration and data  visualization using a QT-based UI. If this option is not passed,  the data will be output to stdout instead.  -h, --help  This will print the usage of the demo.  EXAMPLE USAGE  First execute this script to load kernel modules required:  ./loadmodules.sh  General usage:  ./encode -h    H264 HP video encode only @720p resolution with OSD:  ./encode -v test.264 -y 3 -o    H264 HP video encode from s-video and G.711 speech encode:  ./encode -v test.264 -s test.g711 -I 2  MPEG4 SP video encode only in CIF NTSC resolution with OSD:  ./encode -v test.mpeg4 -r 352x240 -o  MPEG4 SP video encode at 1Mbps with keyboard interface on D1 PAL display:  ./encode -v test.mpeg4 -b 1000000 -k -y 2 |

#### writerThrFxn

|  |  |
| --- | --- |
| /\* Open the output video file \*/  outFile = fopen(envp->videoFile, "w"); | 打开字符文件？为什么不是二进制文件？ |

设计时，应该是将EMIF传输数据设备，由底层封装为一个文件接口，此处调用文件接口，即可将数据写入。

此处的写的数据大小是？

#### captureThrFxn内存分配参数

|  |
| --- |
| #define Buffer\_Memory\_Params\_DEFAULT\_DEFINE { \  Memory\_CONTIGPOOL, \  Memory\_NONCACHED, \  Memory\_DEFAULTALIGNMENT, \  0 \  } |
| #define Buffer\_Memory\_Params\_DEFAULT\_DEFINE { \  Memory\_CONTIGPOOL, \  Memory\_NONCACHED, \  Memory\_DEFAULTALIGNMENT, \  0 \  } |

#### captureThrFxn分辨率支持

这块有点乱，需要整理下。

|  |
| --- |
| /\* We only support D1, 720P and 1080P input \*/  if (videoStd != VideoStd\_D1\_NTSC && videoStd != VideoStd\_D1\_PAL  && videoStd != VideoStd\_720P\_60 && videoStd != VideoStd\_720P\_50 &&  videoStd != VideoStd\_1080I\_30) {  ERR("Need D1/720P/1080P input to this demo\n");  cleanup(THREAD\_FAILURE);  } |
| /\* Capture at half frame rate if using COMPONENT input at 720P \*/  if ((envp->videoStd == VideoStd\_720P\_60)  && (envp->videoInput == Capture\_Input\_COMPONENT)) {  cAttrs.videoStd = VideoStd\_720P\_30;  }  else {  cAttrs.videoStd = envp->videoStd;  }  /\*If its component input and video std is 1080I\_30 then make it 1080I\_60.\*/  if (cAttrs.videoStd == VideoStd\_1080I\_30 && cAttrs.videoInput  == Capture\_Input\_COMPONENT) {  cAttrs.videoStd = VideoStd\_1080I\_60;  } |
| /\* In DM365 the capture driver does not support 720P-30, but the  LSP has an option to reduce the capture rate to half. So when user  sets for 720P-30, we treat it as 720P-60 except we set LSP capture  rate to half. \*/  /\* NOTE: make execption for DM368 camera input which supports 720P-30\*/  if (attrs->videoStd == VideoStd\_720P\_30 && attrs->videoInput !=  Capture\_Input\_CAMERA) {  halfRateCapture = TRUE;  attrs->videoStd = VideoStd\_720P\_60;  } |

#### Dmai\_roundUp

|  |
| --- |
| /\*\*  \* @brief Macro that can be used to round up 'val' to the closest  \* multiple of the value passed in 'rounding'.  \*  \* @param[in] val The value to be rounded  \* @param[in] rounding The closest multiple of which will be returned. Must be  \* a power of 2.  \*  \* @retval Rounded result  \*/  #define Dmai\_roundUp(val, rounding) ( ( (val) & ((rounding) - 1) ) ? \  (((val) & ~((rounding) - 1)) + (rounding)) : (val) ) |

#### ColorSpace\_Type

|  |
| --- |
| ColorSpace\_Type colorSpace = ColorSpace\_YUV420PSEMI; |
| switch (colorSpace) {  case ColorSpace\_RGB888:  bpp = 32;  break;  case ColorSpace\_YUV444P:  bpp = 24;  break;  case ColorSpace\_RGB565:  case ColorSpace\_UYVY:  bpp = 16;  break;  case ColorSpace\_GRAY:  case ColorSpace\_YUV422P:  case ColorSpace\_YUV420P:  case ColorSpace\_YUV420PSEMI:  case ColorSpace\_YUV422PSEMI:  bpp = 8;  break;    case ColorSpace\_2BIT:  bpp = 2;  break; |

### 关键数据结构

该部分列出在视频处理中的关键数据结构。

#### Capture\_Input视频输入

|  |  |
| --- | --- |
| typedef enum {  /\*\* @brief S-Video video input \*/  Capture\_Input\_SVIDEO = 0,  /\*\* @brief Composite video input \*/  Capture\_Input\_COMPOSITE,  /\*\* @brief Component video input \*/  Capture\_Input\_COMPONENT,  /\*\* @brief Camera/Imager card video input  \* @remarks only applicable on DM368  \*/  Capture\_Input\_CAMERA,  Capture\_Input\_COUNT  } Capture\_Input; | S-video输入  复合视频、模拟视频  分量视频、数字视频  照相机  ..... |

#### VideoStd\_Type视频标准格式

红色标注的是我们需要支持的场景。模拟视频、数字视频

|  |  |
| --- | --- |
| /\*  \* WARNING! When adding a member to this enumerated type, make sure you  \* add members to the arrays in linux/Capture.c and linux/Display.c.  \*/  /\*\* @brief Video standards \*/  typedef enum {  VideoStd\_AUTO = 0, /\*\*< Automatically select standard (if supported) \*/  VideoStd\_CIF, /\*\*< CIF @ 30 frames per second \*/  VideoStd\_SIF\_NTSC, /\*\*< SIF @ 30 frames per second \*/  VideoStd\_SIF\_PAL, /\*\*< SIF @ 25 frames per second \*/  VideoStd\_VGA, /\*\*< VGA (640x480) @ 60 frames per second \*/  VideoStd\_D1\_NTSC, /\*\*< D1 NTSC @ 30 frames per second \*/  VideoStd\_D1\_PAL, /\*\*< D1 PAL @ 25 frames per second \*/  VideoStd\_480P, /\*\*< D1 Progressive NTSC @ 60 frames per second \*/  VideoStd\_576P, /\*\*< D1 Progressive PAL @ 50 frames per second \*/  VideoStd\_720P\_60, /\*\*< 720P @ 60 frames per second \*/  VideoStd\_720P\_50, /\*\*< 720P @ 50 frames per second \*/  VideoStd\_720P\_30, /\*\*< 720P @ 30 frames per second \*/  VideoStd\_1080I\_30, /\*\*< 1080I @ 30 frames per second \*/  VideoStd\_1080I\_25, /\*\*< 1080I @ 25 frames per second \*/  VideoStd\_1080P\_30, /\*\*< 1080P @ 30 frames per second \*/  VideoStd\_1080P\_25, /\*\*< 1080P @ 25 frames per second \*/  VideoStd\_1080P\_24, /\*\*< 1080P @ 24 frames per second \*/  VideoStd\_QVGA, /\*\*< QVGA @ 30 frames per second \*/  VideoStd\_1080P\_60, /\*\*< 1080P @ 60 frames per second \*/  VideoStd\_1080P\_50, /\*\*< 1080P @ 50 frames per second \*/  VideoStd\_1080I\_60, /\*\*< 1080I @ 60 frames per second \*/  VideoStd\_COUNT  } VideoStd\_Type; | 当增加一个成员时，确保要在相应的数组中也增加成员 |

#### Display\_Std显示标准

|  |  |
| --- | --- |
| typedef enum {  /\*\* @brief v4l2 video standard \*/  Display\_Std\_V4L2 = 0,  /\*\* @brief Fbdev video standard \*/  Display\_Std\_FBDEV,  Display\_Std\_COUNT  } Display\_Std; |  |

#### ColorSpace\_Type色彩空间类型

|  |  |
| --- | --- |
| /\*\*  \* @brief Color formats.  \*/  typedef enum {  ColorSpace\_NOTSET = -1,  /\*\*\* @brief YUV 420 semi planar corresponding to V4L2\_PIX\_FMT\_NV12 in v4l2.  \* This format consists of two planes: one with the Y component  \* and one with the CbCr components interleaved (hence semi).  \* See the LSP documentation for a thorough description of this  \* format.  \*/  ColorSpace\_YUV420PSEMI = 0,  /\*\*\* @brief YUV 422 semi planar corresponding to V4L2\_PIX\_FMT\_YUV422UVP. This  \* format was added to v4l2 by TI because the dm6467 VDCE and VPSS  \* peripherals use this format. The format consists of two planes:  \* one with the Y component and one with the CbCr components  \* interleaved (hence semi) See the LSP VDCE documentation for a  \* thorough description of this format.  \*/  ColorSpace\_YUV422PSEMI,  /\*\* @brief YUV 422 interleaved corresponding to V4L2\_PIX\_FMT\_UYVY in v4l2 \*/  ColorSpace\_UYVY,    /\*\* @brief RGB 888 packed corresponding to V4L2\_PIX\_FMT\_RGB24 in v4l2 \*/  ColorSpace\_RGB888,  /\*\* @brief RGB 565 packed corresponding to V4L2\_PIX\_FMT\_RGB565 in v4l2 \*/  ColorSpace\_RGB565,  /\*\* \* @brief 2 bits per pixel. This is the format used by the VDCE for the  \* bitmap while blending and is documented in the VDCE peripheral  \* guide.  \*/  ColorSpace\_2BIT,    /\*\*\* @brief YUV 420 planar. The format consists of three planes:  \* one with the Y component, one Cb, and one Cr component.  \*/  ColorSpace\_YUV420P,    /\*\*\* @brief YUV 422 planar. The format consists of three planes:  \* one with the Y component, one Cb, and one Cr component.  \*/  ColorSpace\_YUV422P,  /\*\*\* @brief YUV 444 planar. The format consists of three planes:  \* one with the Y component, one Cb, and one Cr component.  \*/  ColorSpace\_YUV444P,    /\*\*\* @brief Gray Scale. The format consist of single Luma plane  \* ignoring the color plane components.  \*  \*/  ColorSpace\_GRAY,  ColorSpace\_COUNT  } ColorSpace\_Type; | 色彩格式  没有设置 |
| const Capture\_Attrs Capture\_Attrs\_DM365\_DEFAULT = {  3,  Capture\_Input\_COMPONENT,  -1,  -1,  -1,  -1,  "/dev/video0",  FALSE,  VideoStd\_AUTO,  -1,  ColorSpace\_YUV420PSEMI,  NULL,  TRUE  }; |  |

#### VideoCodec视频编解码标准

|  |  |
| --- | --- |
| /\* The video encoders supported by this demo \*/  typedef enum VideoCodec {  MPEG4,  H264,  MPEG2,  NUM\_VIDEO\_ENCODERS  } VideoCodec; |  |

#### Capture\_Attrs视频捕获属性

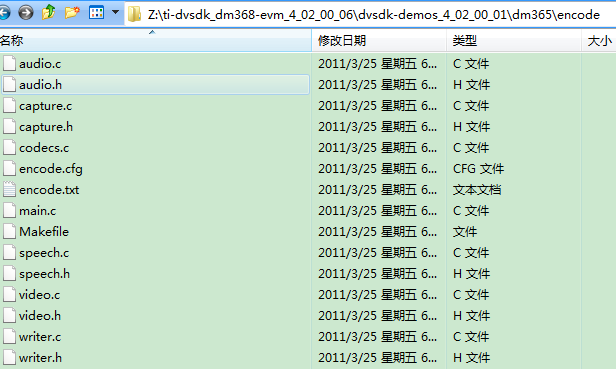
|  |  |
| --- | --- |
| /\*\*  \* @brief Attributes used to create a Capture device driver instance.  \*  \* @see Capture\_Attrs\_DM6446\_DM355\_DEFAULT  \* @see Capture\_Attrs\_DM6467\_DEFAULT  \*/  typedef struct Capture\_Attrs {  /\*\* @brief Number of buffers the driver is to use, either allocated  \* from the device driver or the supplied #BufTab\_Handle. \*/  Int numBufs;  /\*\* @brief Which video input to use. \*/  Capture\_Input videoInput;  /\*\* @brief Width of image to crop (if supported by H/W, otherwise ignored).  \* If set to a negative value, no cropping will be used. This value needs  \* to be even. \*/  Int32 cropWidth;  /\*\* @brief Height of image to crop (if supported by H/W, otherwise ignored).  \* If set to a negative value, no cropping will be used. \*/  Int32 cropHeight;  /\*\* @brief X pos of image to crop (if supported by H/W, otherwise ignored).  \* If set to a negative value, no cropping will be used. \*/  Int32 cropX;  /\*\* @brief Y pos of image to crop (if supported by H/W, otherwise ignored).  \* If set to a negative value, no cropping will be used. \*/  Int32 cropY;  /\*\* @brief Name of the Linux v4l2 capture device to use.  \*  \* @remarks Only applicable on Linux.  \*/  Char \*captureDevice;  /\*\* @brief When using the dm6446 resizer to @ref ti\_sdo\_dmai\_Smooth the  \* captured images, 2 extra rows at the top and 4 extra rows at the  \* bottom needs to be captured (if possible).  \*  **\* @remarks Only applicable for dm6446 / dm357 Linux**.  \*/  Bool smoothPad;  /\*\* @brief To force a certain video standard without auto detection,  \* set this variable to something other than #VideoStd\_AUTO.  \*/  VideoStd\_Type videoStd;  /\*\* @brief To force a certain decoder on dm6467 instead of  \* auto detecting which decoder supports your #Capture\_Attrs.videoInput,  \* set this to a value > -1. See the VPIF device driver guide on which  \* values apply (VIDIOC\_S\_STD ioctl). This is useful when auto detection  \* doesn't work, e.g. when two EVM:s are interconnected.  \*  \* @remarks The detected #Capture\_Input on the decoder still needs to  \* correspond to the #Capture\_Attrs.videoInput set, or the function will  \* fail.  **\* @remarks Only applicable for dm6467 Linux.**  \*/  Int decoderIdx;    /\*\* @brief Output color format  \*  **\* @remarks Only applicable for dm365 Linux.**  \*/  ColorSpace\_Type colorSpace;  /\*\* @brief Capture image size  \*  \* @remarks Only height and width elements are used.  \* @remarks Set it to NULL, if resolution needs to be auto detected.  **\* @remarks Only applicable for dm365 Linux.**  \*/  BufferGfx\_Dimensions \*captureDimension;  /\*\* @brief Set the resizer/previewer to "on the fly mode". In this  \* mode the capture device is chained with previewer/resizer to resize  \* and do chroma conversion on the fly.  \*  **\* @remarks Only applicable for dm365 Linux.**  \* @remarks If this flag is set for composite input, the captured  \* frames will be de-interlaced automatically.  \* @remarks If this flag is enabled the kernel command line parameter  \* dm365\_imp.oper\_mode should be set to '1', otherwise '0'.  \*/  Bool onTheFly;  } Capture\_Attrs; | 飞行模式，该模式下取像设备连接到预览与缩放模块来在飞行时缩放和色彩转换。  如果在模拟输入时设置该标志，那么捕获视频会自动设置为去隔行。 |
| /\*\*  \* @brief Default attributes for video window 0 on dm365.  \* @code  \* numBufs = 3,  \* videoInput = Capture\_Input\_COMPONENT,  \* cropWidth = -1,  \* cropHeight = -1,  \* cropX = -1,  \* cropY = -1,  \* captureDevice = "/dev/video0",  \* smoothPad = FALSE,  \* videoStd = VideoStd\_AUTO,  \* decoderIdx = -1  \* colorSpace = ColorSpace\_YUV420PSEMI,  \* captureDimension = NULL,  \* onTheFly = TRUE  \* @endcode  \*/ |  |

|  |  |
| --- | --- |
| typedef struct Display\_Attrs {  /\*\* @brief Number of buffers the driver is to use, either allocated  \* from the device driver or the supplied #BufTab\_Handle. \*/  Int numBufs;  /\*\* @brief Device driver standard to use. (see #Display\_Std).  \* @remarks Only applicable on Linux.  \*/  Display\_Std displayStd;  /\*\* @brief Video standard to be used by driver. If Display\_Attrs.videoStd  \* is set to VideoStd\_AUTO then video standard is determined by sysfs.  \*/  VideoStd\_Type videoStd;  /\*\* @brief Which video output to use. If Display\_Attrs.videoOutput is set  \* to Display\_Output\_SYSTEM then video output device is determind by sysfs.  \*/  Display\_Output videoOutput;  /\*\* @brief Name of fbdev or v4l2 display device to use.  \* @remarks Only applicable on Linux.  \*/  Char \*displayDevice;  /\*\* @brief Degrees to rotate the display, valid values are 0, 90, 180  \* and 270.  \* @remarks Only applicable on omap3530/dm3730.  \*/  Int rotation;  /\*\* @brief Display colorspace to use. (see #ColorSpace\_Type). \*/  ColorSpace\_Type colorSpace;  /\*\* @brief Video display width.  \*/  Int width;  /\*\* @brief Video display height.  \*/  Int height;  /\*\* @brief Delay the call to VIDIOC\_STREAMON until application calls  \* #Display\_Put with a display buffer.  \* @remarks Only applicable for V4L2, and currently only supported for  \* dm365 and omap3530/dm3730 Linux.  \*/  Int delayStreamon;  /\*\* @brief Instruct the display to use or simulate a desired framerate.  \* The framerate is represented as a fraction, and both #forceFrameRateNum  \* and #forceFrameRateDen must be set to specify the complete value.  \* Up to three decimal places of precision are supported.  \* @remarks Only applicable for V4L2, and currently only supported for  \* dm365.  \*/  Int forceFrameRateNum;  /\*\* @brief See #forceFrameRateNum.  \* @remarks Only applicable for V4L2, and currently only supported for  \* dm365.  \*/  Int forceFrameRateDen;  /\*\* @brief Do not execute VIDIOC\_STREAMON until application calls  \* Display\_control.  \* @remarks Only applicable for V4L2, and currently only supported on  \* omap3530/dm3730 Linux.  \*/  Int streamonDisable;  } Display\_Attrs; |  |
| const Display\_Attrs Display\_Attrs\_DM365\_VID\_DEFAULT = {  3,  Display\_Std\_V4L2,  VideoStd\_D1\_NTSC,  Display\_Output\_COMPOSITE,  "/dev/video2",  0,  ColorSpace\_YUV420PSEMI,  -1,  -1,  FALSE,  0,  0,  FALSE  };  const Display\_Attrs Display\_Attrs\_DM365\_OSD\_DEFAULT = {  2,  Display\_Std\_FBDEV,  VideoStd\_D1\_NTSC,  Display\_Output\_COMPOSITE,  "/dev/fb0",  0,  ColorSpace\_RGB565,  -1,  -1,  FALSE,  0,  0,  FALSE  };  const Display\_Attrs Display\_Attrs\_DM365\_ATTR\_DEFAULT = {  1,  Display\_Std\_FBDEV,  VideoStd\_D1\_NTSC,  Display\_Output\_COMPOSITE,  "/dev/fb2",  0,  ColorSpace\_RGB565,  -1,  -1,  FALSE,  0,  0,  FALSE  }; |  |

#### Args视频参数

|  |  |
| --- | --- |
| typedef struct Args {  VideoStd\_Type videoStd;  Char \*videoStdString;  Int videoBitRate;  Char \*videoBitRateString;  VideoCodec videoCodec;  Char \*videoCodecString;  Capture\_Input videoInput;  Int passThrough;  Int keyboard;  Int time;  Int osd;  Int32 imageWidth;  Int32 imageHeight;  } Args; | #define DEFAULT\_ARGS {  VideoStd\_720P\_60,  "720P 60Hz",  -1,  NULL,  MPEG4,  NULL,  Capture\_Input\_COUNT,  FALSE,  FALSE,  FOREVER,  FALSE,  0,  0} |

* + 1. encode



|  |  |  |
| --- | --- | --- |
| 文件名 | 功能 | 备注 |
|  |  |  |

* + - 1. main.c

此文件包含代码1200多行，是我学习DVSDK的切入点。两天前还曾经面对大约2GB的代码包一筹莫展，做了两天功课，终于找到一些下手的地方。感谢CSDN那些前辈们的blog。第一次阅读其他公司如此大量代码，发现TI的软件开发规范的确与我们不同。特点：

* + 注释特别少；连函数头说明等都省略了。
  + 无用代码也少，基本上干干净净。
  + 变量的命名还算可以，不会串话。不过有个基本说明还是比较好。

暂时还体会不出巧妙之处，边看边体会。

1. 头文件

包含三类头文件，

* 标准库
* TI库，一般是公用的
* 自定义的头文件，私有的。

1. main()

ok，让我们先看main函数。因代码量大，不会一一列出，仅列出关键代码的说明理解，以及代码大体功能、结构。

|  |  |
| --- | --- |
| Int main(Int argc, Char \*argv[]) | 虽然main函数的确可以输入参数，不过之前的开发习惯中很少用到。只是在偶尔的测试、学习代码中用到。 |

* 变量定义

包含定义和初始化。有几个宏需要说明：

下面这个宏定义了默认的视频输出格式、编解码。等看明白了，逐一说明参数的含义

|  |
| --- |
| Args args = DEFAULT\_ARGS; |
| #define DEFAULT\_ARGS \  { VideoStd\_720P\_60, "720P 60Hz", -1, NULL, MPEG4, NULL, \  Capture\_Input\_COUNT, FALSE, FALSE, FOREVER, FALSE, 0, 0} |
| VideoStd\_720P\_60：视频标准  "720P 60Hz：视频标准  MPEG4：视频编码  Capture\_Input\_COUNT |

* 宏定义

另外几个宏在代码下无定义。从名字看是默认属性，具体待确认。先列出：

|  |
| --- |
| Pause\_Attrs pAttrs = Pause\_Attrs\_DEFAULT;  Rendezvous\_Attrs rzvAttrs = Rendezvous\_Attrs\_DEFAULT;  Fifo\_Attrs fAttrs = Fifo\_Attrs\_DEFAULT; |
|  |
|  |

TI的手柄并没有定义为“p”前缀，应该是将函数指针和普通指针区分开来。

* OSD定义

|  |
| --- |
| UI\_Attrs uiAttrs; |
| 定义的是OSD，这是我们产品将来的用到的特性 |
| typedef struct UI\_Attrs {  Int osd;  VideoStd\_Type videoStd;  } UI\_Attrs; |

* 创建线程时传递的环境变量定义

以下几个结构体比较重要

|  |  |
| --- | --- |
| CaptureEnv captureEnv;  WriterEnv writerEnv;  VideoEnv videoEnv;  SpeechEnv speechEnv;  AudioEnv audioEnv;  CtrlEnv ctrlEnv; |  |

稍后在用到的时候逐一说明各变量的含义和用途。

* 线程环境变量清零

|  |
| --- |
| Dmai\_clear |

* 解析参数并设置环境变量

|  |
| --- |
| parseArgs(argc, argv, &args); |
| Optarg这个变量来自哪里？ |

* usage

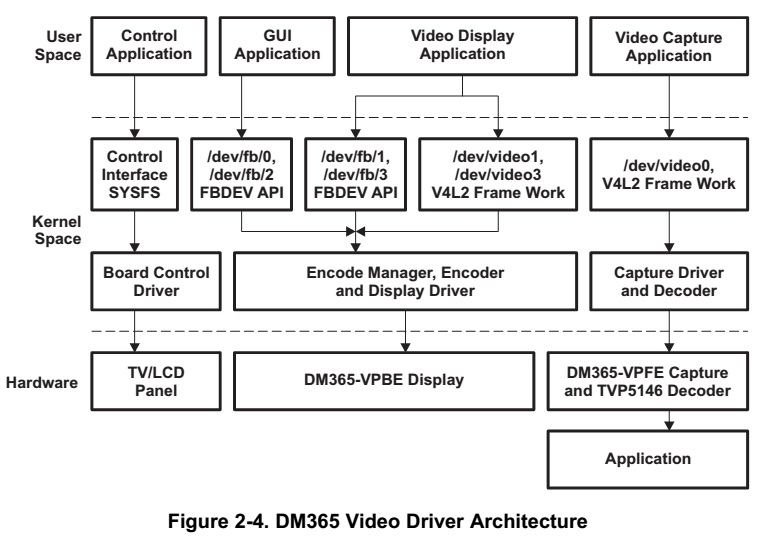
|  |
| --- |
| /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* usage \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  static Void usage(Void)  {  fprintf(stderr, "Usage: encodedecode [options]\n\n"  "Options:\n"  "-y | --display\_standard Video standard to use for display (see below)\n"  "-r | --resolution Captured video resolution ('width'x'height')\n"  " [video standard default]\n"  "-b | --bitrate Bit rate to encode video at [variable]\n"  "-v | --videocodec Video codec to use mpeg4/h264/mpeg2 [mpeg4]\n"  "-p | --passthrough Pass video through without encoding [off]\n"  "-I | --video\_input Video input source [video standard default]\n"  "-k | --keyboard Enable keyboard interface [off]\n"  "-t | --time Number of seconds to run the demo [infinite]\n"  "-o | --osd Show demo data on an OSD [off]\n"  "-h | --help Print this message\n\n"  "Video standards available:\n"  "\t1\tD1 @ 30 fps (NTSC)\n"  "\t2\tD1 @ 25 fps (PAL)\n"  "\t3\t720P @ 60 fps [Default]\n"  "\t7\t480P @ 60 fps\n"  "Video inputs available:\n"  "\t1\tComposite\n"  "\t2\tS-video\n"  "\t3\tComponent\n"  "\t4\tImager/Camera - for DM368\n");  } |

# Davinci Linux Driver Package

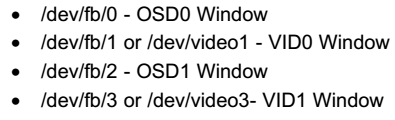
《**LSP 2.10 Davinci Linux driver**》

## videodriver

如图是DM368架构的Davinci视频驱动框架，包括视频显示、捕获、统计、处理部分。



Davinci VPBE的视频显示驱动是字符驱动，兼容frame buffer驱动与V4L2框架。该设备节点创建为FBDev VPBE驱动时：



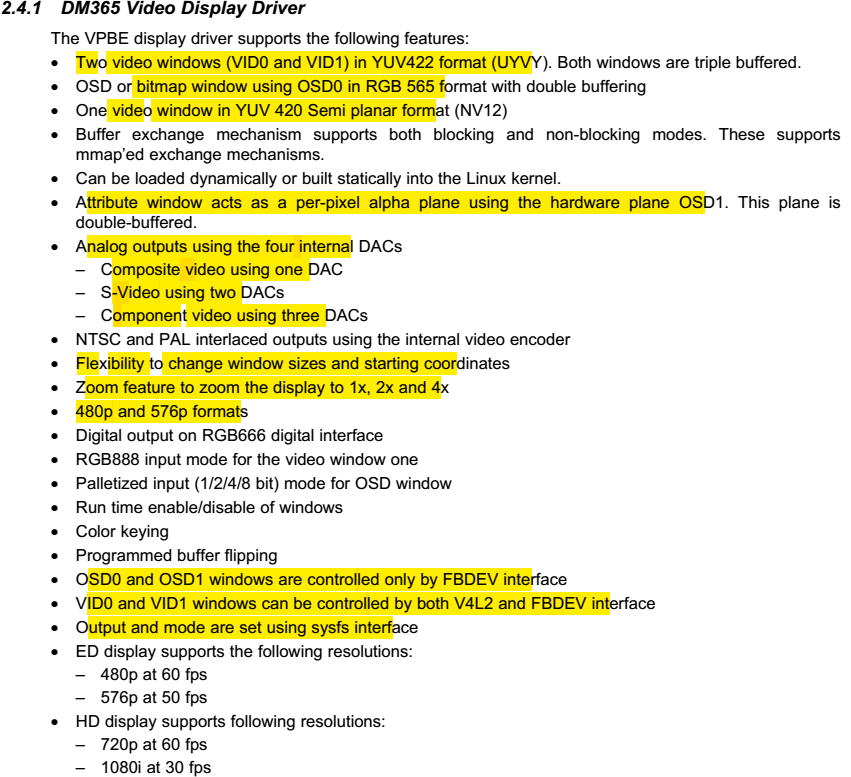
设备节点作为V4L2显示驱动创建时：



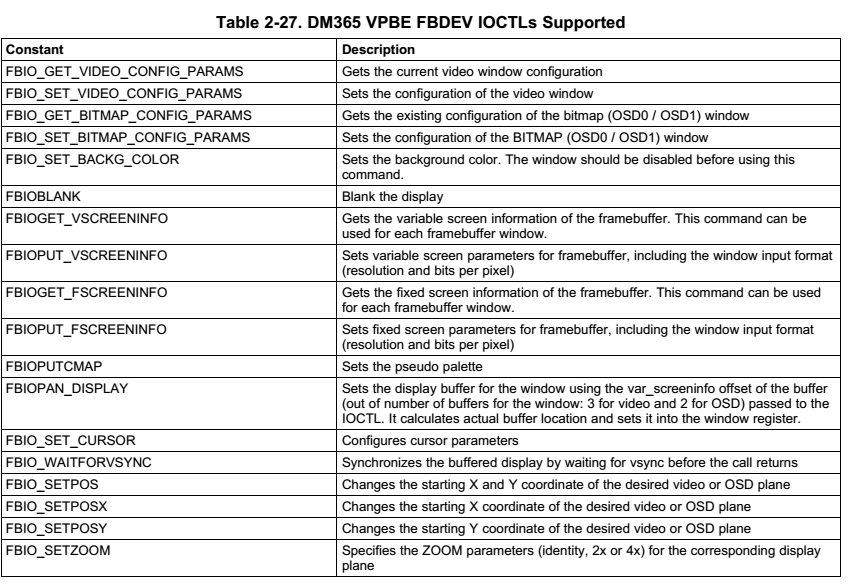
VPFE的视频捕获驱动是一个字符驱动，支持V4L2。VPFE驱动创建的设备节点是/dev/video0。

而previewer、resizer、H3A驱动都是捕获后处理的前端部分。

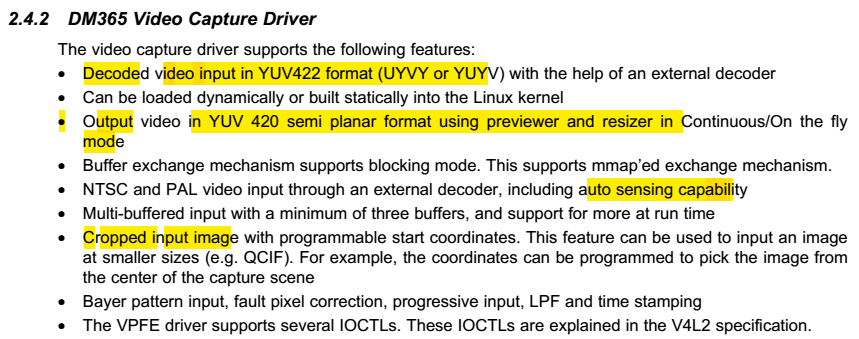
## display driver



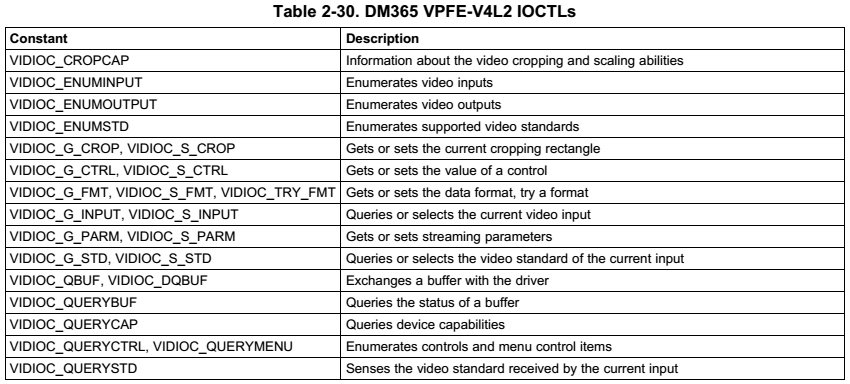
### supported IOCTL

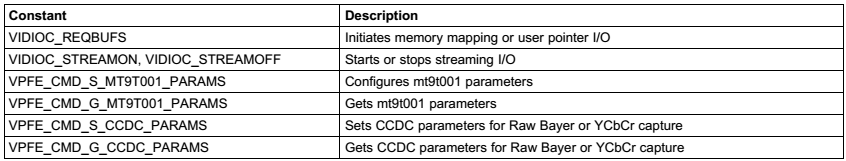


## capture driver



### supported IOCTL





# linux驱动知识

## Sysfs文件系统

           "sysfsis a ram-based filesystem initially based on ramfs. It provides ameans to export kernel data structures, their attributes, and thelinkages between them to userspace.”

            Linux2.6内核引入了sysfs文件系统。sysfs被看成是与proc同类别的文件系统。sysfs把连接在系统上的设备和总线组织成分级的文件,使其从用户空间可以访问到。

          Sysfs被加载在 /sys/目录下,它的子目录包括:

          1）Block:在系统中发现的每个块设备在该目录下对应一个子目录。每个子目录中  
又包含一些属性文件,它们描述了这个块设备的各方面属性,如:设备大小。(loop块设备是使用文件来模拟的)

          2）Bus:在内核中注册的每条总线在该目录下对应一个子目录,如: ide pci scsi usbpcmcia 其中每个总线目录内又包含两个子目录:devices和drivers ,devices目录包含了在整个系统中发现的属于该总线类型的设备,drivers目录包含了注册到该总线的所有驱动。

         3）Class:将设备按照功能进行的分类,如/sys/class/net目录下包含了所有网络接口。

         4）Devices:包含系统所有的设备。

         5）Kernel:内核中的配置参数

         6）Module:系统中所有模块的信息

         7）Firmware:系统中的固件

         8）Fs:描述系统中的文件系统

         9）Power:系统中电源选项

# dm368描述

Linux™ 数字视频软件开发套件 (DVSDK) 使 DaVinci™ 系统集成商能够快速开发基于 Linux 的多媒体应用，它们可以轻松植入达芬奇平台的不同器件中。每个 DVSDK 都包含一套预先测试的操作系统、应用程序框架和具有示例程序的编解码器库，这些程序演示了从外设流入和流出的实时音频和视频数据的解码和编码。针对具有 DSP 内核的达芬奇器件，DVSDK 提供了完整的框架，便于开发人员轻松利用 DSP 加速编解码器，而无需对 DSP 进行编程。DVSDK 完全免费，无需任何运行版税。

此版本 Linux DVSDK 4 版的正式 (GA) 版。支持的处理器：TMS320DM365/368、OMAP-L138 和 TMS320DM3730/25。

## 特性

DM36x Linux DVSDK 是一款免费的非商用版 Linux，附带了用户快速入门所需的所有内容，可能会随生产系统提供。快速入门指南可使您一会儿工夫即可连接硬件，并开始播放基于 GUI 的演示（演示硬件功能）。所有软件（包括源码）和开发工具均可在一小时内安装完毕。 包含的优秀文档用于所有软件组件。例如，驱动程序数据表提供了 Linux 器件驱动器支持片上外设性能信息的功能；多媒体编解码器数据表为特定编解码器列出了 MIPS 和存储器要求。

Linux SDK 包括以下组件：

平台支持包

Linux 内核 2.6.32.17

启动加载程序 (u-boot, UBL) 及其闪存实用程序

多媒体封装

多媒体框架产品 (MFP)

编解码器引擎框架

框架组件

Linux 实用程序 (CMEM)

XDAIS（eXpress DSP 算法可互操作性标准）

DaVinci 多媒体应用程序接口 (DMAI)

H/W 加速编解码器

编码器：H.264、MPEG-4、JPEG、AAC、G711、MPEG-2

解码器：H.264、MPEG-4、JPEG、AAC、G711、MPEG-2、VC1

硬件加速 GStreamer TI 插件

图形包

Qt/Webkit 应用程序框架

## DVSDK v4 的内容

引导加载 (u-boot) - 用于从各种外设中启动 Linux 的软件。

编解码器引擎多媒体堆栈 - 编解码器引擎算法执行框架提供了一组相同的多媒体编解码器 API，称为 xDM，无论编解码器是在 ARM、DSP 还是在专用加速器上运行的。开发人员随后可以基于先进的编解码器引擎构建更为高级的多媒体框架功能，例如 AV 同步或流解析功能。对于专业的 Linux 应用开发人员来说，编解码器引擎使他们能够在 DSP 上远程实例化并执行编解码器和算法，无需编写任何 DSP 代码。

达芬奇多媒体接口 (DMAI) - DMAI 可实现高便携性多媒体应用的开发，它们可以轻松从某个器件或操作系统移到其它器件或操作系统。

多媒体编解码器 - DVSDK 提供了编解码器库以及相关的 AV 剪辑，使开发人员能够立即开发和评估现实应用。这些编解码器可能位于专用硬件 (DM355S) 和/或加速 DSP 中。多数内容现在可与 DVSDK 一起下载。它们是具有生产软件许可协议的生产质量编解码器。

演示 ("Demo") 程序 —Demo 程序展示了解码和编码操作，说明如何将框架、编解码器、视频和音频子系统驱动器相集成以实现完整的应用。

 his release only supports [Ubuntu 10.04 LTS 32-bit](http://www.ubuntu.com/desktop/get-ubuntu/download) as your development host.

# DVSDK的help文档

|  |
| --- |
| ./codec-engine\_2\_26\_02\_11/docs/CodecEngine\_AlgorithmCreator\_UserGuide.pdf  ./codec-engine\_2\_26\_02\_11/docs/CodecEngine\_ServerIntegrator\_UserGuide.pdf  ./xdctools\_3\_16\_03\_36/packages/xdc/cdk/xdcpkg.pdf  ./xdctools\_3\_16\_03\_36/packages/xdc/cdk/xdc.pdf  ./xdctools\_3\_16\_03\_36/packages/xdc/cdk/spruex4.pdf  ./xdctools\_3\_16\_03\_36/packages/xdc/cdk/spru007h.pdf  ./xdctools\_3\_16\_03\_36/packages/xdc/cdk/spraa67.pdf  ./xdctools\_3\_16\_03\_36/packages/xdc/cdk/xs.pdf  ./xdctools\_3\_16\_03\_36/packages/xdc/cdk/langref/xdcSpec.pdf  ./xdctools\_3\_16\_03\_36/packages/xdc/cdk/langref/xdcScript.pdf  ./xdctools\_3\_16\_03\_36/packages/ti/targets/omf/elf/docs/spraab5.pdf  ./xdctools\_3\_16\_03\_36/packages/ti/targets/omf/elf/docs/dwarf-2.0.0-index.pdf  ./xdctools\_3\_16\_03\_36/packages/ti/targets/omf/elf/docs/Debugging\_using\_DWARF.pdf  ./xdctools\_3\_16\_03\_36/packages/ti/targets/omf/elf/docs/elf.pdf  ./xdctools\_3\_16\_03\_36/packages/ti/targets/omf/elf/docs/Errata\_SPRAAB5.pdf  ./xdctools\_3\_16\_03\_36/packages/ti/targets/omf/elf/docs/Dwarf3.pdf  ./xdctools\_3\_16\_03\_36/packages/ti/targets/omf/elf/docs/dwarf-2.0.0.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ittiam/codecs/aaclc\_enc/docs/IA-AAC-LC-Enc-ARM9E-DS.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg2enc/docs/mpeg2\_encoder\_DM365\_Datasheet.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg2enc/docs/mpeg2\_encoder\_dm365\_releasenotes.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg2enc/docs/mpeg2\_encoder\_dm365\_userguide.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg4dec/docs/MPEG4\_Decoder\_DM365\_UserGuide.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg4dec/docs/MPEG4\_Decoder\_DM365\_ReleaseNotes.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg4dec/docs/MPEG4\_Decoder\_DM365\_Datasheet.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/vc1dec/docs/VC1\_Decoder\_dm365\_ReleaseNotes.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/vc1dec/docs/VC1\_Decoder\_dm365\_Datasheet.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/vc1dec/docs/VC1\_Decoder\_dm365\_UserGuide.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/h264enc/docs/h264\_encoder\_dm365\_datasheet.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/h264enc/docs/h264\_encoder\_dm365\_userguide.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/h264enc/docs/h264\_encoder\_dm365\_releasenotes.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg4enc\_hdvicp/docs/mpeg4\_encoder\_dm365\_releasenotes.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg4enc\_hdvicp/docs/mpeg4\_encoder\_dm365\_userguide.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg4enc\_hdvicp/docs/mpeg4\_encoder\_dm365\_datasheet.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/jpegdec/docs/JPEG\_Decoder\_DM365\_UserGuide.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/jpegdec/docs/JPEG\_Decoder\_DM365\_Datasheet.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/jpegdec/docs/JPEG\_Decoder\_DM365\_ReleaseNotes.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg2dec/docs/mpeg2\_decoder\_dm365\_datasheet.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg2dec/docs/mpeg2\_decoder\_dm365\_release\_notes.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg2dec/docs/mpeg2\_decoder\_dm365\_user\_guide.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/h264dec/docs/h264\_decoder\_dm365\_releasenotes.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/h264dec/docs/h264\_decoder\_dm365\_datasheet.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/h264dec/docs/h264\_decoder\_dm365\_userguide.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg4dec\_hdvicp/docs/mpeg4\_decoder\_dm365\_releasenotes.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg4dec\_hdvicp/docs/mpeg4\_decoder\_dm365\_userguide.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg4dec\_hdvicp/docs/mpeg4\_decoder\_dm365\_datasheet.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/jpegenc/docs/JPEG\_Encoder\_DM365\_UserGuide.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/jpegenc/docs/JPEG\_Encoder\_DM365\_Datasheet.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/jpegenc/docs/JPEG\_Encoder\_DM365\_ReleaseNotes.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg4enc/docs/MPEG4\_Encoder\_DM365\_UserGuide.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg4enc/docs/MPEG4\_Encoder\_DM365\_Datasheet.pdf  ./codecs-dm365\_4\_02\_00\_00/packages/ti/sdo/codecs/mpeg4enc/docs/MPEG4\_Encoder\_DM365\_ReleaseNotes.pdf  ./psp/linux-2.6.32.17-psp03.01.01.39/Documentation/DocBook/v4l/vbi\_625.pdf  ./psp/linux-2.6.32.17-psp03.01.01.39/Documentation/DocBook/v4l/fieldseq\_bt.pdf  ./psp/linux-2.6.32.17-psp03.01.01.39/Documentation/DocBook/v4l/crop.pdf  ./psp/linux-2.6.32.17-psp03.01.01.39/Documentation/DocBook/v4l/vbi\_525.pdf  ./psp/linux-2.6.32.17-psp03.01.01.39/Documentation/DocBook/v4l/vbi\_hsync.pdf  ./psp/linux-2.6.32.17-psp03.01.01.39/Documentation/DocBook/v4l/fieldseq\_tb.pdf  ./psp/linux-2.6.32.17-psp03.01.01.39/Documentation/DocBook/dvb/dvbstb.pdf  ./psp/docs/Davinci\_PSP\_03\_01\_Linux\_Installation\_User\_Guide.pdf  ./psp/docs/Davinci\_PSP\_03\_01\_GA\_r39\_Release\_Notes.pdf  ./dvtb\_4\_20\_18/packages/ti/sdo/dvtb/docs/dvtb\_release\_notes.pdf  ./dvtb\_4\_20\_18/packages/ti/sdo/dvtb/docs/dvtb\_user\_guide.pdf  ./xdais\_6\_26\_01\_03/packages/ti/xdais/dm/docs/XDM\_UsersGuide.pdf  ./xdais\_6\_26\_01\_03/packages/ti/xdais/dm/docs/xdm1\_differences.pdf  ./framework-components\_2\_26\_00\_01/docs/FrameworkComponents\_IRES\_RMAN\_AppNote.pdf  ./framework-components\_2\_26\_00\_01/docs/FrameworkComponents\_DSKT2\_UserGuide.pdf  ./framework-components\_2\_26\_00\_01/docs/FrameworkComponents\_UsingDMA\_AppNote.pdf  ./docs/TMS320DM368\_EVM\_Quick\_Start\_Guide.pdf  ./docs/TMS320DM368\_Software\_Developers\_Guide.pdf  ./docs/dvsdk\_4\_02\_00\_06\_dm368\_Release\_Notes.pdf |

# Linux常用命令

## find

find . -name getopt\_long.c -print

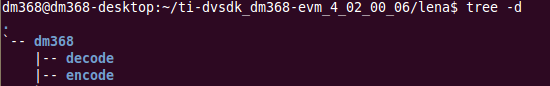
## grep

grep -r "getopt\_long" \*

grep -r "Capture\_Input\_COUNT" \*

# LENA代码结构

## 顶层代码目录



## make修改

|  |
| --- |
| encode/Makefile  decode/Makefile |

* 删除非DM368的选项
* 替换DM365为DM368

下面这个不能替换，否则出错：

|  |
| --- |
| # Platform (board) to build for  XDC\_PLATFORM = ti.platforms.evmDM365 |
| js: "/home/dm368/ti-dvsdk\_dm368-evm\_4\_02\_00\_06/xdctools\_3\_16\_03\_36/packages/xdc/tools/Cmdr.xs", line 51: Error: xdc.tools.configuro: Error: Can't find platform package ti.platforms.evmDM368  make[2]: \*\*\* [encode\_config/compiler.opt] 错误 1  make[1]: \*\*\* [encode] 错误 2  make[1]:正在离开目录 `/home/dm368/ti-dvsdk\_dm368-evm\_4\_02\_00\_06/lena/dm368'  make: \*\*\* [dm368] 错误 2 |

如果保持不变，则MPEG/H.264的库有一个地方命名注意：

|  |
| --- |
| will link with ti.sdo.dmai:lib/dmai\_linux\_dm365.a470MV  will link with ti.sdo.codecs.mpeg2enc:lib/mpeg2venc\_ti\_arm926.a;lib/dma\_ti\_dm365.a  will link with ti.sdo.codecs.h264enc:lib/h264venc\_ti\_arm926.a;lib/h264v\_ti\_dma\_dm365.a |

其中，类似定义在以下位置，说明DM368使用的是DM365的EVMDM365，而没有单独的EVMDM368。

|  |
| --- |
| dmai\_2\_20\_00\_15/Platform.make |
| #------------------------------------------------------------------------------  # DM368 ARM Linux variables.  #------------------------------------------------------------------------------  PLATFORM\_OS\_dm368\_al = linux  PLATFORM\_DEVICE\_dm368\_al = dm368  PLATFORM\_PERIPHERALS\_dm368\_al = dm365  PLATFORM\_XDC\_TARGET\_dm368\_al = gnu.targets.arm.GCArmv5T  PLATFORM\_XDC\_dm368\_al = ti.platforms.evmDM365 |