

TI DVSDK 4.02 Patches for LeopardBoard 368 Installation Guide

Introduction

This guide describes the steps required to install the Texas Instruments TMS320DM368 DVSDK 4.02 onto a Leopard Imaging LeopardBoard 368 for operation with either the LI-5M03 Camera Board with video/line-in audio inputs or the LI-HDI365 Component Video Input Board with component video/line-in audio source inputs—Figure 1 shows the development context for the Leopard Imaging LeopardBoard 368.

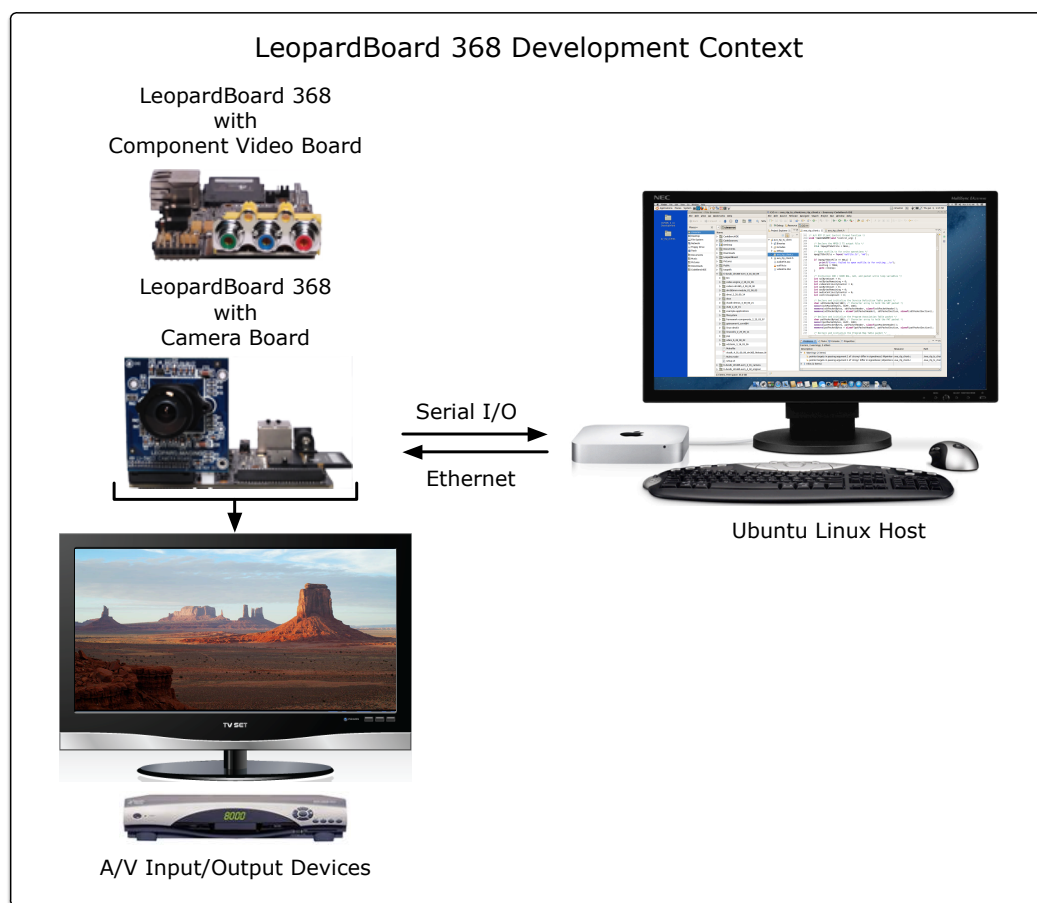


Figure 1: LeopardBoard 368 Development Context.

Features

- Cimarron Systems Digital Media SDK (DMSDK) for the Texas Instruments TMS320DM368

DaVinci Media Processor running DVSDK 4.02 on the Leopard Imaging LeopardBoard 368

- DMSDK A/V RTP Streaming Server Application:
RTP/RTCP server that encodes/streams H.264 video at 1920 x 1080i 30 fps / 1280 x 720p 30 fps and AAC-LC at 32 ksps (*available only in the professional version*).
- DMSDK A/V RTP Streaming Client Application:
RTP/RTCP client that receives, reassembles and writes-to-file H.264 video and AAC-LC audio while it simultaneously encodes reassembled video and audio frames into a MPEG-2 Transport Stream (*available only in the professional version*).
- Implements an Automatic Exposure/White Balance Control Loop (*available only in the professional version*).
- RTP A/V Streaming Client/Server Input/Output via Ethernet (*available only in the professional version*).
- Input from the LI-5M03 Camera Board or LI-HDI365 Component Video Board.
- Texas Instruments TMS320DM365 DVEVM DVSDK 4.02
 - Arago Community Linux Kernel with build system;
 - Platform Support Package (PSP) including U-Boot bootloader, bootloader, memory manager, and interrupt controller; and
 - Codec Engine with fully parameterized H.264, MPEG-4, MPEG-2, AAC-LC, G.711, and JPEG codecs.
 - Texas Instruments Digital Media Application Interface (DMAI) video, audio, and speech encode and decode example applications.
 - Texas Instruments Digital Video Test Bench (DVTB) encode and decode codec test/validation applications.
 - Texas Instruments Qt Embedded GUI Interface Builder and example applications.
 - Texas Instruments GStreamer with DaVinci Plugin and example pipelines.

Before You Begin

It is assumed that you have familiarity with the LeopardBoard 368 as well as the Texas Instruments DVSDK 4.xx development environment. If this is not the case, stop now—then follow the instructions in the Texas Instruments *TMS320DM368 Software Developers Guide* (found in the DVSDK top-level docs directory to ensure that:

- the LeopardBoard 368 can boot from an unmodified version of the DVSDK Linux Kernel;
- and the `make clean`, `make all`, and `make install` operations each run without errors; and
- before beginning the installation process, please be sure to back up all of your development work files—it is also recommended that you begin with a fresh install of DVSDK 4.02.



Notational Conventions

Within this guide, the following notational conventions are used.

Command Line Conventions:

Within this guide, LeopardBoard 368 terminal window entries are indicated as follows:

`root@dm368-evm:~# [command line]`

Within this guide, Ubuntu Linux Host terminal window entries are indicated as follows:

`user@ubuntu:~$ [command line]`

Figure 2 shows the TMS320DM368 DVSDK 4.02 file directory structure highlights including:

- the top-level directory: `~/ti-dvSDK_dm368-evm_4_02_00_06`
- the DMAI components/applications directory: `../dmai_2_20_00_15`
- the demonstration applications directory: `../dvSDK-demos_4_02_00_01`

LeopardBoard 368 DVSDK 4.02 Installation Guide

- the TMS320DM368 DVSDK 4.02 Platform Support Package directory: ../psp

▼ LeopardBoard_368_Public	1 item	folder
▼ ti-dvSDK_dm368-evm_4_02_00_06	26 items	folder
▶ bin	7 items	folder
▶ codec-engine_2_26_02_11	4 items	folder
▶ codecs-dm365_4_02_00_00	1 item	folder
▶ dm365mm-module_01_00_03	4 items	folder
▶ dmai_2_20_00_15	9 items	folder
▶ docs	1 item	folder
▶ dvSDK-demos_4_02_00_01	4 items	folder
▶ dtvb_4_20_18	5 items	folder
▶ etc	1 item	folder
▶ example-applications	2 items	folder
▶ filesystem	1 item	folder
▶ framework-components_2_26_00_01	3 items	folder
▶ gstreamer-ti_svn919	14 items	folder
▶ linux-devkit	9 items	folder
▶ linuxutils_2_26_01_02	9 items	folder
▶ psp	1 item	folder
▶ usr	1 item	folder
▶ xdais_6_26_01_03	16 items	folder
▶ xdctools_3_16_03_36	6 items	folder
ti-dvSDK_dm368-evm_4_02_00_06_dmai_2_20_00_15.patch	83.8 KB	differences between files
ti-dvSDK_dm368-evm_4_02_00_06_leopardboard.patch	123.3 KB	differences between files
ti-dvSDK_dm368-evm_4_02_00_06_Makefile.patch	679 bytes	differences between files
dvSDK_dm368-evm_4_02_00_06_setuplinux	670.1 MB	executable
Makefile	23.2 KB	Makefile
Rules.make	1.9 KB	plain text document
setup.sh	833 bytes	shell script

Figure 2: Texas Instruments TMS320DM368 DVSDK 4.02 File Directory Structure.

Patching then Rebuilding DVSDK 4.02 Components

Using Steps 1a through 1m, download then update the Texas Instruments DVSDK 4.02 (for DVSDK directory details, please see Figure 2).

Step 1a: If you have not already done so, from the Mentor Graphics website, download then install Sourcery G++ Lite 2009q3 in a directory of your choice.

Step 1b: From the Texas Instruments website, download then install the TMS320DM368 DVSDK 4.02 into the ~LeopardBoard_368_Public/ti-dvSDK_dm368-evm_4_02_00_06 directory (if necessary, create the directory).

Step 1c: Connect to the newly installed DVSDK 4.02 directory, i.e., perform execute user@ubuntu:~\$ cd LeopardBoard_368_Public/ti-dvSDK_dm368-evm_4_02_00_06 then, from the Cimarron Systems website support page, download the three LeopardBoard 368 patches then move them into the top level of the directory.

Step 1d: From the Cimarron Systems support webpage, i.e., perform execute user@ubuntu:~\$ cd

LeopardBoard 368 DVSDK 4.02 Installation Guide

LeopardBoard_368_Public/ti-dvSDK_dm368-evm_4_02_00_06 then, from the LeopardBoard 368 Patches folder, move the three patches into the top level of the directory.

Step 1e: Next, apply these three patches as follows:

```
$ patch -p1 < ti-dvSDK_dm368-evm_4_02_00_06_leopardboard.patch
$ patch -p1 < ti-dvSDK_dm368-evm_4_02_00_06_dmai_2_20_00_15.patch
$ patch -p1 < ti-dvSDK_dm368-evm_4_02_00_06_Makefile.patch
```

Step 1f: From the same directory, execute a `make clean` then execute a `make all`—the `make` system will then build the DVSDK 4.02 components—please verify that everything builds with no errors.

Step 1g: From the same directory, execute the following:

```
$ cd psp/prebuilt-images/
$ rm -rf ./*
$ ln -s ../u-boot-2010.12-rc2-bsp03.01.01.39/u-boot.bin u-boot-dm368-evm.bin
$ ln -s ../linux-2.6.32.17-bsp03.01.01.39/arch/arm/boot/uImage uImage-dm368-evm.bin
$ cd ../..
```

Step 1h: Connect a SD Card reader/writer to the Ubuntu Linux Host—check the device node using `dmesg` (in the following command, change `/dev/sdb` to your device node if required)—then, from the DVSDK 4.02 install directory, execute:

```
$ sudo bin/mksdboot.sh --device /dev/sdb --sdk ${PWD}
```

Note: writing the SD Card will take a number of minutes (leaving the partitions unmounted).


Step 1i: In `Rules.make`, comment out `EXEC_DIR=$(HOME)/install/$(PLATFORM)` then add `EXEC_DIR=/media/ROOTFS`.

Step 1j: Reinsert the SD Card, then execute: `sudo make install`

Step 1k: Connect to the SD Card BOOT partition, i.e., execute `$ cd /media/BOOT`

Next, edit the `boot.cmd` file so that it becomes:

```
mmc rescan 0
setenv bootargs 'console=ttyS0, 115200n8 root=/dev/mmcblk0p2 rw ip=192.168.2.127 mem=60M
davinci_enc_mgr.ch0_output=COMPONENT davinci_enc_mgr.ch0_mode=720P-60
davinci_display.cont2_bufsize=6291456 vpfe_capture.cont_bufoffset=6291456
vpfe_capture.cont_bufsize=6291456
video=davincifb:vid0=1920*1080*16,16K:vid1=1920*1080*16,4050K:osd0=1280*720*16,4050K:
osd1=1280*720*4,1215K dm365_imp.oper_mode=0 davinci_capture.device_type=4
vpfe_capture.interface=1 rootwait'
fatload mmc 0 80700000 uImage
bootm 80700000
```

Caution: please ensure that the values contained within the `setenv bootargs '...'` command are assigned to a single line (i.e., each line must not have an intervening CR-LF). Also, you may need to adjust the `boot` args above to fit your system configuration. 

Step 1l: Still connected to the BOOT partition, make `boot.scr` by executing:

```
$ sudo mkimage -A arm -O linux -T script -C none -a 0 -e 0 -n 'Execute Boot Script' -d boot.cmd
boot.scr
```

Next, disconnect from the BOOT partition then eject the SD Card.

Step 1m: Ensure that the LeopardBoard 368 DIP switches are set so that the board boots from the SD Card, i.e., to 1=OFF 2=ON 3=OFF, insert the SD Card, connect the Ethernet, RS-232 Serial, Component Video Output, and 5VDC Power cables, and then power-up the board. After the Linux Kernel boots up (for boot-logs, see Appendix A and B), log in using the password: `root`.

Operation with the LI-5M03 Camera Board

Next, in order to verify that the LI-5M03 Camera Board functions with the LeopardBoard 368, execute:

```
root@dm368-evm:~# cd /usr/share/ti/dvSDK-demos
root@dm368-evm:/usr/share/ti/dvSDK-demos# /etc/init.d/loadmodule-rc restart
./encode -v videoFile.264 -a audioFile.aac -I4 -y5
```

After executing the command lines above, the LeopardBoard 368 terminal window output should look something like the window below (camera video should be displayed on your video monitor):

LeopardBoard 368 Terminal Window

```
root@dm368-evm:/usr/share/ti/dvSDK-demos# ./encode -v videoFile.264 -a audioFile.aac -I4 -y5
Encode demo started.
davinci_resizer davinci_resizer.2: RSZ_G_CONFIG:0:1:124
vpfe-capture vpfe-capture: IPIPE Chained
vpfe-capture vpfe-capture: Resizer present
LeopardBoard: switch to HD imager-MT9P031 video input...
-----Exposure time = 456
-----Exposure time = 456
vpfe-capture vpfe-capture: width = 1920, height = 1080, bpp = 1
vpfe-capture vpfe-capture: adjusted width = 1920, height = 1080, bpp = 1, bytesperline = 1920,
sizeimage = 3110400
vpfe-capture vpfe-capture: width = 1920, height = 1080, bpp = 1
vpfe-capture vpfe-capture: adjusted width = 1920, height = 1080, bpp = 1, bytesperline = 1920,
sizeimage = 3110400
davinci_v4l2 davinci_v4l2.1: Before finishing with S_FMT:
layer.pix_fmt.bytesperline = 1920,
layer.pix_fmt.width = 1920,
layer.pix_fmt.height = 1080,
layer.pix_fmt.sizeimage = 3110400
davinci_v4l2 davinci_v4l2.1: pixfmt->width = 1920,
layer->layer_info.config.line_length= 1920
ARM Load: 65% Video fps: 30 fps Video bit rate: 369 kbps Sound bit rate: 53 kbps Time: 00:00:01 Demo:
Encode Display: 1080I 30Hz Video Codec: H.264 HP Resolution: 1920x1080 ...

ARM Load: 41% Video fps: 30 fps Video bit rate: 925 kbps Sound bit rate: 120 kbps Time: 00:00:02 Demo:
Encode Display: 1080I 30Hz Video Codec: H.264 HP Resolution: 1920x1080 ...

ARM Load: 43% Video fps: 30 fps Video bit rate: 1139 kbps Sound bit rate: 103 kbps Time: 00:00:03 Demo:
Encode Display: 1080I 30Hz Video Codec: H.264 HP Resolution: 1920x1080 ...
...
```

Operation with the LI-HDI365 Component Video Input Board

In order to operate with the LI-HDI365 Component Video Input Board, execute Steps 1j through 1m, however, in the boot.cmd file, remove the vpfe_capture.interface=1 boot arg. Verify that the LI-HDI365 Component Video Input Board functions with the LeopardBoard 368 by doing:

```
root@dm368-evm:/usr/share/ti/dvSDK-demos# /etc/init.d/loadmodule-rc restart
./encode -v videoFile.264 -a audioFile.aac -I3 -y3
```

After executing the command lines above, the LeopardBoard 368 terminal window output should look something like the window generated during operation of the LI-5M03 Camera Board (component video should be displayed on your video monitor).

Running the DMAI Demonstration Applications

Finally, in order to run the Texas Instruments the other DMAI demonstration applications, for example to

LeopardBoard 368 DVSDK 4.02 Installation Guide

decode the video and audio encoded above, execute the following:

```
root@dm368-evm:/usr/share/ti/dvSDK-demos# /etc/init.d/loadmodule-rc restart
./decode -v videoFile.264 -a audioFile.aac -y5
```

The `decode` application will decode the H.264 `videoFile.264` and AAC `audioFile.aac` files then display/output them in 1920 x 1080 resolution video with line-level monophonic audio (please see the `decode.txt` file for other options).

In order to run the `encodeddecode` application, execute the following:

```
root@dm368-evm:/usr/share/ti/dvSDK-demos# /etc/init.d/loadmodule-rc restart
./encodeddecode h264 -I4 -y3
```

In the example above, the `encodeddecode` application will encode, decode, and then output LI-5M03 Camera Board input video on the component video output.

For additional information concerning LeopardBoard 368 hardware components, please see the *LeopardBoard 368 Hardware Guide*. In order to understand DVSDK 4.02 software components, please see the *TMS320DM368 Software Developers Guide* as well as the [LeopardBoard 368 website](#).

Acknowledgements

With respect to the content of the Texas Instruments DVSDK 4.02 including the current patches for the LeopardBoard 368, Cimarron Systems would like to acknowledge the contribution of the developers at [Texas Instruments, Inc.](#); [Ridge Run, LLC](#); and Steve at [SteveGIGJoe Blog](#).

For information regarding Cimarron Systems Digital Media SDKs for Texas Instruments TMS320DM36x Digital Media Processors, please visit us at <http://www.cimarronsystems.com/>.

Contact Information:

Cimarron Systems, LLC
Evergreen, Colorado
telephone: +1 (303) 674-9207
email: info@cimarronsystems.com
website: www.cimarronsystems.com

Please send feedback regarding this document to: support@cimarronsystems.com.

Revision History:

Date	Version	Notes
10/30/2013	version 1.0	Initial version.
1/3/2014	version 1.1	Minor updates.
4/19/2017	version 1.2	Minor updates.

Appendix A

Linux Kernel BootLog with the Camera Board

```
DM36x initialization passed!
TI UBL Version: 1.50
Booting Catalog Boot Loader
BootMode = SD/MMC
Starting SDMMC Copy...
  DONE
Jumping to entry point at 0x81080000.

U-Boot 2010.12-rc2 (Oct 31 2013 - 15:33:35)

Cores: ARM 432 MHz
DDR:   340 MHz
I2C:   ready
DRAM:  128 MiB
NAND:  256 MiB
MMC:   davinci: 0, davinci: 1
Bad block table not found for chip 0
Bad block table not found for chip 0
nand_bbt: Error while writing bad block table -5
NAND read from offset 3c0000 failed -74
*** Warning - readenv() failed, using default environment

Net:   Ethernet PHY: GENERIC @ 0x00
DaVinci-EMAC
Hit any key to stop autoboot:  0
reading boot.scr

583 bytes read
## Executing script at 80600000
reading uImage

2146504 bytes read
## Booting kernel from Legacy Image at 80700000 ...
   Image Name:   Linux-2.6.32.17-davinci1
   Created:      2013-10-31 21:25:59 UTC
   Image Type:   ARM Linux Kernel Image (uncompressed)
   Data Size:    2146440 Bytes = 2 MiB
   Load Address: 80008000
   Entry Point:  80008000
   Verifying Checksum ... OK
   Loading Kernel Image ... OK
OK

Starting kernel ...

Uncompressing Linux..... done, booting the kernel.
Linux version 2.6.32.17-davinci1 (cimarron@ubuntu) (gcc version 4.3.3 (Sourcery G++ Lite
2009q1-203) ) #1 PREEMPT Thu Oct 31 15:25:56 MDT 2013
CPU: ARM926EJ-S [41069265] revision 5 (ARMv5TEJ), cr=00053177
CPU: VIVT data cache, VIVT instruction cache
Machine: DM365 Leopard
Memory policy: ECC disabled, Data cache writeback
DaVinci dm36x_rev1.2 variant 0x8
Built 1 zonelists in Zone order, mobility grouping off. Total pages: 15240
Kernel command line: console=ttyS0,115200n8 root=/dev/mmcblk0p2 rw ip=192.168.2.127 mem=60M
davinci_enc_mgr.ch0_output=COMPONENT davinci_enc_mgr.ch0_mode=720P-60
davinci_display.cont2_bufsize=6291456 vpfe_capture.cont_bufoffset=6291456
vpfe_capture.cont_bufsize=6291456
video=davincifb:vid0=1920*1080*16,16K:vid1=1920*1080*16,4050K:osd0=1280*720*16,4050K:osd1=1280*72
0*4,1215K dm365_imp.oper_mode=0 davinci_capture.device_type=4 vpfe_capture.interface=1 rootwait
PID hash table entries: 256 (order: -2, 1024 bytes)
Dentry cache hash table entries: 8192 (order: 3, 32768 bytes)
Inode-cache hash table entries: 4096 (order: 2, 16384 bytes)
Memory: 60MB = 60MB total
```

LeopardBoard 368 DVSDK 4.02 Installation Guide

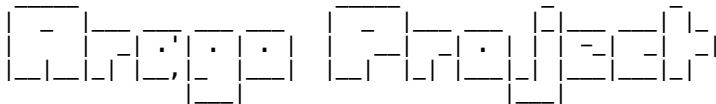
```
Memory: 56220KB available (4020K code, 392K data, 144K init, 0K highmem)
SLUB: Genslabs=11, HWalign=32, Order=0-3, MinObjects=0, CPUs=1, Nodes=1
Hierarchical RCU implementation.
NR_IRQS:245
Console: colour dummy device 80x30
Calibrating delay loop... 215.44 BogoMIPS (lpj=1077248)
Mount-cache hash table entries: 512
CPU: Testing write buffer coherency: ok
DaVinci: 8 gpio irqs
NET: Registered protocol family 16
davinci_serial_init:97: failed to get UART2 clock
bio: create slab <bio-0> at 0
DM365 IPIPE initialized in Continuous mode
SCSI subsystem initialized
usbcore: registered new interface driver usbfs
usbcore: registered new interface driver hub
usbcore: registered new device driver usb
vpss vpss: dm365_vpss vpss probed
vpss vpss: dm365_vpss vpss probe success
dm365_afew_hw_init
ch0 default output "COMPONENT", mode "720P-60"
VPBE Encoder Initialized
cfg80211: Using static regulatory domain info
cfg80211: Regulatory domain: US
        (start_freq - end_freq @ bandwidth), (max_antenna_gain, max_eirp)
        (2402000 KHz - 2472000 KHz @ 40000 KHz), (600 mBi, 2700 mBm)
        (5170000 KHz - 5190000 KHz @ 40000 KHz), (600 mBi, 2300 mBm)
        (5190000 KHz - 5210000 KHz @ 40000 KHz), (600 mBi, 2300 mBm)
        (5210000 KHz - 5230000 KHz @ 40000 KHz), (600 mBi, 2300 mBm)
        (5230000 KHz - 5330000 KHz @ 40000 KHz), (600 mBi, 2300 mBm)
        (5735000 KHz - 5835000 KHz @ 40000 KHz), (600 mBi, 3000 mBm)
cfg80211: Calling CRDA for country: US
LogicPD encoder initialized
Switching to clocksource timer0_1
musb_hdrc: version 6.0, cppi-dma, host, debug=0
musb_hdrc: USB Host mode controller at fec64000 using DMA, IRQ 12
musb_hdrc musb_hdrc: MUSB HDRC host driver
musb_hdrc musb_hdrc: new USB bus registered, assigned bus number 1
usb usb1: configuration #1 chosen from 1 choice
hub 1-0:1.0: USB hub found
hub 1-0:1.0: 1 port detected
NET: Registered protocol family 2
IP route cache hash table entries: 1024 (order: 0, 4096 bytes)
TCP established hash table entries: 2048 (order: 2, 16384 bytes)
TCP bind hash table entries: 2048 (order: 1, 8192 bytes)
TCP: Hash tables configured (established 2048 bind 2048)
TCP reno registered
NET: Registered protocol family 1
RPC: Registered udp transport module.
RPC: Registered tcp transport module.
RPC: Registered tcp NFSv4.1 backchannel transport module.
JFFS2 version 2.2. (NAND) © 2001-2006 Red Hat, Inc.
msgmni has been set to 109
alg: No test for stdrng (krng)
io scheduler noop registered
io scheduler anticipatory registered (default)
Console: switching to colour frame buffer device 160x45
davincifb davincifb.0: dm_osd0_fb: 1280x720x16@0,0 with framebuffer size 4050KB
davincifb davincifb.0: dm_vid0_fb: 1920x0x16@0,0 with framebuffer size 16KB
davincifb davincifb.0: dm_osd1_fb: 1280x720x4@0,0 with framebuffer size 1215KB
davincifb davincifb.0: dm_vid1_fb: 1920x0x16@0,0 with framebuffer size 4050KB
DM365 IPIPEIF probed
imp serializer initialized
davinci_previewer initialized
davinci_resizer initialized
Serial: 8250/16550 driver, 2 ports, IRQ sharing disabled
serial8250.0: ttyS0 at MMIO 0x1c20000 (irq = 40) is a 16550A
console [ttyS0] enabled
serial8250.0: ttyS1 at MMIO 0x1d06000 (irq = 41) is a 16550A
brd: module loaded
console [netcon0] enabled
netconsole: network logging started
```


LeopardBoard 368 DVSDK 4.02 Installation Guide

```
Initializing USB Mass Storage driver...
usbcore: registered new interface driver usb-storage
USB Mass Storage support registered.
usbcore: registered new interface driver usbtest
mice: PS/2 mouse device common for all mice
i2c /dev entries driver
Linux video capture interface: v2.00
ths7303 1-002c: chip found @ 0x58 (DaVinci I2C adapter)
vpfe_init
vpfe-capture: vpss clock vpss_master enabled
vpfe-capture vpfe-capture: v4l2 device registered
vpfe-capture vpfe-capture: video device registered
LeopardBoard: switch to HD imager-MT9P031 video input...
mt9p031 1-0048: Detected a MT9P031 chip ID 1801
mt9p031 1-0048: mt9p031 1-0048 decoder driver registered !!
vpfe-capture vpfe-capture: v4l2 sub device mt9p031 registered
vpfe_register_ccdc_device: DM365 ISIF
DM365 ISIF is registered with vpfe.
af major#: 252, minor# 0
AF Driver initialized
aew major#: 251, minor# 0
AEW Driver initialized
Trying to register davinci display video device.
layer=c20ef000,layer->video_dev=c20ef170
Trying to register davinci display video device.
layer=c20ef400,layer->video_dev=c20ef570
davinci_init:DaVinci V4L2 Display Driver V1.0 loaded
watchdog watchdog: heartbeat 60 sec
davinci_mmc davinci_mmc.0: Using DMA, 4-bit mode
usbcore: registered new interface driver usbhid
usbhid: v2.6:USB HID core driver
Advanced Linux Sound Architecture Driver Version 1.0.21.
No device for DAI tlv320aic3x
No device for DAI davinci-i2s
asoc: tlv320aic3x <=> davinci-i2s mapping ok
ALSA device list:
 #0: DaVinci EVM (tlv320aic3x)
TCP cubic registered
NET: Registered protocol family 17
lib80211: common routines for IEEE802.11 drivers
Clocks: disable unused mmcsd1
Clocks: disable unused spi0
Clocks: disable unused spi1
Clocks: disable unused spi2
Clocks: disable unused spi3
Clocks: disable unused spi4
Clocks: disable unused aemif
Clocks: disable unused pwm0
Clocks: disable unused pwm1
Clocks: disable unused pwm2
Clocks: disable unused pwm3
Clocks: disable unused timer1
Clocks: disable unused timer3
Clocks: disable unused emac
Clocks: disable unused voice_codec
Clocks: disable unused rto
Clocks: disable unused mjcp
davinci_emac_probe: using random MAC addr: f6:ce:68:1c:ad:b0
emac-mii: probed
Waiting for root device /dev/mmcblk0p2...
mmc0: host does not support reading read-only switch. assuming write-enable.
mmc0: new high speed SDHC card at address 1234
mmcblk0: mmc0:1234 SA04G 3.63 GiB
 mmcblk0: p1 p2
IP-Config: Guessing netmask 255.255.255.0
IP-Config: Complete:
    device=eth0, addr=192.168.2.127, mask=255.255.255.0, gw=255.255.255.255,
    host=192.168.2.127, domain=, nis-domain=(none),
    bootserver=255.255.255.255, rootserver=255.255.255.255, rootpath=
kjournald starting. Commit interval 5 seconds
EXT3 FS on mmcblk0p2, internal journal
EXT3-fs: mounted filesystem with writeback data mode.
```

LeopardBoard 368 DVSDK 4.02 Installation Guide

```
VFS: Mounted root (ext3 filesystem) on device 179:2.
Freeing init memory: 144K
INIT: version 2.86 booting
Please wait: booting...
Starting udev
udev: starting version 141
Remounting root file system...
Caching udev devnodes
Populating dev cachemv: cannot rename '/tmp/devices': No such file or directory
NET: Registered protocol family 10
ALSA: Restoring mixer settings...
Configuring network interfaces... done.
Setting up IP spoofing protection: rp_filter.
hwclock: can't open '/dev/misc/rtc': No such file or directory
Thu Mar 24 19:24:00 UTC 2011
hwclock: can't open '/dev/misc/rtc': No such file or directory
INIT: Entering runlevel: 5
Starting system message bus: dbus.
Starting telnet daemon.
Starting syslogd/klogd: done
Starting thttpd.
CMEMK module: built on Oct 31 2013 at 15:27:05
  Reference Linux version 2.6.32
  File /home/cimarron/LeopardBoard_368_Public/ti-dvSDK_dm368-evm_4_02_00_06/
linuxutils_2_26_01_02/packages/ti/sdo/linuxutils/cmем/src/module/cmемk.c
allocated heap buffer 0xc6000000 of size 0x4400000
heap fallback enabled - will try heap if pool buffer is not available
CMEM Range Overlaps Kernel Physical - allowing overlap
CMEM phys_start (0x1000) overlaps kernel (0x80000000 -> 0x83c00000)
cmемk initialized
IRQK module: built on Oct 31 2013 at 15:27:06
  Reference Linux version 2.6.32
  File /home/cimarron/LeopardBoard_368_Public/ti-dvSDK_dm368-evm_4_02_00_06/
linuxutils_2_26_01_02/packages/ti/sdo/linuxutils/irq/src/module/irqk.c
irqk initialized
EDMAK module: built on Oct 31 2013 at 15:27:07
  Reference Linux version 2.6.32
  File /home/cimarron/LeopardBoard_368_Public/ti-dvSDK_dm368-evm_4_02_00_06/
linuxutils_2_26_01_02/packages/ti/sdo/linuxutils/edma/src/module/edmak.c
Calibrating touchscreen (first time only)ts_open: No such file or directory
.
Starting Matrix GUI application.
```



Arago Project <http://arago-project.org> dm368-evm ttyS0

Arago 2011.02 dm368-evm ttyS0

dm368-evm login: root
root@dm368-evm:~#

[Page Intentionally Left Blank]

Appendix B

Linux Kernel BootLog with the Component Video Input

```
DM36x initialization passed!
TI UBL Version: 1.50
Booting Catalog Boot Loader
BootMode = SD/MMC
Starting SDMMC Copy...
  DONE
Jumping to entry point at 0x81080000.

U-Boot 2010.12-rc2 (Oct 31 2013 - 15:33:35)

Cores: ARM 432 MHz
DDR: 340 MHz
I2C: ready
DRAM: 128 MiB
NAND: 256 MiB
MMC: davinci: 0, davinci: 1
Bad block table not found for chip 0
Bad block table not found for chip 0
nand_bbt: Error while writing bad block table -5
NAND read from offset 3c0000 failed -74
*** Warning - readenv() failed, using default environment

Net: Ethernet PHY: GENERIC @ 0x00
DaVinci-EMAC
Hit any key to stop autoboot: 0
reading boot.scr

558 bytes read
## Executing script at 80600000
reading uImage

2146504 bytes read
## Booting kernel from Legacy Image at 80700000 ...
   Image Name:   Linux-2.6.32.17-davinci1
   Created:      2013-10-31 21:25:59 UTC
   Image Type:   ARM Linux Kernel Image (uncompressed)
   Data Size:    2146440 Bytes = 2 MiB
   Load Address: 80008000
   Entry Point:  80008000
   Verifying Checksum ... OK
   Loading Kernel Image ... OK
OK

Starting kernel ...

Uncompressing Linux..... done, booting.
Linux version 2.6.32.17-davinci1 (cimarron@ubuntu) (gcc version 4.3.3 (Sourcery G++ Lite
2009q1-203) ) #1 PREEMPT Thu Oct 31 15:25:56 MDT 2013
CPU: ARM926EJ-S [41069265] revision 5 (ARMv5TEJ), cr=00053177
CPU: VIVT data cache, VIVT instruction cache
Machine: DM365 Leopard
Memory policy: ECC disabled, Data cache writeback
DaVinci dm36x_rev1.2 variant 0x8
Built 1 zonelists in Zone order, mobility grouping off. Total pages: 15240
Kernel command line: console=ttyS0,115200n8 root=/dev/mmcblk0p2 rw ip=off mem=60M
davinci_enc_mgr.ch0_output=COMPONENT davinci_enc_mgr.ch0_mode=720P-60
davinci_display.cont2_bufsize=6291456 vpfe_capture.cont_bufoffset=6291456
vpfe_capture.cont_bufsize=6291456
video=davincifb:vid0=1920*1080*16,16K:vid1=1920*1080*16,4050K:osd0=1280*720*16,4050K:osd1=1280*72
0*4,1215K dm365_imp.oper_mode=0 davinci_capture.device_type=4 rootwait
PID hash table entries: 256 (order: -2, 1024 bytes)
Dentry cache hash table entries: 8192 (order: 3, 32768 bytes)
```

LeopardBoard 368 DVSDK 4.02 Installation Guide

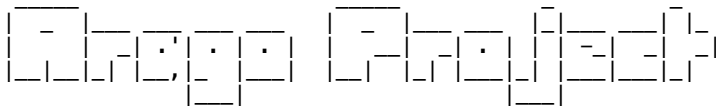
```
Inode-cache hash table entries: 4096 (order: 2, 16384 bytes)
Memory: 60MB = 60MB total
Memory: 56220KB available (4020K code, 392K data, 144K init, 0K highmem)
SLUB: Genslabs=11, HWalign=32, Order=0-3, MinObjects=0, CPUs=1, Nodes=1
Hierarchical RCU implementation.
NR_IRQS:245
Console: colour dummy device 80x30
Calibrating delay loop... 215.44 BogoMIPS (lpj=1077248)
Mount-cache hash table entries: 512
CPU: Testing write buffer coherency: ok
DaVinci: 8 gpio irqs
NET: Registered protocol family 16
davinci_serial_init:97: failed to get UART2 clock
bio: create slab <bio-0> at 0
DM365 IPIPE initialized in Continuous mode
SCSI subsystem initialized
usbcore: registered new interface driver usbfs
usbcore: registered new interface driver hub
usbcore: registered new device driver usb
vpss vpss: dm365_vpss vpss probed
vpss vpss: dm365_vpss vpss probe success
dm365_afew_hw_init
ch0 default output "COMPONENT", mode "720P-60"
VPBE Encoder Initialized
cfg80211: Using static regulatory domain info
cfg80211: Regulatory domain: US
        (start_freq - end_freq @ bandwidth), (max_antenna_gain, max_eirp)
        (2402000 KHz - 2472000 KHz @ 40000 KHz), (600 mBi, 2700 mBm)
        (5170000 KHz - 5190000 KHz @ 40000 KHz), (600 mBi, 2300 mBm)
        (5190000 KHz - 5210000 KHz @ 40000 KHz), (600 mBi, 2300 mBm)
        (5210000 KHz - 5230000 KHz @ 40000 KHz), (600 mBi, 2300 mBm)
        (5230000 KHz - 5330000 KHz @ 40000 KHz), (600 mBi, 2300 mBm)
        (5735000 KHz - 5835000 KHz @ 40000 KHz), (600 mBi, 3000 mBm)
cfg80211: Calling CRDA for country: US
LogicPD encoder initialized
Switching to clocksource timer0_1
musb_hdrc: version 6.0, cppi-dma, host, debug=0
musb_hdrc: USB Host mode controller at fec64000 using DMA, IRQ 12
musb_hdrc musb_hdrc: MUSB HDRC host driver
musb_hdrc musb_hdrc: new USB bus registered, assigned bus number 1
usb usb1: configuration #1 chosen from 1 choice
hub 1-0:1.0: USB hub found
hub 1-0:1.0: 1 port detected
NET: Registered protocol family 2
IP route cache hash table entries: 1024 (order: 0, 4096 bytes)
TCP established hash table entries: 2048 (order: 2, 16384 bytes)
TCP bind hash table entries: 2048 (order: 1, 8192 bytes)
TCP: Hash tables configured (established 2048 bind 2048)
TCP reno registered
NET: Registered protocol family 1
RPC: Registered udp transport module.
RPC: Registered tcp transport module.
RPC: Registered tcp NFSv4.1 backchannel transport module.
JFFS2 version 2.2. (NAND) © 2001-2006 Red Hat, Inc.
msgmni has been set to 109
alg: No test for stdrng (krng)
io scheduler noop registered
io scheduler anticipatory registered (default)
Console: switching to colour frame buffer device 160x45
davincifb davincifb.0: dm_osd0_fb: 1280x720x16@0,0 with framebuffer size 4050KB
davincifb davincifb.0: dm_vid0_fb: 1920x0x16@0,0 with framebuffer size 16KB
davincifb davincifb.0: dm_osd1_fb: 1280x720x4@0,0 with framebuffer size 1215KB
davincifb davincifb.0: dm_vid1_fb: 1920x0x16@0,0 with framebuffer size 4050KB
DM365 IPIPEIF probed
imp serializer initialized
davinci_previewer initialized
davinci_resizer initialized
Serial: 8250/16550 driver, 2 ports, IRQ sharing disabled
serial8250.0: ttyS0 at MMIO 0x1c20000 (irq = 40) is a 16550A
console [ttyS0] enabled
serial8250.0: ttyS1 at MMIO 0x1d06000 (irq = 41) is a 16550A
brd: module loaded
```

LeopardBoard 368 DVSDK 4.02 Installation Guide

```
console [netcon0] enabled
netconsole: network logging started
Initializing USB Mass Storage driver...
usbcore: registered new interface driver usb-storage
USB Mass Storage support registered.
usbcore: registered new interface driver usbtest
mouse: PS/2 mouse device common for all mice
i2c /dev entries driver
Linux video capture interface: v2.00
ths7303 1-002c: chip found @ 0x58 (DaVinci I2C adapter)
vpfe_init
vpfe-capture: vpss clock vpss_master enabled
vpfe-capture vpfe-capture: v4l2 device registered
vpfe-capture vpfe-capture: video device registered
LeopardBoard: switch to tvp7002 HD video input...
tvp7002 1-005c: tvp7002 1-005c decoder driver registered !!
vpfe-capture vpfe-capture: v4l2 sub device tvp7002 registered
LeopardBoard: switch to tvp7002 HD video input...
ths7353 1-002e: chip found @ 0x5c (DaVinci I2C adapter)
ths7353 1-002e: No platform data!!
vpfe-capture vpfe-capture: v4l2 sub device ths7353 registered
vpfe_register_ccdc_device: DM365 ISIF
DM365 ISIF is registered with vpfe.
af major#: 252, minor# 0
AF Driver initialized
aew major#: 251, minor# 0
AEW Driver initialized
Trying to register davinci display video device.
layer=c20d7000,layer->video_dev=c20d7170
Trying to register davinci display video device.
layer=c20d7400,layer->video_dev=c20d7570
davinci_init:DaVinci V4L2 Display Driver V1.0 loaded
watchdog watchdog: heartbeat 60 sec
davinci_mmc davinci_mmc.0: Using DMA, 4-bit mode
usbcore: registered new interface driver usbhid
usbhid: v2.6:USB HID core driver
Advanced Linux Sound Architecture Driver Version 1.0.21.
No device for DAI tlv320aic3x
No device for DAI davinci-i2s
asoc: tlv320aic3x <-> davinci-i2s mapping ok
ALSA device list:
  #0: DaVinci EVM (tlv320aic3x)
TCP cubic registered
NET: Registered protocol family 17
lib80211: common routines for IEEE802.11 drivers
Clocks: disable unused mmcsd1
Clocks: disable unused spi0
Clocks: disable unused spi1
Clocks: disable unused spi2
Clocks: disable unused spi3
Clocks: disable unused spi4
Clocks: disable unused aemif
Clocks: disable unused pwm0
Clocks: disable unused pwm1
Clocks: disable unused pwm2
Clocks: disable unused pwm3
Clocks: disable unused timer1
Clocks: disable unused timer3
Clocks: disable unused emac
Clocks: disable unused voice_codec
Clocks: disable unused rto
Clocks: disable unused mjcp
davinci_emac_probe: using random MAC addr: 86:54:92:f1:b5:ce
emac-mii: probed
Waiting for root device /dev/mmcblk0p2...
mmc0: host does not support reading read-only switch. assuming write-enable.
mmc0: new high speed SDHC card at address 1234
mmcblk0: mmc0:1234 SA04G 3.63 GiB
  mmcblk0: p1 p2
IP-Config: Guessing netmask 255.255.255.0
IP-Config: Complete:
    device=eth0, addr=192.168.2.127, mask=255.255.255.0, gw=255.255.255.255,
```


LeopardBoard 368 DVSDK 4.02 Installation Guide

```
host=192.168.2.127, domain=, nis-domain=(none),
bootserver=255.255.255.255, rootserver=255.255.255.255, rootpath=
kjournald starting. Commit interval 5 seconds
EXT3 FS on mmcblk0p2, internal journal
EXT3-fs: mounted filesystem with writeback data mode.
VFS: Mounted root (ext3 filesystem) on device 179:2.
Freeing init memory: 144K
INIT: version 2.86 booting
Please wait: booting...
Starting udev
Remounting root file system...
Caching udev devnodes
Populating dev cachemv: cannot rename '/tmp/devices': No such file or directory
NET: Registered protocol family 10
ALSA: Restoring mixer settings...
Configuring network interfaces... done.
Setting up IP spoofing protection: rp_filter.
hwclock: can't open '/dev/misc/rtc': No such file or directory
Thu Mar 24 19:24:00 UTC 2011
hwclock: can't open '/dev/misc/rtc': No such file or directory
INIT: Entering runlevel: 5
Starting system message bus: dbus.
Starting telnet daemon.
Starting syslogd/klogd: done
Starting thttpd.
CMEMK module: built on Oct 31 2013 at 15:27:05
Reference Linux version 2.6.32
File /home/cimarron/LeopardBoard_368_Public/ti-dvSDK_dm368-evm_4_02_00_06/
linuxutils_2_26_01_02/packages/ti/sdo/linuxutils/cmek/src/module/cmek.c
allocated heap buffer 0xc6000000 of size 0x4400000
heap fallback enabled - will try heap if pool buffer is not available
CMEM Range Overlaps Kernel Physical - allowing overlap
CMEM phys_start (0x1000) overlaps kernel (0x80000000 -> 0x83c00000)
cmek initialized
IRQK module: built on Oct 31 2013 at 15:27:06
Reference Linux version 2.6.32
File /home/cimarron/LeopardBoard_368_Public/ti-dvSDK_dm368-evm_4_02_00_06/
linuxutils_2_26_01_02/packages/ti/sdo/linuxutils/irq/src/module/irqk.c
irqk initialized
EDMAK module: built on Oct 31 2013 at 15:27:07
Reference Linux version 2.6.32
File /home/cimarron/LeopardBoard_368_Public/ti-dvSDK_dm368-evm_4_02_00_06/
linuxutils_2_26_01_02/packages/ti/sdo/linuxutils/edma/src/module/edmak.c
Calibrating touchscreen (first time only)ts_open: No such file or directory
.
Starting Matrix GUI application.
```



Arago Project <http://arago-project.org> dm368-evm ttyS0

Arago 2011.02 dm368-evm ttyS0

dm368-evm login: root
root@dm368-evm:~#