



IEI Technology Corp.

The background of the top half of the page is a red-to-black gradient. It features a white grid pattern that recedes into the distance. Several spheres of different colors (blue, yellow, grey) are scattered across the grid. On the right side, there is a faint, semi-transparent image of a computer monitor displaying lines of code.

MODEL: HDC-3x Series SDK (Linux)

**A SDK software development kit facilitates development of
Linux-based application for the HDC-3x Series**

User Manual

Rev. 1.17 – 25 March, 2013



Revision

Date	Version	Changes
25 March, 2013	1.17	Updated for software version v1.17: <ul style="list-style-type: none">- Added scale and source select- Removed One Picture Mode- Modified menu (r157)
31 August, 2012	1.17	Updated for new software version v1.17: <ul style="list-style-type: none">- Modified menu (r137)
4 May, 2011	1.02	Added information for the HDC-302E
30 March, 2011	1.01	Added information for the HDC-301 and the HDC-301E and renamed the manual to HDC-3x Series
13 January, 2011	1.00	Initial release

Copyright

COPYRIGHT NOTICE

The information in this document is subject to change without prior notice in order to improve reliability, design and function and does not represent a commitment on the part of the manufacturer.

In no event will the manufacturer be liable for direct, indirect, special, incidental, or consequential damages arising out of the use or inability to use the product or documentation, even if advised of the possibility of such damages.

This document contains proprietary information protected by copyright. All rights are reserved. No part of this manual may be reproduced by any mechanical, electronic, or other means in any form without prior written permission of the manufacturer.

TRADEMARKS

All registered trademarks and product names mentioned herein are used for identification purposes only and may be trademarks and/or registered trademarks of their respective owners.

Table of Contents

1 HDCAPTURE SDK (LINUX)	6
1.1 HDCAPTURE SDK OVERVIEW	7
1.2 SYSTEM REQUIREMENTS	7
1.3 SOFTWARE INSTALLATION	7
1.3.1 Bin File Only.....	7
1.3.2 Source Code	8
1.4 LAUNCH HDCAPTURE SDK.....	10
1.5 VIDEO CAPTURE.....	11
1.5.1 Encoding Setting	11
1.5.2 Encoding	18
1.6 DECODING.....	20
1.6.1 Decoding Setting.....	20
1.6.2 Video I/O Setting	23
1.6.3 Decoding.....	27
2 PROGRAMMING GUIDE	30
2.1 MAIN THREAD	31
2.2 ENCODE	33
2.3 DECODE	34
2.4 FUNCTIONS REFERENCE	35
2.4.1 OnFmbMessage	35
2.4.2 FmbInitInfo	35
2.4.3 FmbReleaseInfo	35
2.4.4 FmbCreatDevInfoAll.....	35
2.4.5 FmbDellDevInfoAll.....	36
2.4.6 FmbGetDeviceTotal.....	36
2.4.7 FmbGetBoardTotal.....	36
2.4.8 FmbSetCurDeviceNum	36
2.4.9 FmbGetCurDeviceNum.....	37
2.4.10 FmbGetCurDevNumOnBoard	37
2.4.11 FMBDeviceNumEnum	37

HDC-3x Series SDK (Linux)

2.4.12 <i>FmbSetFirstDev</i>	38
2.4.13 <i>FmbSetNextDev</i>	38
2.4.14 <i>DrvBootAll</i>	38
2.4.15 <i>MmuxInit</i>	39
2.4.16 <i>MmuxRelease</i>	39
2.4.17 <i>MmuxGetVer</i>	39
2.4.18 <i>MmuxGetCodecSrcResolution</i>	39
2.4.19 <i>MmuxIsCodecSrcConnect</i>	40
2.4.20 <i>MmuxGetOutputSrc</i>	41
2.4.21 <i>MmuxSetOutputSrc</i>	41
3 LINUX OPEN SOURCE CODE	43

Chapter

1

HDCapture SDK (Linux)

HDC-3x Series SDK (Linux)

1.1 HDCapture SDK Overview

The HDCapture SDK is a video capture tool that allows user to capture video through the HDMI input ports in Linux environment. The HDCapture SDK also includes decoding function that decodes the video signal for video output to the HDMI-enabled display device.

This manual includes SDK information for the HDC-3x Series, which includes:

- HDC-301
- HDC-301E
- HDC-302E
- HDC-304E

1.2 System Requirements

The following programs must be installed in order to use the HDCapture SDK in Linux:

- Kernel: Fedora16-64 Distribution (based on Kernel 3.1.0)
- Tool chain: Runs on Fedora16 (binutils 2.18, gcc 4.3.2, glibc 2.9)

1.3 Software Installation

1.3.1 Bin File Only

To install the HDCapture SDK to a system running Linux, please follow the steps below.

Step 1: Copy the **HDCapture_3xx-r157_20121212-6339-3.1.0-7.fc16.x86_64.tar.bz2** (**r157 release code** , **20121212 release date**, **6339 svn version**, **3.1.0-7.fc16.x86_64 kernel version**) file from the utility CD to the system hard drive. The user can also get the latest version from IEI website.

Step 2: Unzip the file by typing:

```
$ tar xf HDCapture_(hdc series)-(version)_(date)-(svn version)_(kernel version).tar.gz
$ tar -jvxf HDCapture_3xx-r157_20121212-6339-3.1.0-7.fc16.x86_64.tar.bz2
```

Step 3: Change to HDCapture directory by typing:

```
$ cd HDCapture_3xx-(version)_(date)-(svn version)
$ cd HDCapture_3xx-r157_20121212-6339
```

Step 4: Login as root by typing:

```
$ su
```

Step 5: Enter the password:

```
password: *****
$
```

Step 6: Install the HDCapture SDK by typing:

```
sh install_ko_hdc3xx.sh
```

Step 7: The system starts to install the HDCapture SDK and shows:

```
Installation start.
Installing fmb_player_apl... done.
Installing firm... done.
Making fmb_driver... `h55fmb.ko' -> `/lib/modules/2.6.43.5-2.fc15.i686/h55fmb.ko'
done.
Making mmux_driver... `media_mux.ko' -> `/lib/modules/2.6.43.5-2.fc15.i686/media_mux.ko'
done.
Unloading old MB86H55 device driver... done.
Loading new MB86H55 device driver... done.
Unloading old media_mux device driver... done.
Loading new media_mux device driver... done.
Installation completed.
```

1.3.2 Source Code

To install the HDCapture SDK to a system running Linux, please follow the steps below.

Step 1: Copy the **HDCapture_SDK_(version)_(date)-(svn version).tar.gz** file from the utility CD to the system hard drive. The user can also get the latest version from IEI website.

HDC-3x Series SDK (Linux)

Step 2: Unzip the file by typing:

```
$ tar xf HDCapture_SDK_(version)_(date)-(svn version).tar.gz
$ tar xf HDCapture_3xx-r157_20121212-6339.tar.gz
```

Step 3: Change to HDCapture directory by typing:

```
$ cd (version)
$ cd r157
```

Step 4: Login as root by typing:

```
$ su
```

Step 5: Enter the password:

```
password: *****
$
```

Step 6: Install the HDCapture SDK by typing:

```
#sh install_hdc3xx.sh
```

Step 7: The system starts to install the HDCapture SDK and shows:

```
Installation start.
Making fmb_player_apl...done.
Installing fmb_player_apl...done.
Installing firm...done.
Making fmb_driver...done.
Making mmux_driver...done.
Unloading old MB86H55 device driver...done.
Loading new MB86H55 device driver...done.
Unloading old media_mux device driver...done.
Loading new media_mux device driver...done.
Installation completed.
#
```

Step 8: Type “exit” to logout from root:

```
# exit
```

1.4 Launch HDCapture SDK

Step 1: To launch the HDCapture SDK, type “hdcapture” as below:

```
$ hdcapture
```

Step 2: The system starts to launch the HDCapture SDK. When the main menu appears, it is ready to use.

```
[dev-0] Boot ...
[dev-1] Boot ...
[dev-2] Boot ...
[dev-3] Boot ...

_____
/ _____/
|                ||
| HDCapture SDK for HDC 3xx series          ||
| SDK Version: r157                         ||
| firm      : 2012-1113-9341-0400          ||
| cpld      : 0304e003                     ||
| fpga      : 0304e003                     ||
| mcu       : 8051A101                      ||
| mmux-driver: 0.0.3.3                     ||
| fmb-driver : 0.0.3.2                     ||
| apl       : 0.1.5.3                       ||
| kernel    : 3.1.0-7.fc16.x86_64         ||
|_____||

< Main menu >

Select following character.

1 Encoding
2 Decoding
3 Decoding(Start pause)

6 Encode settings
7 Decode settings
```

HDC-3x Series SDK (Linux)

```
8 Video I/O settings
9 Exit
fmb-0?>
```



The contents marked in blue are especially for the HDC-302E and the HDC-304E video capture cards.

1.5 Video Capture

There two main procedures to capture video by the HDCapture SDK in Linux. First, setup the encoding setting. Second, start capturing video. These two procedures are described in the following sections.

1.5.1 Encoding Setting

Before capturing the video, please connect the hardware and follow the instruction below to configure the encoding settings.

Step 1: Launch the HDCapture SDK (refer to **Section 1.4**).

Step 2: Type **6** to select “6 Encode settings” from the main menu.

```
fmb-0?>6
```

Step 3: The following message shows.

```
-----
-----
< Device list >
Select following number.

1 /dev/h55fmb0
2 /dev/h55fmb1
3 /dev/h55fmb2
4 /dev/h55fmb3
```

```
fmb-0?>
```

Step 4: Select a port to configure by entering the port number. For example, type **1** to select the first port.

```
fmb-0?>1
```

Step 5: The following message shows.

```
< Encode settings - Sub menu >
      'Z'<-----
Select following number. (Current settings)

1 Output port setting      (File ./stream-0.mpg)
2 Video resolution setting (1920x1080_59.94p)
3 Video bitrate setting    (CBR, 6000kbps)
4 Audio format             (MPEG1L2, 256kbps)
5  Video scale             (Not Use)
fmb-0?>
```

**NOTE:**

The user can enter **Z** at anytime to go back to the previous page.

Step 6: **Configure the file name of the encoded file.** To configure the file name, type **1** to select the “Output port setting”.

```
fmb-0?>1
```

Step 7: The following message shows.

```
[[[ Encode settings - Sub menu -> Output port ]]]
      'Z'<----->Enter
Select following number. (* is current setting)

1 * File
fmb-0?>
```

Step 8: Type **1** to select “File”.

HDC-3x Series SDK (Linux)

```
fmb-0?>1
```

Step 9: The following message shows the current file name is “stream-0.mpg”.

```
[[[ Encode settings - Sub menu -> Output port -> File ]]]
          ~~~~~->Enter

Input filepath of stream.
Current filepath is ./stream-0.mpg.
fmb-0?>
```

Step 10: Type the new file name. Take “video1.mpg” as an example.

```
fmb-0?>video1.mpg
```

Step 11: The encoding setting sub menu appears. The file name has been changed to “video1.mpg”.

```
< Encode settings - Sub menu >
          'Z'<-----

Select following number. (Current settings)

1 Output port setting      (File video1.mpg)
2 Video resolution setting (1920x1080_59.94p)
3 Video bitrate setting    (CBR, 6000kbps)
4 Audio format             (MPEG1L2, 256kbps)
5  Video scale             (Not Use)
fmb-0?>
```

Step 12: Video resolution settings. The encoding video resolution is automatically detected by the hardware and can not be changed from the HDCapture SDK. If the video resolution is changed and does not match the current settings list in the SDK sub-menu, please type **2** to select “Video resolution setting”.

```
fmb-0?>2
```

Step 13: When the following message shows, press **Enter** to reload the current setting.

```
[[[ Encode settings - Sub menu -> Video resolution ]]]
          'Z'<----->Enter
```

Select following number.

1920x1080_60p

* 1920x1080_59.94p

1920x1080_50p

1920x1080_60i

1920x1080_59.94i

1920x1080_50i

1440x1080_60i

1440x1080_59.94i

1440x1080_50i

1280x720_60p

1280x720_59.94p

1280x720_50p

720x480_60i

720x480_59.94i

720x480_50i

(* is current setting)

Please change the input source resolution if you want to use select others. Then press "Enter" to reload.

fmb-0?>



NOTE:

The contents marked in blue are especially for the HDC-302E and the HDC-304E video capture cards. The HDC-301 and HDC-301E can not detect the resolution automatically. The user needs to select the input source resolution manually.

Step 14: Type **Z** to go back to the sub-menu of the encode setting.

fmb-0?>z

HDC-3x Series SDK (Linux)

```

< Encode settings - Sub menu >
          'Z'<-----
Select following number. (Current settings)

1 Output port setting      (File video1.mpg)
2 Video resolution setting (1920x1080_59.94p)
3 Video bitrate setting    (CBR, 6000kbps)
4 Audio format              (MPEG1L2, 256kbps)
5  Video scale              (Not Use)
fmb-0?>

```

Step 15: Configure the video Scale,

```
fmb-0?>5
```

Step 16: The following message shows.

```

[[[ Encode settings - Sub menu -> Video scaler ]]]
          'Z'<----->Enter
Select following number. (* is current setting)

1.  * Not Use
2.  1440x1080
3.  720x480
5.  320x240
fmb-0?>5
-----
< Encode settings - Sub menu >
          'Z'<-----
Select following number. (Current settings)

1  Output port setting      (File ./stream-0.mpg)
2  Video resolution setting(1920x1080_59.94i)
3  Video bitrate setting    (CBR, 768kbps)
4  Audio format              (MPEG1L2, 256kbps)

```

```
5 Video scale (320x240)
```

**NOTE:**

The video scale support

1080(60i) -> 480(60i)

1080(59.94i) -> 480(59.94i)

1080(50i) -> 480(50i)

1080(60i) -> 320x240(30p)

1080(59.94i) -> 320x240(29.97p)

1080(50i) -> 320x240(25p)

720(60p) -> 480 (60i)

720(59.94p) -> 480 (59.94i)

720(50p) -> 480 (50i)

720(60p) -> 320x240(30p)

720(59.94p) -> 320x240(29.97p)

720(50p) -> 320x240(25p)

Step 17: Type **Z** to go back to the sub-menu of the encode setting.

```
fmb-0?>z
```

```
< Encode settings - Sub menu >
```

```
'Z'<-----
```

```
Select following number. (Current settings)
```

- | | |
|----------------------------|--------------------|
| 1 Output port setting | (File video1.mpg) |
| 2 Video resolution setting | (1920x1080_59.94p) |
| 3 Video bitrate setting | (CBR, 6000kbps) |
| 4 Audio format | (MPEG1L2, 256kbps) |
| 5 Video scale | (Not Use) |

```
fmb-0?>
```


HDC-3x Series SDK (Linux)

Step 18: Configure the video bitrate. To configure the video bitrate, type **3** to select the “Video bitrate setting”.

```
fmb-0?>3
```

Step 19: The following message shows.

```
[[[ Encode settings - Sub menu -> Video ratecontrol -> Video bitrate ]]]
          'Z'<----->Enter
Select following number. (* is current setting)
  1 * CBR
  2 VBR
fmb-0?>
```

Step 20: To configure the video bitrate as CBR, type **1** to select “CBR”. To configure the video bitrate as VBR, type **2** to select “VBR”. Now we take CBR as an example.

```
fmb-0?>1
```

Step 21: The following message shows the current CBR value.

```
[[[ Encode settings - Sub menu -> Video rate control -> Video bitrate ]]]
          'Z'<----->Enter
Input bitrate in kbps. (6000kbps -- 24000kbps(I) or 30000kbps(P))
Current value is 6000kbps.
fmb-0?>
```

Step 22: To change the value, enter a bitrate.

```
fmb-0?>24000
```

```
< Encode settings - Sub menu >
          'Z'<-----
Select following number. (Current settings)

  1 Output port setting      (File video1.mpg)
  2 Video resolution setting (1920x1080_59.94p)
  3 Video bitrate setting    (CBR, 24000kbps)
  4 Audio format             (MPEG1L2, 256kbps)
```

```
5 Video scale (Not Use)
fmb-0?>
```


NOTE:

The audio format of the encoding and decoding settings is set by default and can not be changed.

1.5.2 Encoding

After configuring encoding settings, the user can start capturing the video by following the steps below.

Step 1: Access to the main menu. Type **1** to select “1 Encoding” from the main menu.

```
fmb-0?>1
```

Step 2: The following message shows. The option marked with * is the selected port for capturing video. The unsupported port appears with an “unknown” message. Type the port number to select or deselect the port. In this example, port 1 is selected.

```
< Device list >
      'Z'<-----
Select following number. (* is current setting)

1 < * > /dev/h55fmb0 link : 1920x1080@59.94p
2 < > /dev/h55fmb1 no link : unknown
3 < > /dev/h55fmb2 link : 1920x1080@59.94p
4 < > /dev/h55fmb3 no link : unknown

y Ok
z Return
fmb-0?>
```

HDC-3x Series SDK (Linux)



The contents marked in blue are especially for the HDC-302E and the HDC-304E video capture cards.

Step 3: After selecting the port(s), type **y**.

```
fmb-0?>y
```

Step 4: The current settings of the selected port shows as following.

```
[[[ Encoding - confirm -> Encoding ]]]
      'Z'<----->Enter
Current settings
Stream output      File ./stream-0.mpg
Video format       1920x1080
Video framerate    59.94p
Video rate control CBR
Video bitrate      6000kbps
Audio format       MPEG1L2
Audio bitrate      256kbps
Scale Format        Not Use
Press <Enter> key to start encoding.
fmb-0?>
```

Step 5: Press **Enter**. The system shows the following message and starts capturing video.

```
[[[ Encoding ]]]
Press <Enter> key to stop.
fmb-0?>
```

Step 6: Press **Enter** again to stop capturing video.

1.6 Decoding

There are three main procedures to decode video through the HDCapture SDK in Linux. First, configure the decoding setting. Second, configure the video output port setting. Third, start decoding video. These three procedures are described in the following sections.

1.6.1 Decoding Setting

Please connect the hardware and follow the instruction below to configure the decoding settings before decoding the video.

Step 1: Launch the HDCapture SDK (refer to **Section 1.4**).

Step 2: Type **7** to select “7 Decode settings” from the main menu.

```
fmb-0?>7
```

Step 3: The following message shows.

```
-----  
-----  
< Device list >  
Select following number.  
    === Card 1: HDC-304e ===  
1 /dev/h55fmb0  
2 /dev/h55fmb1  
3 /dev/h55fmb2  
4 /dev/h55fmb3  
z Return  
fmb-0?>
```

Step 4: Select a port to configure by entering the port number. For example, type **1** to select the first port.

```
fmb-0?>1
```

Step 5: The following message shows.

```
< Encode settings - Sub menu >  
    'Z'<-----
```

HDC-3x Series SDK (Linux)

```
Select following number. (Current settings)

1 Input port setting      (File ./stream-0.mpg)
2 Video resolution setting (1920x1080_59.94i)
3 Audio format            (MPEG1L2)

fmb-0?>
```



NOTE:

The user can enter **Z** at anytime to go back to the previous page.

Step 6: Configure the file name of the decoded file. To configure the file name, type 1 to select the “Input port setting”.

```
fmb-0?>1
```

Step 7: The following message shows.

```
[[[ Encode settings - Sub menu -> Input port ]]]
          'Z'<----->Enter

Select following number. (* is current setting)

1 * File

fmb-0?>
```

Step 8: Type 1 to select “File”.

```
fmb-0?>1
```

Step 9: The following message shows the current file name is “stream-0.mpg”.

```
[[[ Decode settings - Sub menu -> Input port -> File]]]
          'Z'<----->Enter

Input filepath of stream.

Current filepath is ./stream-0.mpg.

fmb-0?>
```

Step 10: Type the new file name. Take “video1.mpg” as an example.

```
fmb-0?>video1.mpg
```

Step 11: The following message shows.

```
[[[ Decode settings - Sub menu -> Input port -> File -> Start position ]]]
                                     'Z'<----->Enter

Input start position in byte.
Current value is 0 (Max.18474912)
fmb-0?>
```

Step 12: Setup the start point to decode the video. Type **0** to decode the video from the beginning.

```
fmb-0?>0
```

Step 13: The decoding setting sub menu shows. The file name has been changed to “video1.mpg”.

```
< Decode settings - Sub menu >
                                     'Z'<-----

Select following number. (Current settings)

1 Input port setting      (File video1.mpg)
2 Video resolution setting (1920x1080_59.94i)
3 Audio format            (MPEG1L2)

fmb-0?>
```

Step 14: **Configure video resolution settings.** Type **2** to select “Video resolution setting” to configure the video resolution.

```
fmb-0?>2
```

Step 15: When the following message shows, select the resolution of the video which will be decoded.

```
[[[ Decode settings - Sub menu -> Video resolution ]]]
                                     'Z'<----->Enter

Select following number.
```

HDC-3x Series SDK (Linux)

```

1  1920x1080_60p
2 * 1920x1080_59.94p
3  1920x1080_50p
4  1920x1080_60i
5  1920x1080_59.94i
6  1920x1080_50i
i  1280x720_60p
j  1280x720_59.94p
k  1280x720_50p
l  720x480_60i
m  720x480_59.94i
n  720x480_50i
(* is current setting)
fmb-0?>

```



NOTE:

The audio format of the encoding and decoding settings is set by default and can not be changed.

1.6.2 Video I/O Setting



NOTE:

This section is only for the HDC-302E and the HDC-304E video capture cards.

The HDC-302E and the HDC-304E both support HDMI output port(s) that broadcast encoded video bypass or display via decoded port. To configure the video output port which displays the decoded video, please follow the steps below.

Step 1: Type **8** to select “8 Video I/O settings” from the main menu.

```
fmb-0?>8
```

Step 2: The following message shows.

```
-----  
< Mode list >  
Select following character.  
0 Video output  
1 Video input  
  
z Return
```

Step 3: Type **1** to select Video input

```
fmb-0?>1
```

```
< Device list >  
Select following character.  
  
===== Card 1: HDC-304e =====  
0 Video input      [Video-0]  
1 Video input      [Video-1]  
2 Video input      [Video-2]  
3 Video input      [Video-3]  
z Return  
  
fmb-0?>
```

Step 4: Type **0** to select Card 1: HDC-304e Channel 0 source

```
fmb-0?>0
```

Step 5: The following message shows.

```
You Select dev-0  
-----  
< Device list >  
  
                'Z'<----->Enter  
Select following number. (* is current setting)
```


HDC-3x Series SDK (Linux)

```
0 * Video 0
1 Video 1
2 Video 2
3 Video 3
```

Step 6: Card 1: HDC-304e Channel 0 source change to [Video-1]

```
< Device list >
Select following character.

===== Card 1: HDC-304e =====
0 Video input      [Video-1]
1 Video input      [Video-1]
2 Video input      [Video-2]
3 Video input      [Video-3]

z Return
```

Step 7: Type Z

```
fmb-0?>z
```

The following message shows

```
-----
< Mode list >
Select following character.

0 Video output
1 Video input

z Return
```

Step 8: Type 0 to select Video output

```
fmb-0?>0
```

Installed with HDC-302E:

```
< Device list >
Select following character.

    === Card 1: HDC-302e ===
1 Video output      [Video-0]
2 Video output      [Video-1]

z Return
fmb-0?>
```

Installed with HDC-304E:

```
< Device list >
Select following character.

1 Video output      [Codec-0]

fmb-0?>
```

Step 9: Type **1** to select the first video output port. If two HDC-3x Series capture cards are installed, there will be two video output ports to select.

```
fmb-0?>1
```

Step 10: The following message shows.

```
< Device list >

    'Z'<----->Enter

Select following number. (* is current setting)

1 Bypass Mode
2 H.264 Mode

fmb-0?>
```

Step 11: Type **2** to select “H.264 Mode” from the sub-menu.

```
fmb-0?>2
```

HDC-3x Series SDK (Linux)

Step 12: The following message shows. Select the codec chip that is used for decoding the video. Please note that the codec chip options may vary by the installed capture card.

```
< Device list >
          'Z'<----->Enter
Select following number. (* is current setting)
1 * Codec chip 0
2   Codec chip 1
3   Codec chip 2
4   Codec chip 3
fmb-0?>
```



NOTE:

To broadcast the encoding video directly, please select “1 Bypass mode” from **Step 5**. When the following message shows, select the input port that is used for encoding the video. Please note that the input port options may vary by the installed capture card.

```
< Device list >
          'Z'<----->Enter
Select following number. (* is current setting)
1 * Input 0
2   Input 1
3   Input 2
4   Input 3
fmb-0?>
```

1.6.3 Decoding

After configuring decoding settings and video I/O setting, the user can start decoding the video by following the steps below.

Step 1: Access to the main menu. Type **2** to select “2 Decoding” from the main menu.

```
fmb-0?>2
```

Step 2: The following message shows.

```
< Device list >
Select following number.

1 /dev/h55fmb0
2 /dev/h55fmb1
3 /dev/h55fmb2
4 /dev/h55fmb3

fmb-0?>
```

Step 3: Select a port by entering the port number. The selected port must be same with the port selected in the decoding setting. The first port is selected in the encoding setting (**Section 1.6.1 Step 4**), so type **1** to select the first port.

```
fmb-0?>1
```

Step 4: The current settings of the selected port shows as following.

```
[[[ Decoding - confirm -> Decoding ]]]
      'Z'<----->Enter

Current settings
Stream input      File ./video1.mpg
Start file position 0 / 18474912 byte
Video format      1920x1080
Video framerate   59.94i
Audio format      MPEG1L2
Press <Enter> key to start decoding.

fmb-0?>
```

Step 5: Press **Enter**. The system shows the following message and starts decoding video.

```
[[[ Decoding ]]]
Select following character.
```

HDC-3x Series SDK (Linux)

```

p  Pause
s  Slow
m  Mute (OFF)
e  Stop and return
fmb-0?>

```

Step 6: While decoding, the user can type **P** to pause, type **S** to slow down or type **M** to mute the video.

Step 7: Type **e** to stop decoding video.

```
fmb-0?>e
```

```

[[[ Decoding - finished -> Main menu ]]]
      ~~~~~->Enter

Finished decoding.
Press <Enter> key to return to Main menu.
fmb-0?>

```



NOTE:

The third option of the main menu, Decoding (Start pause), is similar with **Section 1.6.3 Decoding**. The only difference is when press Enter in **Step 5**, the system will not start decoding right away until the user type the decoding command (d).

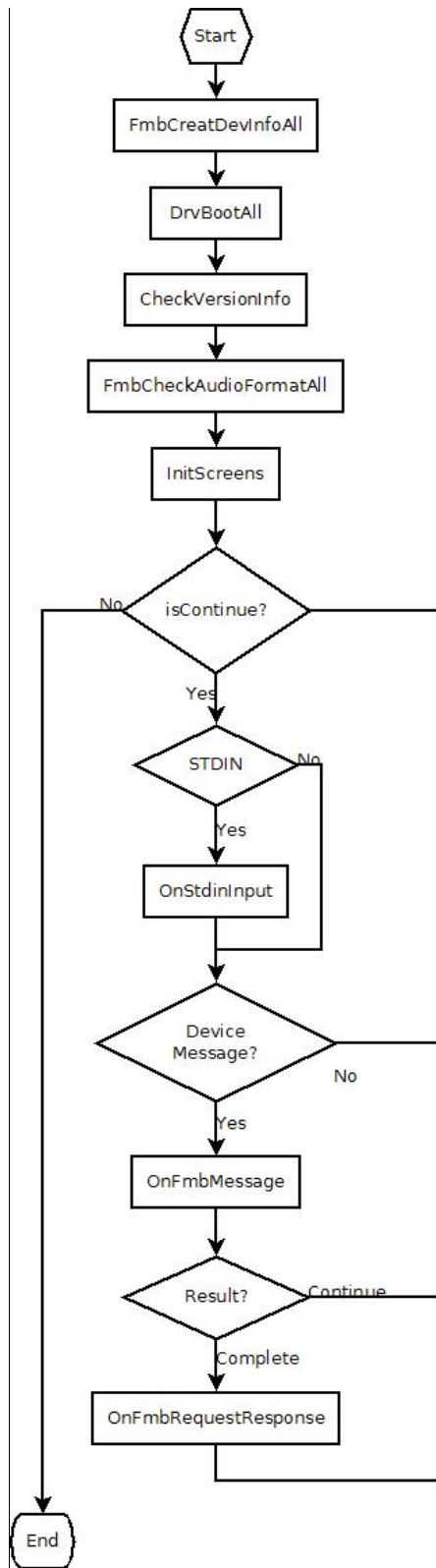
Chapter

2

Programming Guide

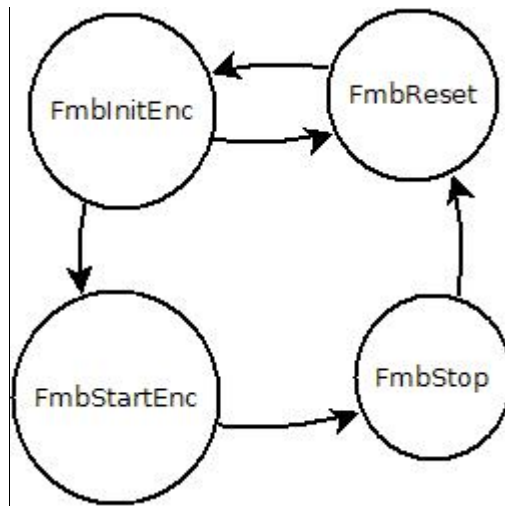
2.1 Main Thread

- Step 1:** Initialize global variables (FmbInitInfo).
- Step 2:** Set the default of all devices (FmbCreatDevInfoAll) and create a thread for each device (fmbRecvThread). This thread will receive Events sent by codec chip, and send the message to the main thread through the pipe.
- Step 3:** Write firmware to the codec chip and boot codec chip (DrvBootAll).
- Step 4:** Read the current version of the codec chip (CheckVersionInfo).
- Step 5:** Set the audio compression format supported by the current version of the codec chip (FmbCheckAudioFormatAll).
- Step 6:** Initialize the screen (InitScreens).
- Step 7:** Wait for the keyboard input and execute the commands of the codec chip setting (OnStdinInput)
- Step 8:** Receive the message sent by FmbRecvThread. Determine the current state of the codec chip and execute the corresponding actions (OnFmbMessage).
- Step 9:** Determine whether the device setup is completed and go to next step (OnFmbRequestResponse).
- Step 10:** Repeat **Step 7 ~ Step 9**.



2.2 Encode

- void FmbReset(void);
- void FmbInitEnc(void);
- void FmbStartEnc(void);
- void FmbStop(void);

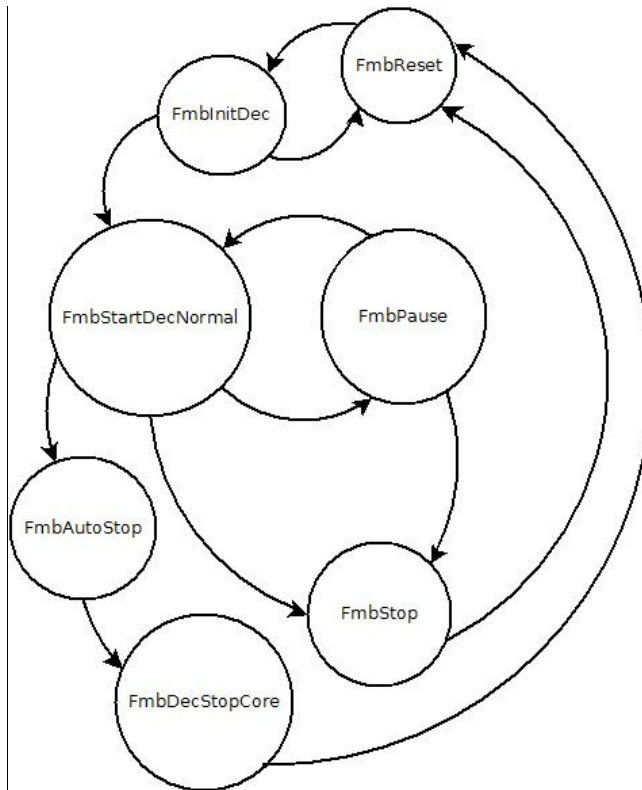


NOTE:

Be sure to use OnFmbMessage () function to confirm the codec chip return the message to switch the state (return FMBEnmResultRequestComplete), then continue to the next action.

2.3 Decode

- void FmbReset(void);
- void FmbInitDec(void);
- void FmbStartDecNormal(void);
- void FmbPause(void);
- void FmbSetMuteMode(void);
- void FmbStop(void);
- void FmbAutoStop(void);
- void FmbDecStopCore(void);



NOTE:

Be sure to use OnFmbMessage () function to confirm the codec chip return the message to switch the state (return FMBEnmResultRequestComplete), then continue to the next action.

2.4 Functions Reference

2.4.1 OnFmbMessage

Description: receive the returned message from codec chip to determine the current state

Prototype: enum FMBResultEnum OnFmbMessage (int fd)

Parameters: fd: device (codec chip) handle

Return Value: Current state: FMBEnmResultContinue, FMBEnmResultRequestComplete
enum FMBResultEnum {FMBEnmResultContinue, FMBEnmResultRequestComplete}

2.4.2 FmbInItInfo

Description: card initialized

Prototype: int FmbInItInfo (void)

Parameters: NONE

Return Value: TRUE: function succeeded; FALSE: functioned failed

2.4.3 FmbReleaseInfo

Description: release the card resources

Prototype: void FmbReleaseInfo (void)

Parameters: NONE

Return Value: NONE

2.4.4 FmbCreatDevInfoAll

Description: create and initialize all the codec chip resources

Prototype: int FmbCreatDevInfoAll(void)

Parameters: NONE

Return Value: TRUE: function succeeded; FALSE: function failed

2.4.5 FmbDellDevInfoAll

Description: release all of codec chip resources

Prototype: void FmbDellDevInfoAll(void)

Parameters: None

Return Value: None

2.4.6 FmbGetDeviceTotal

Description: get the current number of codec chip

Prototype: int FmbGetDeviceTotal(void)

Parameters: None

Return Value: codec chip number

2.4.7 FmbGetBoardTotal

Description: get the current number of card

Prototype: int FmbGetBoardTotal (void)

Parameters: None

Return Value: card number

2.4.8 FmbSetCurDeviceNum

Description: set the codec chip number

Prototype: void FmbSetCurDeviceNum(enum FMBDeviceNumEnum deviceNum)

Parameters: enum FMBDeviceNumEnum, device number

```
enum FMBDeviceNumEnum {  
    FMBEnmDevice_NONE = -1,  
};
```

HDC-3x Series SDK (Linux)

```
FMBEnmDevice_0 = 0,  
FMBEnmDevice_1,  
FMBEnmDevice_2,  
FMBEnmDevice_3,  
FMBEnmDevice_4,  
FMBEnmDevice_5,  
FMBEnmDevice_6,  
FMBEnmDevice_7,  
FMBEnmDeviceMax  
};
```

Return Value: None

2.4.9 FmbGetCurDeviceNum

Description: get the chip number of the current operating codec (numbering by the order of all the codec on the board)

Prototype: enum FMBDeviceNumEnum FmbGetCurDeviceNum(void);

Parameters: None

Return Value: enum FMBDeviceNumEnum, device number

2.4.10 FmbGetCurDevNumOnBoard

Description: get the chip number of the current operating codec (numbering by the order of the codec on a board)

Prototype: enum FMBDeviceNumEnum FmbGetCurDevNumOnBoard (void);

Parameters: None

Return Value: enum FMBDeviceNumEnum, device number

2.4.11 FMBDeviceNumEnum

Description: get the chip number of the current operating codec

Prototype: enum FMBDeviceNumEnum FmbGetCurDeviceNum(void)

Parameters: None

Return Value: enum FMBDeviceNumEnum, device number

2.4.12 FmbSetFirstDev

Description: set to switch the current operating device to the first available device code

Prototype: enum FMBDeviceNumEnum FmbSetFirstDev(void)

Parameters: None

Return Value: enum FMBDeviceNumEnum: return the device number set, return FMBEnmDevice_NONE if there is no available devices

2.4.13 FmbSetNextDev

Description: set to switch the current operating device to the next available device number of the current device

Prototype: enum FMBDeviceNumEnum FmbSetNextDev(void)

Parameters: None

Return Value: enum FMBDeviceNumEnum: return the device number set, return FMBEnmDevice_NONE if there is no available devices

2.4.14 DrvBootAll

Description: load the firmware of all the codec chip and start the codec chip

Prototype: int DrvBootAll(void)

Parameters: None

Return Value: 0: function succeeded, other values: function failed

HDC-3x Series SDK (Linux)

2.4.15 MmuxInit

Description: initialize the CPLD to control handle

Prototype: MmuxHandle MmuxInit(int deviceNum)

Parameters: deviceNum: request to initialize the version of card number and strat from 0

Return Value: MmuxHandle

2.4.16 MmuxRelease

Description: release CPLD and control handle

Prototype: void MmuxRelease(MmuxHandle hdMmux)

Parameters: hdMmux: get handle when MmuxInit executes

Return Value: None

2.4.17 MmuxGetVer

Description: get the current version of CPLD, FPGA and driver

Prototype: int MmuxGetVer(MmuxHandle hdMmux, struct MmuxVer *p_Ver)

```
struct MmuxVer {
    unsigned long    CPLD;
    unsigned long    FPGA;
    char             driver[MMUX_VER_LEN];
};
```

Parameters: hdMmux: handle obtained when MmuxInit executes, p_Ver: the structure indicator of stored information

Return Value: 0: get version successfully, the other value failed

2.4.18 MmuxGetCodecSrcResolution

Description: get the current resolution of the codec chip input

Prototype: enum MMUX_RESOLUTION MmuxGetCodecSrcResolution(MmuxHandle hdMmux, int codecNum);

```
enum MMUX_RESOLUTION {
    MMUX_RESOLUTION_1080P_60,
    MMUX_RESOLUTION_1080P_59,
    MMUX_RESOLUTION_1080P_50,
    MMUX_RESOLUTION_1080I_60,
    MMUX_RESOLUTION_1080I_59,
    MMUX_RESOLUTION_1080I_50,
    MMUX_RESOLUTION_1080P_24,
    MMUX_RESOLUTION_1080P_23,
    MMUX_RESOLUTION_720P_60,
    MMUX_RESOLUTION_720P_59,
    MMUX_RESOLUTION_720P_50,
    MMUX_RESOLUTION_720P_24,
    MMUX_RESOLUTION_720P_23,
    MMUX_RESOLUTION_480i,
    MMUX_RESOLUTION_576i_50,
    MMUX_RESOLUTION_UNKNOWN = 0x1f,
};
```

Parameters: hdMmux: Get handle when MmuxInit executes, codecNum: Request the codec chip number

Return Value: the current resolution (enum MMUX_RESOLUTION)

2.4.19 MmuxIsCodecSrcConnect

Description: request whether there is signal input from the current codec chip

Prototype: int MmuxIsCodecSrcConnect(MmuxHandle hdMmux, int codecNum)

Parameters: hdMmux: Get handle when MmuxInit executes, codecNum: request the codec chip number

HDC-3x Series SDK (Linux)

Return Value: 0: no signal, 1: signal input

2.4.20 MmuxGetOutputSrc

Description: get the current source set of the video output

Prototype: enum MMUX_OUTPUT_SRC MmuxGetOutputSrc(MmuxHandle hdMmux, int outputNum)

```
enum MMUX_OUTPUT_SRC {
    MMUX_OUT_NONE = -1,
    MMUX_OUT_IN_0 = 0,
    MMUX_OUT_IN_1,
    MMUX_OUT_IN_2,
    MMUX_OUT_IN_3,
    MMUX_OUT_CODEEC_0,
    MMUX_OUT_CODEEC_1,
    MMUX_OUT_CODEEC_2,
    MMUX_OUT_CODEEC_3,
    MMUX_OUT_SRC_MAX,
};
```

Parameters: Get handle when MmuxInit executes, outputNum: request the output number (currently always 0)

Return Value: enum MMUX_OUTPUT_SRC: the sources of current set

2.4.21 MmuxSetOutputSrc

Description: set the source of the video output

Prototype: int MmuxSetOutputSrc(MmuxHandle hdMmux, int outputNum, enum MMUX_OUTPUT_SRC outputSrc)

Parameters: hdMmux: Get handle when MmuxInit executes, outputNum: request the output number (currently always 0), outputSrc: set the image source

Return Value: 0: set succeeded

Chapter

3

Linux Open Source Code

Linux Open Source Code

This product from IEI contains software originating from Fujitsu and its suppliers and also contains component from the open source community. Software code is subject to the GNU General Public License. See the GNU General Public License for the exact terms and conditions of this license.

The GPL Code used in this products are distributed without any warranty and are subject to the copyrights of their authors. You can obtain the open source code in the user manual CD-ROM or at <http://www.ieiworld.com> under "Download" page.

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

HDC-3x Series SDK (Linux)

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

HDC-3x Series SDK (Linux)

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.



4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

HDC-3x Series SDK (Linux)

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

HDC-3x Series SDK (Linux)

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
```

```
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
```

```
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for  
details.
```

The hypothetical commands ``show w'` and ``show c'` should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than ``show w'` and ``show c'`; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program  
`Gnomovision' (which makes passes at compilers) written by James Hacker.
```

```
<signature of Ty Coon>, 1 April 1989
```

```
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.