

# **Users Guide**

for

**MPEG4-Encoder** 

(2008)

Part Number: ADSP-EDU-BF531(MPEG4-Encoder)

Update: December, 2008, Beijing, China

www.openadsp.com 印印 环源社区



【文档简介】

基于 BF53X 处理器的 MPEG4-Encoder Demo 说明文档

#### DEMO目的

介绍如何使用 ADI 提供的 MPEG4\_Encoder 库

二. 硬件平台说明

开发板: ADSP-BF53X-SSK 仿真器: AD-HP560ICE-M 软件环境: Visual DSP++5.0

三. 实验内容及原理



四. 核心代码片段分析 typedef struct nallist\_tag { unsigned char \*ptr; int numbytes;

} nallist\_t;

typedef struct naltable\_tag {
 nallist\_t nallist[12];
 int numnal;
 int grosslength;
} naltable\_t;
//以上的 2 个数据结构是用来接收编码结果

www.openadsp.com IDSP 新語注区

```
四维电子
```

```
write_to_file(FILE* out, naltable_t * pn)//把编码后的数据存入 PC 文件中
{
 int n=0, j, len,i;
 char *stream;
                    // pn->numnal 记录了需要存入文件的块
for (j=0;j<pn->numnal;j++)
数
     {
         stream = (char*)pn->nallist[j].ptr; //需要存入文件的数据的首地址
                                     //需要存入文件的数据的个数
         len = pn->nallist[j].numbytes;
         n = fwrite (stream, 1, len, out);
     }
}
*******
                                                    Definitions
*
                                          Function
*******/
void main()
{
  int result;
  Setup_Flags();
  Init_EBIU();
  Init_SDRAM();
  Init_Flash();
  int num_of_frames = 30; //实际需要编码的帧数
                                  //记录当前的编码帧号
  int fno = 0;
                               //记录返回需要写入文件的数据结构
  naltable_t * test;
  char * Outfiles [1] = \{(char^*)-1\};
  char * Infiles[1] = {(char*)-1};
  FILE *frec, *fwrite;
  Outfiles[0] = "../enc-test.m4v";
  Infiles[0] = "../enc-demo-320x240.eyuv";
                                  // 打开输入文件
  frec = fopen(Infiles[0], "rb");
                     www.openadsp.com
```

Open

ADSP FFFFFF

Open A DCD	四维电子							
fwrite = fopen(Outfiles[0], "wb"); // 初始化编码需要内存地址和参数	//打开输出文件							
<pre>test = Init_encoder((unsigned char * )0;</pre>	x4,(unsigned char * )0x20000);							
<pre>write_to_file(fwrite,test);</pre>	//把返回的数据写入 PC 文件							
while (num_of_frames != 0 && fno < 1 {	num_of_frames)							
char * read_data;								
int n;								
<pre>read_data = start_encoder();</pre>	//本帧编码的初始							
$n = fread (read_data, 1,0x1C200, f)$	rec); //从文件中读入一帧图像							
test = Do_encoder(); $\overline{A_{y}^{I}}$	//进行编							
write_to_file(fwrite,test); 回结果写入文件	//把编码返							
stop_encoder(); 束本帧的编码	//结							
<pre>fno ++; printf("number %d framer encode }</pre>	is done \n",fno-1);							
test = //结束本次所有的编码	Close_encoder();							
write_to_file(fwrite,test); 入文件的结尾	//写							
fclose(frec);								
//关闭读取文件								
fclose(fwrite); //关闭写入文件								
fno = 0;								
return 0;								
www.openadsp.com 印印 研題社区								





#### 五. 用户使用说明

1. 用户可调用函数接口

```
      Init_encoder 代码中的函数原型是:

      naltable_t * Init_encoder(unsigned char * output, unsigned char * input)

      功能:
      初始化编码需要内存地址和参数。

      参数:
      unsigned char * output: 编码输出的结构起始地址;

      需要保证起始地址往后的 0x4000 的空间给 output 独享。

      unsigned char * input:
      编码输入的结构起始地址;

      需要保证起始地址往后的 0x1c200 的空间给 input 独享。

      返回:
      naltable_t *: 返回的数据结构的地址。

      start_encoder、Do_encoder 、Close_encoder 函数都会返回一个 naltable_t 结构的指针,这个指针所指向的数据都是需要写入 PC 文件中的。

      存入文件的方法可以参照 write_to_file () 函数。
```

#### 2. 用户使用库的要求

1. 函数只能对 YUV420 存储格式的图像进行编码,
图像大小为 320×240×3/2= 0X1C200bit;
2 编码时需要使用的数据空间,需要按照初始化的要求,空间给的小会造成
无法正常运行。
3 用户在自己生成工程中使用该解码库,必须向工程中添加
encodervdsp533.ldf、 libmpeg4enc_33.dlb 、 mp4enc_lib.dlb。
因为解码库的内存分配完全依赖 encodervdsp533.ldf 文件。
4 如果用户更改了工程属性,那么该操作会反映给 LDF 文件,那么需要把
原始的 LDF 再次拷贝到工程下。
5 如果用户的图像输入不是文件方式,只需要将文件读取的操作,改为向目
标地址填写图像数据的功能。填入的数据存储方式一定要为 YUV420。
如图:YUV每个元素占用8个bit

## 四维电子



Υ	Y	Y	Y	Y	Y	Y	Y	Y	Y.	Y	Y		Y	Y	Y	Y
γ	Y	Y	Y	γ	Y	Y	γ	Y	Y	Y	γ	1	γ	Y	Y	γ
Y	Y	Y	Y	Υ	Y	Y	Υ	Y	Y	Y	Υ	]	Y	Y	Y	Υ
Y	Y	Y	Y	Υ	Y	Y	Υ	Y	Y	Υ	Υ	]	Y	Y	Y	Υ
γ	Y	Y	Y	γ	Y	Y	γ	Υ	Y	γ	γ	]	γ	Y	Y	γ
Υ	Y	Y	Y	Υ	Y	Y	Υ	Y	Y	Y	Υ		Υ	Y	Y	Υ
Y	Y	Y	Y	Υ	Y	Y	Y	Y	Y	Y	Y		Y	Y	Y	Y
γ	Y	Y	Y	Υ	Y	Y	Υ	Y	Y	γ	Υ		γ	Y	Y	γ
				•				•						·		
				:				:						:		
	<b>I</b>			·			<b>I</b>			1	<b>I</b>	7		<u>.</u>		
Ŷ	Y	Y	Y	Y	Y	Y	Y	Y	Y	Ŷ	Y		Ŷ	Y	Y	Y
Υ 	Y	Y	Y	Y	Y V	Y	Y	Y N	Y	Y	Y		Y	Ľ.	Y	Y
<u>Y</u>	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y	Y	Y
Ŷ	Y	Y	Y	Υ	Υ	Y	Υ	Y	Y	Ϋ́	Y	••••••	Y	Y	Y	Y
U	U	υ	U	υ	υ	]			.U	υ						
υ	υ	U	υ	υ	U	]			<u>u</u>	υ						
U	U	U	U	υ	U	<u> </u>			U	U						
υ	υ	U	U	υ	U	]			U	U						
				:												
				•												
				İn		1					1					
U U	U	u	u	U	U	1			u	u						
-	1		T	1-		1	• • • •		-		ן ר					
V	۷.	<b>Y</b>	Y	V.	Υ.		• • • •	• • • •	<u>Y</u>	V	-					
V	V	V	V	V	V				<u>v</u>	V	-					
V	V	V	V	٧	V	<b> </b>	• • • •	• • • •	<u>v</u>	V	-					
V	۷	V	۷	۷	V	]		• • • •	٧.	۷						

#### 六. 操作步骤

- 1. 安装 Elecard StreamEye Tools 2.9.1.70328.exe,中间包含了播放编码后文件格式的驱动和工具
- 2. 连接好 BF533 板卡和仿真器。
- 3. 连接好 BF533 板卡电源和仿真器电源、仿真器 USB 电缆。
- 4. 打开 BF533 和仿真器电源。
- 5. 通过 VDSP5.0 打开 mp4enc\_DEMO 目录下的 mp4enc\_DEMO.dpj 工程文件。
- 6. 通过 VDSP 通过仿真器连接板卡。
- 7. Load 该工程的 DXE 文件。
- 8. 按 F5 运行代码。
- 9. 待执行完毕后,查看工程目录下是否存在 enc-test.m4v 文件。

www.openadsp.com 印印 研題社区



10. YUV 的文件可以用 ADViewer.exe 查看, ADViewer.exe 在打开 enc-demo-320x240.eyuv 文件 时的配置如下图。

- 11.具体的打开设置如下图。
- 12. 可以用 PC 机的播放软件(如: 暴风影音, 需要安装 Elecard StreamEye Tools 2.9.1.70328.exe) 打开 enc-test.m4v 文件, 看视频图像是否和原始视频图像一致, 如果是则表示 demo 代码运 行成功。

I YUV Parameters
Parameters         Directory         E:\encode\disk\mpeg4enc         File name         enc-demo-320x240.eyuv         Dimensions       Format         Width       320         Height       240         Format       Format of input file         Fames       YUV420         YUV422       Format of input file         Fames       YUV422         YUV422       Repeat         YUV444       YUV400 (monochrome)         RGB       MJPEG
OK Cancel

### 【联系我们】

- 联系人: 陈工
- 联系电话: 15011475977
- 电子邮箱: <u>sale@openadsp.com</u>
- 传真: 010-64811482