

## 基於晶心科技 Andes 平臺的 AAC 移植

晶心科技是亞洲首家提供原創性 32 位元微處理器 IP 的公司，並提供 32 位元處理器系統晶片設計平臺(Processor-based SoC Platforms)。隨著電子產業產品日趨多功能化，要求處理器及設計平臺有更佳的整合性、設計彈性，以及高效能、低功率。晶心科技以彈性配置平臺(Configurable Platforms)，推出了 Andes Core N9，N10，N12 三種系列的 32 位處理器軟硬核 IP，其中 N9 主要面向 MCU 市場。基於晶心處理器系統平台，晶心科技針對不同音訊應用為客戶提供完整的解決方案。其中包括將多種音訊格式的編、解碼器（開源），例如 MP3、AAC、WMA、G729 等移植到 Andes 平臺上。利用 nds32（基於 Andes Core）架構的優勢和針對音訊處理的擴展指令集，以及演算法上的改進，對這些轉碼器做了進一步的優化，使其不僅佔用的存儲空間較小而且具備了較高的運行性能。本文以 AAC 解碼器為例，介紹如何將 AAC 解碼器移植到 Andes 平臺。

### 1. AAC 簡介：

AAC 是 Advanced Audio Coding 的簡寫，是數位音訊的一種失真壓縮格式，由 [Fraunhofer IIS](#)，[AT&T Bell Laboratories](#)，[Dolby](#)，[Sony](#) 和 [Nokia](#) 等這些公司共同開發。像一些副檔名 m4a，m4b，m4p，m4v，m4r，mp4，aac，3gp 都是屬於 AAC 編碼的檔。

AAC 應用很廣泛，它是一些高階數位產品的預設支援音訊格式，如 Apple 公司的 iPhone，iPod，iPad，Sony 公司的 PlayStation 3，Android 作業系統的手機等。AAC 可以在比 MP3 檔案小 30% 的前提下提供更好的音質，被手機界評為“21 世紀資料壓縮方式”。

### 2. 環境及軟體介紹

#### 2.1 系統環境：

linux：fedora 8

原始程式碼：<http://sourceforge.net/projects/faac/>（當前版本 faad2-2.7.tar.bz2）

#### 2.2 開發環境及流程：

##### 2.2.1 開發環境

AndeSight v1.4。

AndeSight 是晶心科技提供的一套基於 nds32 架構開發嵌入式工程的圖形化的整合式開發環境。主要由 AndeSight IDE, AndESLive 和 nds32 工具集 3 個部分組成。

AndeSight IDE 為工程師提供了各種友好的介面，包括對嵌入式工程做編輯，編譯，運行，除錯或者評測等等操作。

AndESLive 提供了基於 nds32 架構的模擬器和一種圖形化的虛擬 SoC 構建模型，它與 AndeSight IDE 相結合為使用者提供了一個虛擬的硬體平臺。這個虛擬評估平臺提供 Andes 自行定義 ISA 的多組系列 32 位元微處理器 IP 以及各種週邊設備 IP，並且支援使用者自訂 IP 模型。

AndESLive 配合 AndeSight IDE 不僅使得 SoC 設計者能在計畫初期就開始軟體設計、偵錯、最優化等工作，並對系統架構及功能進行檢驗，而且使硬體工程師和軟體工程師具有一樣的能力去定制和修改他們各自的系統模型，可以有效的控制 NRE(NonRecurring Engineering)成本，讓軟體工程師在拿到硬體原型之前，即可以進行軟體的開發和優化。

nds32 工具集提供了一套在 Andes 平臺上開發嵌入式應用程式所需要的標準工具，例如交叉編譯器、交叉偵錯器、交叉連結器等等。nds32 工具集中對應不同的 Andes Core 型號，不同的系統函式程式庫以及大小端形式等條件，提供了各種對應的工具集。

##### 2.2.2 開發流程

利用 Andes 提供的上述工具，可以方便的進行開發，下圖是 Andes 平臺開發流程。

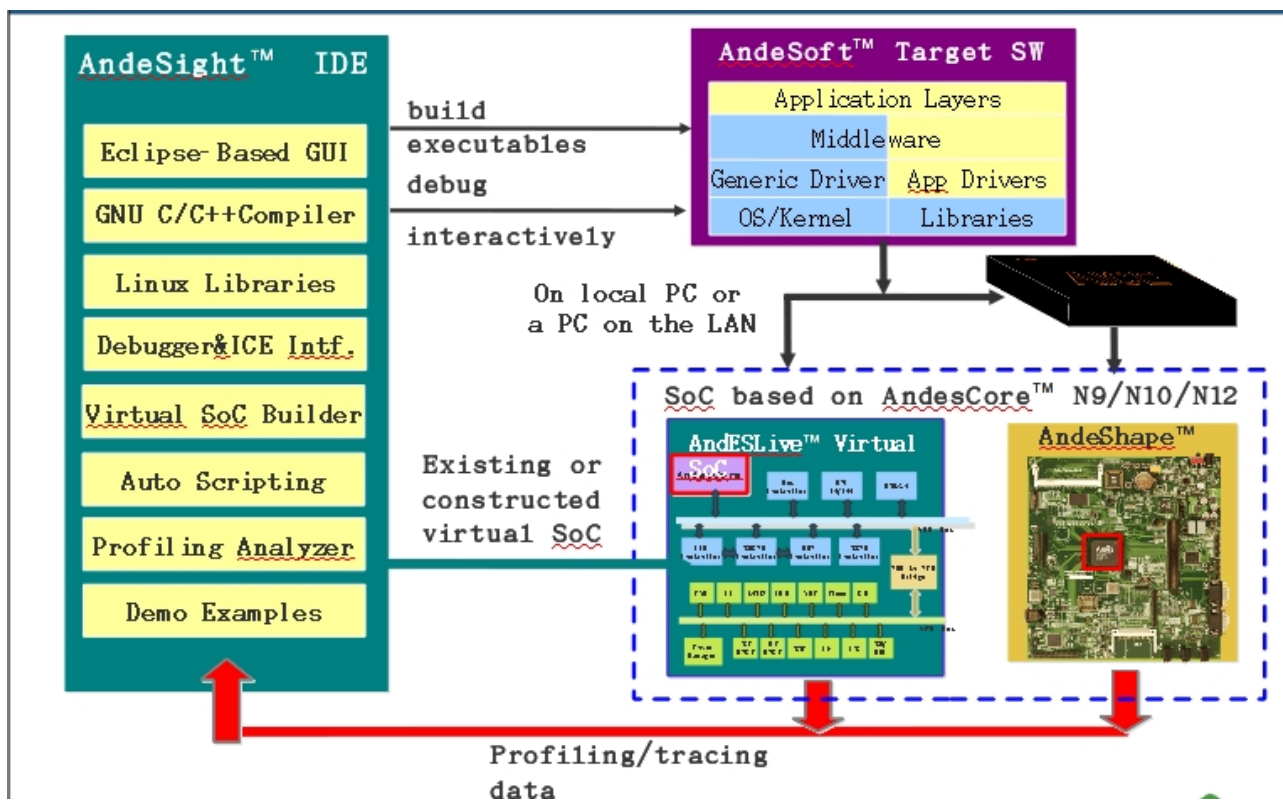


圖 1 Andes 平臺開發流程

### 3： 操作模式

Andes 平臺可以通過命令列模式操作，同時也可以通過 AndeSight v1.4 為用戶提供的整合式開發環境（IDE）模式，以方便使用者的開發。

#### 3.1: 命令列操作模式

##### 3.1.1： 操作過程

1)： 在命令列模式下，進入到你下載的 faad2-2.7.tar.bz2 的目錄，解壓這個檔案夾：tar -jxvf faad2-2.7.tar.bz2

2)： 進入解壓好的檔案目錄裡面，如目錄為 faad2-2.7,則 cd faad2-2.7

3)： 在正式執行 configure 這個命令之前，先 autoconf 一下，執行 autoconf -vif 即可

4)： 接下來要執行 ./configure。預設情況下，它是將檔案安裝在 /usr/local 下面。如果你想安裝到其它地方,如/opt 下面，你只需要在 ./configure --prefix="/opt", 可以用 ./configure --help 查看其它的設置，我們選擇默認形式。

5)： 編譯其中的一個檔案 config.sub（用 vi 或 gedit 等），這是一個 script 檔找到其中的一個 case 語句，是配置 basic\_machine 的，加入  
nds32\*)

```
basic_machine=nds32-unknown
os=linux
;;
```

這樣的 case 語句，然後保存，退出。

6)： 在當前的 shell 裡設定一個環境變數，export NDS32\_GCC\_CFLAGS=-static，該變數是 Andes 平臺編譯時需要的一個參數。

7)： 添加路徑，在 ~/.bash\_profile 中添加你的 AndeSight1.4 所在的路徑到你 PATH 變數中去，PATH="You AndeSight install path": \$PATH, 再 source 一下這個檔。

8)： 配置交叉編譯環境 ./configure --prefix="/usr/local" --host=nds32le-elf --disable-shared

```
--enable-static --without-pic --with-mp4v2 CC=nds32le-elf-gcc CFLAGS=-static LDFLAGS=-static  
CXX=nds32le-elf-g++ CXXFLAGS=-static
```

9) : make

10) : make install。這樣 faad 的解碼器將安裝你/usr/local/下面了，或者在由-prefix 指定的路徑裡。

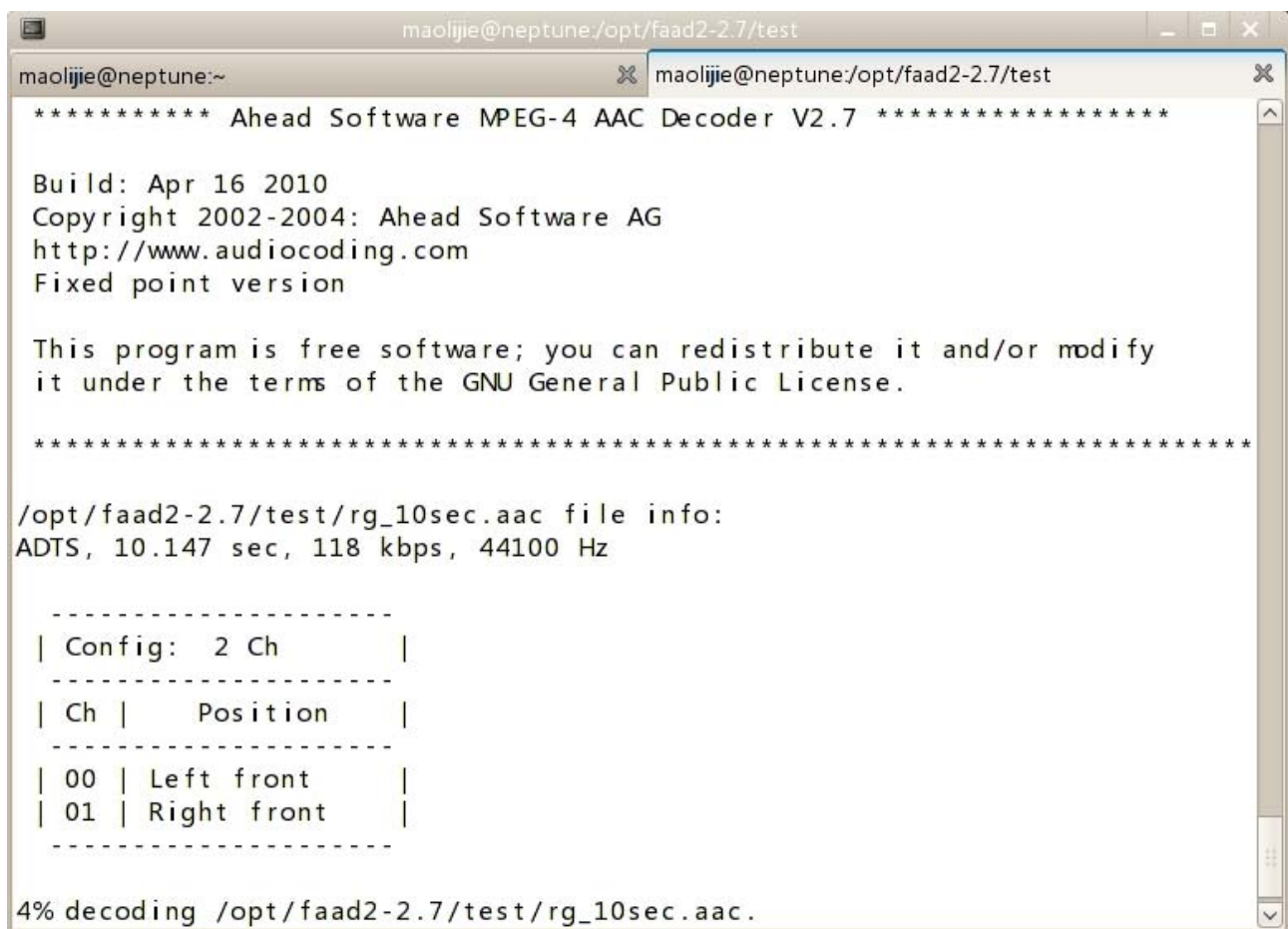
11) : 這時 faad 已經在 Andes 平臺上安裝好了。但是要讓它在我們在平臺運行。還需要借助於模擬器，我們通常稱之為 sid。這個 sid 需要配置一些參數，如：選擇什麼樣的 cpu, 選擇什麼樣的指令集等等，這裡我們把這些配置都寫檔案裡，當執行模擬器時，為了執行的方便，只需要用一個指令檔(script file)，將相關的配置添加進去就行了。在這裡我們在配置文檔--sid-conf 裡加入

```
set cpu-loader file "faad"
```

這樣的設置

12) : 配置完成。運行我們的 sid 腳本就能得到我 wav 檔。

13) : 解碼時，產生的圖形如下：



```
maolijie@neptune:/opt/faad2-2.7/test  
***** Ahead Software MPEG-4 AAC Decoder V2.7 *****  
  
Build: Apr 16 2010  
Copyright 2002-2004: Ahead Software AG  
http://www.audiocoding.com  
Fixed point version  
  
This program is free software; you can redistribute it and/or modify  
it under the terms of the GNU General Public License.  
  
*****  
  
/opt/faad2-2.7/test/rg_10sec.aac file info:  
ADTS, 10.147 sec, 118 kbps, 44100 Hz  
  
-----  
| Config:  2 Ch          |  
-----  
| Ch |      Position      |  
-----  
| 00 | Left front         |  
| 01 | Right front        |  
-----  
  
4% decoding /opt/faad2-2.7/test/rg_10sec.aac.
```

圖 2 命令列模式產生結果

14) : 我們完成了將 aac, mp4 等格式的檔解碼成為 wav 等格式的檔，你想要聽到聲音，需要用如 mplayer 這樣的播放機播放出來。

### 3.2 : GUI 介面操作模式

我們用 Andes 平臺最新的整合式開發環境(IDE)AndeSight1.4。

#### 3.2.1 : 操作過程

1) : 重複第一部分命令列操作模式的 1 -> 8 步驟，目的是產生 makefile 檔。

2) : 建立一個標準 STD C++ project 命名為 faad-static。

3) : 設置環境變數,從 Properties → c/c++ Make Project → Enviroment 中添加 :  
NDS32\_GCC\_CFLAGS 值為-static。

4) : 建立編譯目標過程,從 Creat a new Make target 中建立 : all , install。

5) : build all , built install。這和命令列模式 make 和 make install 有一樣的效果。

6) : 建立一個虛擬運行平臺 VEP,也就是在 AndeSight 中通過 AndESLive 提供的虛擬 SoC 平臺進行模擬,從 Target-> Fork VEP Target 我們選擇 Andes-demo,vep。

7) : Run as →Program 欄中的 C/C++ Application 裡添加我們生成的 faad 檔  
/usr/local/bin/faad

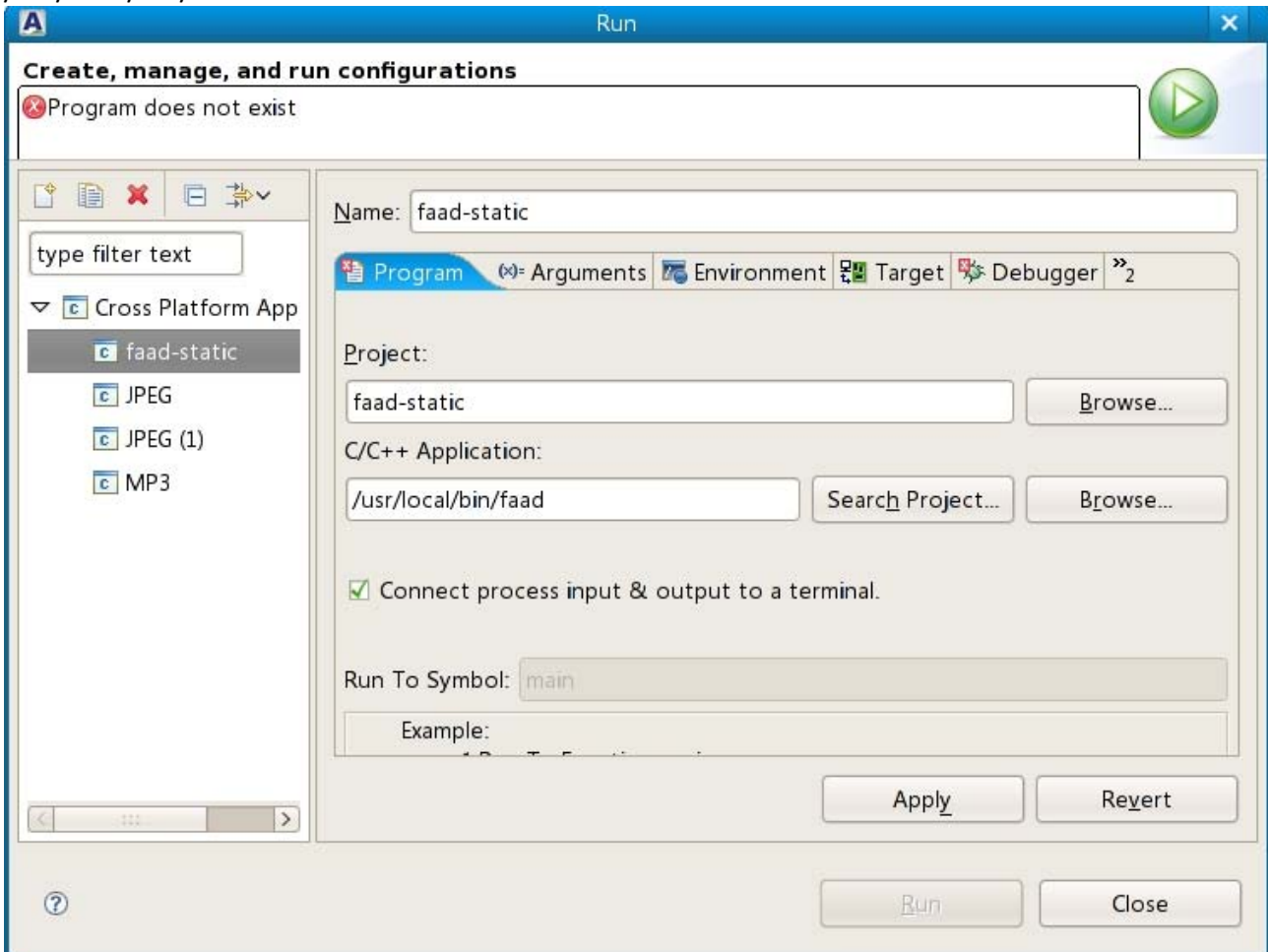


圖 3 添加執行檔

8) : 在 Arguments 欄為這個可執行檔加上適當的參數,如輸入輸出檔案名和路徑等。

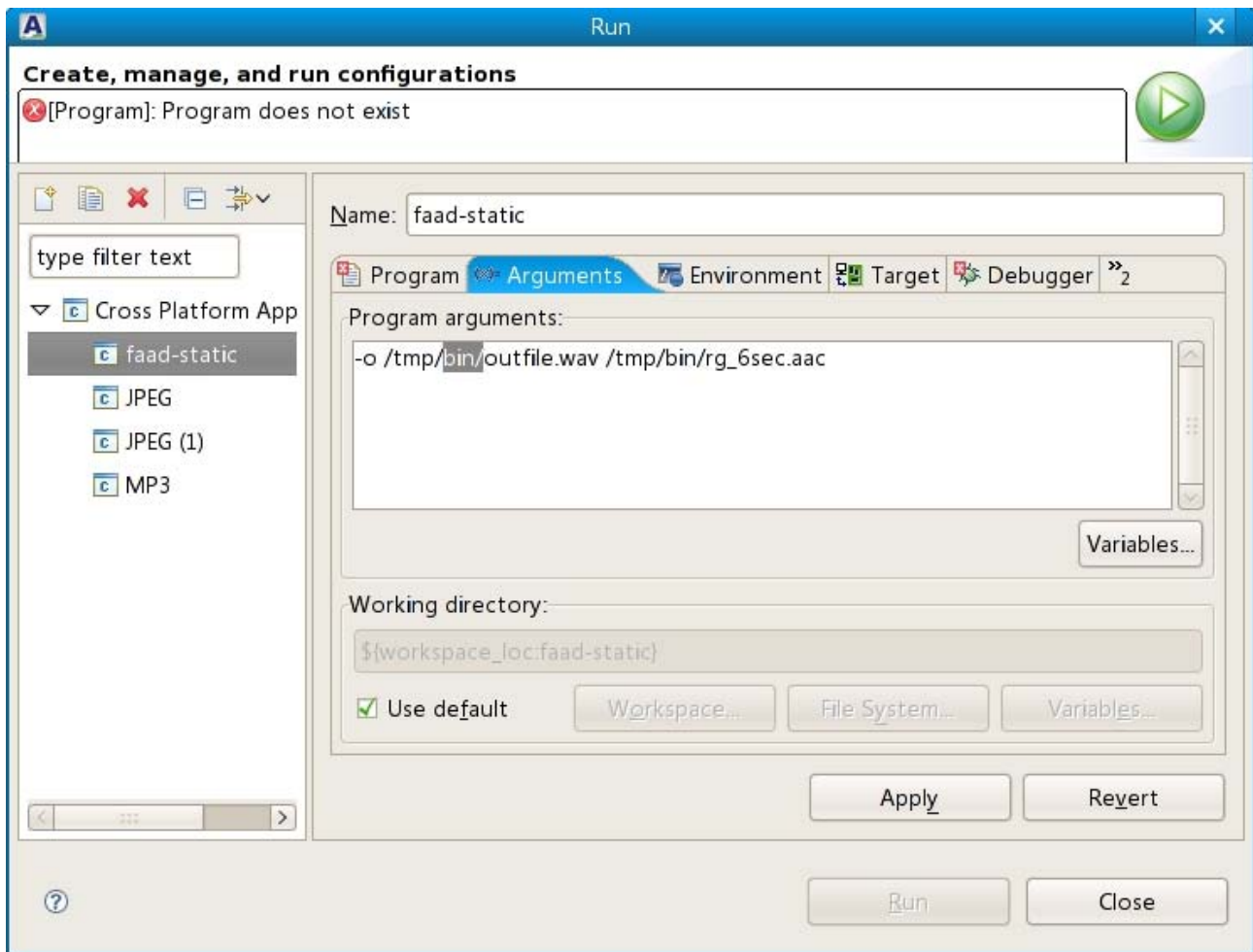


圖 4 添加運行參數

9) : 點擊 Apply , 程式就開始運行了。

10) : 運行的過程如下圖



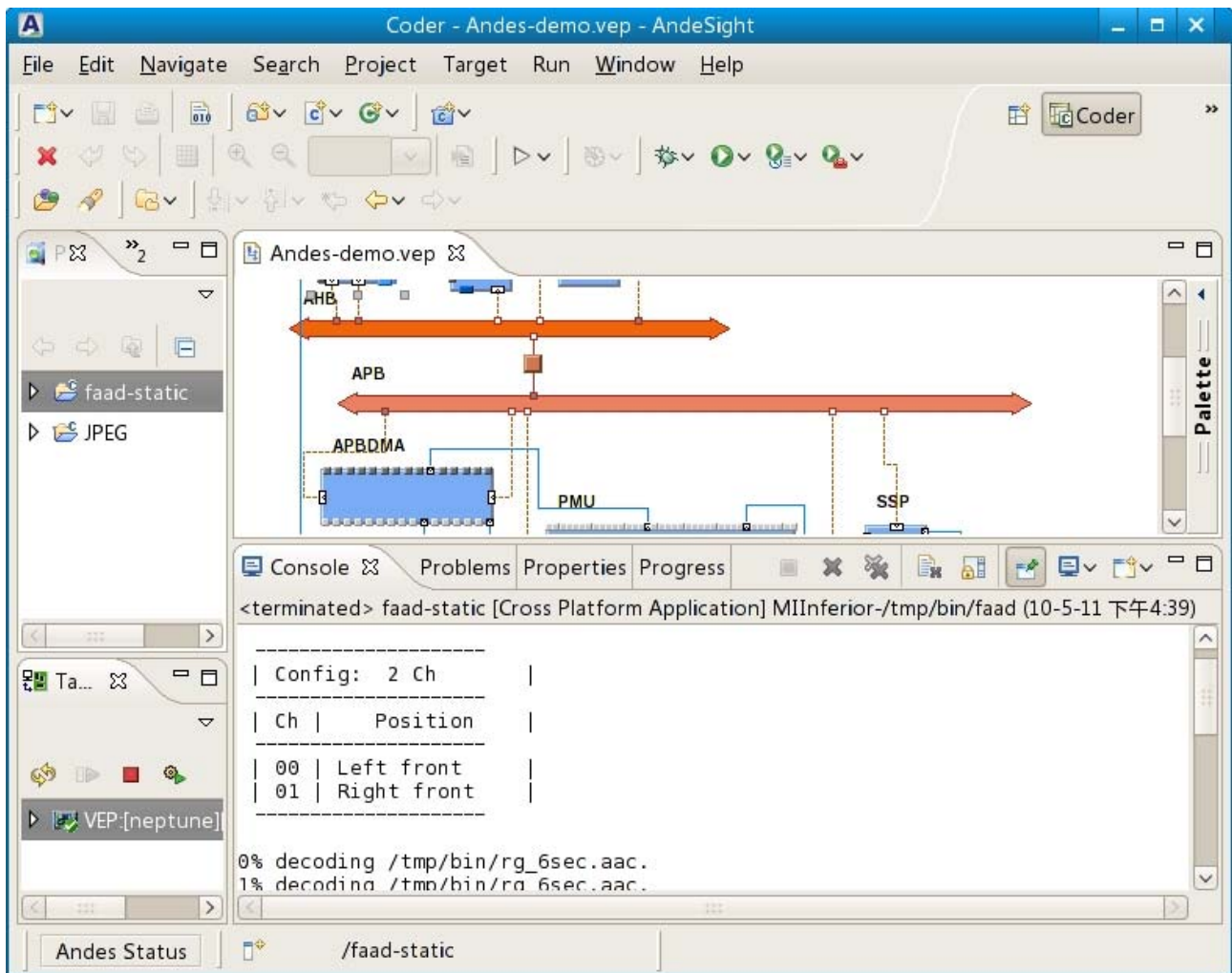


圖 5 IDE 模式運行結果

11) : 在我們第 8) 步中設置的檔參數路徑中，就得到了想要的 .wav 文件。该 wav 文件可以用 mplayer 进行播放,这就验证了 faad 的移植是正确的。

參考：

1：晶心科技網站：[www.andestech.com](http://www.andestech.com)

2：維基百科：[http://en.wikipedia.org/wiki/Advanced\\_Audio\\_Coding](http://en.wikipedia.org/wiki/Advanced_Audio_Coding)