The Design and Implementation of Electronic Program Guide In Digital TV Set-Top-Box

[a1] Yu-bao Cui, [b2] Li Zhao, [a3] Li-yuan Liu, [a3] Hua-ling Chen

[a] North China Institute of Aerospace Engineering Computer Science Department Langfang, China
[b] Hebei Technical College of Petroleum Profession Computer Science Department Langfang, China

Abstract

EPG system is very important for Set-Top-Box of digital TV. At first, this article briefly introduces the business information standard of digital TV. Then, the paper mainly discusses the system structure of EPG in digital TV set-top-box. Three main layers of this struture, including Application layer, realization layer and driver layer, are described one by one. Most important is that the design and implementation of EPG management module has been discussed, especially management of program subscription and management of Timer.

Index Terms: Set-top-box;demultiplexing; EPG; program management

1. Introduction

In term of the mainstream development trend for current digital TV, high-definition TV has already become one of the major technical orientations. To explore quality digital TV receiver is the key to ensure high definition, sound audio and visual effects as well as good services. This paper examines the design and implementation of digital TV set-top box EPG receiver system, which is based on Linux embedded operation system and ST7109 platform that supports high-definition digital TV decoding. It aims to provide users with convenient and swift means of high definition TV. Being an important component of digital TV data application system, EPG offers nice Graphics User Interface (GUI). Through EPG, users can freely and effectively choose and prescribe any TV programs they like. Therefore, the design of EPG and its functions and GUI determine whether the receiver will be accepted or not.

2. Business information standard of digital TV

Choosing Digital TV programs differs from the traditional analog TV, in that for traditional ones, each chancel corresponds to a program, users can watch program so long as it is modulated to an appropriate frequency, while the digital ones adopts digital signal modulation, so several sets of programs can be used at a

* Corresponding author.
E-mail address: [1] cuiyubao@163.com, [2] zhaoli_lf@126.com,[3]liuliyuanlf@163.com

frequency point. Viewing options also changes, if one wants to watch certain program, it is necessary to modulate the digital signal at first, then through signal demutiplexing process to extract the compressed program from transport stream, and decode at last. In MPEG-2 Transport Stream (TS), Program Specific Information (PSI) regulates different programs and the process of multiplexing into single TS from various components. PSI provides components physical information of the current stream, and Service Information (SI)[1-4] in DVB further expands it, adding useful service information such as type of program, service providers, and interactive relations among programs.

A.   MPEG-2 standard

In the MPEG-2 standard, in order to make one or more audio, video or other types of basic data stream into single or multiple data streams so as to store and transfer, it is required to pack once again and recode, at the same time, various time markers and system control information should be inserted into code stream and sent to the channel coding and modulation at last. In this way, two types of data stream has been formed, namely, transport stream and program stream. Transport stream, TS for short, consists one or several set of programs, each program has a or several original streams, combined with other streams like streams of video, audio, PSI and other data packets. TS can be widely used in storage of audio& video data, non-linear editing system of TV programs and related networks.

Defined in MPEG-2, PSI is used to describe construct structure of TS and plays a vital role. PSI information is made up of 5 types of tables, Program Association Table (PAT), Conditional Access Table (CAT), Program Map Table (PMT), Network Information Table (NIT) and describer.

B.   Digital video broadcasting SI standard

Being an important part of digital video broadcasting code stream, SI data can help users select information like program rating, program profile, time-lapse description, etc, and it is the basis to form Electronic Program Guide (EPG). In this standard, SI also provides other multiplexed stream services and event information. All these data is composed of 9 tables, namely, business Group Association Table (BAT), Service Description Table (SDT), Event Information Table (EIT), Running Status Table (RST), Time and Date Table (TDT), Time Offset Table (TOT), Stuff Table (ST), Select Information Table (SIT), and Discontinuous Iinformation Table (DIT) [5].

3. Software system structure

Based on Linux operating system, this system's major functions include EPG, program management, program searching, system settings. The whole system is divided into 3 layers, namely, application layer, realization layer and driver layer, as shown in the following figure. Lower layer supports upper layer, arrows indicate supporting relationship among them. The following paper will explain for each layer from the bottom up to the top [6].
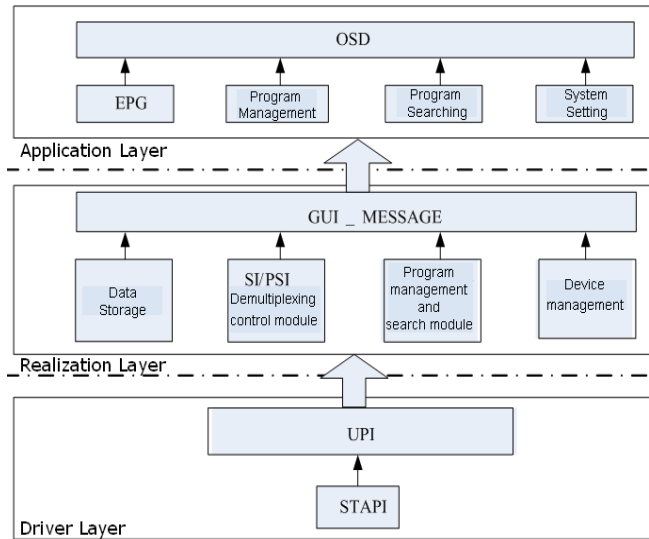
Fig. System Hierarchy

A. *Driver layer*

It can be seen from Fig. that OSD lies at the top application layer, it is the concrete implementation of the interface, and it needs support from the bottom layer to realize functions. To obtain required data, it is must to demultiplex TS stream to extract related information from PSI/SI tables. And it is necessary to encapsulate PTI application interface of STAPI, and this encapsulated layer is called driver layer, which provides unified interface for demultiplexing. When platform is changed, only one layer needs to be modified, which greatly enhance the portability of a project. Similarly, program switching, modifying and storing of program information and system information also need the support of STAPI at the bottom layer, hence, it is necessary to encapsulate other drivers, for instance, video decoder, audio decoder, Eeprom, flash, and so on. Therefore, STAPI has a unified encapsulation, UPI for short.

B. *The implementation layer*

This layer provides upper layers with support, including data storage, demultiplexing control module, program management and search module, device management, and the interface among application layers.
(1) Data storage
After completion of system settings, data will be stored in FLASH or EEPROM.
(2) Demultiplexing control module
The main task in the process of de-multiplexing is to control PTI. Through packing some PIT interfaces to control PTI. This paper gives its main focus on PSI/SI demultiplexing and parsing and it will not discuss STAPI of audio and video. Demultiplexing module implements Demultiplexing control, and parsing module carries out parsing. Demultiplexing module and parsing module combine into one module because of the close relationship. Parsing module can be implemented by a callback function, in this process, PSI/SI data is acquired and processed, available for upper application.
(3) Program management and search module
This layer realizes programs' management and searching and responds to the operation of OSD. For example, when after delete operation, relative delete interfaces of this layer will be called. Program information like the storage and revision also needs this layer to manage. As for program searching, this layer provides relative

interfaces to complete task. Besides, the system setting module is responsible for the maintenance of system information, while device management module is responsible for the management of decoding of audio and video.

(4) Device management

This layer completes analysis and decomposition tasks of audio and video information.

(5) UGI_MESSAGE interface

GUI_MESSAGE interface provides a universal interface, which is easy to add on, manage and expand. It is responsible for transferring information to each module which creates its own information receiving task. And information will be received and processed by processing function. Messages are defined as follows:

```
typedef struct _GUI_MESSAGE
{
GUI_Module                  Module;
GUI_CommandType        CommandType;
GUI_INT32                   Command;
void                              *DataPtr;
void                              *FuncPtr;
}APP_MESSAGE
```

Module is used to define types of information, for example, if EPG information is the one to be extracted, the information type should be set as EPG_TYPE [7]. If the type for program management is PRO_TYPE, then the system information should be set as SYS_TYPE. Each module complete different tasks, so different function information should be distributed and and applied accordingly.

CommandType is introduced to indicate the flow or performing of the data. Here the performance refers to data reading and writing, etc. In this way the transmission direction can be controlled.

Command is defined as what data to access.

DataPtr data storage address

FuncPtr data callback handler is used to call for the OSD, when data accessed needs to be displayed in OSD. OSD only needs to register relative display interfaces of FuncPtr.

Through this interface, OSD can distinguish itself form other data processing partition, which is conducive to maintain, expand and transplant of project.

C.   Application layer

This layer provides user interfaces, including four modules, namely, EPG, program management, program searching and system setting. OSD is provided in the form of menu to users, and users can select certain function. The part will be examined in the following paper [8].

4. Design and implementation of the system

In this section, EPG management module will be discussed.

A.   Display form of program information

The way to present program information to users is the external manifestation of the system. EPG interface present information via two lists, one is program name list, the other is EIT information list. Program name list shows program names and service id through list control, when the program name is a lit long, letters with their focus on item will be shown in a roll way. As for those program information shown on OSD layer and not necessary to care about how to obtain the information and its current display, it is necessary to provide a drawing tool and a small window which can play video in EPG interface. EIT information list shows contents in the table. When programs are switched, needs to be shown simultaneously with program list so as to be seen when users

look through. At the same time, program EIT also needs to be updated into the information list of EIT [9].

B.   Management of program subscription

As for EPG system, the function of program subscription is a good application. It can be divided into four management modules: program information storage, the creation of Timer, information acquisition of subscription program, and responding processing when there is no subscription program information.

In order to manage scheduled program, a special gadget is defined, which can store play time and order of play, as well as callback function. It is defined as follows:

typedef void (*TimerNotify)(unsigned short*);//type of callback function
typedef struct   PURCHASE_PROGRAM_T{
CC_TIME   time;
unsigned short   Parameter[TimerMAXpara];
TimerNotify   TimerFunc;
}Purchase_Program_t, *Purchase_Program_p;

Time: is used to store play time, accurate to second level.

TimerFunc: is used to store registered callback function, in this way, it is easy to expand scheduled capacities.

Parameter: is used to store program information, and it is only necessary to store the play number, which corresponds to a single program.

When computer is booted up, the system will automatically receive TDT list to initialize system clock. After initialization, information will be read from flash into memory, and check whether there is any program that expires during this process. If do exist, the overdue program will be deleted. As for those almost due programs, registered callback function needs to be booted up, which passes prescribed program information to Timer and restore in Flash after checking. When TV is on, the new programs added by users will be examined to see whether it is overdue, if so, users will be warned not able to subscribe. Then it will check whether the number of subscribed programs are up to the top amount, if so, users will be noted not able to further subscribe, if not, subscribed programs will be registered to Timer.

C.   Management of Timer

In the previous structure of program subscription, the callback function of Timer will start scheduled play function after detecting that time is close to, so the Timer also needs subscription information like time, therefore, a timer structure is needed to define as follows:

typedef struct APP_TIMER_T{
CC_TIME   time;
TimerNotify   TimerFunc;
unsigned short   TimerPara[TimerMAXpara];
unsigned char   timerInuse;
}APP_Timer_t, *APP_Timer_p;

Time: is to store the scheduled program time;

TimerFunc: is to subscribe callback function. Timer needs to be registered either for program subscription oradd-on;

TimerPara: subscribe other program information like program number;

timerInuse: is the symbol to check whether Timer is used or not.

The creation of Timer is managed through a task that includes such operations as add Timer operation, delete Timer, update Timer, and Timer task processing.

5. Conclusion

EPG(Electronic program guide) to provide users with an easy to use, friendly interface, you can quickly access a program, users can also watch one or more of the functional channels, or even all channels will broadcast the show recently. At the same time, EPG can provide classification, can help users browse and select the types of programs. Good design of EPG management module can improve the function of digital calbe TV. Meanwhile, in order to have a better performance, EPG shoud be designed and implemented with a friendly GUI. That is the direction of our effort now and in the future.

References

[1]   DVB. EN 300-468 Services Information Criterion of System (DVB-SI) [S]. 2000.
[2]   DVB. EN 300-472 Tranmission and Criterion of graphics   & word in System Byte Stream (DVB-TXT) [S]. 2000.
[3]   DVB. EN 301-192 DVB Criterion of Data Broadcasting (DVB-DATA), 2000.
[4]   DVB. EN 50083-9 Professional Devices and Interfaces of   CATV/SMATV and DVB/MPEG-2 (DVB-PI), 2002.
[5]   R.S.Sherratt. Performance and conformance results of the deterministic DVB-Tequalizer. IEEE Transactions on Consumer Electronics[J], 2004, 50(1): 95~99.
[6]   DVB. ETSI EN 300-708 V1.2.1 Television systems, Data transmission within Teletex, 2003.
[7]   Li-jin Xu. SI of DVB and Application of EPG. Wired TV Technology(in Chinese). 2005, 2(10): 76~79.
[8]   Xue-ling Zhu. MIDDLEWARE BASED ON MESSAGE QUEUE APPLIED IN IPTV(in Chinese): Beijing University of Post and Telecommunications Thesis of Master Degree, 2007.
[9]   Han-guang Dan, Xian-min Zhang, THE DESIGN OF ELECTRONIC PROGRAM GUIDE IN EMBEDDED SYSTEM [J] (in Chinese). Computer Application and Software, 2005, 9(4): 34~35.