

Standard for RealTek DVD Recordable

Date: 7/13/2011

DVD-VENUS System Software File Structure

RealTek specification on DVD Recordable Technology

Specification for DVD-VENUS: Venus System Software File Structure

Note

This document is subject to review.

Yuantao Wu
y_wu@realcomtec.com

DVD-Venus Specification

Version	Date	Author
0.0.1	7/19/2004	Yuantao Wu

Change List	Version	Description
--------------------	----------------	--------------------

Table of Contents

Table of Contents	ii
1 List of Figures	ii
2 Introduction	2
2.1 Purpose.....	2
2.2 Scope.....	2
2.3 Glossary	2
2.4 References	2
2.5 Overview of Document	2
3 Visualize Software Components in the System CPU.....	2
4 Visualize Software Components Under Subversion	2
5 File Structure Design Rules.....	2
5.1 Includes	2
5.2 Lib	2
5.3 Document	2
5.4 Src	2

1 List of Figures

Figure 1 Software Components in System CPU..... 2

Figure 2 Software components under Subversion for the system CPU.....4

2 Introduction

2.1 Purpose

This document describes the file structure under Subversion for the Venus DVR (digital video recorder) project. It helps the individual program developers to design and organize their codes systematically.

2.2 Scope

It covers most of the software components running on the system CPU inside the DVR system. For those modules running on the Audio/Video CPUs, separate documents will be available.

2.3 Glossary

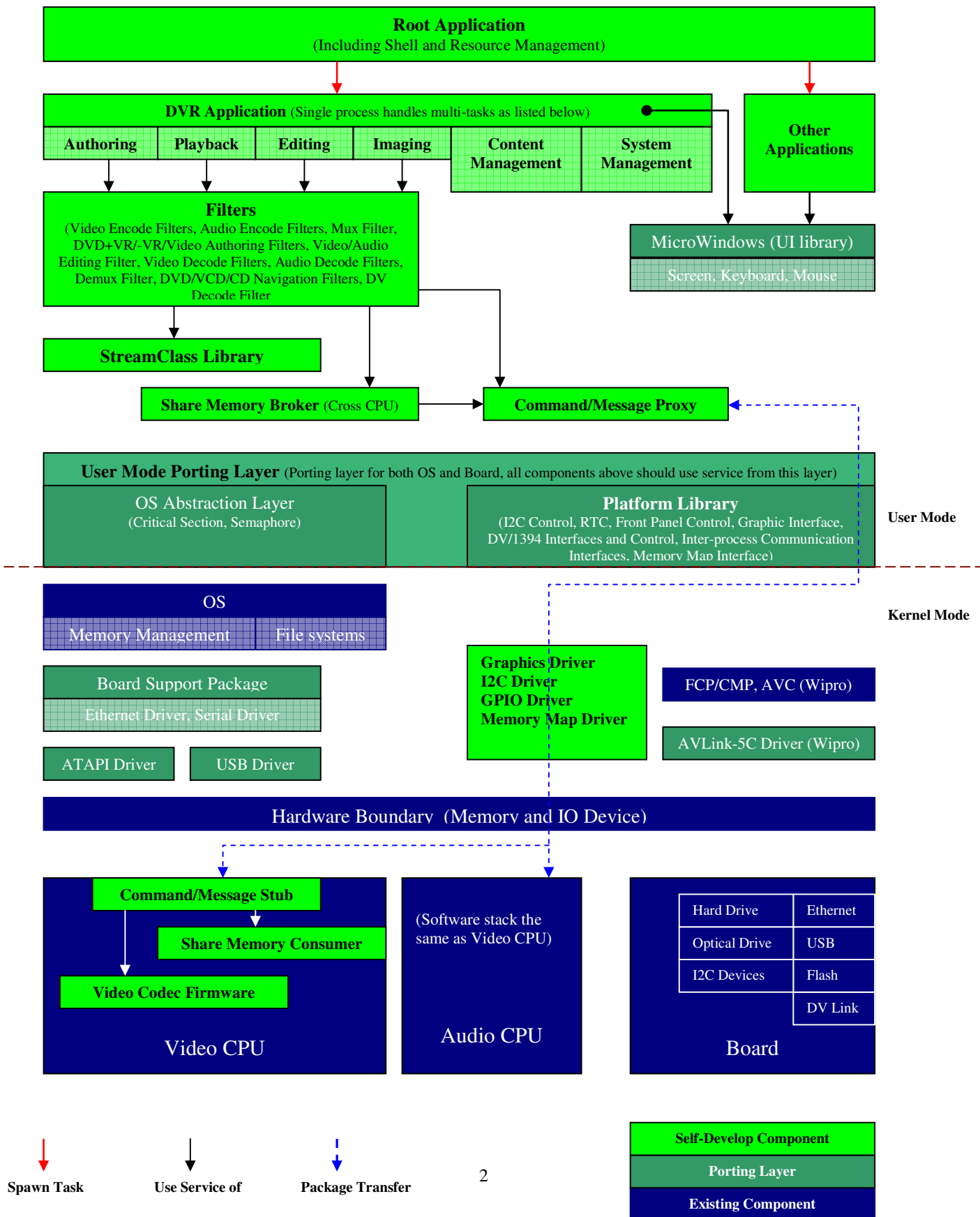
- StreamClass: A media streaming architecture bases on filter connection. See [Stream Class](#) Specification.
- Flow: A collection of Filters and Pins that can perform a media-streaming task. For example, DVD playback flow.
- OSAL: Operating System Abstraction Layer.

2.4 References

- [Operating System Abstraction Layer \(OSAL\)](#)
- [Stream Class Specification](#)
-

2.5 Overview of Document

3 Visualize Software Components in the System CPU



4 Visualize Software Components Under Subversion

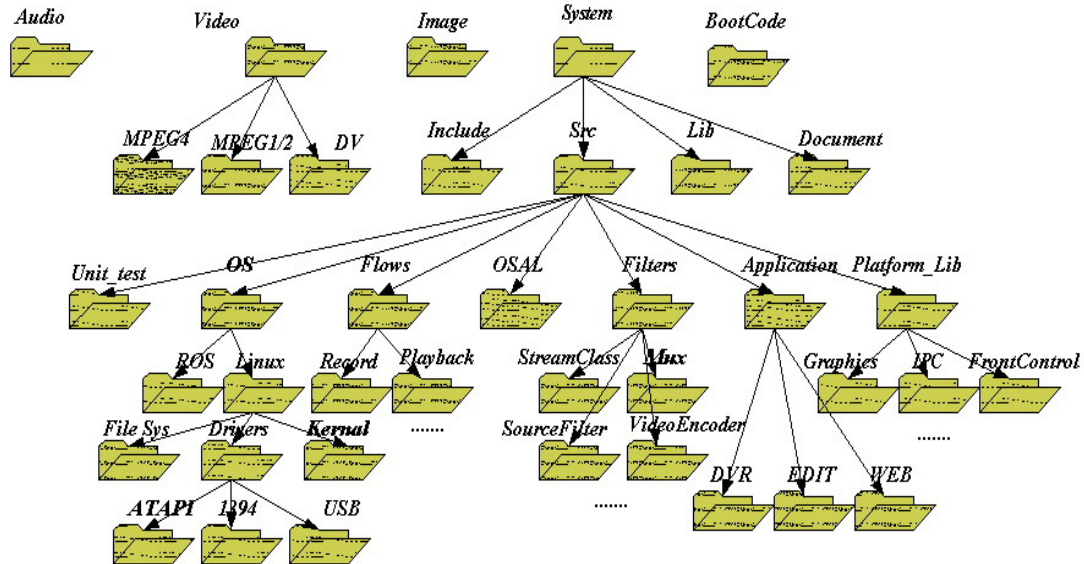


Figure2 Software components under Subversion for the system CPU

5 File Structure Design Rules

VENUS DVR system is a very complicated software system, involving many modules developed by different individuals. A well-structured file system, which reflects the completed functionality defined in the previous chapter, will lighten the developers' work and shorten their learning cycles significantly. On the other hand, it is quite dangerous and painful if the file structure is modified in the later develop stage. Therefore, a clean file structure should be defined and followed as early as possible.

There are three CPUs (Audio, Video and System) running on the VENUS with different operating systems. Different codes are developed separately for those CPUs. Five directories under root directory(//trunk) are created accordingly: one for each CPU, one for the boot up code(/BootCode) and one for the final executable image(/Image). Different file organization may be applied for each CPU. For example, video directory can be further divided into three sub-directories based on the different compression standard: MPEG1/2, MPEG4 and DV. However, this document only covers the rules for the modules that will be running with the system CPU.

There will be five sub-directories under System: Include, Src, Lib, Exe, Document.

5.1 Includes

There will be many header files for the VENUS system and located everywhere. It may become programmer's headache to find the correct header files that need to be included,

especially for the third party developer in the future. Therefore, the ***Includes*** directory will only have several “big” header files, serving as a container that contains the completed header files for the whole system. The purpose of directory is to simplify the developer’s work. For example, StramClass.h may exist, which includes all the header files located under /src/filters/StreamClass for the above purpose.

5.2 Lib

An empty directory which only holds the library that is dynamically generated during the compiling time.

5.3 Document

All the documents related to the system software should be placed here. Since they are normally saved under /trunk/Spec, soft links will be generated in this directory.

5.4 Src

All the source files for the modules running on the system CPU are located here. Based on their functionality, /OS, /OSAL, /Flows, /Filters, /Application, /Platform_lib, and /Unit_test are created under /src. Some new directories may be added in the future if necessary. The definition for the most of the directories is straight forward. Only Unit_test is explained here.

The purpose of the Unit_test directory is to hold the test code for one or several modules during the developing phase. For example, a DVDProgMux directory may exist under unit_test to hold the test C++ files and data files for the DVD Program Mux Filter.