08-6280-RN-ZCH66

MAR 18, 2009

5.1

Release Notes for WMV9 Decoder on ARM11 ELINUX

ABSTRACT:

Release Notes for WMV9 Decoder on ARM11 ELINUX

KEYWORDS:

Multimedia codecs, WMV9, Windows Media Video 9

APPROVED:

Wang Zening

Revision History

VERSION	DATE	AUTHOR	CHANGE DESCRIPTION
1.0	06-May-2005	Prachi/Anurag	Release notes template
2.0	26-May-2005	Prachi	Release
3.0	16-Sep-2005	Puneet Gulati	Build Procedure changes for RVDS2.2
4.0	06-Feb-2006	Lauren Post	Using new format
4.1	31-Mar-2006	Prachi	Updated
4.2	24-June-2006	Sriram	Updated
4.3	13- Feb- 2007	Abhishek Mehrotra	Updated
5.0	06-Dec-2007	Ellick Chen	Update for MAD release
5.1	18-Mar-2009	Eagle Zhou	Add support of device mxc_mem

Table of Contents

Introduction

1.1 Purpose

The purpose of this document is to provide information on the package contents, instructions on building library and test applications and test execution on ARM11 ELINUX.

1.2 Scope

The scope is restricted to information on the package contents and instructions for building and testing. This document does not provide architecture or details about the APIs provided in the package. Performance data will be provided in another document as detailed in the Requirements Book.

1.3 Audience Description

The reader is expected to have basic understanding of video processing and windows media video coding standard.

1.4 References

1.4.1 Standards

- WMV Version 9.0, Windows Media Video V9 Decoding Specification, revision 87
- WMT Version 9.0, Functional Specification, Recommended Media Decoding – rev 8.1.

1.4.2 References

- Arm codec coding guidelines
- Advanced System format (ASF) Specification, Revision 01.20.02, Microsoft Corporation, June 2004

1.4.3 Freescale Multimedia References

- WMV9 Decoder Requirements Book – wmv9_dec_reqb.doc
- WMV9 Decoder Test Plan – wmv9_dec_test_plan.doc
- WMV9 Decoder Release notes – wmv9_dec_release_notes.doc
- WMV9 Decoder Test Results – wmv9_dec_test_results.doc
- WMV9 Decoder Interface Header – wmv9mp_dec_api.h
- WMV9 Decoder Application Code – wmv9_testapp.c

1.5 Definitions, Acronyms, and Abbreviations

TERM/ACRONYM	DEFINITION
AVC	Advanced Video Coding
API	Application Programming Interface
ARM	Advanced RISC Machine
ASF	Advanced System Format,
FSL	Freescale
ISO	International Standards Organization
ITU	International Telecommunication Union
MPEG	Moving Pictures Expert Group
NAL	Network Abstraction Layer
RVDS	RealView Development Suite
SP	Simple Profile
RVDS	ARM RealView Development Suite
TBD	To Be Determined
UNIX	Linux PC x/86 C-reference binaries
WMV	Windows Media Video

1.6 Document Location

docs/wmv9mp_dec

2 Release History

RELEASE NUMBER	DELIVERABLES	FEATURES
1.0		• Engineering Release
2.0	• Documentation • Application Interface header file • ELINUX and RVDS libraries and test applications • UNIX/Linux x86 Reference library and test application • Makefiles and Source code for library and test application including optimized assembler for the ELINUX and RVDS libraries. • Test vectors	• Reference version of the library using only C source. • Assembly optimized code for ARM11. • Addition of debug logs. • Support to build decoder application for board. • Enhanced Application to display the decoded frames in LCD (SW color conversion) •
2.2	Same	• Shared Library Support • Upgrade to RVDS 2.2
3.0	• Documentation • Application Interface header file • ELINUX libraries and test applications Makefiles and Source code for library and test application including optimized assembler for the ELINUX libraries.	• Optimized library for Arm11 • Improved Error Handling/protection
5.1	• same	• Mxc_mem device support(please get more detail from BSP)

Table 1. Details of the Release

2.1 Assumptions and Known Problems

None

2.2 Contacts

Please report any problems, contact Freescale customer representative.

3 List of Deliverables

3.1 Documentation

Base directory: /docs

Subdirectory	Files
/WMV9mp_dec	wmv9_dec_api.doc wmv9_dec_reqb.doc wmv9_dec_test_plan.doc wmv9_dec_test_results.doc wmv9_dec _release_notes.doc

3.2 Public Headers

Base directory: /

Subdirectory	File
ghdr	wmv9mp_dec_api.h

3.3 Test Application Source

Base directory: /testvideo_testwmv9_dut

Subdirectory	Files
/	“Makefile” makefile for building ELINUX libraries.
/src	*.c, application code.
/hdr	*.h, application header files

3.4 Library Source

Base directory: /src/ WMV9MP_dec

Subdirectory	Files
/	Makefile “Makefile” for building ELINUX libraries.
/c_src	*.c, WMV9 decoder source code
/arm_asm	*.s WMV9 decoder assembly source
/hdr	*.h, WMV9 decoder library header files

3.5 Common Makefiles

Base Directory: / build

Makefile	Description

Makefile.init	This is a common makefile included in the codec library makefile for building the libraries. This file includes common options used by all codecs. Following flags can be overwritten or added to in the codec library makefile 1. Path to toolchain tools (TC_ROOT) 2. GNU header file path (HEADER_PATHS) 3. GNU library path (LIB_PATHS) 4. GNU Compiler/Assembler Options (GNU_CFLAGS, GNU_AFLAGS) 5. Endian Flags 6. Optimization Flags(OPTIM_LEVEL, OPTIM_TYPE) 7. Common options for ELINUX (CFLAGS,AFLAGS) 8. Build specific flags 9. Source directory of 'C' code 10. Source directory of 'assembly(.s)' code 11. Object directory for .o files 12. Codec header path 13. SHARED_ELINUX builds for libraries that must be linked using the toolchain because of external library includes.
common_testapp.mk	This is the common makefile included in the codec test makefile for building the test application. This file includes the common options used by the all the codecs. Following flags can be overwritten or added to in the codec test makefile 1. Toolchain path depending on the build option 2. Compiler Flags 3. Linker flags 4. Paths for c_source, exe and object directories 5. Codec header files' INCLUDES path 6. Endian Flags 7. CODEC_LIB generation

4 Software Setup & Tools used

- ARM RVDS 2.2 should be installed in the PC.
- Freescale Linux OS Release L26.1.16 must be running on the evaluation board (Argon+, MX31).
- Intel based Red Hat Linux Machine must have the Montavista toolchain installed on it.
  - `MontaVista 3.4.3-25.0.36.0501313 2005-08-21`
- 'Cygwin' **Version** CYGWIN_NT-5.1, a freely downloadable linux emulator is installed in PC - **http://www.cygwin.com/**.
- 'make' utility available for targeted platforms

5 Build Procedure

All the required makefiles are provided under individual directories. The library can be built for windows / target processor (ARM1136J-S). The details for the build procedure are described below. (Note that this supports WMV9 Main Profile on Little Endian mode only).

5.1 Library

To build the library, run 'make' on 'Makefile' from the library directory. The makefile shall create the required directory to hold the object files. The makefile can be used if you want to build the library only. The following options are available to build the library.

Options

a) **BUILD options**:

BUILD= ARM11ELINUX : This is the default option and builds both static library 'libwmv9mp_dec_MP_arm11_ELINUX.a' and shared library 'libwmv9mp_dec_MP_arm11_ELINUX.so', for testing on the board.
Note: make sure that FSL_API=1 is defined when making with this option.

Eg: make BUILD= ARM11ELINUX FSL_API=1

b) **clean options**:

- **clean**: Deletes all the object files and libraries.

c) **other options**:

- NO_KEV=1 This is used when output is not required. By default, kev is generated.

Note: Make appropriate changes in file 'Makefile.init' at directory 'build' for the location of toolchains.

The library that is built is saved as libwmv9mp_dec_MP_arm11_ELINUX.a and libwmv9mp_dec_MP_arm11_ELINUX.so for board build. These libraries are saved in the current directory (the same directory in which the source and assembly directories are listed).

Target	Compilation Environment	Build Options	Library name
Board	PC (Using Cygwin)	BUILD= ELINUX FSL_API=1	libwmv9mp_dec_MP_arm11_ELINUX.a libwmv9mp_dec_MP_arm11_ELINUX.so

5.2 Test Application

To build the test application, run 'make' on 'Makefile' from the test directory. This makefile can create executables for testing on board for ARM11. The executable wmv9mp_dec_MP_arm11_ELINUX for board is stored under the test/exe directory. The makefile shall create the required directory structure to hold the object files and executables. The following commands should be invoked so as to build the executables.

Options

1) **BUILD options**:
   - **BUILD=ARM11ELINUX FSL_API=1** : This is the default option and builds the executable 'wmv9mp_dec_MP_arm11_ELINUX', for the board.

     Eg: make BUILD=ARM11ELINUX FSL_API=1 (for board)

2) **LIBRARY options**:
   - **LIB_TYPE= STATIC**: This option builds the ELINUX test application linked with the ELINUX static library 'libwmv9mp_dec_MP_arm11_ELINUX.a'.If nothing is specified, the executable links with shared library 'libwmv9mp_dec_MP_arm11_ELINUX.so'

     Eg: make LIB_TYPE=STATIC

3) **clean options**:
   - **clean**: Deletes all the object files and executables.

4) **other options**:
   - DISPLAY_LINUX=1 Builds the application with display capabilities. If built without this option, display would not be enabled. This is supported only for board.
   - NO_KEV=1 This is used when output is not required. By default, kev is generated.

Note:

In 'common_testapp.mk' at directory 'ARM11/common', the paths for the compiling and linking tools are hard coded for the current set-up. These paths may not be the same in the user's directory set up. Hence, the 'common_testapp.mk' should be modified to point to the directories where the linking and compilation tools are present before building the application for board.

The following table summarises the build options,

Target	Compilation Environment	Build options	executable name
Board	Redhat Linux Machine	BUILD=ARM11ELINUX FSL_API=1 LIB_TYPE= STATIC	wmv9mp_dec_MP_arm11_ELINUX

6 Test Application Execution

To know the options provided by the test application, run the executable without any argument. It shall print a brief summary of all the options available.

a. -i <file_name> :input file name
b. –o <out fdir> :Output directory, for kev save only
c. –e <num frames> : numbers of frames to be decoded

6.1 ELINUX

wmv9mp_dec_MP_arm11_ELINUX –i <input vector>

7 Pre compilation Options

The following C options need to be set

C Defines	Description	Remarks
ADS	To run on RVDS. Disable to run on UNIX.	
ALL_PROFILE	doesn't use WMV9_SIMPLE_ONLY	
ARM11		defined for all builds
LITTLE_ENDIAN	To run the code as Little Endian.	
LOG_TIMING	Performance logging	
DISPLAY_LINUX	Used for display	
SAVE_KEV	Set when NO_KEV is not set	Default case – used for ELINUX builds only
USE_ASM		Used for all builds
USE_DISPLAY	Displays video output	
WMV9_SIMPLE_ONLY	Set when `ALL_PROFILE is not set`	