JPEG Encoder/Decoder User's Manual (Linux)

S3C6400/6410

August 29, 2008

REV 3.10

Important Notice

The information in this publication has been carefully checked and is believed to be entirely accurate at the time of publication. Samsung assumes no responsibility, however, for possible errors or omissions, or for any consequences resulting from the use of the information contained herein.

Samsung reserves the right to make changes in its products or product specifications with the intent to improve function or design at any time and without notice and is not required to update this documentation to reflect such changes.

This publication does not convey to a purchaser of semiconductor devices described herein any license under the patent rights of Samsung or others.

Samsung makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Samsung assume any liability arising out of the application or use of any product or circuit and specifically disclaims any and all liability, including without limitation any consequential or incidental damages.

"Typical" parameters can and do vary in different applications. All operating parameters, including "Typicals" must be validated for each customer application by the customer's technical experts.

Samsung products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, for other applications intended to support or sustain life, or for any other application in which the failure of the Samsung product could create a situation where personal injury or death may occur.

Should the Buyer purchase or use a Samsung product for any such unintended or unauthorized application, the Buyer shall indemnify and hold Samsung and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, expenses, and reasonable attorney fees arising out of, either directly or indirectly, any claim of personal injury or death that may be associated with such unintended or unauthorized use, even if such claim alleges that Samsung was negligent regarding the design or manufacture of said product

S3C6400/6410 RISC Microprocessor
JPEG User's manual

Samsung Electronics Co., Ltd.
San #24 Nongseo-Dong, Giheung-Gu
Yongin-City Gyeonggi-Do, Korea
446-711

Home Page: http://www.samsungsemi.com/

E-Mail: mobilesol.cs@samsung.com

Printed in the Republic of Korea

Revision History

Revision No	Description of Change	Refer to	Author(s)	Date
1.00	Initial Version	-	Jiun Yu	2007-07-20
2.00	Change buffer allocation	-	Jiun Yu	2007-12-22
2.10	Some bugs are fixed		Jiun Yu	2008-01-16
3.00	Supporting JPEG driver on S3C6410		Jiun Yu	2008-07-05
3.10	Tested on linux kernel 2.6.24		Jiun Yu	2008-08-29

Contents

1 Introduction

1.1 Purpose

The purpose of the document is to describe the JPEG Encoder/Decoder API for easy portability into different platforms by developers.

1.2 Scope

The scope of this document is to describe

- Software architecture of Encoder/Decoder
- Data structures and API used for Encoder/Decoder
- Usage example of Encoder/Decoder

1.3 Intended Audience

Intended Audience	Tick whenever Applicable
Project Manager	Yes
Project Leader	Yes
Project Team Member	Yes
Test Engineer	Yes

1.4 Definitions, Acronyms, and Abbreviations

Abbreviations	Description
JPEG	Joint Photographic Exports Grout
MCU	Minimum Coded Unit
EXIF	Exchangeable Image File Format
API	Application Program Interface

1.5 References

Number	Reference	Description
1	SMDK6400_Linux_2.6.16_JPEG_PortingGuide_REV1_00_20070720.doc	OS porting guide
2	S3C6400_WinCE6.0_WM6.0_JPEG_API_REV3.0_20071109.doc	API specification

2 Software Architecture

2.1 Overview

Software architecture of JPEG Encoder/Decoder package mainly comprises of two major modules:

- Common JPEG API
- JPEG Encoder/Decoder Device Driver

Common JPEG API provides the same interface to user application even if operating system and Codec(H/W or S/W) is different. Figure 1. shows relationship among user application, API, device driver and codec

Common JPEG API consists of 5 operations

- Initialize : Initialize encoder/decoder. i.e. initialize memory and default variables.
- Buffer Management : Get input/output buffer address
- Execute : Execute encoding/decoding process
- Configuration : Set/Get parameters to execute encoding/decoding
- Finalization : Release encoder/decoder resources

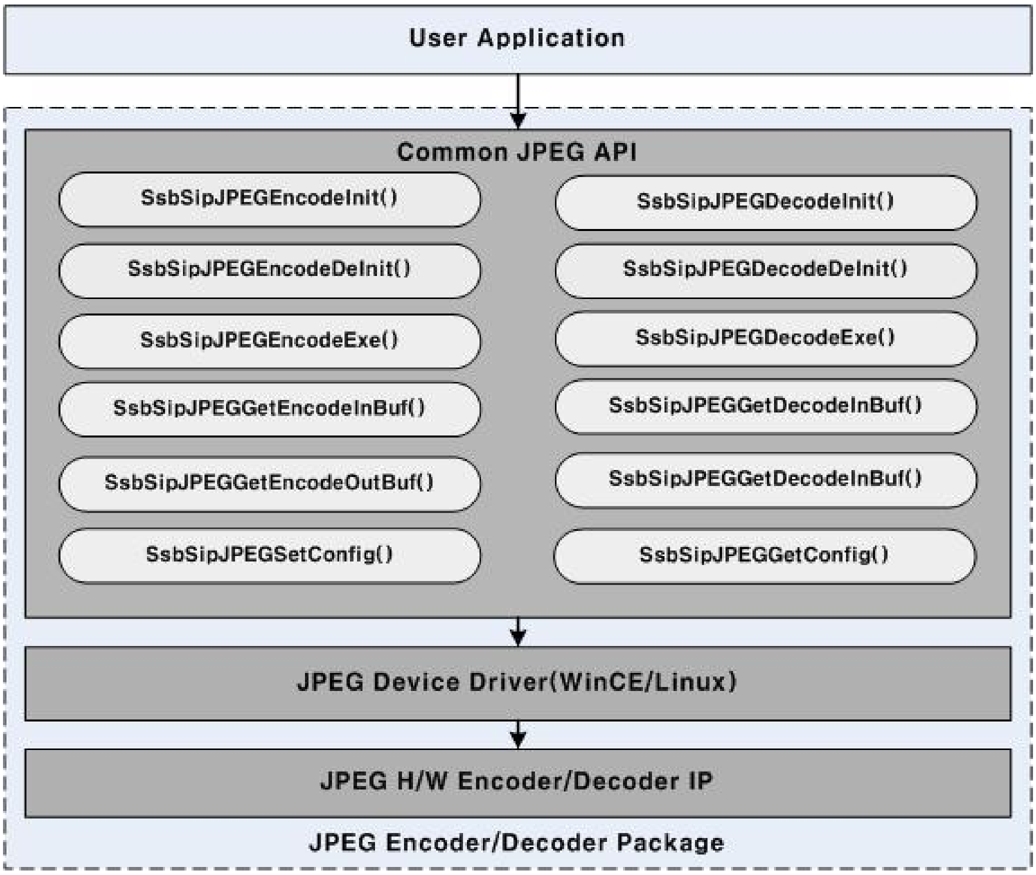


Figure . Architecture of JPEG Package

2.2 Features

- Compression/decompression up to UXGA(1600*1200).
- Encoding format: interleaved YCBYCR (input format of JPEG engine).

- Decoding format: YCbCr4:4:4, YCbCr4:2:2, YCbCr4:2:0, YCbCr4:1:1 or Gray.
- Support Exif file and Thumbnail encoding.
- The resolution of thumbnail is up to 320*240.
- Support 4 different image quality levels during encoding.
- Support of direct compression from the camera output.
- Support of compression of memory data in interleaved YCBYCR

3 Package Guidelines

3.1 Directory Structure

Directory	Files	Description
/jpeg_app/	*.c	JPEG API source file, test file
/jpeg_app/	*.h	JPEG API header file
/jpeg_drv/	*.c, *.h	JPEG Device Driver file
/jpeg_app/testVector/testInp	*.jpg, *.yuv	Test input vector
/jpeg_app/testVector/testOut	*.jpg, *.yuv	Test output vector
/doc	*.doc, *.pdf	JPEG documents

4 How to Test JPEG Encoder/Decoder

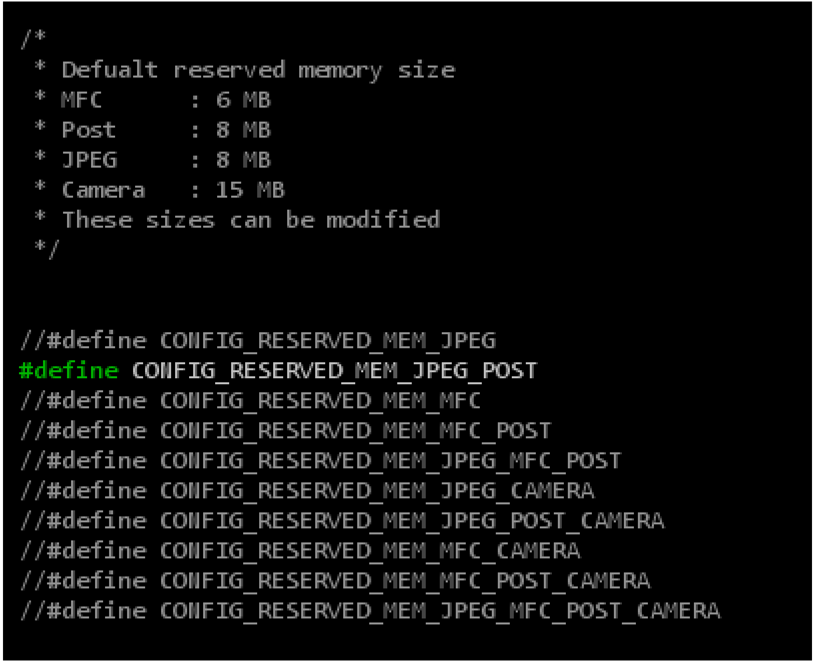
4.1 Procedure to build API and test application

1. kernel compilation
2. module compilation
3. Test application compilation
4. loading kernel
5. insert module into kernel
6. Execute test application binary
7. Output file is in */testVectors/testOut*

4.2 Procedure to test

4.2.1 Kernel configuration and building

Some devices need reserved memory because they have to allocate physically continuous memory. So you must setup reserved memory layout using below file named "reserved_mem.h" before kernel compilation.



```
/*
 * Defualt reserved memory size
 * MFC      : 6 MB
 * Post     : 8 MB
 * JPEG     : 8 MB
 * Camera   : 15 MB
 * These sizes can be modified
 */


//#define CONFIG_RESERVED_MEM_JPEG
#define CONFIG_RESERVED_MEM_JPEG_POST
//#define CONFIG_RESERVED_MEM_MFC
//#define CONFIG_RESERVED_MEM_MFC_POST
//#define CONFIG_RESERVED_MEM_JPEG_MFC_POST
//#define CONFIG_RESERVED_MEM_JPEG_CAMERA
//#define CONFIG_RESERVED_MEM_JPEG_POST_CAMERA
//#define CONFIG_RESERVED_MEM_MFC_CAMERA
//#define CONFIG_RESERVED_MEM_MFC_POST_CAMERA
//#define CONFIG_RESERVED_MEM_JPEG_MFC_POST_CAMERA
```

[NOTE]
1. This file is in
Include/asm-arm/arch-s3c64xx/reserved_mem.h (Linux2.6.16)
Include/asm-arm/arch-s3c2410/reserved_mem.h (Linux2.6.21 and 2.6.24)

2. For detailed information about how to build Linux kernel and how to download kernel image and cramfs, please refer to related porting guide documents.

4.2.2 Module compilation

JPEG device driver should be compiled as a kernel module.

1. Modify Makefile

You should modify "Makefile" to set configuration according to your environment.

```
[root@localhost s3c-linux-2.6.21]# cd [module directory]/Multimedia_DD/JPEG_V1.01/jpeg_drv

[root@localhost jpeg_drv]# vi Makefile
```

```
##################################################
# Makefile for JPEG Driver
# 2007 (C) Samsung Electronics
# Author : Jiun. Yu <jiun.yu@samsung.com>
##################################################

KERNEL_DIR := /home/mobile/workspace/s3c-linux-2.6.24
TOPDIR     := /home/mobile/workspace/s3c-linux-2.6.24

obj-m           := s3c_jpeg.o

s3c_jpeg-y := LogMsg.o JPGOpr.o JPGMisc.o JPGMem.o s3c-jpeg.o

PWD                     := $(shell pwd)

here:
        (cd $(KERNEL_DIR); make SUBDIRS=$(PWD) modules)

clean:
        rm -rf *.ko
        rm -rf *.mod.*
        rm -rf .*.cmd
        rm -rf *.o
        rm -rf Module.*
```

2. Module compilation

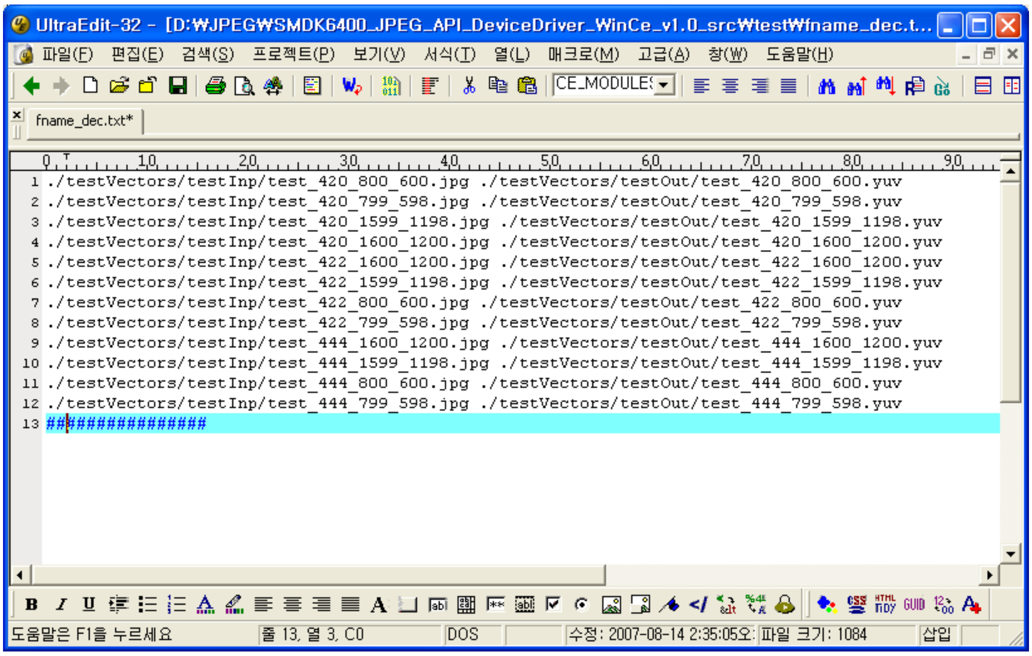
After compilation, you can find newly created files. "s3c_jpeg.ko" file is module of JPEG device driver.

```
[root@localhost jpeg_drv]# make
```

4.2.3 Test application compilation

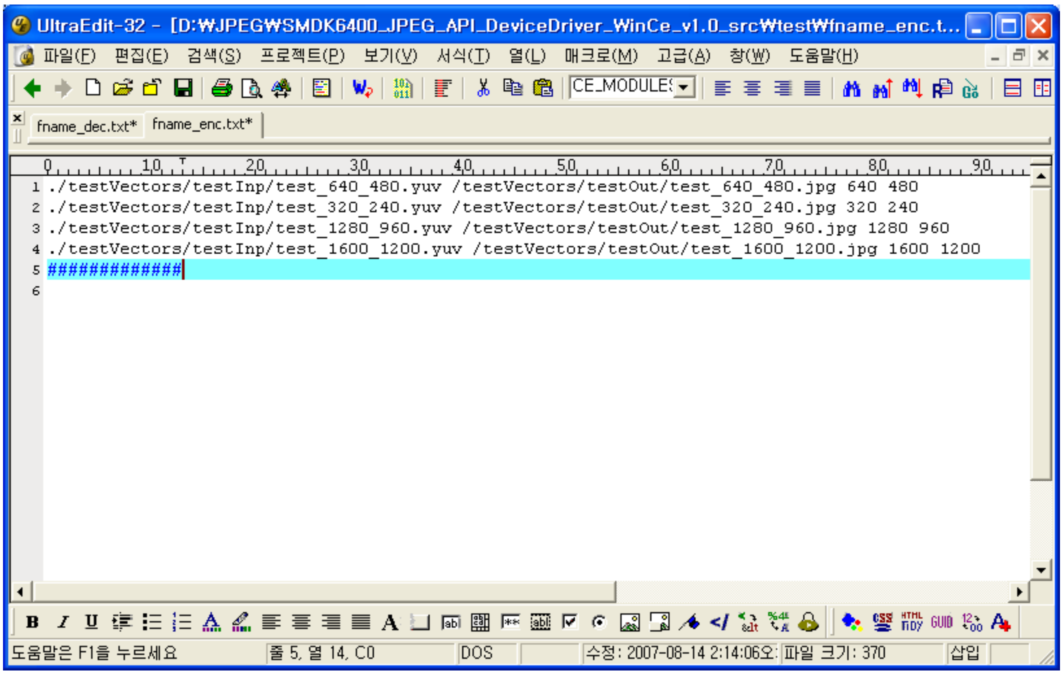
4.2.3.1 Setting control file

1. fname_dec.txt
    - control file which is used for decoder test
    - format : <input file name> <output file name>
    - termination character is '#'


UltraEdit-32 - [D:₩JPEG₩SMDK6400_JPEG_API_DeviceDriver_WinCe_v1.0_src₩test₩fname_dec.t...
파일(F) 편집(E) 검색(S) 프로젝트(P) 보기(V) 서식(T) 열(L) 매크로(M) 고급(A) 창(W) 도움말(H)
CE_MODULES
fname_dec.txt*
도움말은 F1을 누르세요
줄 13, 열 3, C0
DOS
수정: 2007-08-14 2:35:05오: 파일 크기: 1084
삽입

```
./testVectors/testInp/test_420_800_600.jpg ./testVectors/testOut/test_420_800_600.yuv
./testVectors/testInp/test_420_799_598.jpg ./testVectors/testOut/test_420_799_598.yuv
./testVectors/testInp/test_420_1599_1198.jpg ./testVectors/testOut/test_420_1599_1198.yuv
./testVectors/testInp/test_420_1600_1200.jpg ./testVectors/testOut/test_420_1600_1200.yuv
./testVectors/testInp/test_422_1600_1200.jpg ./testVectors/testOut/test_422_1600_1200.yuv
./testVectors/testInp/test_422_1599_1198.jpg ./testVectors/testOut/test_422_1599_1198.yuv
./testVectors/testInp/test_422_800_600.jpg ./testVectors/testOut/test_422_800_600.yuv
./testVectors/testInp/test_422_799_598.jpg ./testVectors/testOut/test_422_799_598.yuv
./testVectors/testInp/test_444_1600_1200.jpg ./testVectors/testOut/test_444_1600_1200.yuv
./testVectors/testInp/test_444_1599_1198.jpg ./testVectors/testOut/test_444_1599_1198.yuv
./testVectors/testInp/test_444_800_600.jpg ./testVectors/testOut/test_444_800_600.yuv
./testVectors/testInp/test_444_799_598.jpg ./testVectors/testOut/test_444_799_598.yuv
################
```

2. fname_enc.txt
   - control file which is used for encoder test
   - format : <input file name> <output file name> <width of input> <height of input>
   - termination character is '#'


UltraEdit-32 - [D:₩JPEG₩SMDK6400_JPEG_API_DeviceDriver_WinCe_v1.0_src₩test₩fname_enc.t...
파일(F) 편집(E) 검색(S) 프로젝트(P) 보기(V) 서식(T) 열(L) 매크로(M) 고급(A) 창(W) 도움말(H)
CE_MODULES
fname_dec.txt*
fname_enc.txt*
도움말은 F1을 누르세요
줄 5, 열 14, C0
DOS
수정: 2007-08-14 2:14:06오: 파일 크기: 370
삽입

```
./testVectors/testInp/test_640_480.yuv /testVectors/testOut/test_640_480.jpg 640 480
./testVectors/testInp/test_320_240.yuv /testVectors/testOut/test_320_240.jpg 320 240
./testVectors/testInp/test_1280_960.yuv /testVectors/testOut/test_1280_960.jpg 1280 960
./testVectors/testInp/test_1600_1200.yuv /testVectors/testOut/test_1600_1200.jpg 1600 1200
#############

```

4.2.3.2 Encoder

1. Simple Encoding
   - set *#define TEST_ENCODE 1* at /jpeg_app/ test.c
   - set *#define TEST_ENCODE_WITH_EXIF 0* at /jpeg_app/test.c
   - set *#define TEST_ENCODE_WITH_THUMBNAIL 0* at /jpeg_app/test.c
   - Build test project.
   - Execute test.exe

2. Encoding with Exif

- set *#define TEST_ENCODE 1* at /jpeg_app/test.c
- set *#define TEST_ENCODE_WITH_EXIF 1* at /jpeg_app/test.c
- set *#define TEST_ENCODE_WITH_THUMBNAIL 0* at /jpeg_app/test.c
- Build test project.
- Execute test.exe

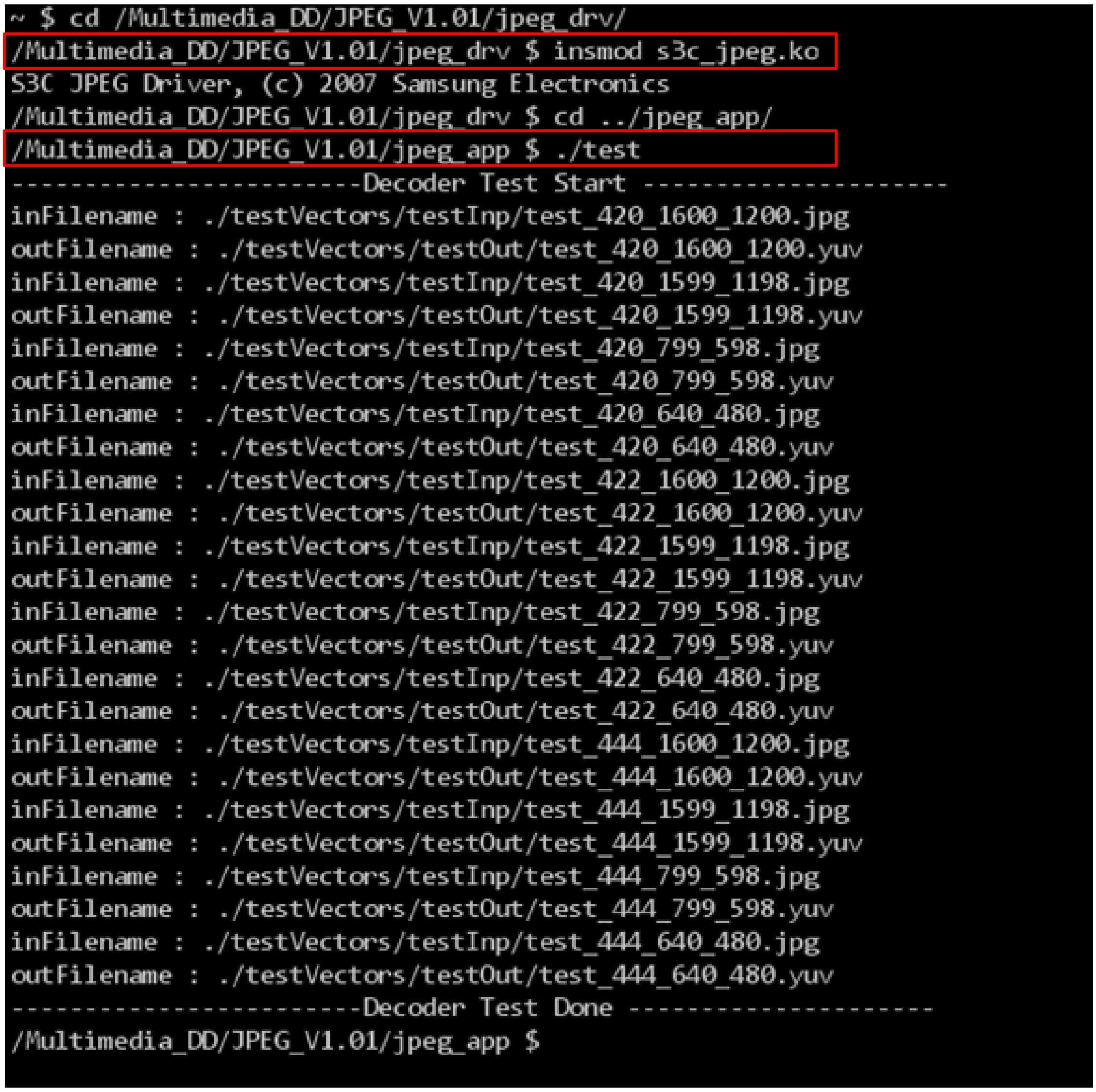
3. Encoding with Thumbnail
   - set *#define TEST_ENCODE 1* at /jpeg_app/test.c
   - set *#define TEST_ENCODE_WITH_EXIF 1* at /jpeg_app/test.c
   - set *#define TEST_ENCODE_WITH_THUMBNAIL 1* at /jpeg_app/test.c
   - Build test project.
   - Execute test.exe
   - When create thumbnail, *TEST_ENCODE_WITH_EXIF* must to be *1*

4.2.3.3 Decoder

1. JPEG to YCBYCR
   - set *#define TEST_ENCODE 0* at /jpeg_app/test.c
   - set *#define TEST_DECODE 1* at / jpeg_app /test.c
   - set *#define TEST_DECODE_OUTPUT_YCBYCR 1* at / jpeg_app /test.c
   - set *#define TEST_DECODE_OUTPUT_YUV422 0* at / jpeg_app /test.c
   - Build test project.
   - Execute test.exe

2. JPEG to YUV422
   - set *#define TEST_ENCODE 0* at / jpeg_app /test.c
   - set *#define TEST_DECODE 1* at / jpeg_app /test.c
   - set *#define TEST_DECODE_OUTPUT_YUV422 1* at / jpeg_app /test.c
   - set *#define TEST_DECODE_OUTPUT_YCBYCR 0* at / jpeg_app /test.c
   - Build test project.
   - Execute test.exe

4.3 Test on board



```
~ $ cd /Multimedia_DD/JPEG_V1.01/jpeg_drv/
/Multimedia_DD/JPEG_V1.01/jpeg_drv $ insmod s3c_jpeg.ko
S3C JPEG Driver, (c) 2007 Samsung Electronics
/Multimedia_DD/JPEG_V1.01/jpeg_drv $ cd ../jpeg_app/
/Multimedia_DD/JPEG_V1.01/jpeg_app $ ./test
------------------------Decoder Test Start ---------------------
inFilename : ./testVectors/testInp/test_420_1600_1200.jpg
outFilename : ./testVectors/testOut/test_420_1600_1200.yuv
inFilename : ./testVectors/testInp/test_420_1599_1198.jpg
outFilename : ./testVectors/testOut/test_420_1599_1198.yuv
inFilename : ./testVectors/testInp/test_420_799_598.jpg
outFilename : ./testVectors/testOut/test_420_799_598.yuv
inFilename : ./testVectors/testInp/test_420_640_480.jpg
outFilename : ./testVectors/testOut/test_420_640_480.yuv
inFilename : ./testVectors/testInp/test_422_1600_1200.jpg
outFilename : ./testVectors/testOut/test_422_1600_1200.yuv
inFilename : ./testVectors/testInp/test_422_1599_1198.jpg
outFilename : ./testVectors/testOut/test_422_1599_1198.yuv
inFilename : ./testVectors/testInp/test_422_799_598.jpg
outFilename : ./testVectors/testOut/test_422_799_598.yuv
inFilename : ./testVectors/testInp/test_422_640_480.jpg
outFilename : ./testVectors/testOut/test_422_640_480.yuv
inFilename : ./testVectors/testInp/test_444_1600_1200.jpg
outFilename : ./testVectors/testOut/test_444_1600_1200.yuv
inFilename : ./testVectors/testInp/test_444_1599_1198.jpg
outFilename : ./testVectors/testOut/test_444_1599_1198.yuv
inFilename : ./testVectors/testInp/test_444_799_598.jpg
outFilename : ./testVectors/testOut/test_444_799_598.yuv
inFilename : ./testVectors/testInp/test_444_640_480.jpg
outFilename : ./testVectors/testOut/test_444_640_480.yuv
------------------------Decoder Test Done ---------------------
/Multimedia_DD/JPEG_V1.01/jpeg_app $
```