

# **JPEG Loopback on the DM642 EVM**

*Video and Imaging Systems*

## **ABSTRACT**

The software demonstrates real-time D1 JPEG encoding and decoding of images on a DM642 EVM. The JPEG standard pertains to compression of still images. Performing JPEG at the rate of 30 frames per second, in isolation, as individual images, is considered motion JPEG (MJPEG). The demonstration uses:

- JPEG encoder library optimized for a DM642 EVM capable of real-time D1 encoding
- JPEG decoder library optimized for a DM642 EVM capable of real-time D1 decoding
- JPEG encoder and decoder library integrated with the IDMA layer specification
- JPEG encoder and decoder library implemented using XDAIS interfaces
- Sample integration of JPEG encoder and decoder under RF-5 framework to demonstrate JPEG loopback (encode + decode) at programmable quality and frame rate

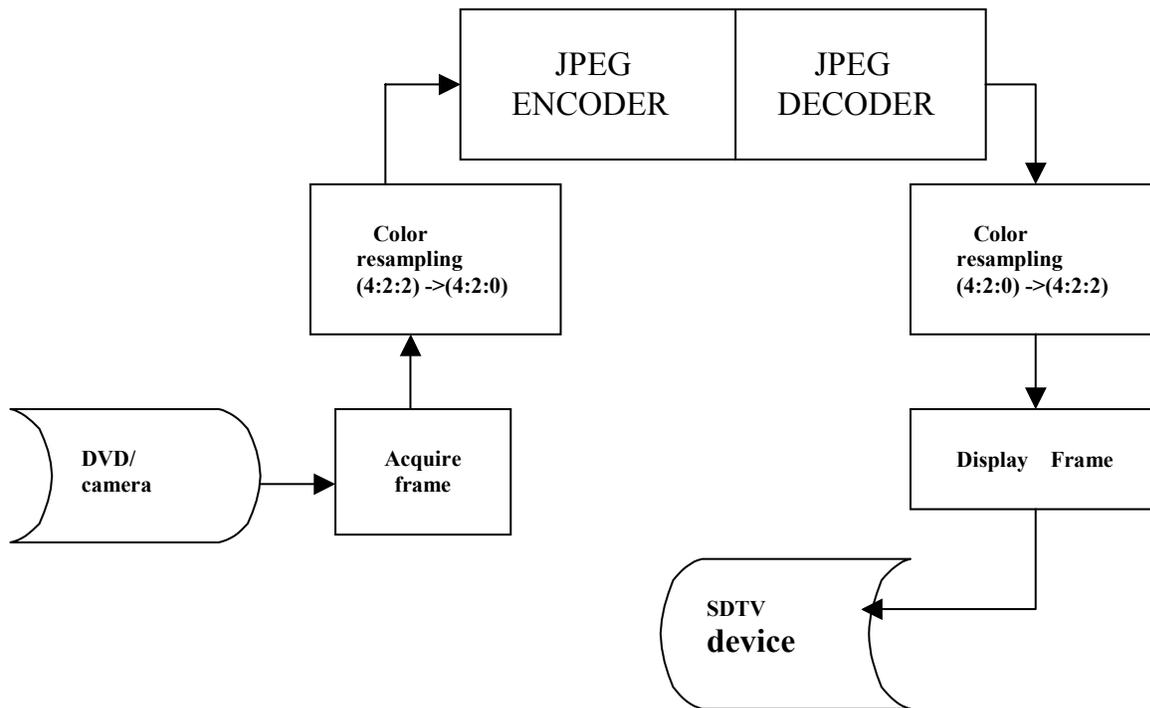
## **Contents**

<b>Software Architecture/Data Flow</b> .....	<b>2</b>
Data Flow Diagram for the Demonstration.....	2
Framework Flowchart.....	2
<b>System Requirements/Configuration</b> .....	<b>4</b>
Software Requirements.....	4
Hardware Requirements .....	4
<b>Hardware Setup</b> .....	<b>4</b>
<b>Demonstration Execution</b> .....	<b>5</b>
Controlling Parameters of the Demonstration .....	6
<b>Demonstration Code and Build Procedure</b> .....	<b>6</b>
Build Procedure.....	6
Troubleshooting .....	7
<b>Known Bugs and Constraints</b> .....	<b>7</b>
<b>Performance of JPEG Encoder and JPEG Decoder</b> .....	<b>7</b>

## **Figures**

<b>Figure 1. Data Flow Diagram</b> .....	<b>2</b>
<b>Figure 2. Hardware Setup</b> .....	<b>5</b>
<b>Figure 3. Directory Structure for jpeg_loopback</b> .....	<b>6</b>

## Software Architecture/Data Flow



**Figure 1. Data Flow Diagram**

### Data Flow Diagram for the Demonstration

The data flow in the demonstration this sequence:

1. A frame is captured from the input source (DVD/camera), and the acquired frame data, in YUV 4:2:2 format, is resampled to YUV 4:2:0 format.
2. The frame is fed to the JPEG encoder library, which generates a JPEG-encoded image at a desired quality set by the user.
3. The encoded JPEG image is then accepted as an input by the JPEG decoder. The JPEG decoder produces a decoded YUV 4:2:0 image.
4. The decoded YUV 4:2:0 image is converted to a YUV 4:2:2 image.

### Framework Flowchart

This demonstration uses RF-5 framework to integrate the JPEG encoder and decoder library. The demonstration uses a four-task setup. The fourth task is a control task, which uses a mailbox to send messages to the process task. (The fourth task is not shown in Figure 1 because of space limitations.) The process task receives messages from this mailbox and adjusts the frame rate at which the application runs or changes the quality of the encoded image, based on the nature of the message. Before coming to the DSP BIOS™ task scheduler, the demonstration code initializes various modules used in the system. These include:

- Board and processor
  - The system performs DSP BIOS™ initialization and CSL initialization.
  - The L2 cache mode is set to 128K cache.
  - EMIFA CE0 and EMIF CE1 space are enabled for caching.
  - The DMA priority queue lengths are set to maximum.
  - Priority for L2 request is set as high.
  - DMA manager is initialized with allocated internal and external heap.
- RF-5 modules
  - The system initializes the channel module of RF-5.
  - The system initializes the ICC and SCOM modules of RF-5 required for intercell communication and messaging.
  - Channel setup is performed with the internal and external heap buffers.
- Capture and display channels
  - An instance of capture channel is created and started.
  - An instance of display channel is created and started.

After these initializations, the system enters the four-task system managed by the DSP BIOS™ scheduler. These four tasks use the SCOM module of RF-5 to communicate with each other:

- Input task
 

The input task is responsible for acquiring the frames from the NTSC input device. It uses FVID\_exchange calls provided by the driver to acquire a frame. The acquired frame is in YUV 4:2:2 format. It is resampled to YUV 4:2:0 format. It then ends the message to the process task with the frame pointer embedded in the message. The task then waits for the message from the process task to continue.
- Process task
 

The process task has two cells in this demonstration. The first cell is a JPEG encoder cell that accepts an input YUV 4:2:0 image and produces a compressed JPEG image at a desired quality. The second cell is a JPEG decoder cell that accepts a JPEG-encoded image and produces a decoded image. The decoded image produced by the JPEG decoder is in YUV4:2:0 format. After the completion of the JPEG decode, the process task posts a message to both the input and the output tasks indicating that the buffers passed and are now available for reuse.
- Output task
 

The output task is responsible for displaying the frames on the NTSC output device. It uses FVID\_exchange calls provided by the driver to display a frame on the NTSC output device. The acquired frame is in YUV 4:2:0 format and is resampled to YUV 4:2:2 format prior to display. The task then waits for a message from the process task to continue.

- Control task

The control task is responsible for controlling the parameters that are variable within this demonstration. These include the frame rate at which the JPEG demonstration is executed and the quality factor of the demonstration. The control task checks for a change in value in the parameters defined in a global structure, External Control, (visible to the user in global space). It then copies the values of the changed parameters into a local structure, External Control\_prev, and posts messages in a mailbox to the process thread. The processing task periodically checks for messages and calls the corresponding cell's control function.

## System Requirements/Configuration

### Software Requirements

- Microsoft Windows NT (SP6)/Microsoft Windows 2000 (SP1 and SP2)
- Code Composer Studio™ Integrated Development Environment (IDE) version 2.20.18
- Driver software (DDK 1.1)

### Hardware Requirements

- Pentium machines with 450 MHz, 64MB RAM (minimum)
- DM642 EVM
- NTSC TV for display purposes
- Camera/DVD for NTSC capture purposes
- XDS 510/560 emulator

### Hardware Setup

To run the demonstration, the hardware must be setup properly, as shown in Figure 2.

- The DM642 EVM must be connected to the appropriate power source.
- The input video port (for composite video) must be connected to NTSC input.
- The source (DVD/camera) must be connected using RCA cable.
- The output video port (for composite video) must be connected to NTSC
- The output device (SDTV) must be connected using RCA cable.
- The XDX510/560 emulator must be connected to the JTAG pins to download the demonstration code to the board and control it from Code Composer Studio™ (IDE).

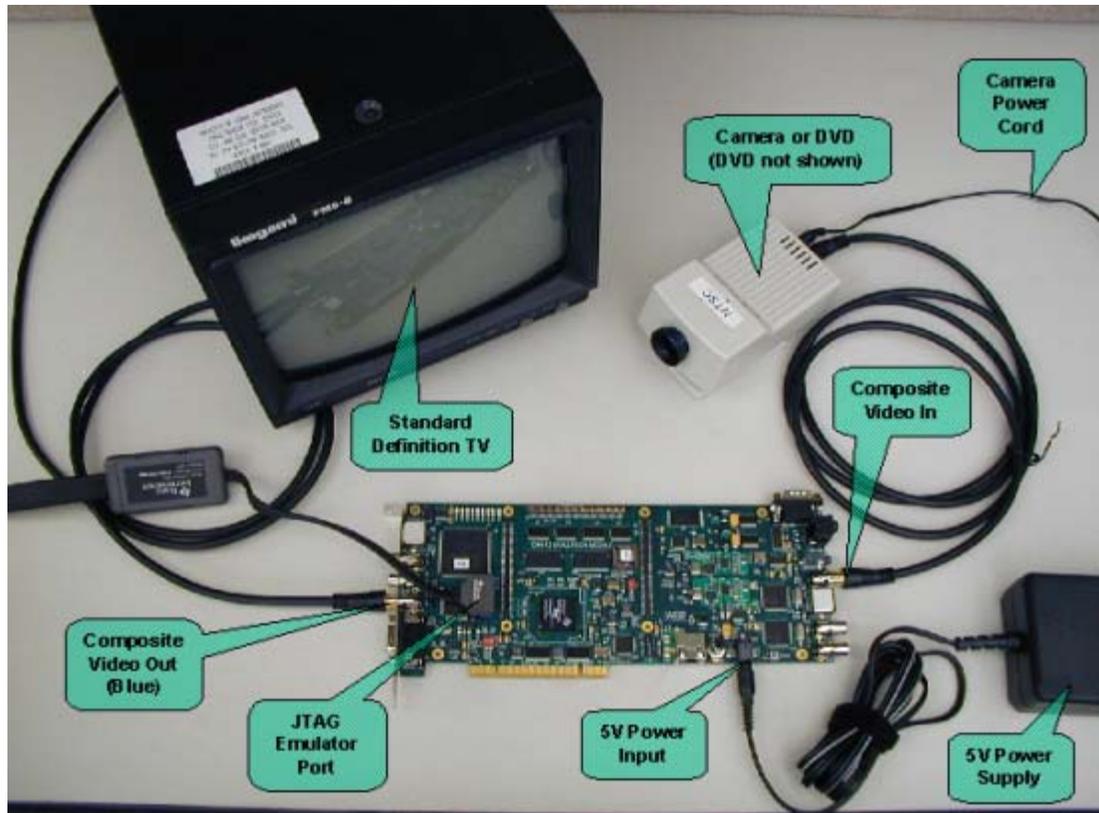


Figure 2. Hardware Setup

## Demonstration Execution

To run the demonstration:

1. Connect the NTSC input device (camera/DVD) using proper RCA cables.
2. Connect the NTSC output device (SDTV) using proper RCA cables.
3. Power up the DM642 EVM board.
4. Start Code Composer Studio™ IDE version 2.20.18.
5. Check the color bar on the output device.
6. Go to the bin folder under the jpeg\_loopback directory and load .out.
7. Once the program is loaded, go to the Debug Menu in Code Composer Studio™ IDE and press the Run option (F5).
8. On the output screen, watch the JPEG reconstructed image with the TI logo at the top-right corner of the screen.

## Controlling Parameters of the Demonstration

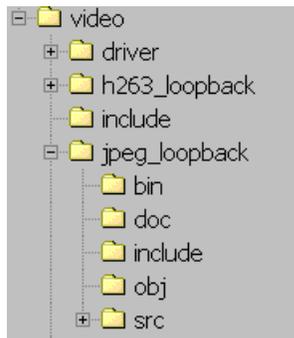
1. To control the quality or the frame rate of the demonstration, make sure that the `thrControl.c` file is built with the full symbolic debug feature (-g).
2. Add the `externalControl` structure in the watch window in Code Composer Studio™ IDE.
3. Now, expand the structure in the control window. This structure contains two fields: `frameRatio` and `quality`.

`frameRatio` is a ratio, as the name implies, and is a divide-down factor with a value of 1 that provides 30-output frames/sec and a value of 10 that provides 3-output frames/sec.

The second field, `quality`, allows the quality of the resulting JPEG decoded image to be varied. The allowed values for quality are of the form  $1 \leq \text{quality} \leq 100$  with both end values being legal.

## Demonstration Code and Build Procedure

The directory structure for `jpeg_loopback` is located under `evmdm642\examples\video`.



**Figure 3. Directory Structure for jpeg\_loopback**

## Build Procedure

1. Start Code Composer Studio™ IDE version 2.20.18.
2. Open the `jpeg_loopback` project (`jpeg_loopback_lib.pjt`) in the `examples\video\jpeg_loopback` folder.
3. Go to Project->Build and rebuild the project.
4. Compiler options used under preprocessor are `CHIP_DM642=1`, `C6000`, and `UTL_DBGLEVEL=70`.
5. Build the project and load the executable from the `bin` directory `jpeg_loopback.out`.
6. Press F5 to watch the decoded JPEG image.

## Troubleshooting

Compile the project with the `TEST_FEEDBACK_LOOP` preprocessor directive that is provided to disable the MPEG-2 encoder from the demonstration and test the loopback part of the setup. In case of any problems, this symbol can be defined in the preprocessor options of the project compilation setting and to test the setup part of the demonstration.

## Known Bugs and Constraints

- The quality factor must be in the range 1 to 100.
- The frame rate ratio must be a positive number greater than or equal to 1.
- The decoder checks whether a decoded image is a valid JPEG and returns a negative error code on receiving an incorrect JPEG stream.

## Performance of JPEG Encoder and JPEG Decoder

The performance of JPEG encoder and decoder images is content- and quality-dependent. The figures shown here are for a quality setting of 75 and reasonable complexity images, which are typical.

- D1 4:2:0 encode for typical images at a quality setting of 75% uses 23% of a 600-mHz C64x DSP.
- D1 4:2:0 decode for typical images at a quality setting of 75% uses 20% of a 600-mHz C64x DSP.

## IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

<b>Products</b>		<b>Applications</b>	
Amplifiers	<a href="http://amplifier.ti.com">amplifier.ti.com</a>	Audio	<a href="http://www.ti.com/audio">www.ti.com/audio</a>
Data Converters	<a href="http://dataconverter.ti.com">dataconverter.ti.com</a>	Automotive	<a href="http://www.ti.com/automotive">www.ti.com/automotive</a>
DSP	<a href="http://dsp.ti.com">dsp.ti.com</a>	Broadband	<a href="http://www.ti.com/broadband">www.ti.com/broadband</a>
Interface	<a href="http://interface.ti.com">interface.ti.com</a>	Digital Control	<a href="http://www.ti.com/digitalcontrol">www.ti.com/digitalcontrol</a>
Logic	<a href="http://logic.ti.com">logic.ti.com</a>	Military	<a href="http://www.ti.com/military">www.ti.com/military</a>
Power Mgmt	<a href="http://power.ti.com">power.ti.com</a>	Optical Networking	<a href="http://www.ti.com/opticalnetwork">www.ti.com/opticalnetwork</a>
Microcontrollers	<a href="http://microcontroller.ti.com">microcontroller.ti.com</a>	Security	<a href="http://www.ti.com/security">www.ti.com/security</a>
		Telephony	<a href="http://www.ti.com/telephony">www.ti.com/telephony</a>
		Video & Imaging	<a href="http://www.ti.com/video">www.ti.com/video</a>
		Wireless	<a href="http://www.ti.com/wireless">www.ti.com/wireless</a>

Mailing Address: Texas Instruments  
Post Office Box 655303 Dallas, Texas 75265

Copyright © 2003, Texas Instruments Incorporated