JPEG Codec Library for Wii Specifications

Version 1.0.1

The content of this document is highly confidential and should be handled accordingly.

RVL-06-0269-001-C
Released: October 29, 2008

Confidential

Table of Contents

Code

Tables

Figures

Revision History

Version	Revision Date	Description
1.0.1	2008/03/11	Revised the types for the `TMCCJPEGJpegSize` structure member variables and various function arguments, and improved error tolerance.
1.0.0	2008/01/18	Initial version.

1 Overview

This version of Middleware provides encoding and decoding capabilities using algorithms that conform to ISO JPEG (ISO/IEC 10918-1).

1.1 Encoding Capabilities and Limitations

This version of middleware features the following encoding capabilities and limitations.

- Supports only the JPEG baseline (ISO/IEC 10918-1 Sequential DCT-based mode, the interchange format); encoding to extension formats is not possible
- Supports the input of RGB565 or RGBA8 texture formats
- Uses a YUV sub-sampling format; 4:2:0 (interleaved three-component Y,U,V with sub-sampling of 2:1:1 horizontally and 2:1:1 vertically).
- Does not support a restart interval marker
- Capable of determining the image quality factor when a JPEG output size is specified
- Capable of determining the JPEG output size when an image quality factor is specified
- Allows encoding by specifying output limit values

1.2 Decoding Capabilities and Limitations

This version of Middleware features the following decoding capabilities and limitations:

- Supports only the JPEG baseline (ISO/IEC 10918-1 Sequential DCT-based mode, the interchange format); JPEG streams encoded using extension formats cannot be decoded
- Supports the output of RGB565 and RGBA8 texture formats
- Supports restart using RST markers on the error stream
- Can change number of lines using DNL markers
- Supports the following YUV sub-sampling formats:
  - 4:1:1 (interleaved three-component Y,U,V with sub-sampling of 4:1:1 horizontally and 1:1:1 vertically)
  - 4:2:2 (interleaved three-component Y,U,V with sub-sampling of 2:1:1 horizontally and 1:1:1 vertically)
  - 4:2:0 (interleaved three-component Y,U,V with sub-sampling of 2:1:1 horizontally and 2:1:1 vertically)
  - 2:1:1 (interleaved three-component Y,U,V with sub-sampling of 1:1:1 horizontally and 2:1:1 vertically)
  - 4:4:4 (interleaved three-component Y,U,V with sub-sampling of 1:1:1 horizontally and 1:1:1 vertically)
  - 4:0:0 (single-component Y data)
- Capable of getting `Exif` information; for details, see section 2.3.2.1 Getting Exif Information

- Capable of enlarging the JPEG stream of thumbnails; for details, see section 2.3.2.2 Enlarging Thumbnails
- Capable of specifying a region to be enlarged in the source image; for details, see section 2.3.2.3 Specifying the Area to Be Enlarged
- Allows changing the size of the output texture image by specifying a resolution reduction level; for details, see section 2.3.2.4 Specifying the Resolution Reduction Level

2 Details

2.1 Work Memory Management

This version of Middleware requires the types of work memory described below. Ensure the application allocates each work memory type from the heap memory area and uses arguments to pass its address to functions. Do not free any work memory areas until all API processes have ended.

2.1.1 Work Memory for Parameters

To execute processing for a single frame, this version of Middleware uses a work memory of the size given by `TMCC_JPEG_WORKAREA_SIZE` (7168 bytes). Memory should be allocated before each encoding or decoding process is started. Specify the start address of the work area using the first argument, `pWorkArea`, of each function.

2.1.2 Work Memory for Encoding

To execute an encoding process using this version of Middleware, allocate a work memory sufficient for one frame of the input texture image and a work memory for the output JPEG stream. The output JPEG stream requires a work memory size of at least `TMCC_JPEG_JPEGBUFFER_MINIMAM_SIZE` (607 bytes).

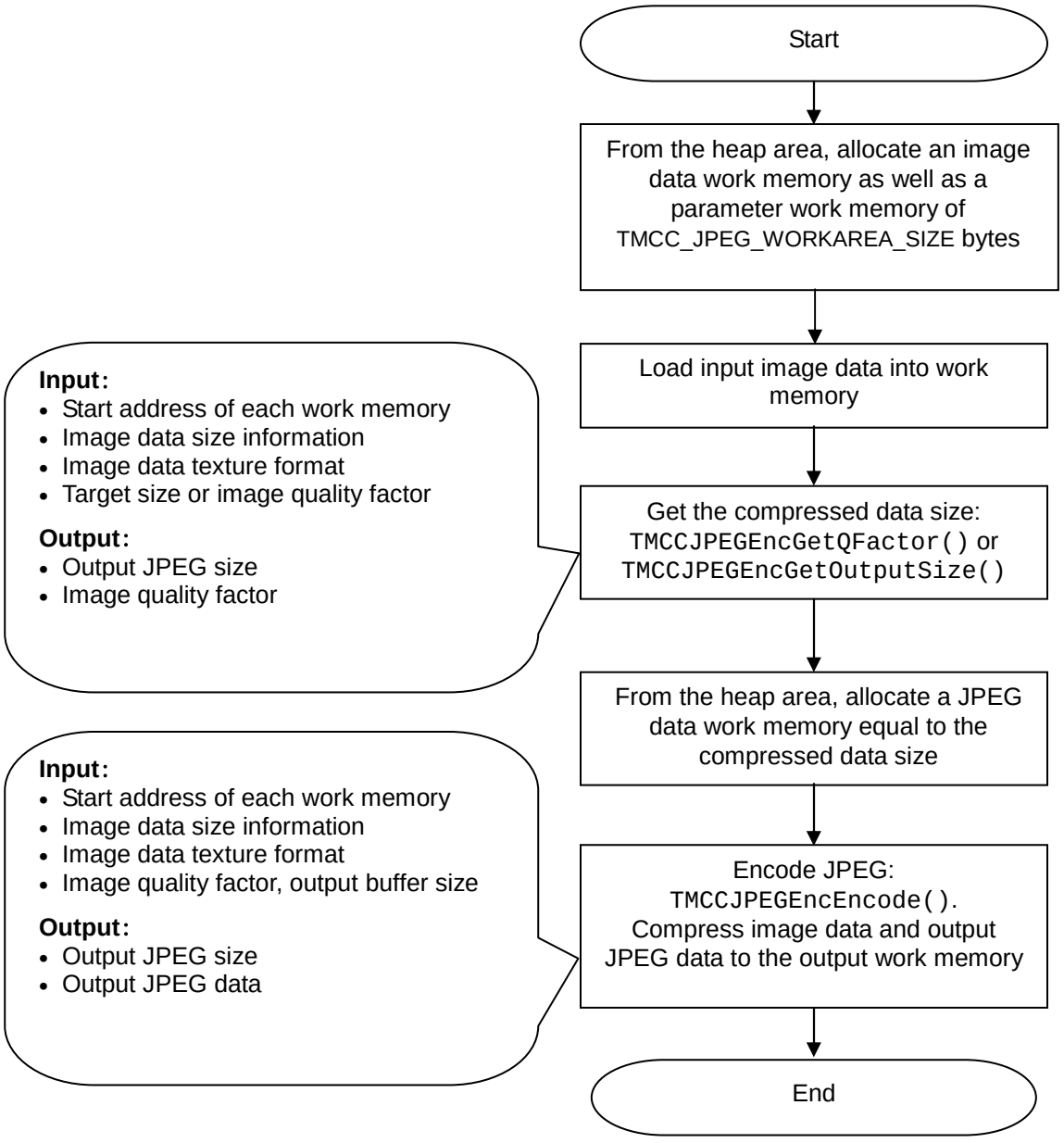
2.1.3 Work Memory for Decoding

To execute a decoding process using this version of Middleware, allocate a work memory sufficient for one frame of the output texture image and a work memory for the input JPEG stream. The start address of the input JPEG stream must have a 32-byte alignment.

2.2 Flowchart

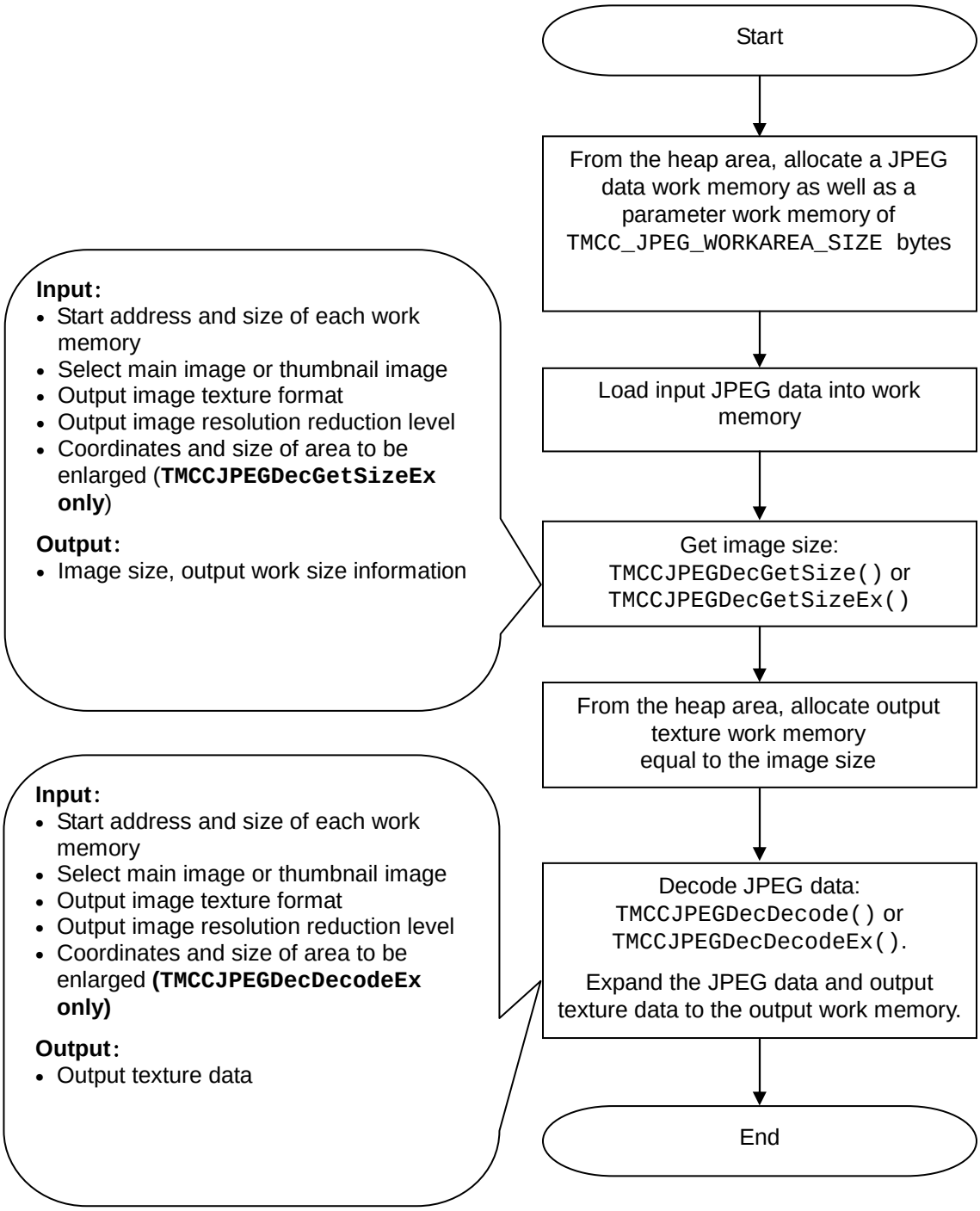
2.2.1 Encoding Images

Figure 2-1 Encoding Images



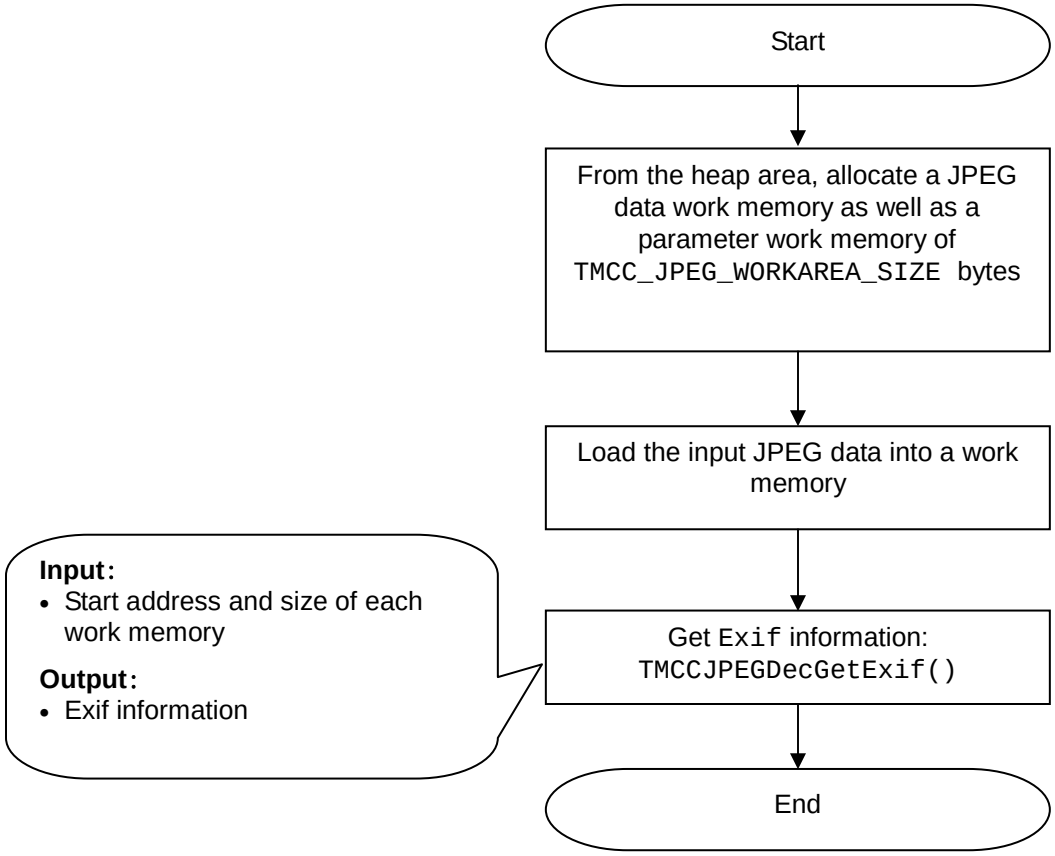
2.2.2 Decoding Images

Figure 2-2 Decoding Images



2.2.3 Getting Exif Information

Figure 2-3 Getting Exif Information



2.3 Optional Features

2.3.1 Encoder

2.3.1.1 Image Quality Factor

The image quality factor set for the encoder controls the image quality level and compression ratio of the output JPEG after compression. As shown below, the value of the image quality factor is proportional to the image quality level and inversely proportional to the compression ratio. The minimum value allowed for the image quality factor is 1, while the maximum value is 500. Output data that exceeds a compression ratio given by an image quality factor of 1 cannot be obtained with this encoder. Furthermore, output data that exceeds an image quality level given by an image quality factor of 500 cannot be obtained.

- Image quality factor of 1: maximum compression, minimum image quality level
- Image quality factor of 500: minimum compression, maximum image quality level

Reference data are shown below. (Sample image: 640x480 pixels with the data size of 921656 bytes after compression).

Table 2-1 Compression Size and Compression Ratio by Image Quality Factor

Image Quality Factor	Data Size in Bytes (Compression Ratio)
1	9433 (98 times)
100	56613 (16 times)
200	83618 (11 times)
300	106420 (9 times)
400	125660 (7 times)
500	145737 (6 times)

2.3.1.2 Checking the Image Quality Factor and Data Size After Compression

1. Getting the image quality factor and actual output size based on the output compression size.
   The `TMCCJPEGEncGetQFactor` function returns the image quality factor and actual compressed output data size that approaches the output size. For details on this function, see section 3.1.2 TMCCJPEGEncGetQFactor. Depending on the image, the difference between the actual output size and the output size given in accordance with the image may differ.
2. Getting the actual output size from the image quality factor.
   The `TMCCJPEGEncGetOutputSize` function returns the actual output size when a given image quality factor is input to the encoder. Compressed data is not output. For details on this function, see section 3.1.3 TMCCJPEGEncGetOutputSize.

2.3.2 Decoder

2.3.2.1 Getting Exif Information

The decoder conforms to JEIDA Standard `Exif` Format, version 2.1.

A list of obtainable information is given below. The information listed is obtained using the `TMCCJPEGDecGetExif` function. For details on this function, see section 3.2.5 TMCCJPEGDecGetExif.

The `TMCCJPEGExifIFD0Info` structure stores information related to the main image in the JPEG stream. The `TMCCJPEGExifIFD1Info` structure stores information related to the thumbnail image in the JPEG stream. For details on these structures, see section 4.2 TMCCJPEGExifInfo.

Note: Corresponding tag information may not be recorded in the input JPEG stream file. In such case, the initial value of 0 is retained as the output value. In addition, the image title, name of the image input device manufacturer, and model name of the image input device parameters of the `TMCCJPEGExifIFD0Info` structure are not obtained. This is because they are ASCII strings and no upper limit has been set for the maximum number of characters.

Table 2-2 TMCCJPEGExifIFD0Info Parameters

Tag Name	Field Name	TAG(Dec)	TAG(Hex)	Type	Count
Image title	ImageDescription	270	10E	ASCII	Any
Name of image input device manufacturer	Make	271	10F	ASCII	Any
Model name of image input device	Model	272	110	ASCII	Any
Image orientation	Orientation	274	112	SHORT	1
Horizontal image resolution	XResolution	282	11A	RATIONAL	1
Vertical image resolution	YResolution	283	11B	RATIONAL	1
Horizontal and vertical image resolution unit	ResolutionUnit	296	128	SHORT	1
Playback gradation curve properties	TransferFunction	301	12D	SHORT	3x256
File modification date and time	DateTime	306	132	ASCII	20
YCC pixel construction (Y and C position)	YCbCrPositioning	531	213	SHORT	1
Offset value to Exif Private tag	Exif IFD Pointer	34665	8769	LONG	1
Exif version	ExifVersion	36864	9000	ASCII	4
Meaning of each component	ComponentsConfiguration	37121	9101	BYTE	4
Corresponding Flash Pix version	FlashPixVersion	40960	A000	ASCII	4
Color space information	ColorSpace	40961	A001	SHORT	1
Effective image width	PixelXDimension	40962	A002	SHORTorLONG	1
Effective image height	PixelYDimension	40963	A003	SHORTorLONG	1

Table 2-3 TMCCJPEGExifIFD1Info Parameters

Tag Name	Field Name	TAG(Dec)	TAG(Hex)	Type	Count
Compression type	Compression	259	103	SHORT	1
Horizontal image resolution	Xresolution	282	11A	RATIONAL	1
Vertical image resolution	Yresolution	283	11B	RATIONAL	1
Horizontal and vertical image resolution unit	ResolutionUnit	296	128	SHORT	1
Offset to JPEG SOI	JPEGInterchangeFormat	513	201	LONG	1
JPEG data size in bytes	JPEGInterchangeFormatLength	514	202	LONG	1

2.3.2.2 Enlarging Thumbnails

The Exif format JPEG thumbnail data included in JPEG data can be enlarged.

To get the thumbnail JPEG image size, call the TMCCJPEGDecGetSize function with 1 specified for the thumbnail argument. To enlarge the thumbnail JPEG, call the TMCCJPEGDecDecode function with 1 specified for the thumbnail argument. If there is no JPEG compressed thumbnail data included in the JPEG data, these functions will return the TMCC_JPEG_ERR_NO_THUMBNAIL error code.

When enlarging thumbnails, the area to be enlarged (this option is described next in 2.3.2.3) cannot be specified. Only 1 can be specified for the resolution reduction level, given in 2.3.2.4. If a value other than 1 is specified for resolution when either of the above two functions is used, the error code TMCC_JPEG_OPT_ERR will be returned.

2.3.2.3 Specifying the Area to Be Enlarged

Any area in the compressed source image can be specified for enlargement.

The TMCCJPEGDecGetSizeEx and TMCCJPEGDecDecodeEx functions correspond to this feature. Set the respective arguments srcX, srcY, srcWidth, and srcHeight as shown in Table 2-4.

Table 2-4 Area Specification

Input Parameters	Setting Contents
srcX	Represents the upper left X-coordinate of the specified area when the upper left corner of the source image is taken as (0, 0).
srcY	Represents the upper left Y-coordinate of the specified area when the upper left corner of the source image is taken as (0, 0).
srcWidth	Represents the width of the specified area from srcX (width of the output image).
srcHeight	Represents the height of the specified area from srcX (height of the output image).

Input Limitations

- `srcX` and `srcY` can only be specified in multiples of 32 (including 0)
- `srcWidth` and `srcHeight` may be set to any value that is either at least 32 and a multiple of 32, or represents a length extending to the lower right corner of the source image
- If input does not satisfy the above conditions, error code `TMCC_JPEG_OPT_ERR` is returned.

  In addition, the same error code is returned if `srcX` + `srcWidth` or `srcY` + `srcHeight` exceed the source image width.

  Taking a 50x50 source image as an example, the following values can be set for `srcX`, `srcY`, `srcWidth`, and `srcHeight`:

  - The specifiable values for `srcX` and `srcY` are (0,0),(0,32), (32,0), and (32,32)

  The specifiable values for `srcWidth` and `srcHeight` are given below for each possible pair of `srcX` and `srcY`:

  - When (`srcX`,`srcY`) are (0,0), `srcWidth`=32 or 50 and `srcHeight`=32 or 50
  - When (`srcX`,`srcY`) are (0,32), `srcWidth`=32 or 50 and `srcHeight`=18
  - When (`srcX`,`srcY`) are (32,0), `srcWidth`=18 and `srcHeight`=32 or 50
  - When (`srcX`,`srcY`) are (32,32), `srcWidth`=18 and `srcHeight`=18 only

2.3.2.4 Specifying the Resolution Reduction Level

The width and height of the output image can be specified as 1/1, 1/2, 1/4, or 1/8 of the compressed image. This specification can be made using the `resolution` argument of the `TMCCJPEGDecGetSize`, `TMCCJPEGDecGetSizeEx`, `TMCCJPEGDecDecode`, or `TMCCJPEGDecDecodeEx` functions.

The resolution of the area specified using the area to be enlarged specification option given in 2.3.2.3 can be reduced using the `TMCCJPEGDecGetSizeEx` or `TMCCJPEGDecDecodeEx` function. Only `1` can be legally set for `resolution` in the thumbnail enlargement option given in 2.3.2.2.

If an invalid value is entered, error code `TMCC_JPEG_OPT_ERR` will be returned.

Table 2-6 Resolution Reduction Level

Specified Value	Setting Contents
1	1/1 width of output image, 1/1 height, 1/1 total number of pixels
2	1/2 width of output image, 1/2 height, 1/4 total number of pixels
4	1/4 width of output image, 1/4 height, 1/16 total number of pixels
8	1/8 width of output image, 1/8 height, 1/64 total number of pixels

3 API Functions

3.1 JPEG Encoding Functions

3.1.1 TMCCJPEGEncEncode

Code 3-1 TMCCJPEGEncEncode Function

```
s32 TMCCJPEGEncEncode(
        void *pWorkArea,
        void *pTex,
        u16 srcWidth,
        u16 srcHeight,
        GXTexFmt texFormat,
        u32 QFactor,
        void *pJpegBuffer,
        u32 jpegBufferSize,
        u32 *poutputSize
);
```

Arguments

`void *pWorkArea`	[in] Start of work area
`void *pTex`	[in] Start address of texture data
`u16 srcWidth`	[in] Width of source image
`u16 srcHeight`	[in] Height of source image
`GXTexFmt texFormat`	[in] Texture format
`u32 QFactor`	[in] Image quality factor
`void *pJpegBuffer`	[out] Output buffer
`u32 jpegBufferSize`	[in] Output buffer size
`u32 *poutputSize`	[out] Actual output size

- Set the start address of the work area in `pWorkArea`
- Set the start address of the input texture data in `pTex`
- Specify the width and height of the input texture's active image in `srcWidth` and `srcHeight`
  If the texture size and active image area do not match, pad the excess amount with garbage data.
  `TMCC_JPEG_OPT_ERR` is the error code returned for input outside of the 1 to 65535 range.
- Specify the texture format in `texFormat`
  `TMCC_JPEG_TEXT_ERR` is the error code returned if anything other than `GX_TF_RGB565` or `GX_TF_RGBA8` is specified.
- Specify the image quality factor in `Qfactor`
  For information on the image quality factor, see section 2.3.1.1 Image Quality Factor..
  `TMCC_JPEG_OPT_ERR` is the error code returned for input outside the 1 to 500 range.
- Specify the start address of the work memory for output JPEG data in `pJpegBuffer`

- Specify the size of work memory for output JPEG data in `jpegBufferSize`

  `TMCC_JPEG_ENC_BUF_OVER` is the error code returned if it appears that the output JPEG size may exceed the size specified here.

  Be sure to allocate and set a work memory whose byte size is at least that given by `TMCC_JPEG_JPEGBUFFER_MINIMAM_SIZE`.

- Size of the output JPEG data is stored in `poutputSize`

Return Values

0 if successful termination; error code if negative.

Zero is returned if execution ends normally; error code (negative value) is returned if execution ends abnormally. For error codes, see section 5.1 Error Codes.

Description

This function performs JPEG encoding on the input texture image data and outputs compressed data.

3.1.2 TMCCJPEGEncGetQFactor

Code 3-2 TMCCJPEGEncGetQFactor Function

```
s32 TMCCJPEGEncGetQFactor(
        void *pWorkArea,
        void *pTex,
        u16 srcWidth,
        u16 srcHeight,
        GXTexFmt texFormat,
        u32 TargetSize,
        u32 *poutputSize,
        u32 *pQFactor
);
```

Arguments

`void *pWorkArea`	[in] Start address of work area
`void *pTex`	[in] Start address of texture data
`u16 srcWidth`	[in] Width of source image
`u16 srcHeight`	[in] Height of source image
`GXTexFmt texFormat`	[in] Texture format
`u32 TargetSize`	[in] Required output size
`u32 *poutputSize`	[out] Actual output size
`u32 *pQFactor`	[out] Image quality factor

- Set the start address of the work area in `pWorkArea`
- Set the start address of the input texture data in `pTex`
- Specify the width and height of the input texture's active image in `srcWidth` and `srcHeight`

  If the texture size and active image area do not match, pad the excess amount with garbage data. The error code `TMCC_JPEG_OPT_ERR` is returned for input outside the 1 to 65535 range.

- Specify the texture format in `texFormat`

  `TMCC_JPEG_TEXT_ERR` is the error code returned if anything other than `GX_TF_RGB565` or `GX_TF_RGBA8` is specified.

- Specify the output size after compression in `TargetSize`

- `poutputSize` stores the size that can be output using the obtained `pQFactor`
- `pQFactor` stores the image quality factor for compressed data of the byte size given by `poutputSize`

  For information about the image quality factor, see section 2.3.1.1 Image Quality Factor.

Return Values

0 if successful termination; error code if negative.

Zero is returned if execution ends normally; error code (negative value) is returned if execution ends abnormally. For error codes, see section 5.1 Error Codes.

Description

To get JPEG output of an arbitrary data size, this function determines the image quality factor and actual output size. It gets the image quality factor (`*pQFactor`) needed to compress the texture image data that has been input to the target size (`TargetSize`), and it gets the actual output size (`*poutputSize`). The difference between the target size and the actual obtained output size depends on the input image.

3.1.3 TMCCJPEGEncGetOutputSize

Code 3-3 TMCCJPEGEncGetOutputSize Function

```
s32 TMCCJPEGEncGetOutputSize(
        void *pWorkArea,
        void *pTex,
        u16 srcWidth,
        u16 srcHeight,
        GXTexFmt texFormat,
        u32 *poutputSize,
        u32 QFactor
);
```

Arguments

`void *pWorkArea`	[in] Start address of work area
`void *pTex`	[in] Start address of texture data
`u16 srcWidth`	[in] Width of source image
`u16 srcHeight`	[in] Height of source image
`GXTexFmt texFormat`	[in] Texture format
`u32 *poutputSize`	[out] Actual output size
`u32 QFactor`	[in] Image quality factor

- Set the start address of the work area in `pWorkArea`
- Set the start address of the input texture data in `pTex`
- Specify the width and height of the input texture's active image in `srcWidth` and `srcHeight`

  If the texture size and active image area do not match, pad the excess amount with garbage data. `TMCC_JPEG_OPT_ERR` is the error code returned for input outside the 1 to 65535 range.

- Specify the texture format in `texFormat`

`TMCC_JPEG_TEXT_ERR` is the error code returned if anything other than `GX_TF_RGB565` or `GX_TF_RGBA8` is specified.

- `poutputSize` stores the size that can be output using the specified `QFactor`
- Set the image quality factor in `QFactor`
  For information on the image quality factor, see section 2.3.1.1 Image Quality Factor.
  `TMCC_JPEG_OPT_ERR` is the error code returned if input is outside the 1 to 500 range.

Return Values

0 if successful termination; error code if negative.

Zero is returned if execution ends normally; error code (negative value) is returned if execution ends abnormally. For error codes, see section 5.1 Error Codes.

Description

This function determines the actual output size for the given image quality factor. It returns the actual output size (`*poutputSize`) when the input texture image data is compressed under the specified image quality factor (`QFactor`).

3.2 JPEG Decoding Functions

3.2.1 TMCCJPEGDecDecode

Code 3-4 TMCCJPEGDecDecode Function

```
s32 TMCCJPEGDecDecode(
        void *pWorkArea,
        void *pJpegStream,
        u32 jpegStreamSize,
        void *pTexBuffer,
        u32 texBufferSize,
        GXTexFmt texFormat,
        s32 resolution,
        u8 thumbnail
);
```

Arguments

`void *pWorkArea`	[in] Start address of work area
`void *pJpegStream`	[in] Start address of JPEG data
`u32 jpegStreamSize`	[in] Size of JPEG data
`void *pTexBuffer`	[out] Start address of texture buffer
`u32 texBufferSize`	[in] Size of texture buffer
`GXTexFmt texFormat`	[in] Texture format
`s32 resolution`	[in] Resolution reduction parameter
`u8 thumbnail`	[in] Toggle for expanding thumbnail data or not

- Set the start address of the work area in `pWorkArea`
- Set the start address of the input JPEG data in `pJpegStream`

  JPEG data's start address must be 32-byte aligned; if the address is not 32-byte aligned, error code `TMCC_JPEG_OPT_ERR` is returned.
- Specify the size of input JPEG data in `jpegStreamSize`
- Specify the start address of work memory for the output texture data in `pTexBuffer`
- Specify the size of work memory for the output texture data in `texBufferSize`

  `TMCC_JPEG_ERR_BUFFER` is the error code returned if it appears that the output texture size may exceed this work memory size.
- Specify the texture format in `texFormat`

  `TMCC_JPEG_TEXT_ERR` is the error code returned if anything other than `GX_TF_RGB565` or `GX_TF_RGBA8` is specified.
- Set the resolution reduction conversion level in `resolution`.

  For information on the resolution reduction conversion level, see section 2.3.2.4 Specifying the Resolution Reduction Level. `TMCC_JPEG_OPT_ERR` is the error code returned if a value other than 1, 2, 4, or 8 is specified.
- Enter a value of `1` for `thumbnail` when enlarging the thumbnail

  The main image is enlarged when `0` is specified.

  For information on enlarging thumbnails, see section 2.3.2.2 Enlarging Thumbnails. `TMCC_JPEG_OPT_ERR` is the error code returned if a value other than 0 or 1 is entered.

Return Values

0 if successful termination; error code if negative.

Zero is returned if execution ends normally; error code (negative value) is returned if execution ends abnormally. For error codes, see section 5.1 Error Codes.

Description

Enlarges the input JPEG data and outputs texture data.

3.2.2 TMCCJPEGDecDecodeEx

Code 3-5 TMCCJPEGDecDecodeEx Function

```
s32 TMCCJPEGDecDecodeEx(
        void *pWorkArea,
        void *pJpegStream,
        u32 jpegStreamSize,
        void *pTexBuffer,
        u32 texBufferSize,
        GXTexFmt texFormat,
        s32 resolution,
        u16 srcX,
        u16 srcY,
        u16 srcWidth,
        u16 srcHeight
);
```

Arguments

`void *pWorkArea`	[in] Start address of work area
`void *pJpegStream`	[in] Start address of JPEG data
`u32 jpegStreamSize`	[in] Size of JPEG data
`void *pTexBuffer`	[out] Start address of texture buffer
`u32 texBufferSize`	[in] Size of texture buffer
`GXTexFmt texFormat`	[in] Texture format
`s32 resolution`	[in] Resolution reduction parameter
`u16 srcX`	[in] Upper left X-coordinate of the area to be enlarged
`u16 srcY`	[in] Upper left Y-coordinate of the area to be enlarged
`u16 srcWidth`	[in] Width of area to be enlarged
`u16 srcHeight`	[in] Height of area to be enlarged

- Set the start address of the work area in `pWorkArea`
- Set the start address of the input JPEG data in `pJpegStream`
  The JPEG data's start address must be 32-byte aligned; if the data is not 32-byte aligned, error code `TMCC_JPEG_OPT_ERR` is returned.
- Specify the size of input JPEG data in `jpegStreamSize`
- Specify the start address of work memory for the output texture data in `pTexBuffer`
- Specify the size of work memory for the output texture data in `texBufferSize`
  `TMCC_JPEG_ERR_BUFFER` is the error code returned if it appears that the output texture size may exceed this work memory size.
- Specify the texture format in `texFormat`
  `TMCC_JPEG_TEXT_ERR` is the error code returned if anything other than `GX_TF_RGB565` or `GX_TF_RGBA8` is specified.
- Set the resolution reduction conversion level in `resolution`
  For information on the resolution reduction conversion level, see section 2.3.2.4 Specifying the Resolution Reduction Level. `TMCC_JPEG_OPT_ERR` is the error code returned if a value other than 1, 2, 4, or 8 is specified.
- Specify the upper left X-coordinate of the area to be enlarged in `srcX`
- Specify the upper left Y-coordinate of the area to be enlarged in `srcY`
- Specify the width of the area to be enlarged in `srcWidth`
- Specify the height of the area to be enlarged in `srcHeight`
  For information on these four arguments, see section 2.3.2.3 Specifying the Area to Be Enlarged.

Return Values

0 if successful termination; error code if negative.

Zero is returned if execution ends normally; error code (negative value) is returned if execution ends abnormally. For error codes, see section 5.1 Error Codes.

Description

This function enlarges the input JPEG data and outputs texture data. It is possible to specify an area in the source image to be enlarged. For details on specifying the area to be enlarged, see section 2.3.2.3 Specifying the Area to Be Enlarged.

3.2.3 TkMCCJPEGDecGetSize

Code 3-6 TMCCJPEGDecGetSize Function

```
s32 TMCCJPEGDecGetSize(
        void *pWorkArea,
        void *pJpegStream,
        u32 jpegStreamSize,
        GXTexFmt texFormat,
        s32 resolution,
        u8 thumbnail,
        TMCCJPEGJpegSize *jpegSize
);
```

Arguments

`void *pWorkArea`	[in] Start address of work area
`void *pJpegStream`	[in] Start address of JPEG data
`u32 jpegStreamSize`	[in] Size of JPEG data
`GXTexFmt texFormat`	[in] Texture format
`s32 resolution`	[in] Resolution reduction parameter
`u8 thumbnail`	[in] Toggle for expanding thumbnail data or not
`TMCCJPEGJpegSize *jpegSize`	[out] Pointer to the image size information structure

- Set the start address of the work area in `pWorkArea`
- Set the start address of the input JPEG data in `pJpegStream`
  The start address of the JPEG data must be 32-byte aligned; if the data is not 32-byte aligned, error code `TMCC_JPEG_OPT_ERR` is returned.
- Specify the size of input JPEG data in `jpegStreamSize`
- Specify the texture format in `texFormat`
  `TMCC_JPEG_TEXT_ERR` is the error code returned if anything other than `GX_TF_RGB565` or `GX_TF_RGBA8` is specified.
- Set the resolution reduction conversion level in `resolution`
  For information on the resolution reduction conversion level, see section 2.3.2.4 Specifying the Resolution Reduction Level. `TMCC_JPEG_OPT_ERR` is the error code returned if a value other than 1, 2, 4, or 8 is specified.
- Enter a value of `1` for `thumbnail` when enlarging the thumbnail. The main image is enlarged when `0` is specified.
  For information on enlarging thumbnails, see section 2.3.2.2 Enlarging Thumbnails. `TMCC_JPEG_OPT_ERR` is the error code returned if a value other than 0 or 1 is entered.
- `jpegSize` stores the size information of the JPEG data
  For information on the `TMCCJPEGJpegSize` structure, see section 4.1 TMCCJPEGJpegSize.

Return Values

0 if successful termination; error code if negative.

Zero is returned if execution ends normally; error code (negative value) is returned if execution ends abnormally. For error codes, see section 5.1 Error Codes.

Description

This function gets the number of valid pixels in the input JPEG data.

3.2.4 TMCCJPEGDecGetSizeEx

Code 3-7 TMCCJPEGDecGetSizeEx Function

```
s32 TMCCJPEGDecGetSizeEx(
        void *pWorkArea,
        void *pJpegStream,
        u32 jpegStreamSize,
        GXTexFmt texFormat,
        s32 resolution,
        u16 srcX,
        u16 srcY,
        u16 srcWidth,
        u16 srcHeight,
        TMCCJPEGJpegSize *jpegSize
);
```

Arguments

`void *pWorkArea`	[in] Start address of work area
`void *pJpegStream`	[in] Start address of JPEG data
`u32 jpegStreamSize`	[in] Size of JPEG data
`GXTexFmt texFormat`	[in] Texture format
`s32 resolution`	[in] Resolution reduction parameter
`u16 srcX`	[in] Upper left X-coordinate of area to be enlarged
`u16 srcY`	[in] Upper left Y-coordinate of the area to be enlarged
`u16 srcWidth`	[in] Width of area to be enlarged
`u16 srcHeight`	[in] Height of area to be enlarged
`TMCCJPEGJpegSize *jpegSize`	[out] Pointer to the image size information structure

- Set the start address of the work area in `pWorkArea`
- Set the start address of the input JPEG data in `pJpegStream`
  JPEG data's start address must be 32-byte aligned; if the data is not 32-byte aligned, the error code `TMCC_JPEG_OPT_ERR` is returned.
- Specify the size of input JPEG data in `jpegStreamSize`
- Specify the texture format in `texFormat`
  Error code `TMCC_JPEG_TEXT_ERR` is returned if anything other than `GX_TF_RGB565` or `GX_TF_RGBA8` is specified.

- Set the resolution reduction conversion level in `resolution`
  For information on the resolution reduction conversion level, see section 2.3.2.4 Specifying the Resolution Reduction Level. `TMCC_JPEG_OPT_ERR` is the error code returned if a value other than 1, 2, 4, or 8 is specified.
- Specify the upper left X-coordinate of the area to be enlarged in `srcX`
- Specify the upper left Y-coordinate of the area to be enlarged in `srcY`
- Specify the width of the area to be enlarged in `srcWidth`
- Specify the height of the area to be enlarged in `srcHeight`
  For information on these four arguments, see section 2.3.2.3 Specifying the Area to Be Enlarged.
- `jpegSize` stores the size information of the JPEG data.
  For information on the `TMCCJPEGJpegSize` structure, see section 4.1 TMCCJPEGJpegSize.

Return Values

0 if successful termination; error code if negative.

Zero is returned if execution ends normally, while an error code (negative value) is returned if execution ends abnormally. For error codes, see section 5.1 Error Codes.

Description

This function gets the number of pixels in the specified area in the input JPEG data. For details on specifying the area to be enlarged, see section 2.3.2.3 Specifying the Area to Be Enlarged.

3.2.5 TMCCJPEGDecGetExif

Code 3-8 TMCCJPEGDecGetExif Function

```
s32 TMCCJPEGDecGetExif(
        void *pWorkArea,
        void *pJpegStream,
        u32 jpegStreamSize,
        TMCCJPEGExifInfo *exifInfo
);
```

Arguments

`void *pWorkArea`	[in] Start address of work area
`void *pJpegStream`	[in] Start address of JPEG data
`u32 jpegStreamSize`	[in] Size of JPEG data
`TMCCJPEGExifInfo *exifInfo`	[out] `Exif` information

- Set the start address of the work area in `pWorkArea`
- Set the start address of the input JPEG data in `pJpegStream`
  The start address of the JPEG data must be 32-byte aligned. `TMCC_JPEG_OPT_ERR` is the error code returned if the data is not 32-byte aligned.
- Specify the size of input JPEG data in `jpegStreamSize`
- `exifInfo` stores `Exif` information
  For information on the `TMCCJPEGExifInfo` structure, see section 4.2 TMCCJPEGExifInfo.

Return Values

0 if successful termination; error code if negative.

Zero is returned if execution ends normally; error code (negative value) is returned if execution ends abnormally. For error codes, see section 5.1 Error Codes.

Description

This function gets the `Exif` information in the input JPEG data. For details on the obtained `Exif` information, see section 2.3.2.1 Getting Exif Information. `TMCC_JPEG_NG` is the error code returned if there is no `Exif` information in the input data.

3.3 Getting the Middleware Version

```
u32 TMCCJPEGGetVer();
```

Arguments

None.

Return Values

Returns the current version of this Middleware.

Description

Gets the current version of this Middleware.

4 Structures

4.1 TMCCJPEGJpegSize

Code 4-1 TMCCJPEGJpegSize Structure

```
typedef struct {
        u16 width;           /* Width of decompressed image */
        u16 height;          /* Height of decompressed image */
        u32 texBufferSize;   /* texture output size */
} TMCCJPEGJpegSize;
```

Description

This structure stores JPEG size information parameters. It is used as an argument by the `TMCCJPEGDecGetSize` and `TMCCJPEGDecGetSizeEx` functions. The `width` and `height` member variables represent the width and height of the JPEG image and may differ from the texture data size. The `texBufferSize` member variable represents the size of the work memory used for output texture data. This is used as the size of the output work memory allocated when the `TMCCJPEGDecDecode` or `TMCCJPEGDecDecodeEx` decoding functions are called.

4.2 TMCCJPEGExifInfo

Code 4-2 TMCCJPEGExifInfo Structure

```
typedef struct {
        TMCCJPEGExifIFD0Info image;      /* Tag data for main image */
        TMCCJPEGExifIFD1Info thumb;      /* Tag data for thumbnail image */
} TMCCJPEGExifInfo;
```

Description

This structure stores `Exif` information parameters. As member variables, it includes `TMCCJPEGExifIFD0Info`, the structure for storing main image information, and `TMCCJPEGExifIFD1Info`, the structure for storing thumbnail image information.

For `Exif` information specifications, see section 2.3.2.1 Getting Exif Information.

4.2.1 TMCCJPEGExifIFD0Info

Code 4-3 TMCCJPEGExifIFD0Info Structure

```
typedef struct {
    /* Image title (ASCII) */
        s8 *ImageDescription;
    /* Name of manufacturer of image input device (ASCII) */
        s8 *Make;
    /* Model name of image input device (ASCII) */
        s8 *Model;
    /* Image direction (unsigned 16-bit integer) */
        u16 Orientation;
    /* Horizontal image resolution (unsigned 32-bit fraction, [0]/[1]) */
        u32 XResolution[2];
```

```
    /* Vertical image resolution (unsigned 32-bit fraction, [0]/[1]) */
       u32 YResolution[2]
    /* Horizontal and vertical image resolution unit (unsigned 16-bit integer) */
       u16 ResolutionUnit;
    /* Playback gradation curve properties (unsigned 16-bit integer 3*256) */
       u16 TransferFunction[3][256];
    /* File modification date and time (ASCII"YYYY:MM:DD HH:MM:SS") */
       s8 DateTime[20];
    /* YCC pixel construction <Y and C position> (unsigned 16-bit integer) */
       u16 YCbCrPositioning;
    /* Exif tag */
       u32 ExifIFDPointer;
    /* Exif version (ASCII string without NULL character) */
       s8 ExifVersion[4];
    /* Meaning of each component (unsigned 8-bit integer) */
       u8 ComponentsConfiguration[4];
    /* Corresponding Flash Pix version (ASCII string without NULL character) */
       s8 FlashPixVersion[4];
    /* Color space information (unsigned 16-bit integer) */
       u16 ColorSpace;
    /* Effective image width (unsigned 32- or 16-bit integer */
       u32 PixelXDimension;
    /* Effective image height (unsigned 32- or 16-bit integer */
       u32 PixelYDimension;
} TMCCJPEGExifIFD0Info;
```

Description

This structure stores `Exif` information related to the main image.

For `Exif` information specifications, see section 2.3.2.1 Getting Exif Information.

4.2.2 TMCCJPEGExifIFD1Info

Code 4-4 TMCCJPEGExifIFD1Info Structure

```
typedef struct{
   /* Compression type (unsigned 16-bit integer) */
      u16 Compression;
   /* Horizontal image resolution (unsigned 32-bit fraction, [0]/[1]) */
      u32 XResolution[2];
   /* Vertical image resolution (unsigned 32-bit fraction, [0]/[1]) */
      u32 YResolution[2];
   /* Horizontal and vertical image resolution unit (unsigned 16-bit integer) */
      u16 ResolutionUnit;
   /* Offset to JPEG SOI (unsigned 32-bit integer) */
      u32 JPEGInterchangeFormat;
   /* JPEG data length in bytes (unsigned 32-bit integer) */
      u32 JPEGInterchangeFormatLength;
} TMCCJPEGExifIFD1Info;
```

Description

This structure stores `Exif` information related to the thumbnail image.

For `Exif` information specifications, see section 2.3.2.1 Getting Exif Information.

5 Constant Definitions

5.1 Error Codes

The error codes returned by this version of Middleware are shown below.

- Code returned when execution ends normally:

```
TMCC_JPEG_OK                    0x00      /* Normal end */
```

- Two error codes returned when incorrect parameter values are passed as arguments to API functions:

```
TMCC_JPEG_OPT_ERR               -0x01     /* User option error */
TMCC_JPEG_TEXT_ERR              -0x04     /* Texture format error */
```

- Code returned when the encoder output size exceeds the output buffer size; increase the buffer size when this error is encountered:

```
TMCC_JPEG_ENC_BUF_OVER          -0x03     /* Output size limit during encoding */
```

- Codes caused by problems with input JPEG stream data:

```
TMCC_JPEG_NG                    -0x02     /* Abnormal termination */
TMCC_JPEG_ERR_SOF               -0x10     /* SOF marker error */
TMCC_JPEG_ERR_SOI               -0x20     /* SOI marker error */
TMCC_JPEG_ERR_EOI               -0x21     /* EOI marker error */
TMCC_JPEG_ERR_SOS               -0x22     /* SOS marker error */
TMCC_JPEG_ERR_RSTm              -0x23     /* RSTm marker error */
TMCC_JPEG_ERR_NOMC              -0x2f     /* unsupported marker encountered */
TMCC_JPEG_ERR_DHT               -0x40     /* DHT error */
TMCC_JPEG_ERR_DQT               -0x41     /* DQT error */
TMCC_JPEG_ERR_DRI               -0x42     /* DRI error */
TMCC_JPEG_ERR_DNL               -0x43     /* DNL error */
TMCC_JPEG_ERR_COM               -0x44     /* COM error */
TMCC_JPEG_ERR_APPn              -0x45     /* APP0-F error */
TMCC_JPEG_ERR_FHEADER           -0x50     /* SOF header error */
TMCC_JPEG_ERR_SHEADER           -0x51     /* SOS header error */
TMCC_JPEG_ERR_BITSTREAM         -0x64     /* JPEG stream error */
TMCC_JPEG_ERR_FORMAT            -0x70     /* JPEG unsupported format error */
TMCC_JPEG_ERR_EXIF_BUFFER       -0x91     /* EXIF buffer error */
TMCC_JPEG_ERR_BUFFER            -0x90     /* JPEG buffer error */
TMCC_JPEG_ERR_NO_THUMBNAIL      -0xA0     /* Thumbnail does not exist */
TMCC_JPEG_ERR_EXIF_FORMAT       -0xA1     /* EXIF configuration error */
TMCC_JPEG_RET_ERR_USRFUNC       -0xF0     /* User-defined function error */
TMCC_JPEG_ERR_JPEG_COPY_BUFFER  -0xF1     /* Error while copying data */
```

5.2 Other Constants

Other constants used by this version of Middleware are shown below.

5.2.1 Memory Size

```
/* Parameter work size (in bytes) used when encoding or decoding */
   TMCC_JPEG_WORKAREA_SIZE              0x1C00
/* Minimum size (in bytes) of the output JPEG stream buffer used when encoding */
   TMCC_JPEG_JPEGBUFFER_MINIMAM_SIZE    0x25F
```