To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: http://www.renesas.com

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (http://www.renesas.com)

Send any inquiries to http://www.renesas.com/inquiry.

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: “Standard”, “High Quality”, and “Specific”. The recommended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as “Specific” without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as “Specific” or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is “Standard” unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

   “Standard”: Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.

   “High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.

   “Specific”: Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

RENESAS

Application Note

Multimedia Processor for Mobile Applications

Image Processor Unit (IPU)

EMMA Mobile 1

Document No. S19898EJ1V0AN00
Date Published Aug, 2009

PREFACE

Purpose The purpose of this document is to introduce the usage of EMMA Mobile 1 Image Processor Unit.

Organization This document includes the following:

- Chapter 1. Overview
- Chapter 2. Usage of Image Processor Unit
- Chapter 3. Example of Image Processor Unit Operation
- Appendix. Image Processor Unit Driver Function

Notation Here explains the meaning of following words in text:

Note	Explanation of item indicated in the text
Caution	Information to which user should afford special attention
Remark	Supplementary information

Related document The following tables list related documents.

Reference Document

Document Name	Version/date	Author	Description
S19268EJ1V0UM00_1chip.pdf	1st edition	NECEL	User's Manual
S19265EJ1V0UM00_ASMUGIO.pdf	1st edition	NECEL	User's Manual
S19264EJ1V0UM00_IPU.pdf	1st edition	NECEL	User's Manual
S19907EJ1V0AN00_GD.pdf f	1st edition	NECEL	GD Spec
S19899EJ1V0AN00_LCD.pdf	1st edition	NECEL	Application Note
S19906EJ1V0AN00_IMC.pdf	1st edition	NECEL	Application Note

Disclaimers

- **The information contained in this document is subject to change without prior notice in the future. Refer to the latest applicable data sheet(s) and User's Manual when designing a product for mass production.**
- No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Electronics. NEC Electronics assumes no responsibility for any errors that may appear in this document.
- NEC Electronics does not assume any liability for infringement of patents, copyrights or other intellectual property rights of third parties by or arising from the use of NEC Electronics products listed in this documents or any other liability arising from the use of such products. No license, express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC Electronics or others.
- Descriptions of circuits, software and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The incorporation of these circuits, software and information in the design of a customers' equipment shall be done under the full responsibility of the customer. NEC Electronics assume no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.
- While NEC Electronics endeavors to enhance the quality, reliability and safety of NEC Electronics products, customers agree and acknowledge that possibility of defects thereof cannot be eliminated entirely. To minimize risks of damage to property or injury (including death) to persons arising from defects in NEC Electronics products, customers must incorporate sufficient safety measures in their design, such as redundancy, fire-containment and anti-failure features.

Note)

1. "NEC Electronics" as used in this document means NEC Electronics Corporation and also includes its majority-owned subsidiaries.
2. "NEC Electronics products" means any product developed or manufactured by or for NEC Electronics (as defined above)
3. All trademarks or registered trademarks are the property of their respective owners. Registered trademarks ® and trademarks™ are not noted in this document.

CONTENTS

Chapter 1 Overview 9

1.1 Introduction 9
1.2 Development Environment 9

Chapter 2 Usage of Image Processor Unit 10

2.1 Image Processor Unit Function 10
2.2 Normal Procedure of IPU - Rotator 14
2.3 Detail of IPU-ROT Normal Procedure 15
2.3.1 IPU (Rotator) Initialization 15
2.3.2 Get Source Image Data 15
2.3.3 Configure Rotator Registers 15
2.3.4 Start Rotation 16
2.3.5 Rotation Processing 16
2.3.6 Display the Rotated Image 16
2.3.7 Reset IPU (Rotator) 16
2.4 Normal Procedure of IPU - Image Processor 17
2.5 Detail of IPU-IMG Normal Procedure 18
2.5.1 IPU (IMG) Initialization 18
2.5.2 Get Source Image Data 18
2.5.3 Configure Image Processor Registers 18
2.5.4 Start IMG Processing 20
2.5.5 Image Processing 20
2.5.6 Display the Destination Image 20
2.5.7 Reset IPU (Image Processor) 20
2.6 Normal Procedure of IPU – Graphics DMA 21
2.7 Detail of IPU-GDMA Normal Procedure 22
2.7.1 IPU (GDMA) Initialization 22
2.7.2 Get Source Image Data 22
2.7.3 Configure GDMA Registers 22
2.7.4 Start GDMA Processing 23
2.7.5 GDMA Processing 23
2.7.6 Display the Destination Image 23
2.7.7 Reset IPU (GDMA) 24

Chapter 3 Example of IPU Operation 25

3.1 Outline of IPU Operation Example 25
3.1.1 Data Flow Chart of IPU Examples 26
3.1.2 IPU Initialization and Reset 27
3.2 IPU Sample – Rotator 28

3.2.1 Operation Flow 29
3.2.2 Operation Detail 30
3.2.2.1 IPU Initialization 30
3.2.2.2 Initialize LCD and IMC 30
3.2.2.3 Read Source Data 30
3.2.2.4 Display the Source Image 30
3.2.2.5 Configure ROT Registers for Rotation 31
3.2.2.6 Start Rotator Processing 37
3.2.2.7 Rotator Processing 37
3.2.2.8 Display the rotated image in LCD panel 37
3.2.2.9 Reset IPU 37
3.2.2.10 Release LCD and IMC 37
3.3 IPU Sample – Image Processor Function 38
3.3.1 Operation Flow 39
3.3.2 Operation Detail 40
3.3.2.1 IPU Initialization 40
3.3.2.2 Initialize LCD and IMC 40
3.3.2.3 Read Source Data 40
3.3.2.4 Configure IMG 41
3.3.2.5 Start Image Processing 45
3.3.2.6 IMG Processing 45
3.3.2.7 Display the processed image in LCD panel 45
3.3.2.8 Reset IPU 45
3.3.2.9 Release LCD and IMC 45
3.4 IPU Sample – GDMA ROP Function 46
3.4.1 Operation Flow 47
3.4.2 Operation Detail 48
3.4.2.1 IPU Initialization 48
3.4.2.2 Initialize LCD and IMC 48
3.4.2.3 Read Source Data 48
3.4.2.4 Configure GDMA 49
3.4.2.5 Start GDMA Processing 50
3.4.2.6 GDMA Processing 50
3.4.2.7 Display the processed image in LCD panel 50
3.4.2.8 Reset IPU 50
3.4.2.9 Release LCD and IMC 50

Appendix A Image Processor Unit Driver Function 51

A.1 IPU API function list 51
A.2 Type Define 51
A.2.1 Naming rule and coding rule 51
A.2.2 Structure 51
A.3 Function Detail 52

A.3.1 Initialization Function 52
A.3.2 Reset IPU Function 54
A.3.3 Enable/Disable ROT Register Update Function 55
A.3.4 Configure ROT Registers 56
A.3.5 Start ROT Processing 57
A.3.6 Get ROT Status 58
A.3.7 Enable/Disable IMG Register Update Function 59
A.3.8 Configure IMG Registers 60
A.3.9 Start IMG Processing 61
A.3.10 Get IMG Status 62
A.3.11 Enable/Disable GDMA Register Update Function 63
A.3.12 Configure ROT Registers 64
A.3.13 Start GDMA Processing 65
A.3.14 Get GDMA Status 66

ANNEX Modification History 67

LIST OF TABLES

Table 1-1 Hardware Environment 9
Table 1-2 Software Environment..... 9
Table 2-1 IPU-ROT Function 10
Table 2-2 IPU-IMG Resizer Function 11
Table 2-3 IPU-IMG Converter Function 11
Table 2-4 IPU-IMG Color Dithering Function 12
Table 2-5 IPU-IMG Pixel Packing Function 12
Table 2-6 IPU-GDMA Function 13
Table 3-1 List of IPU Examples..... 25
Table 3-2 IPU-ROT Register Setting for YUV422 90° Rotation 31
Table 3-3 IPU-ROT Register Setting for YUV422 180° Rotation 32
Table 3-4 IPU-ROT Register Setting for YUV422 270° Rotation 32
Table 3-5 IPU-ROT Register Setting for YUV420 90° Rotation 33
Table 3-6 IPU-ROT Register Setting for YUV420 180° Rotation 34
Table 3-7 IPU-ROT Register Setting for YUV420 270° Rotation 34
Table 3-8 IPU-ROT Register Setting for RGB565 90° Rotation..... 35
Table 3-9 IPU-ROT Register Setting for RGB565 180° Rotation..... 36
Table 3-10 IPU-ROT Register Setting for RGB565 270° Rotation..... 36
Table 3-11 Setting IPU-IMG Registers for IPU-IMG Sample 1 41
Table 3-12 Setting IPU-IMG Registers for IPU-IMG Sample 2 43
Table 3-13 Setting IPU-GDMA Registers for IPU-GDMA Sample..... 49
Table A-1 IPU Driver Function List..... 51
Table A-2 Structure List 51

LIST OF FIGURES

Figure 2-1 Normal Procedure of IPU-ROT 14
Figure 2-2 Normal Procedure of IPU-IMG 17
Figure 2-3 Normal Procedure of IPU-GDMA 21
Figure 3-1 Data Flow Chart of IPU Examples 26
Figure 3-2 Operation Flow of IPU Initialization and Reset 27
Figure 3-3 Operation Flow of IPU Rotator Sample 29
Figure 3-4 Operation Flow of IPU Image Processor Sample 39
Figure 3-5 Operation Flow of IPU Graphics DMA Sample 47
Figure A-1 IPU Initialization 53
Figure A-2 IPU Reset 54
Figure A-3 Enable/Disable ROT Register Update Function 55
Figure A-4 ROT Configure Function 56
Figure A-5 Start ROT Processing 57
Figure A-6 Get ROT Status 58
Figure A-7 Enable/Disable IMG Register Update Function 59
Figure A-8 IMG Configure Function 60
Figure A-9 Start IMG Processing 61
Figure A-10 Get IMG Status 62
Figure A-11 Enable/Disable GDMA Register Update Function 63
Figure A-12 GDMA Configure Function 64
Figure A-13 Start GDMA Processing 65
Figure A-14 Get GDMA Status 66

Chapter 1 Overview

1.1 Introduction

In this document, the below contents of EMMA Mobile 1 Image Processor Unit (IPU) will be described.

1) the normal process procedure of Image Processor Unit
2) usage sample of Image Processor Unit

As additional, the EMMA Mobile 1 Image Processor Unit driver interface of EMMA Mobile 1 evaluation program will be explained in Appendix.

About detail of Image Processor Unit, please refer to "**EMMA Mobile 1 IPU User's Manual**".

1.2 Development Environment

- Hardware environment of this project is listed as below.

Table 1-1 Hardware Environment

Name	Version	Maker
EMMA Mobile 1 evaluation board (PSKCH2Y-S-0016-01)	-	NEC Electronics
PARTNER-Jet ICE ARM	M20	Kyoto Microcomputer Co. Ltd

- Software used in this project is listed as below.

Table 1-2 Software Environment

Name	Version	Maker
GNUARM Toolchain	V4.3.2	GNU
WJETSET-ARM	V5.10a	Kyoto Microcomputer Co. Ltd

Chapter 2 Usage of Image Processor Unit

2.1 Image Processor Unit Function

EMMA Mobile 1 Image Processor Unit (IPU) supports the below functions:

1) image rotation;

Table 2-1 IPU-ROT Function

Item	Valuable Setting
Rotate angle	0º, 90º, 180º, or 270º
Image size	Max: 4092 pixels x 4092 pixels
Input data format	YUV422 (Semi-Planar/Interleave) YUV420 (Semi-Planar/Planar) RGB565 RGB888
Output data format	YUV422 (Semi-Planar/Interleave) YUV420 (Semi-Planar/Planar) RGB565 RGB888
32bit data endian switch	Little Endian and Big Endian
Byte lane switch	32-bit units; U,V or Y, UV order

Note:

About input data format and output data format relation and limitation, please refer "**EMMA Mobile 1 IPU User's Manual**".

2) image resize (enlarge or reduce)

Table 2-2 IPU-IMG Resizer Function

Item	Valuable Setting
Image size NOTE	Max: 8188 pixels x 8188 pixels
Input data format	YUV422 (Semi-Planar/Planar/Interleave) YUV420 (Semi-Planar/Planar) RGB565 RGB888
Output data format	YUV422 (Semi-Planar/Planar/Interleave) YUV420 (Semi-Planar/Planar) RGB565 RGB666 RGB888
Resize range	Horizontal: 1/40 ~ x4 Vertical: 1/40 ~ x4

Note:

For each data format (such as YUV420/YUV422/RGB565/RGB888 etc), the size limitation is different. More detail, please refer "**EMMA Mobile 1 IPU User's Manual**".

3) image color space converter (YUV ←→ RGB)

Table 2-3 IPU-IMG Converter Function

Item	Valuable Setting
Image size	Max: 8188 pixels x 8188 pixels
Input data format	YUV422/420 RGB565/888
Output data format	YUV422/420 RGB565/888 RGB666 (with pixel packing)
Brightness setting	Enable for YUV → RGB conversion

4) color dithering

Table 2-4 IPU-IMG Color Dithering Function

Item	Valuable Setting
Image size	Max: 8188 pixels x 8188 pixels
Input data format	RGB565/888 YUV422/420 (only when use YUV → RGB conversion)
Output data format	RGB565 RGB666 (with pixel packing)
Brightness setting	Enable for YUV → RGB conversion

5) pixel packing (RGB565/888 → RGB666)

Table 2-5 IPU-IMG Pixel Packing Function

Item	Valuable Setting
Image size	Max: 8188 pixels x 8188 pixels
Input data format	RGB565/888 YUV422/420 (only when use YUV → RGB conversion)
Output data format	RGB666

6) overlay

Overlay the front-image and back-image, can be used with the other functions of IPU. Support color mask and transparency.

7) Vertical/horizontal mirror flip

8) Endian switch

9) Byte lane switch

10) Graphics DMA function

This function can improve the image processing speed.

Supports the below operations:

- Memory to memory (M2M): copy and rectangle fill function;
- Raster operation;
- Color mask function;
- Byte lane switch function;

Table 2-6 IPU-GDMA Function

Item	Valuable Setting
Image size	Max: 8188 pixels x 8188 pixels
Input data format	RGB565/666/888 YUV422 (Semi-Planar/Planar/Interleave) YUV420 (Semi-Planar/Planar)
Output data format	RGB565/666/888 YUV422 (Semi-Planar/Planar/Interleave) YUV420 (Semi-Planar/Planar)

2.2 Normal Procedure of IPU - Rotator

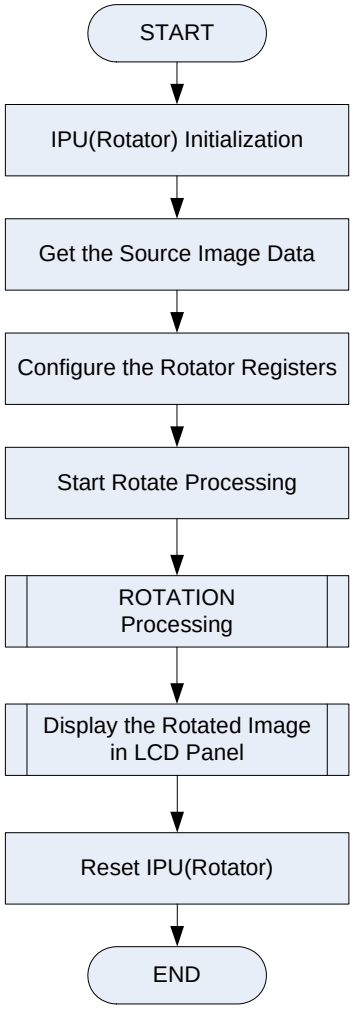

Figure 2-1 Normal Procedure of IPU-ROT

Note:

1. About the explanation of all the Rotator registers mentioned in this document, please refer to "**EMMA Mobile 1 IPU User's Manual**".
2. About the explanation of all the ASMU registers mentioned in this document, please refer to "**EMMA Mobile 1 SMU/GIO Interface User's Manual**".

2.3 Detail of IPU-ROT Normal Procedure

2.3.1 IPU (Rotator) Initialization

When do rotator initialization, cancel reset IPU clock by setting ASMU registers as below.
Related Register:

AHBCLKCTRL0;
APBCLKCTRL0;
GCLKCTRL0ENA;
GCLKCTRL0;
RESETREQ0ENA;
RESETREQ0;

Caution:
For EMMA Mobile 1 IPU interface, the clock should be supplied to the whole interface at the same time. So when supply clock for Rotator, the clock of Image Processor and Graphics DMA are supplied also.

2.3.2 Get Source Image Data

For each sample, the related YUV or RGB image will be read as the source data.

2.3.3 Configure Rotator Registers

Before start rotation, rotator registers need to be configured, such as source data address, destination address, rotator mode, image data format etc.
The following registers need to be set as necessary in random sequence.
Related Registers:

ROT_DUAL_FF;
ROT_MODE;
ROT_FRAME;
ROT_SRCSIZE;
ROT_DSTSIZE;
ROT_SRCYADR_A/B/C;
ROT_SRCUVADR_A/B/C;
ROT_SRCVADR_A/B/C;
ROT_DSTYADR_A/B/C;
ROT_DSTUVADR_A/B/C;
ROT_DSTVADR_A/B/C;
ROT_SRCHSIZE;
ROT_SRCVSIZE;
ROT_ENSET;

ROT_FORMAT;
ROT_SRCBYTE;
ROT_DSTBYTE;

2.3.4 Start Rotation

Issue the rotation processing request to start rotator.
Related Register:
ROT_REQ

2.3.5 Rotation Processing

The source data will be rotated according the IPU-ROT registers' configuration. The rotated image data will be store in the specified destination address.
When rotation is completed, the value of register "ROT_STATUS" will cleared ("1" → "0"). And a processing end interrupt is issued.
Gets the rotation processing result, by checking the processing status and the interrupt status.

2.3.6 Display the Rotated Image

Check the operation result by display the rotated image in LCD panel.
EMMA Mobile 1 LCD controller only supports RGB565 and RGB666 format data input.

Remark:
In this step, some IMC and LCDC modules' interface will be called for the displaying.
More detail about these two modules, please refer to "**EMMA Mobile 1 IMC Application Note**" and "**EMMA Mobile 1 LCDC Application Note**".

2.3.7 Reset IPU (Rotator)

Before exit from rotation operation, reset IPU module.
Related Registers:
AHBCLKCTRL0;
APBCLKCTRL0;
GCLKCTRL0ENA;
GCLKCTRL0;
RESETREQ0ENA;
RESETREQ0;

2.4 Normal Procedure of IPU - Image Processor

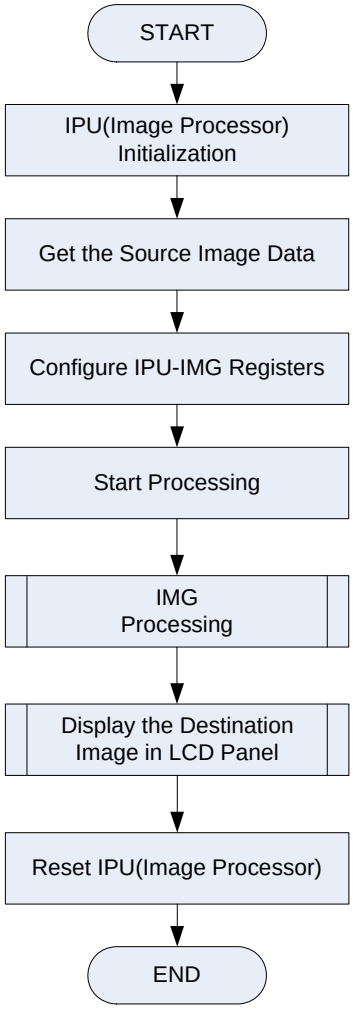

Figure 2-2 Normal Procedure of IPU-IMG

Note:

1. About the explanation of all the IPU-IMG registers mentioned in this document, please refer to "**EMMA Mobile 1 IPU User's Manual**".
2. About the explanation of all the ASMU registers mentioned in this document, please refer to "**EMMA Mobile 1 SMU/GIO Interface User's Manual**".

2.5 Detail of IPU-IMG Normal Procedure

2.5.1 IPU (IMG) Initialization

When do IMG initialization, cancel reset IPU clock by setting ASMU registers as below.

Related Register:

AHBCLKCTRL0;
APBCLKCTRL0;
GCLKCTRL0ENA;
GCLKCTRL0;
RESETREQ0ENA;
RESETREQ0;

Note:

The operations of this step are same with" **2.3.1 IPU (Rotator) Initialization**"

2.5.2 Get Source Image Data

For each sample, the related YUV or RGB image will be read as the source data.

2.5.3 Configure Image Processor Registers

Before start image processor, some registers need to be configured, such as source data address, destination address, function mode, image data format etc.

The following registers need to be set as necessary in random sequence.

Related Registers:

IMG_DUAL_FF;
IMG_MODE;
IMG_FORMAT;
IMG_SRCSIZE_B/F;
IMG_DSTSIZE;
IMG_SRCYRGBADR_B/F;
IMG_SRCUVADR_B/F;
IMG_SRCVADR_B/F;
IMG_DSTYRGBADR;
IMG_DSTUVADR;
IMG_DSTVADR;
IMG_SRCHSIZE_B/F;
IMG_SRCVSIZE_B/F;
IMG_DSTHSIZE;
IMG_DSTVSIZE;
IMG_OFFSETX;

IMG_OFFSETY;
IMG_MASKCOLOR;
IMG_ALPHA;
IMG_HSTEP;
IMG_VSTEP;
IMG_HFOLD;
IMG_VFOLD;
IMG_PEL_ENDIAN;
IMG_RGBYUV_CONF (/ IMG_YUVRGB_CONF);
IMG_RGBYUV00 (/ IMG_YUVRGB00);
IMG_RGBYUV01 (/ IMG_YUVRGB01);
IMG_RGBYUV02 (/ IMG_YUVRGB02);
IMG_RGBYUV10 (/ IMG_YUVRGB10);
IMG_RGBYUV11 (/ IMG_YUVRGB11);
IMG_RGBYUV12 (/ IMG_YUVRGB12);
IMG_RGBYUV20 (/ IMG_YUVRGB20);
IMG_RGBYUV21 (/ IMG_YUVRGB21);
IMG_RGBYUV22 (/ IMG_YUVRGB22);
IMG_ENSET;
IMG_INDATABYTE_B;
IMG_INDATABYTE_F;
IMG_OUTDATABYTE;
IMG_R_BRITNESS;
IMG_G_BRITNESS;
IMG_B_BRITNESS;

Note:

Some registers are unnecessary to be set under different operation modes.

2.5.4 Start IMG Processing

Issue the processing request to start image processor.
Related Register:
IMG_REQ

2.5.5 Image Processing

The source data will be processed according the IPU-IMG registers' configuration.
The processed image data will be store in the specified destination address.
When processing is completed, the value of register "IMG_ACK" will cleared ("1" → "0"). And a processing end interrupt is issued.
Gets the image processing result, by checking the processing status and the interrupt status.

2.5.6 Display the Destination Image

Check the operation result by display the destination image in LCD panel.

Note:
The operation of this step is similar with" **2.3.6 Display the Rotated Image**"

2.5.7 Reset IPU (Image Processor)

Before exit from image process operation, reset IPU.
Related Registers:
AHBCLKCTRL0;
APBCLKCTRL0;
GCLKCTRL0ENA;
GCLKCTRL0;
RESETREQ0ENA;
RESETREQ0;

2.6 Normal Procedure of IPU – Graphics DMA

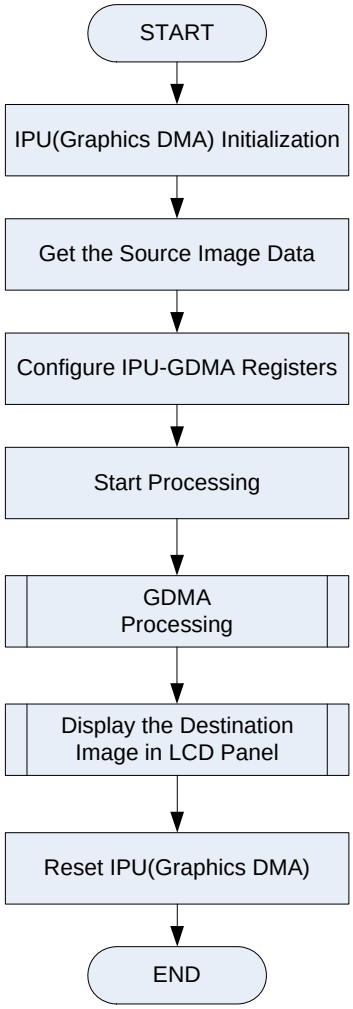

Figure 2-3 Normal Procedure of IPU-GDMA

Note:

1. About the explanation of all the IPU-GDMA registers mentioned in this document, please refer to "**EMMA Mobile 1 IPU User's Manual**".
2. About the explanation of all the ASMU registers mentioned in this document, please refer to "**EMMA Mobile 1 SMU/GIO Interface User's Manual**".

2.7 Detail of IPU-GDMA Normal Procedure

2.7.1 IPU (GDMA) Initialization

When do GDMA initialization, cancel reset IPU clock by setting ASMU registers as below.

Related Register:

AHBCLKCTRL0;
APBCLKCTRL0;
GCLKCTRL0ENA;
GCLKCTRL0;
RESETREQ0ENA;
RESETREQ0;

Note:

The operation of this step is same with" **2.3.1 IPU (Rotator) Initialization**"

2.7.2 Get Source Image Data

For each sample, the related YUV or RGB image will be read as the source data.

2.7.3 Configure GDMA Registers

Before start GDMA processing, some registers need to be configured, such as source data address, destination address, function mode, image data format etc.

The following registers need to be set as necessary in random sequence.

Related Registers:

DMA_DUAL_FF;
DMA_MODE;
DMA_FORMAT;
DMA_SRCSIZE_1;
DMA_SRCSIZE_2;
DMA_DSTSIZE;
DMA_SRCYRGBADR_1;
DMA_SRCYRGBADR_2;
DMA_DSTYRGBADR;
DMA_SRCUVADR_1;
DMA_SRCUVADR_2;
DMA_DSTUVADR;
DMA_SRCVADR_1;
DMA_SRCVADR_2;
DMA_DSTVADR;
DMA_HSIZE;

DMA_VSIZE;
DMA_MASKCOLOR;
DMA_FILLDATA;
DMA_SRCBYTE_1;
DMA_SRCBYTE_2;
DMA_DSTBYTE;
DMA_AUTO_SCAN;
DMA_ENSET;

Note:
Some registers are unnecessary to be set under different operation modes.

2.7.4 Start GDMA Processing

Issue the processing request to start GDMA.
Related Register:
DMA_REQ

2.7.5 GDMA Processing

The source data will be processed according the IPU-GDMA registers' configuration.
The processed image data will be store in the specified destination address.
When processing is completed, the value of register "DMA_ACK" will cleared ("1" → "0"). And a transfer end interrupt is issued.
Gets the GDMA processing result, by checking the processing status and the interrupt status.

2.7.6 Display the Destination Image

Check the operation result by display the destination image in LCD panel.

Note:
The operation of this step is similar with" **2.3.6 Display the Rotated Image**"

2.7.7 Reset IPU (GDMA)

Before exit from graphics DMA process operation, reset IPU.

Related Registers:

AHBCLKCTRL0;

APBCLKCTRL0;

GCLKCTRL0ENA;

GCLKCTRL0;

RESETREQ0ENA;

RESETREQ0;

Chapter 3 Example of IPU Operation

3.1 Outline of IPU Operation Example

EMMA Mobile 1 IPU supports image rotator, image process, and graphics DMA function.
So in this chapter, the below examples of these functions will be described:

Table 3-1 List of IPU Examples

No.		Example detail
1	Rotator	YUV422 Interleave: 0º, 90º, 180º, 270º
		YUV420 Planar: 0º, 90º, 180º, 270º
		RGB565: 0º, 90º, 180º, 270º
2	Image Processor	Example 1: Reducing 600x400 RGB565 image to 480x360 → back image, Converting 320x240 YUV420 Planar image to 320x240 RGB565 image → front image, Then overlay. [resize + color space conversion (YUV→RGB) + overlay]
		Example 2: Enlarging and converting 100x100 YUV422 to 256x192 RGB666 → back image, Overlay with 320x240 RGB888 image (with color masking and pixel packing) → front image [resize + color space conversion (YUV→RGB) + overlay + pixel packing]
3	Graphics DMA	ROP function: Mix source image 1 (RGB565 320x240) and source image 2 (RGB565 600x400).

The detail of the examples will be described in the following chapters.

3.1.1 Data Flow Chart of IPU Examples

In the IPU examples, the data is transmitted as below:

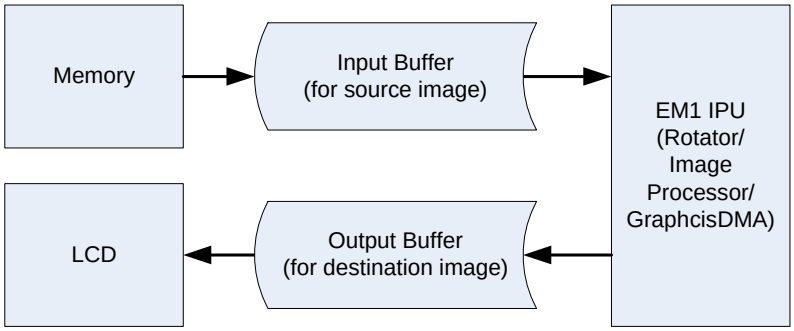

Figure 3-1 Data Flow Chart of IPU Examples

As shown in the figure, the source image data is read out from external memory and stored into the input buffer(s), then set the IPU-ROT/IPU-IMG/IPU-GDMA as necessary and start to perform the processing.

The processed image data is stored into the output buffer, then display in LCD panel.

3.1.2 IPU Initialization and Reset

For all samples of IPU-ROT, IPU-IMG and IPU-GDMA, the process of IPU initialization/reset is same. Process flow is below.

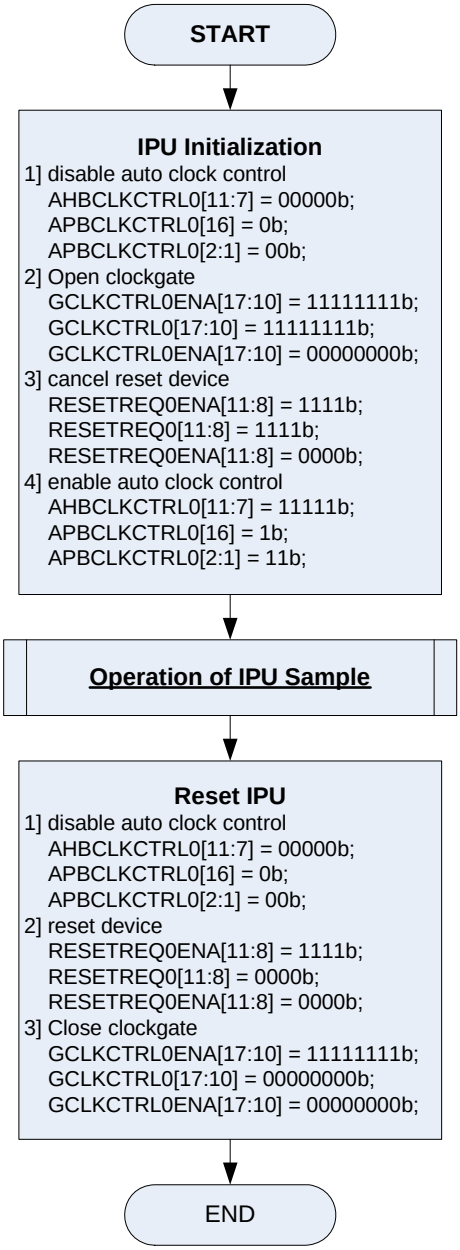

Figure 3-2 Operation Flow of IPU Initialization and Reset

3.2 IPU Sample – Rotator

In this sample, IPU-Rotator function is checked. The below samples are performed.

Detail of this sample:

Sample 1:

Input image: 100x100 YUV422 (Interleave)

Output image: 100x100 YUV422 with rotate angle 90°, 180°, and 270°.

Sample 2:

Input image: 320x240 YUV420 (Planar)

Output image: 320x240 YUV420 with rotate angle 90°, 180°, and 270°.

Sample 3:

Input image: 320x240 RGB565

Output image: 320x240 RGB565 with rotate angle 90°, 180°, and 270°.

Note:

Between each rotation operation, there is 1s delay for image displaying in LCD panel.

3.2.1 Operation Flow

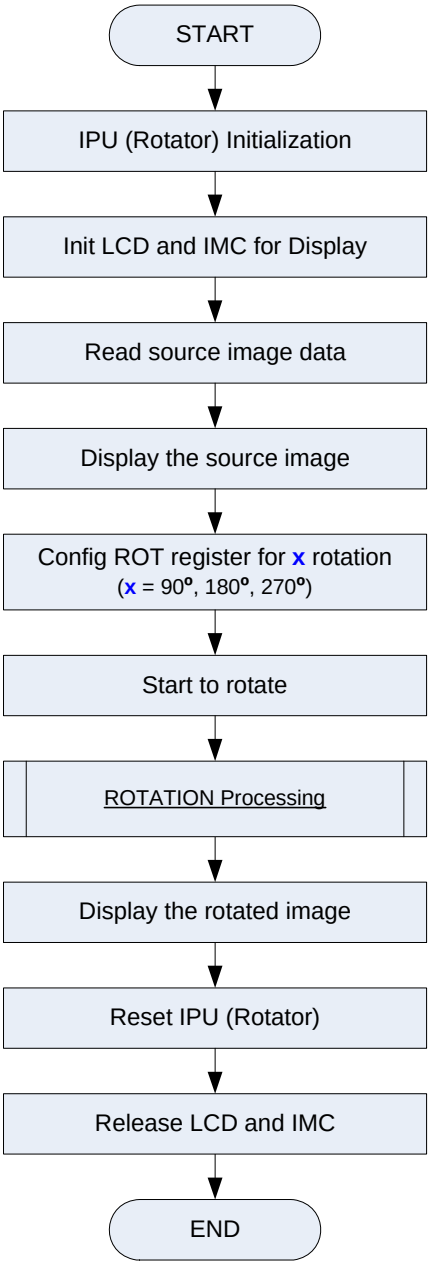

Figure 3-3 Operation Flow of IPU Rotator Sample

3.2.2 Operation Detail

3.2.2.1 IPU Initialization

Initialize IPU modules for rotation operation.
Refer to "**3.1.2 IPU Initialization and Reset**".

3.2.2.2 Initialize LCD and IMC

In this sample, LCD and IMC module will be called to display the rotated image. So need to initialize LCD and IMC module.
More detail about these two modules, please refer to "**EMMA Mobile 1 IMC Application Note**" and "**EMMA Mobile 1 LCDC Application Note**".

3.2.2.3 Read Source Data

The specified image data are read out as the source image data. In IPU-ROT samples, YUV422 Interleave data, YUV420 Planar data and RGB565 data are used.

3.2.2.4 Display the Source Image

In order to compare with the rotated image, the original image is displayed before rotation.

3.2.2.5 Configure ROT Registers for Rotation

ROT register configurations of these samples are listed as below.

Sample 1: YUV422 Interleave image rotation

- Setting for 90º rotation:

Table 3-2 IPU-ROT Register Setting for YUV422 90º Rotation

Item	Setting	Explanation
ROT_DUAL_FF NOTE1	0x0000_0000	REG_EN = 0b: disable register update reserve function (default)
ROT_MODE	0x0000_0039	ROT_MOD = 01b: rotate 90º ; SRCYUV = 0b: not care ; (default) DSTENDIAN = 1b: Little Endian ; SRCENDIAN = 1b: Little Endian ; NEWEN = 1b: use additional function ;
ROT_FRAME	0x0000_0005	SRCSEL = 01b: A frame ; (default) DSTSEL = 01b: A frame ; (default)
ROT_SRCSIZE	0x0000_00C8	200 (= 100x2)
ROT_DSTSIZE	0x0000_00C8	200 (= 100x2)
ROT_SRCYADR_A	0x3100_0000	0x3100_0000
ROT_DSTYADR_A	0x3200_0000	0x3200_0000
ROT_SRCUVADR_A	0x0	0
ROT_DSTUVADR_A	0x0	0
ROT_SRCHSIZE	0x0000_0064	100
ROT_SRCVSIZE	0x0000_0064	100
ROT_FORMAT	0xAA	DSTFMT = 1010b: YUV422 Interleave ; SRCFMT = 1010b: YUV422 Interleave ;
ROT_SRCBYTE	0x0000_00E4	(default)
ROT_DSTBYTE	0x0000_00E4	(default)
ROT_INTENSET	0x0000_0001	DMASTOPEN = 0b: not enable ; (default) DMAERREN = 0b: not enable ; (default) ROTENDEN = 1b: enable rotation end interrupt ;

Note:

1. During the rotation processing, set register “ROT_DUAL_FF” at the first time only.
2. Setting of the other registers, which not listed in the upper table, should use the default value.

- Setting for 180º rotation (Only list difference setting with 90º rotation):

Table 3-3 IPU-ROT Register Setting for YUV422 180º Rotation

Item	Setting	Explanation
ROT_MODE	0x0000_003A	ROT_MOD = 10b: rotate 180º ; SRCYUV = 0b: not care ; (default) DSTENDIAN = 1b: Little Endian ; SRCENDIAN = 1b: Little Endian ; NEWEN = 1b: use additional function ;
ROT_FRAME	0x0000_0009	SRCSEL = 01b: A frame ; (default) DSTSEL = 10b: B frame ;
ROT_DSTYADR_A	0x3210_0000	0x3210_0000

- Setting for 270º rotation (Only list difference setting with 90º rotation):

Table 3-4 IPU-ROT Register Setting for YUV422 270º Rotation

Item	Setting	Explanation
ROT_MODE	0x0000_003B	ROT_MOD = 11b: rotate 270º ; SRCYUV = 0b: not care ; (default) DSTENDIAN = 1b: Little Endian ; SRCENDIAN = 1b: Little Endian ; NEWEN = 1b: use additional function ;
ROT_FRAME	0x0000_000D	SRCSEL = 01b: A frame ; (default) DSTSEL = 11b: C frame ;
ROT_DSTYADR_A	0x3220_0000	0x3220_0000

Sample 2: YUV420 Planar image rotation

- Setting for 90º rotation:

Table 3-5 IPU-ROT Register Setting for YUV420 90º Rotation

Item	Setting	Explanation
ROT_DUAL_FF NOTE1	0x0000_0000	REG_EN = 0b: disable register update reserve function (default)
ROT_MODE	0x0000_0031	ROT_MOD = 01b: rotate 90º ; SRCYUV = 0b: not care ; DSTENDIAN = 0b: Big Endian ; (default) SRCENDIAN = 1b: Little Endian ; NEWEN = 1b: use additional function ;
ROT_FRAME	0x0000_0005	SRCSEL = 01b: A frame ; (default) DSTSEL = 01b: A frame ; (default)
ROT_SRCSIZE	0x0000_0140	320
ROT_DSTSIZE	0x0000_00F0	240
ROT_SRCYADR_A	0x3110_0000	0x3110_0000
ROT_DSTYADR_A	0x3200_0000	0x3200_0000
ROT_SRCUVADR_A	0x3111_2C00	0x3110_0000 + 320x240
ROT_DSTUVADR_A	0x3201_2C00	0x3200_0000 + 320x240
ROT_SRCVADR_A	0x3111_7700	0x3111_2C00 + 320x240/4
ROT_DSTVADR_A	0x3201_7700	0x3201_2C00 + 320x240/4
ROT_SRCHSIZE	0x0000_0140	320
ROT_SRCVSIZE	0x0000_00F0	240
ROT_FORMAT	0x0000_0044	SRCFMT = 0100b: YUV420 Planar ; DSTFMT = 0100b: YUV420 Planar ;
ROT_SRCBYTE	0x0000_00E4	(default)
ROT_DSTBYTE	0x0000_00E4	(default)
ROT_INTENSET	0x0000_0001	DMASTOPEN = 0b: not enable ; (default) DMAERREN = 0b: not enable ; (default) ROTENDEN = 1b: enable rotation end interrupt ;

Note:

1. During the rotation processing, set register “ROT_DUAL_FF” at the first time only.
2. Setting of the other registers, which not listed in the upper table, should use the default value.

- Setting for 180º rotation (Only list difference setting with 90º rotation):

Table 3-6 IPU-ROT Register Setting for YUV420 180º Rotation

Item	Setting	Explanation
ROT_MODE	0x0000_0032	ROT_MOD = 10b: rotate 180º ; SRCYUV = 0b: not care ; DSTENDIAN = 0b: Big Endian ; (default) SRCENDIAN = 1b: Little Endian ; NEWEN = 1b: use additional function ;
ROT_FRAME	0x0000_0009	SRCSEL = 01b: A frame ; (default) DSTSEL = 10b: B frame ;
ROT_DSTSIZE	0x0000_0140	320
ROT_DSTYADR_B	0x3210_0000	0x3210_0000
ROT_DSTUVADR_B	0x3211_2C00	0x3210_0000 + 320x240
ROT_DSTVADR_B	0x3211_7700	0x3211_2C00 + 320x240/4

- Setting for 270º rotation (Only list difference setting with 90º rotation):

Table 3-7 IPU-ROT Register Setting for YUV420 270º Rotation

Item	Setting	Explanation
ROT_MODE	0x0000_0033	ROT_MOD = 11b: rotate 270º ; SRCYUV = 0b: not care ; DSTENDIAN = 0b: Big Endian ; (default) SRCENDIAN = 1b: Little Endian ; NEWEN = 1b: use additional function ;
ROT_FRAME	0x0000_000D	SRCSEL = 01b: A frame ; (default) DSTSEL = 11b: C frame ;
ROT_DSTSIZE	0x0000_00F0	240
ROT_DSTYADR_C	0x3220_0000	0x3220_0000
ROT_DSTUVADR_C	0x3221_2C00	0x3220_0000 + 320x240
ROT_DSTVADR_C	0x3221_7700	0x3221_2C00 + 320x240/4

Sample 3: RGB565 image rotation

- Setting for 90º rotation:

Table 3-8 IPU-ROT Register Setting for RGB565 90º Rotation

Item	Setting	Explanation
ROT_DUAL_FF NOTE1	0x0000_0000	REG_EN = 0b: disable register update reserve function (default)
ROT_MODE	0x0000_0021	ROT_MOD = 01b: rotate 90º ; SRCYUV = 0b: not care ; (default) DSTENDIAN = 0b: Big Endian ; (default) SRCENDIAN = 0b: Big Endian ; (default) NEWEN = 1b: use additional function ;
ROT_FRAME	0x0000_0005	SRCSEL = 01b: A frame ; (default) DSTSEL = 01b: A frame ; (default)
ROT_SRCSIZE	0x0000_0280	640 (=320 x 2)
ROT_DSTSIZE	0x0000_01E0	480 (=240 x 2)
ROT_SRCYADR_A	0x3120_0000	0x3120_0000
ROT_DSTYADR_A	0x3200_0000	0x3200_0000
ROT_SRCHSIZE	0x0000_00F0	320
ROT_SRCVSIZE	0x0000_0140	240
ROT_FORMAT	0x0000_0022	SRCFMT = 0010b: RGB565 ; DSTFMT = 0010b: RGB565 ;
ROT_SRCBYTE_CMP	0x0000_00E4	(default)
ROT_DSTBYTE_CMP	0x0000_00E4	(default)
ROT_INTENSET	0x0000_0001	DMASTOPEN = 0b: not enable ; (default) DMAERREN = 0b: not enable ; (default) ROTENDEN = 1b: enable rotation end interrupt ;

Note:

1. During the rotation processing, set register “ROT_DUAL_FF” at the first time only.
2. Setting of the other registers, which not listed in the upper table, should use the default value.

- Setting for 180º rotation (Only list difference setting with 90º rotation):

Table 3-9 IPU-ROT Register Setting for RGB565 180º Rotation

Item	Setting	Explanation
ROT_MODE	0x0000_0022	ROT_MOD = 10b: rotate 180º ; SRCYUV = 0b: not care ; (default) DSTENDIAN = 0b: Big Endian ; (default) SRCENDIAN = 0b: Big Endian ; (default) NEWEN = 1b: use additional function ;
ROT_FRAME	0x0000_0009	SRCSEL = 01b: A frame ; (default) DSTSEL = 10b: B frame ;
ROT_DSTSIZE	0x0000_0280	640 (=320 x 2)
ROT_DSTYADR_B	0x3210_0000	0x3210_0000

- Setting for 270º rotation (Only list difference setting with 90º rotation):

Table 3-10 IPU-ROT Register Setting for RGB565 270º Rotation

Item	Setting	Explanation
ROT_MODE	0x0000_0023	ROT_MOD = 11b: rotate 270º ; SRCYUV = 0b: not care ; (default) DSTENDIAN = 0b: Big Endian ; (default) SRCENDIAN = 0b: Big Endian ; (default) NEWEN = 1b: use additional function ;
ROT_FRAME	0x0000_000D	SRCSEL = 01b: A frame ; (default) DSTSEL = 11b: C frame ;
ROT_DSTSIZE	0x0000_01E0	480 (= 240 x 2)
ROT_DSTYADR_C	0x3220_0000	0x3220_0000

3.2.2.6 Start Rotator Processing

Issue the processing request to start IPU-ROT by set “ROT_REQ” to 1.

3.2.2.7 Rotator Processing

The source image data will be rotated according to the setting of IPU-ROT registers.
When rotation is completed, the value of register “ROT_STATUS” will become “0”.
Checks the value for ensure the rotation is completed.

3.2.2.8 Display the rotated image in LCD panel

Call IMC interface to display the rotated image.

3.2.2.9 Reset IPU

Reset IPU as description in “**3.1.2 IPU Initialization and Reset**”.

3.2.2.10 Release LCD and IMC

Release LCD and IMC resource used by the ROT samples.
Actually needn’t to do anything for IMC;
Only need to stop and power off LCD.

3.3 IPU Sample – Image Processor Function

In this chapter, there are two samples to evaluate image processor functions, such as resize/converter/dithering/overlay etc.

Detail of this sample:

Sample 1:

Show how to use resizer combine with color space converter function.

Back image: 600x400 RGB565
Will be reduced to 480x360;

Front image: 320x240 YUV420 Planar;
Will be converter to 320x240 RGB565;

Output image: 480x360 RGB565.

Sample 2:

Show how to use overlay combine with resizer, color space converter and pixels packing function.

Back image: 100x100 YUV422 Interleave;
Will be enlarged and converted to 256x192 RGB666;

Front image: 320x240 RGB888 image;

Output image: 320x240 RGB666 data.

When do overlay, pixel packing, color mask and transparency function are used.

3.3.1 Operation Flow

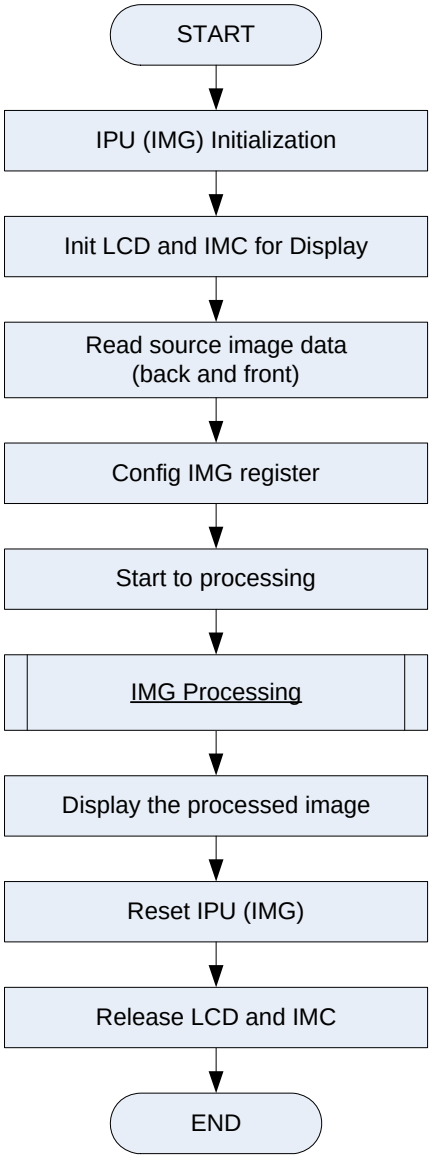

Figure 3-4 Operation Flow of IPU Image Processor Sample

3.3.2 Operation Detail

3.3.2.1 IPU Initialization

Initialize IPU modules for image process operation.
Refer to "**3.1.2 IPU Initialization and Reset**".

3.3.2.2 Initialize LCD and IMC

In this sample, LCD and IMC module will be called to display the processed image. So need to initialize LCD and IMC module.

Note:
This operation is same with "**3.2.2.2 Initialize LCD and IMC**".

3.3.2.3 Read Source Data

The specified image data are read out as the source image data.
In IPU-IMG sample 1, RGB565 data (back image) and YUV420 Planar data (front image) are used.
In IPU-IMG sample 2, YUV422 Interleave data (back image) and RGB888 data (front image) are used.

3.3.2.4 Configure IMG

IMG register configurations of these samples are listed as below.

Sample 1:

Table 3-11 Setting IPU-IMG Registers for IPU-IMG Sample 1

Item	Setting	Explanation
IMG_DUAL_FF	0x0000_0000	REG_EN = 0b: disable register update reserve function ; (default)
IMG_MODE	0x0011_2700	SRCFMT_F = 0b: not care ; (default) SRCFMT_B = 0b: not care ; (default) DSTFMT = 0b: not care ; (default) DITHER_F = 0b: not care ; (default) DITHER_B = 0b: not care ; (default) PELPACK = 0b: disable pixel packing ; (default) RFILTER_F = 0b: not care ; (default) RFILTER_B = 0b: not care ; (default) OP_MODE = 127H: OP13 ; UDINV = 0b: don't flip ; (default) RLINV = 0b: don't flip ; (default) MSK_COL = 0b: not care ; (default) NEWEN = 1b: use additional function ;
IMG_SRCSIZE_B	0x0000_04B0	1200 (= 600x2)
IMG_SRCYRGBADR_B	0x3100_0000	0x3100_0000
IMG_SRCHSIZE_B	0x0000_0258	600
IMG_SRCVSIZE_B	0x0000_0190	400
IMG_SRCSIZE_F	0x0000_0280	320
IMG_SRCYRGBADR_F	0x3120_0000	0x3120_0000
IMG_SRCUVADR_F	0x3121_2C00	0x3120_0000 + 320x240
IMG_SRCVADR_F	0x3121_7700	0x3121_2C00 + 320x240/4
IMG_SRCHSIZE_F	0x0000_0140	320
IMG_SRCVSIZE_F	0x0000_00F0	240
IMG_DSTSIZE	0x0000_03C0	960 (= 480x2)
IMG_DSTYRGBADR	0x3200_0000	0x3200_0000
IMG_DSTHSIZE	0x0000_01E0	480
IMG_DSTVSIZE	0x0000_0168	360
IMG_OFFSETX	0x0000_0000	0: Horizontal offset ;
IMG_OFFSETY	0x0000_0000	0: Vertical offset ;
IMG_HSTEP	0x0000_0140	320 (= 256 / (480/600))
IMG_VSTEP	0x0000_011C	284 (= 256 / (360/400))
IMG_HFOLD	0x0000_0033	51 (= (480/600) x 64)
IMG_VFOLD	0x0000_0039	57 (= (360/400) x 64)

IMG_FORMAT	0x0000_0224	DSTFMT = 0010b: RGB565 ; SRCFMT_B = 0010b: RGB565 ; SRCFMT_F = 0100b: YUV420 Planar ;
IMG_INDATABYTE	0x0000_00E4	(default)
IMG_OUTDATABYTE	0x0000_00E4	(default)
IMG_PEL_ENDIAN	0x0000_0003	OUT_ENDI = 1: Little Endian ; IN_ENDI = 1: Little Endian.
IMG_YUVRGB_CONF NOTE2	0x0000_0000	(default)
IMG_YUVRGB00 NOTE2	0x0000_0100	(default)
IMG_YUVRGB01 NOTE2	0x0000_0000	(default)
IMG_YUVRGB02 NOTE2	0x0000_015F	(default)
IMG_YUVRGB10 NOTE2	0x0000_0100	(default)
IMG_YUVRGB11 NOTE2	0x0000_0856	(default)
IMG_YUVRGB12 NOTE2	0x0000_08B3	(default)
IMG_YUVRGB20 NOTE2	0x0000_0100	(default)
IMG_YUVRGB21 NOTE2	0x0000_01BB	(default)
IMG_YUVRGB22 NOTE2	0x0000_0000	(default)
IMG_ENSET	0x0000_0001	IMG_ENSETFLT = 0b ; IMG_ENSETSTOP = 0b ; IMG_ENSETERROR = 0b ; IMG_ENSETSTAT = 1b: enable processing end interrupt ;
IMG_FFCLR	0x0000_0001	IMG_FFCLRFLT = 0b ; IMG_FFCLRSTOP = 0b ; IMG_FFCLRERROR = 0b ; IMG_FFCLRSTAT = 1b: clear processing end interrupt source ;

Note:

1. Setting of the other registers, which not listed in the upper table, should use the default value.
2. There are several kinds of arithmetic algorithms for YUV→RGB converter. About the implementation and the typical setting values, please refer "**EMMA Mobile 1 IPU User's Manual**".

Sample 2:

Table 3-12 Setting IPU-IMG Registers for IPU-IMG Sample 2

Item	Setting	Explanation
IMG_DUAL_FF	0x0000_0000	REG_EN = 0b: disable register update reserve function ; (default)
IMG_MODE	0x0011_5038	SRCFMT_F = 0b: not care ; (default) SRCFMT_B = 0b: not care ; (default) DSTFMT = 0b: not care ; (default) DITHER_F = 1b: enable color dithering ; DITHER_B = 1b: enable color dithering ; PELPACK = 1b: enable pixel packing ; RFILTER_F = 0b: not care ; (default) RFILTER_B = 0b: not care ; (default) OP_MODE = 150H: OP18 ; UDINV = 0b: don't flip ; (default) RLINV = 0b: don't flip ; (default) MSK_COL = 0b: enable color masking ; (default) NEWEN = 1b: use additional function ;
IMG_SRCSIZE_B	0x0000_00C8	200 (= 100x2)
IMG_SRCYRGBADR_B	0x3100_0000	0x3100_0000
IMG_SRCHSIZE_B	0x0000_0064	100
IMG_SRCVSIZE_B	0x0000_0064	100
IMG_SRCSIZE_F	0x0000_0280	960 (= 320x3)
IMG_SRCYRGBADR_F	0x3110_0000	0x3110_0000
IMG_SRCHSIZE_F	0x0000_0140	320
IMG_SRCVSIZE_F	0x0000_00F0	240
IMG_DSTSIZE	0x0000_02D0	720 (= 320x18/8)
IMG_DSTYRGBADR	0x3200_0000	0x3200_0000
IMG_DSTHSIZE	0x0000_0100	256
IMG_DSTVSIZE	0x0000_00C0	192
IMG_OFFSETX	0x0000_0000	0: Horizontal offset ;
IMG_OFFSETY	0x0000_0000	0: Vertical offset ;
IMG_HSTEP	0x0000_0064	100 (= 256 / (256/100))
IMG_VSTEP	0x0000_0085	133 (= 256 / (192/100))
IMG_HFOLD	0x0000_00A3	163 (= (256/100) x 64)
IMG_VFOLD	0x0000_007A	122 (= (192/100) x 64)
IMG_MASKCOLR	0x0000_0033	51
IMG_ALPHA	0x0000_0014	20
IMG_FORMAT	0x0000_01A0	DSTFMT = 0001b: RGB666 ; SRCFMT_B = 1010b: YUV422 Interleave ; SRCFMT_F = 0000b: RGB888 ;

IMG_INDATABYTE	0x0000_00E4	(default)
IMG_OUTDATABYTE	0x0000_00E4	(default)
IMG_PEL_ENDIAN	0x0000_0003	OUT_ENDI = 1: Little Endian ; IN_ENDI = 1: Little Endian.
IMG_YUVRGB_CONF NOTE2	0x0000_0000	(default)
IMG_YUVRGB00 NOTE2	0x0000_0100	(default)
IMG_YUVRGB01 NOTE2	0x0000_0000	(default)
IMG_YUVRGB02 NOTE2	0x0000_015F	(default)
IMG_YUVRGB10 NOTE2	0x0000_0100	(default)
IMG_YUVRGB11 NOTE2	0x0000_0856	(default)
IMG_YUVRGB12 NOTE2	0x0000_08B3	(default)
IMG_YUVRGB20 NOTE2	0x0000_0100	(default)
IMG_YUVRGB21 NOTE2	0x0000_01BB	(default)
IMG_YUVRGB22 NOTE2	0x0000_0000	(default)
IMG_ENSET	0x0000_0001	IMG_ENSETFLT = 0b ; IMG_ENSETSTOP = 0b ; IMG_ENSETERROR = 0b ; IMG_ENSETSTAT = 1b: enable processing end interrupt ;
IMG_FFCLR	0x0000_0001	IMG_FFCLRFLT = 0b ; IMG_FFCLRSTOP = 0b ; IMG_FFCLRERROR = 0b ; IMG_FFCLRSTAT = 1b: clear processing end interrupt source ;

Note:

1. Setting of the other registers, which not listed in the upper table, should use the default value.
2. There are several kinds of arithmetic algorithms for YUV→RGB converter. About the implementation and the typical setting values, please refer "**EMMA Mobile 1 IPU User's Manual**".

3.3.2.5 Start Image Processing

Issue the processing request to start IPU-IMG by set “IMG_REQ” to 1.

3.3.2.6 IMG Processing

The source image data will be processed according to the setting of IPU-IMG registers.
When image processing is completed, the value of register “IMG_ACK” will become “0”.
Checks the value for ensure the processing is completed.

3.3.2.7 Display the processed image in LCD panel

Call IMC interface to display the processed image.

3.3.2.8 Reset IPU

Reset IPU as description in “**3.1.2 IPU Initialization and Reset**”.

3.3.2.9 Release LCD and IMC

Release LCD and IMC resource used by the ROT samples.
Actually needn’t to do anything for IMC;
Only need to stop and power off LCD.

3.4 IPU Sample – GDMA ROP Function

This sample will show how to use the ROP function of IPU-GDMA.

Detail of this sample:

Source image 1:	320x240 RGB565
Source image 2:	600x400 RGB565
Destination image:	320x240 RGB565
	(Source image 1 and 2 do overlay with mask color)

3.4.1 Operation Flow

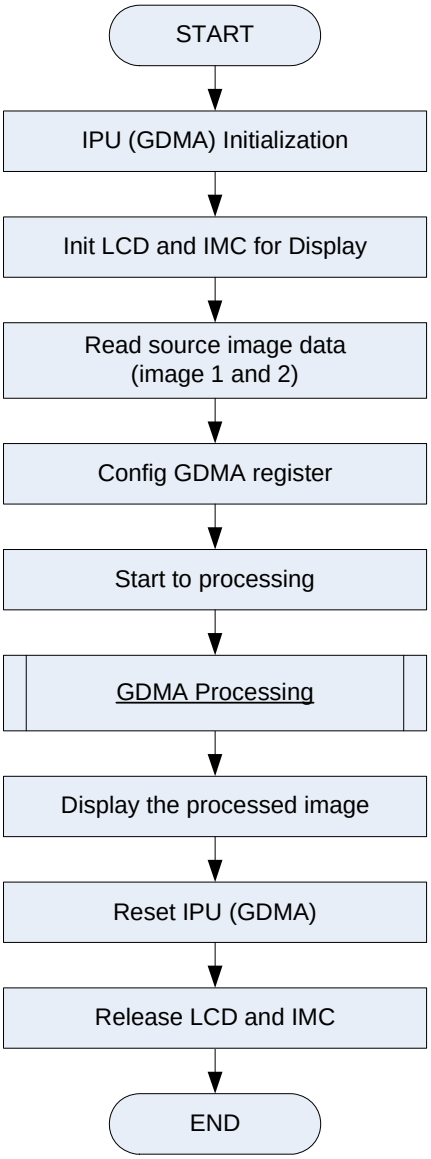

Figure 3-5 Operation Flow of IPU Graphics DMA Sample

3.4.2 Operation Detail

3.4.2.1 IPU Initialization

Initialize IPU modules for GDMA operation.
Refer to "**3.1.2 IPU Initialization and Reset**".

3.4.2.2 Initialize LCD and IMC

In this sample, LCD and IMC module will be called to display the processed image. So need to initialize LCD and IMC module.

Note:
This operation is same with "**3.2.2.2 Initialize LCD and IMC**".

3.4.2.3 Read Source Data

The specified image data are read out as the source image data.
In this sample, RGB565 600x400 image and RGB565 320x240 image are used as source image 1 and source image 2.

3.4.2.4 Configure GDMA

GDMA register configurations of this sample are listed as below.

Table 3-13 Setting IPU-GDMA Registers for IPU-GDMA Sample

Item	Setting	Explanation
DMA_DUAL_FF	0x0000_0000	REG_EN = 0b: disable register update reserve function ; (default)
DMA_MODE	0x0000_0068	MASKOBJ = 0b: source image 1 ; KEY = 1b: enable mask color ; OP = 10b: ROP ; ROP = 1000b: SRC1 and SRC2 ;
DMA_SRCSIZE_1	0x0000_0280	640 (= 320x2)
DMA_SRCYRGBADR_1	0x3100_0000	0x3100_0000
DMA_SRCSIZE_2	0x0000_04B0	1200 (= 600x2)
DMA_SRCYRGBADR_2	0x3110_0000	0x3110_0000
DMA_DSTSIZE	0x0000_0280	640 (= 320x2)
DMA_DSTYRGBADR	0x3200_0000	0x3200_0000
DMA_HSIZE	0x0000_0140	320
DMA_VSIZE	0x0000_00F0	240
DMA_MASKCOLR	0x0000_00C8	200
DMA_FORMAT	0x0000_0002	2: RGB565
DMA_FILLDATA	0x0000_0000	(default)
DMA_SRCBYTE_1	0x0000_00E4	(default)
DMA_SRCBYTE_2	0x0000_00E4	(default)
DMA_DSTBYTE	0x0000_00E4	(default)
DMA_AUTO_SCAN	0x0000_0000	(default)
DMA_ENSET	0x0000_0001	DMA_ENSETAHBERR = 0b ; DMA_ENSETSTAT = 1b: enable DMA processing end interrupt ;
DMA_FFCLR	0x0000_0001	DMA_ FFCLRAHBERR = 0b ; DMA_FFCLRSTAT= 1b: clear DMA processing end interrupt ;

Note:

1. Setting of the other registers, which not listed in the upper table, should use the default value.

3.4.2.5 Start GDMA Processing

Issue the processing request to start IPU-GDMA by set “DMA_REQ” to 1.

3.4.2.6 GDMA Processing

The source image1 and source image 2 will be mixed according to the setting of IPU-GDMA registers.
When processing is completed, the value of register “DMA_ACK” will become “0”.
Checks the value for ensure the processing is completed.

3.4.2.7 Display the processed image in LCD panel

Call IMC interface to display the processed image.

3.4.2.8 Reset IPU

Reset IPU as description in “**3.1.2 IPU Initialization and Reset**”.

3.4.2.9 Release LCD and IMC

Release LCD and IMC resource used by the ROT samples.
Actually needn’t to do anything for IMC;
Only need to stop and power off LCD.

Appendix A. Image Processor Unit Driver Function

A.1 IPU API function list

The following table shows the IPU functions:

Table A-1 IPU Driver Function List

Type	Function Name	Function Detail
Driver Function	em1_ipu_rot_dual_off	Enable or disable register update function
	em1_ipu_rot_cfg	Configure Rotator registers
	em1_ipu_rot_do	Perform rotation processing
	em1_ipu_rot_get_status	Get current status of Rotator
	em1_ipu_img_dual_off	Enable or disable register update function
	em1_ipu_img_cfg	Configure IMG registers
	em1_ipu_img_do	Perform IMG processing
	em1_ipu_img_get_status	Get current status of IMG
	em1_ipu_gdma_dual_off	Enable or disable register update function
	em1_ipu_gdma_cfg	Configure GDMA registers
	em1_ipu_gdma_do	Perform GDMA processing
	em1_ipu_gdma_get_status	Get current status of GDMA
	em1_ipu_init	Initialize IPU module
	em1_ipu_release	Reset IPU module

A.2 Type Define

A.2.1 Naming rule and coding rule

About naming rule and coding rule, please refer to "**GD_SPEC_EM1_AN&TP.pdf**"

A.2.2 Structure

Table A-2 Structure List

Structure Name	Detail
_ROT_MODE	Mode register of Rotator
_ROTIO_ROTATION	All necessary register setting of Rotator
_IMG_MODE	Mode register of IMG
_IMG_DATA	Data info of IMG, ex. address, size etc.
_IMGIO_CONVERT	All necessary register setting of IMG
_GDMA_MODE	Mode register of GDMA
_GDMA_DATA	Data info of GDMA, ex. address, size etc.
_IPU_GDMA	All necessary register setting of GDMA

A.3 Function Detail

A.3.1 Initialization Function

[Function Name]

em1_ipu_init

[Format]

DRV_RESULT em1_ipu_init (void);

[Argument]

None

[Function Return]

DRV_ERR_STATE;

DRV_OK;

[Function Flow]

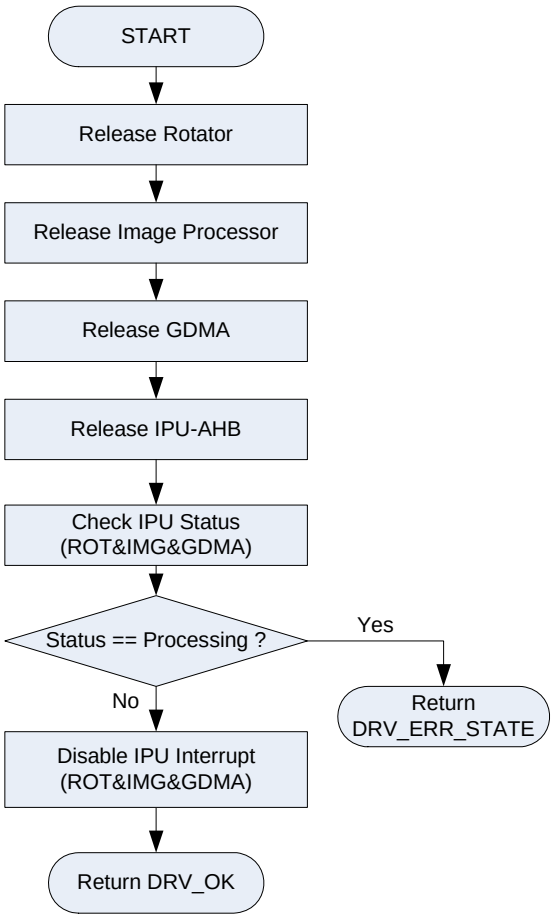

Figure A-1 IPU Initialization

[Note]

None

A.3.2 Reset IPU Function

[Function Name]

em1_ipu_release

[Format]

void em1_ipu_release (void);

[Argument]

None

[Function Return]

None

[Function Flow]

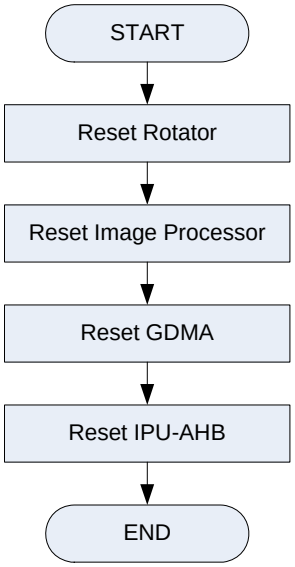

Figure A-2 IPU Reset

[Note]

None

A.3.3 Enable/Disable ROT Register Update Function

[Function Name]

em1_ipu_rot_dual_off

[Format]

void em1_ipu_rot_dual_off (uchar flag);

[Argument]

Parameter	Type	I/O	Detail
flag	uchar	I	Enable or disable flag

[Function Return]

None

[Function Flow]

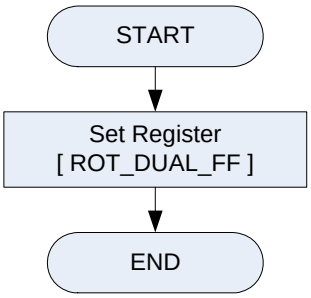

Figure A-3 Enable/Disable ROT Register Update Function

[Note]

Call this function first time only.

A.3.4 Configure ROT Registers

[Function Name]

em1_ipu_rot_cfg

[Format]

DRV_RESULT em1_ipu_rot_cfg (_ROTIO_ROTATION* rot);

[Argument]

Parameter	Type	I/O	Detail
rot	_ROTIO_ROTATION *	I	All necessary setting for rotation

[Function Return]

DRV_ERR_CONFIG;

DRV_OK;

[Function Flow]

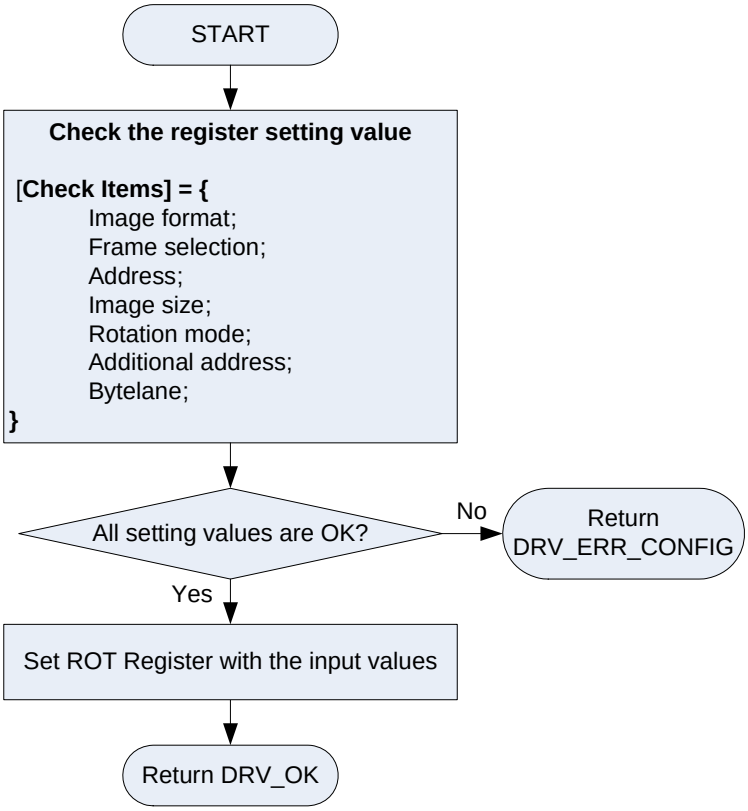

Figure A-4 ROT Configure Function

[Note]

None

A.3.5 Start ROT Processing

[Function Name]

em1_ipu_rot_do

[Format]

void em1_ipu_rot_do (void);

[Argument]

None

[Function Return]

None

[Function Flow]

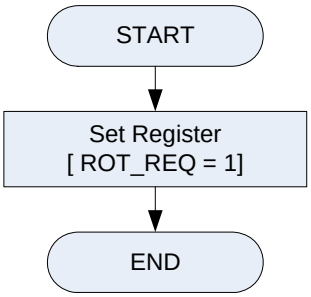

Figure A-5 Start ROT Processing

[Note]

None

A.3.6 Get ROT Status

[Function Name]

em1_ipu_rot_get_status

[Format]

void em1_ipu_rot_get_status (void);

[Argument]

None

[Function Return]

None

[Function Flow]

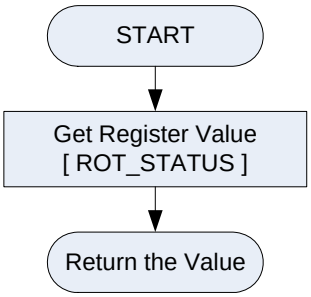

Figure A-6 Get ROT Status

[Note]

None

A.3.7 Enable/Disable IMG Register Update Function

[Function Name]

em1_ipu_img_dual_off

[Format]

void em1_ipu_img_dual_off (uchar flag);

[Argument]

Parameter	Type	I/O	Detail
flag	uchar	I	Enable or disable flag

[Function Return]

None

[Function Flow]

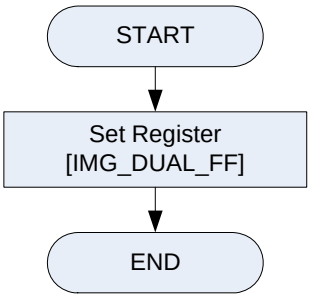

Figure A-7 Enable/Disable IMG Register Update Function

[Note]

Call this function first time only.

A.3.8 Configure IMG Registers

[Function Name]

em1_ipu_img_cfg

[Format]

DRV_RESULT em1_ipu_img_cfg (_IMGIO_CONVERT* img);

[Argument]

Parameter	Type	I/O	Detail
img	_IMGIO_CONVERT *	I	All necessary setting for image processing

[Function Return]

DRV_ERR_CONFIG;

DRV_OK;

[Function Flow]

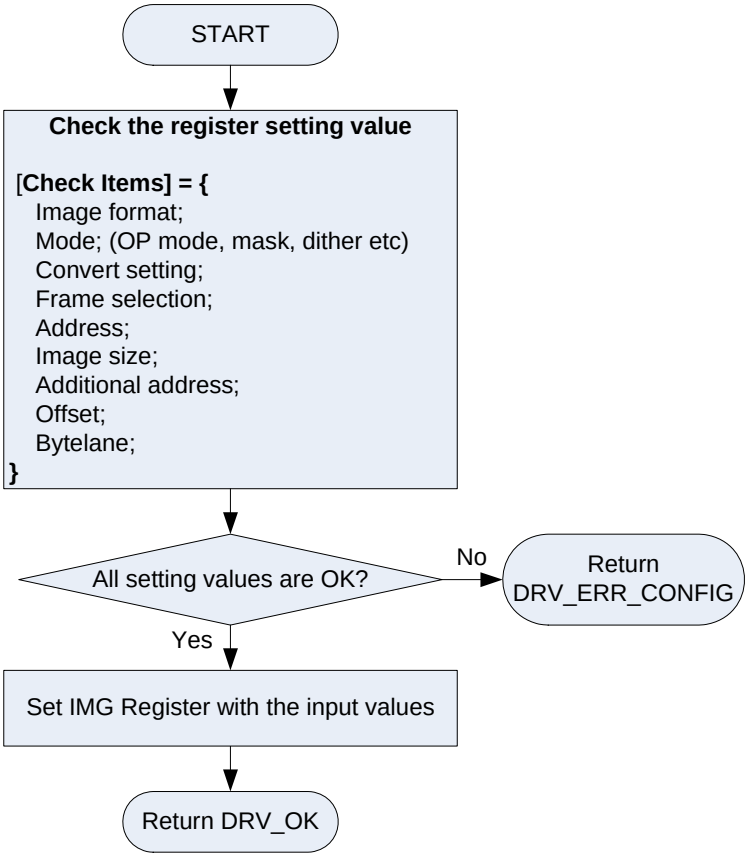

Figure A-8 IMG Configure Function

[Note]

None

A.3.9 Start IMG Processing

[Function Name]

em1_ipu_img_do

[Format]

void em1_ipu_img_do (void);

[Argument]

None

[Function Return]

None

[Function Flow]

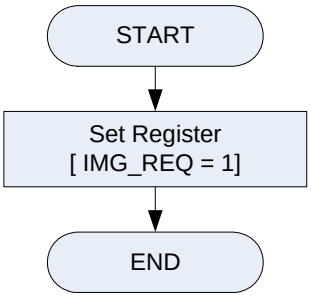

Figure A-9 Start IMG Processing

[Note]

None

A.3.10 Get IMG Status

[Function Name]

em1_ipu_img_get_status

[Format]

void em1_ipu_img_get_status (void);

[Argument]

None

[Function Return]

None

[Function Flow]

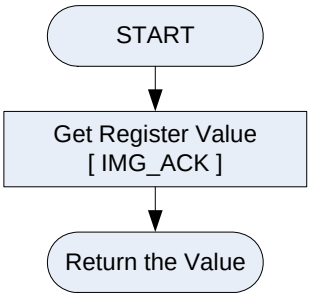

Figure A-10 Get IMG Status

[Note]

None

A.3.11 Enable/Disable GDMA Register Update Function

[Function Name]

em1_ipu_gdma_dual_off

[Format]

void em1_ipu_gdma_dual_off (uchar flag);

[Argument]

Parameter	Type	I/O	Detail
flag	uchar	I	Enable or disable flag

[Function Return]

None

[Function Flow]

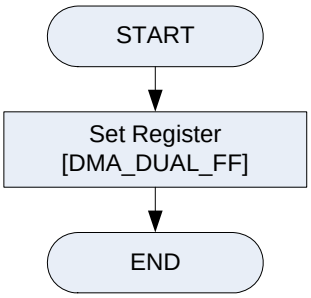

Figure A-11 Enable/Disable GDMA Register Update Function

[Note]

Call this function first time only.

A.3.12 Configure ROT Registers

[Function Name]

em1_ipu_rot_cfg

[Format]

DRV_RESULT em1_ipu_gdma_cfg (_IPU_GDMA* cfg);

[Argument]

Parameter	Type	I/O	Detail
cfg	_IPU_GDMA *	I	All necessary setting for GDMA operation

[Function Return]

DRV_ERR_CONFIG;

DRV_OK;

[Function Flow]

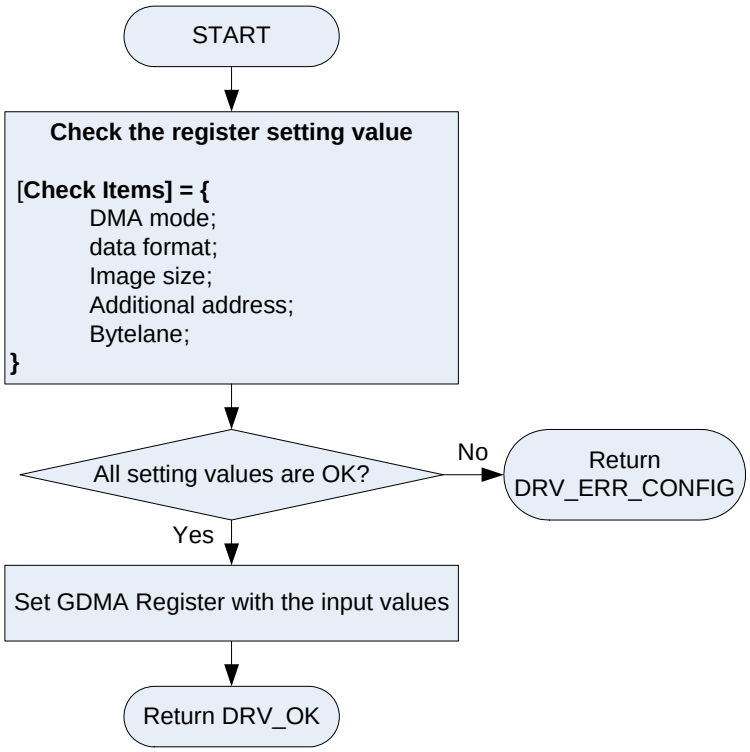

Figure A-12 GDMA Configure Function

[Note]

None

A.3.13 Start GDMA Processing

[Function Name]

em1_ipu_gdma_do

[Format]

void em1_ipu_gdma_do (void);

[Argument]

None

[Function Return]

None

[Function Flow]

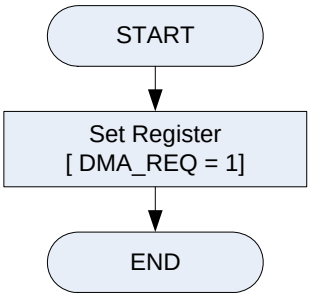

Figure A-13 Start GDMA Processing

[Note]

None

A.3.14 Get GDMA Status

[Function Name]

em1_ipu_gdma_get_status

[Format]

void em1_ipu_gdma_get_status (void);

[Argument]

None

[Function Return]

None

[Function Flow]

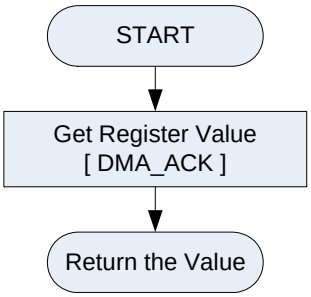

Figure A-14 Get GDMA Status

[Note]

None

ANNEX Modification History

Number	Modification Contents	Author	Date
Ver 1.00	New version		Aug,4,2009