User's Manual

CONFIDENTIAL

Linux Interface Specification U-Boot

User's Manual: Software

R-Car H3/M3/M3N/E3/D3 Series

Renesas Electronics
www.renesas.com

Rev.3.00 Dec. 2021

Notice

1. 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 or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving 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, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, 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 and impractical, you are responsible for evaluating the safety of the final products or systems 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. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall 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. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.
(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan

www.renesas.com

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/.

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Trademark

- Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.
- Arm is a registered trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.
- Other company names and product names mentioned herein are registered trademarks or trademarks of their respective owners.
- Registered trademark and trademark symbols (® and ™) are omitted in this document

How to Use This Manual

- **[Readers]**
  This manual is intended for engineers who develop products which use the R-Car H3/M3/M3N/E3/D3 processor.

- **[Purpose]**
  This manual is intended to give users an understanding of the functions of the R-Car H3/M3/M3N/E3/D3 processor device driver and to serve as a reference for developing hardware and software for systems that use this driver.

- **[How to Read This Manual]**
  It is assumed that the readers of this manual have general knowledge in the fields of electrical
  — engineering, logic circuits, microcontrollers, and Linux.
  → Read this manual in the order of the CONTENTS.
  — To understand the functions of a multimedia processor for R-Car H3/M3/M3N/E3/D3
  → See the R-Car H3/M3/M3N/E3/D3 User's Manual.
  — To know the electrical specifications of the multimedia processor for R-Car H3/M3/M3N/E3/D3
  → See the R-Car H3/M3/M3N/E3/D3 Data Sheet.

- **[Conventions]**
  The following symbols are used in this manual.
  Data significance: Higher digits on the left and lower digits on the right
  Note: Footnote for item marked with Note in the text
  Caution: Information requiring particular attention
  Remark: Supplementary information
  Numeric representation: Binary ... ××××, 0b××××, or ××××B
  Decimal ... ××××
  Hexadecimal ... 0x×××× or ××××H
  Data type: Double word … 64 bit
  Word … 32 bits
  Half word ... 16 bits
  Byte ... 8 bits

Table of Contents

1. Overview

This manual explains about U-Boot for R-CarH3-SiP/M3-SiP/M3N-SiP/E3/D3 System Evaluation Board. U-Boot is a boot loader.

1.1 Overview

This boot loader is a boot loader for R-CarH3-SiP/M3-SiP/M3N-SiP/E3/D3 System Evaluation Board to initialize the hardware and boot the Linux kernel.

This boot loader is based on Das U-Boot v2020.10

1.2 Function

The main functions of U-Boot are as follows.

- Setting hardware in chip and board dependence
- Starting of Arm Cortex-A57 MPCore on R-Car H3/M3/M3N
- Starting of Arm Cortex-A53 MPCore on R-Car E3/D3
- I-Cache / D-Cache
- Clock Pulse Generator (CPG)
- Pin Function Controller (PFC)
- Loading kernel from a boot device, and starting
- Executing the boot command which uses a serial console

A boot command is used from a serial console and the following things are possible.

- Save environment variables to eMMC devices (R-Car H3/M3/M3N/E3/D3)
- Read/Write functions for FAT and ext2-4 file system on a SD Card and eMMC devices
- Read/Write functions for FAT and ext2-4 file system on a USB Storage Media (USB 2.0 and USB 3.0)
- Network connection over an USB-Ethernet adapter device

Please refer to 5.2 for the supported boot commands.

The following table shows U-Boot functions in detail

Table 1-1 U-Boot functions

Function	Summary
Evaluation board identification and device tree file selection	U-Boot detects board type and select device tree file to use and outputs that information to the log at startup The format when outputting to the log is as follows CPU: Renesas Electronics XXXXXX rev ZZZ Model: Renesas YYYYYY board based on XXXXXX ZZZ XXXXXX: CPU Name YYYYYY: Board Name ZZZ: Board Revision The implementation depending on the information obtained from the IPL: "DTB containing DRAM layout from BL2 to BL33 via register x3". BL33 here is U-Boot. Based on that, U-Boot selects the device tree to be used. This means that the same U-Boot image can be used for a specific System Evaluation Board
Arm Core setting - Arm Cortex-A57 MPCore [R-Car H3/M3/M3N] - Arm Cortex-A53 MPCore [R-Car E3/D3]	Instruction cache, Data cache are available with this boot loader At the time of power activation, U-Boot executes on CPU0, other CPUs are in shutdown state (power domain OFF) and not operate during U-Boot operation U-Boot uses Arm generic timer. The Arm generic timer control is started by IPL and U-Boot uses it in active state
CPG/MSSR setting	Initial clock supply setting. Set the clock supply state at modules to the CPG (to supply module power) Clock supply of the used module will be stopped before the kernel starts
Local memory setting for functions	This boot loader is executable on - LPDDR4-SDRAM on R-Car H3/M3/M3N - DDR3/DDR3L-SDRAM on R-CarE3/D3 System Evaluation Board U-Boot uses main memory as a work area. For the memory map, refer to 4.4 for more detail. U-Boot set following functions: MMU, debug trace, address relocation, memory allocation, board information, global data, fdt, and stack
PFC/GPIO setting	Set the PFC configuration (Pin functions for LSI pins) Set the GPIO configuration (Power supply and function controls)
Serial console output setting - SCIF2 [R-Car H3/M3/M3N/E3/D3]	Set the UART configuration for standard Input/Output devices In: serial@XXXXXXXX Out: serial@XXXXXXXX Err: serial@XXXXXXXX XXXXXXXX: e6e88000 [R-Car H3/M3/M3N/E3/D3]
Network setting - AVB0 - [R-Car H3/M3/M3N/E3/D3]	U-Boot controls On-chip EthernetAVB and Ethernet PHY on the R-Car H3-SiP/M3-SiP/M3N-SiP/E3/D3 System Evaluation Board. - [R-Car H3-SiP/M3-SiP/M3N-SiP System Evaluation Board] Supports 1Gbps and 100 Mbps transfer - [R-Car E3/D3 System Evaluation Board] Supports 100 Mbps transfer, not support 1Gbps transfer (EthernetAVB hardware specification restriction). U-Boot setups the network and support for TFTP Transfer

	Net: eth0: ethernet@e6800000 Note that the TX clock internal delay mode is not supported on R-Car E3/D3
Boot command execution	U-Boot supports to load kernel from a boot device and start it. U-Boot support to config the boot environment via U-Boot environment variable bootargs (kernel boot parameters) and bootcmd (Command line to boot Linux Kernel) U-Boot also supports various boot commands via a serial console. Please refer to 5.2 for the detail
Support MMC	U-Boot controls Multi Media Card Interface on R-Car H3/M3/M3N/E3/D3 and supports: - Read/Write functions for FAT and ext2-4 file system on an eMMC devices - In case of non-secure boot, save environment variables to eMMC devices (Use second boot partition area of eMMC) - Transfer mode: 1bit, 4bit, 8bit - Bus speed mode: Backward-compatible (25MHz), high-speed (26MHz, 52MHz), HS200 (200MHz) Not support: - Bus speed mode: HS400 (200MHz)
Support SD Memory card	U-Boot controls SD Card Host Interface on R-Car H3/M3/M3N/E3 and supports: - Read/Write functions for FAT and ext2-4 file system on a SD Card devices - Transfer mode: 1bit, 4bit - Bus speed mode: Default Speed (25MHz), High Speed (50MHz), UHS-I: SDR104 (208MHz)/ SDR50 (100MHz)/ SDR25 (50MHz)/ SDR12 (25MHz) Not support: - Bus speed mode: UHS-I: DDR50 (50MHz), UHS-II - SDIO devices
Support USB device	U-Boot controls USB 2.0 Host controller (EHCI) on R-Car H3/M3/M3N/E3/D3, USB 3.0 Host controller (XHCI) on R-Car H3/M3/M3N/E3 and supports: - Read/Write functions for FAT and ext2-4 file system on a USB Storage Media (USB 2.0 and USB 3.0) - Network connection over an USB-Ethernet adapter device Not support USB 1.x devices

1.3 Supported module

The following describes connectors related to some modules controlled by U-Boot and their support status

1.3.1 Serial Interface

This boot loader uses SCIF2 on R-Car H3/M3/M3N/E3/D3 for serial console. Configure the terminal setting is as follows:

[Setting value] baud rate 115200bps, 8bit data, parity none, stop 1 bit, flow control none

Table 1-2 SCIF Connector (R-Car H3/M3/M3N)

Channel	Connector	Support status	Remark
SCIF1	CN26 USB 2.0 type micro AB receptacle	No	
SCIF2	CN25 USB 2.0 type micro AB receptacle	Yes	Source clock: S3D4 (66.6MHz)

Table 1-3 SCIF Connector (R-Car E3)

Channel	Connector	Support status	Remark
SCIF2	CN25 USB 2.0 type micro AB receptacle	Yes	Source clock: S3D4C (66.6MHz)

Table 1-4 SCIF Connector (R-Car D3)

Channel	Connector	Support status	Remark
SCIF2	CN25 USB 2.0 type micro AB receptacle	Yes	Source clock: S3D4C (66.6MHz)

1.3.2 Ethernet Interface

This boot loader uses On-chip EthernetAVB ch0 and set Ethernet PHY on R-CarH3-SiP/M3-SiP/M3N-SiP/E3/D3 System Evaluation Board

Table 1-5 Ethernet PHY Device (R-Car H3/M3/M3N/E3/D3)

Vendor	Product	Interface	Support status	Remark
Microchip	KSZ9031RNXVx	RGMII	Yes	

Table 1-6 Ethernet PHY Connector (R-Car H3/M3/M3N/E3/D3)

Channel	Connector	Support status	Remark
AVB0	CN22 RJ-45	Yes	

1.3.3 SD Card Host/MMC memory Interface

Table 1-7 SDHI Connector (R-Car H3/M3/M3N)

Channel	Connected to	Support status	Remark
SDHI0	CN13 SD Card Connector	Yes	
SDHI2(MMC0)	eMMC Connector	Yes	
SDHI3	CN12 SD Card Connector	Yes	

Table 1-8 SDHI Connector (R-Car E3)

Channel	Connected to	Support status	Remark
SDHI0	CN13 SD Card Connector	Yes	
SDHI1	CN14 microSD Card Connector	Yes	
SDHI3(MMC1)	eMMC Connector	Yes	

Table 1-9 SDHI Connector (R-Car D3)

Channel	Connected to	Support status	Remark
SDHI2(MMC0)	eMMC Connector	Yes	

1.3.4 USB 2.0/3.0 Interface

This boot loader supports USB Storage Media (USB 2.0 and USB 3.0) and USB-Ethernet adapter device (Refer to 5.3.3)

This boot loader does not support USB 1.x devices

R-Car D3 does not support USB 3.0 Interface

Table 1-10 USB 2.0 Connector (R-Car H3)

Channel	Connector	Support status	Remark
USB2.0 ch0	CN9 USB 2.0 type micro AB receptacle	No	
USB2.0 ch1	CN10 Lower Part USB 2.0 type A receptacle	Yes	
USB2.0 ch2	CN10 Upper Part USB 2.0 type A receptacle	Yes	
USB2.0 ch3	CN37 USB 2.0 type micro AB receptacle	No	On R-CarH3-SiP System Evaluation Board Salvator-XS only

Table 1-11 USB 2.0 Connector (R-Car M3/M3N)

Channel	Connector	Support status	Remark
USB2.0 ch0	CN9 USB 2.0 type micro AB receptacle	No	
USB2.0 ch1	CN10 Lower Part USB 2.0 type A receptacle	Yes	

Table 1-12 USB 2.0 Connector (R-Car E3/D3)

Channel	Connector	Support status	Remark
USB2.0 ch0	CN9 USB 2.0 type micro AB receptacle	Yes	

Table 1-13 USB 3.0 Connector (R-Car H3/M3/M3N/E3)

Channel	Connector	Support status	Remark
USB3.0 ch0	CN11 USB 3.0 type A receptacle	Yes	

1.4 Reference

1.4.1 Standards

The following table shows the referenced documentations on standards.

Table 1-14 Related Standards

Reference No.	Issue	Title	Edition	Date
-	SD Card Association	SD Specifications Part 1 Physical Layer Simplified Specification	4.10	Jan. 22, 2013
JESD84-B51	JEDEC STANDARD Multi Media Card Association	Embedded Multi-Media Card (e•MMC) Electrical Standard (5.1)	5.1	Feb. 2015
-	IEEE Standards Association	IEEE Standard 802.3	2000	Mar. 30, 2000
-	USB Implementers Forum, Inc	Universal Serial Bus Specification	2.0	Apr. 27, 2000
-	Philips Semiconductors	THE I2C-BUS SPECIFICATION	2.1	Jan. 2000

1.4.2 Related documents

The following table shows the document related to U-Boot.

Table 1-15 Related document (R-Car H3/M3/M3N/E3/D3)

Reference No.	Issue	Title	Edition	Date
-	Renesas Electronics	R-Car Series, 3rd Generation User's Manual: Hardware	Rev.2.20	Jul. 30, 2020
-	Renesas Electronics	R-CarH3-SiP System Evaluation Board Salvator-X Hardware Manual RTP0RC7795SIPB0011S	Rev.1.09	May. 11, 2017
-	Renesas Electronics	R-CarM3-SiP System Evaluation Board Salvator-X Hardware Manual RTP0RC7796SIPB0011S	Rev.0.04	Oct. 3, 2016
-	Renesas Electronics	R-CarH3-SiP/M3-SiP/M3N-SiP System Evaluation Board Salvator-XS Hardware Manual	Rev.2.04	Jul. 17, 2018
-	Renesas Electronics	R-CarE3 System Evaluation Board Ebisu Hardware Manual RTP0RC77990SEB0010S	Rev.0.03	Apr. 11, 2018
-	Renesas Electronics	R-CarE3 System Evaluation Board Ebisu-4D (E3 board 4xDRAM) Hardware Manual	Rev.1.01	Jul. 19, 2018
-	Renesas Electronics	R-CarD3 System Evaluation Board Hardware Manual RTP0RC77995SEB0010S	Rev.1.20	Jul. 25, 2017

1.5 Restrictions

- None.

1.6 Notice

- Read operations from USB2.0 storage media are supported. However, it is not possible to read data directly from USB to a memory area exceeding 32 bits (i.e. over 0x0600000000). As hardware specifications, only support to handle data within 32 bits address area.
- USB-Ethernet function is not available with the default setting. To enable it, follow the settings in 5.3.3
- Ethernet fixed-link is not available with the default setting. To enable it, follow the settings in 5.3.4
- U-Boot serial console will not be available when its work memory in SDRAM overwritten.

2. Terminology

The following table shows the terminology and abbreviated name related to this boot loader.

Table 2-1 Term / Abbreviation list

Term / Abbreviation	Contents
U-Boot	Boot loader for R-CarH3-SiP/M3-SiP/M3N-SiP/E3/D3 System Evaluation Board explained in this manual.
Salvator-X, Salvator-XS, Ebisu, Draak	Abbreviated name of R-CarH3-SiP/M3-SiP/M3N-SiP/E3/D3 System Evaluation Board. Using with the source code of U-Boot.
IPL	Initial Program Loader.
ATF	Arm Trusted Firmware
CPG	Clock Pulse Generator
MSSR	Module Standby, Software Reset
PFC	Pin Function Controller
GPIO	General-Purpose Input/Output Ports
DVFS	Dynamic Voltage Frequency Scaling
MMC	Multi Media Card
eMMC	Embedded Multi Media Card
SD	Secure Digital
SDIO	Secure Digital Input/Output
SDHI	SD card host interface H/W module
USB	Universal Serial Bus
OHCI	Open Host Controller Interface
EHCI	Enhanced Host Controller Interface
XHCI	Extensible Host Controller Interface
EthernetAVB	Ethernet Audio Video Bridging
RGMII	Reduced Gigabit Media Independent Interface
SCIF	Serial Communications Interface with FIFO

3. Operating environment

3.1 Hardware Environment

This boot loader can operate in the following hardware environment.

Table 3-1 Hardware specification (R-Car H3/M3/M3N/E3/D3)

Name	Version	Manufacturer
R-CarH3-SiP System Evaluation Board Salvator-X	-	Renesas Electronics
R-CarM3-SiP System Evaluation Board Salvator-X	-	Renesas Electronics
R-CarH3-SiP/M3-SiP/M3N-SiP System Evaluation Board Salvator-XS	-	Renesas Electronics
R-CarE3 System Evaluation Board Ebisu	-	Renesas Electronics
R-CarE3 System Evaluation Board Ebisu-4D	-	Renesas Electronics
R-CarD3 System Evaluation Board Draak	-	Renesas Electronics

3.2 State Transition Diagram

There is no state transition diagram for this module.

4. Detail

4.1 File configuration

The file list that modified or added for R-Car H3/M3/M3N/E3/D3 is shown in a Table 4-1.

Table 4-1 File list for R-Car H3/M3/M3N/E3/D3

File path (Below "rcar-bsp-u-boot/" is indicated)	File name	Add / Modify	Contents
arch/arm/dts/	r8a77951.dtsi	Add	Device tree source for r8a77951
	r8a77960.dtsi	Modify	Device tree source for r8a77960
	r8a77961.dtsi	Add	Device tree source for r8a77961
	r8a77965.dtsi	Modify	Device tree source for r8a77965
	r8a77990.dtsi	Modify	Device tree source for r8a77990
	r8a77995.dtsi	Modify	Device tree source for r8a77995
	r8a779x-u-boot.dtsi	Modify	U-Boot specific device tree source
	r8a77951-u-boot.dtsi	Add	U-Boot specific device tree source
	r8a77960-u-boot.dtsi	Modify	U-Boot specific device tree source
	r8a77961-u-boot.dtsi	Add	U-Boot specific device tree source
	r8a77965-u-boot.dtsi	Modify	U-Boot specific device tree source
	r8a77990-u-boot.dtsi	Modify	U-Boot specific device tree source
	r8a77995-u-boot.dtsi	Modify	U-Boot specific device tree source
	salvator-common.dtsi	Modify	Device tree source for Salvator-X/XS board
	salvator-x.dtsi	Modify	Device tree source for Salvator-X board
	salvator-xs.dtsi	Add	Device tree source for Salvator-XS board
	r8a77951-salvator-x.dts	Add	Device tree source for r8a77951
	r8a77951-salvator-x-u-boot.dts	Add	U-Boot specific device tree source
	r8a77951-salvator-xs.dts	Add	Device tree source for r8a77951
	r8a77951-salvator-xs-u-boot.dts	Add	U-Boot specific device tree source
	r8a77960-salvator-x.dts	Modify	Device tree source for r8a77960
	r8a77960-salvator-x-u-boot.dts	Modify	U-Boot specific device tree source
	r8a77960-salvator-xs.dts	Add	Device tree source for r8a77960
	r8a77960-salvator-xs-u-boot.dts	Add	U-Boot specific device tree source
	r8a77961-salvator-xs.dts	Add	Device tree source for r8a77961
	r8a77961-salvator-xs-u-boot.dts	Add	U-Boot specific device tree source
	r8a77965-salvator-x.dts	Modify	Device tree source for r8a77965
	r8a77965-salvator-x-u-boot.dts	Modify	U-Boot specific device tree source
	r8a77965-salvator-xs.dts	Add	Device tree source for r8a77965
	r8a77965-salvator-xs-u-boot.dts	Add	U-Boot specific device tree source
	r8a77990-ebisu.dts	Modify	Device tree source for r8a77990
	r8a77990-ebisu-u-boot.dts	Modify	U-Boot specific device tree source
	r8a77995-draak.dts	Modify	Device tree source for r8a77995
	r8a77995-draak-u-boot.dts	Modify	U-Boot specific device tree source
arch/arm/lib/	crt0_64.S	Modify	Change to not relocate
	bootm.c	Modify	Disable LMB reservation
arch/arm/mach-rmobile/	cpu_info.c	Modify	CPU functions

	cpu_info-rcar.c	Modify	CPU functions
	memmap-gen3.c	-	DRAM area configuration
	lowlevel_init_gen3.S	Modify	Save boot parameters from ATF
board/renesas/salvator-x/	salvator-x.c	Modify	Salvator-X/XS Board initialize and setting
board/renesas/ebisu/	ebisu.c	Modify	Ebisu Board initialize and setting
board/renesas/draak/	draak.c	Modify	Draak Board initialize and setting
board/renesas/rcar-common/	common.c	Modify	DRAM/Board configuration
cmd/	load.c	Modify	Add guard program code area
	net.c	Modify	Network command support (Fix cache)
configs/	rcar3_salvator-x_defconfig	Modify	Default configuration for r8a77951/ r8a77960/r8a77961/r8a77965 on Salvator-X/XS board
	r8a77990_ebisu_defconfig	Modify	Default configuration for r8a77990 on Ebisu/Ebisu-4D board
	r8a77995_draak_defconfig	Modify	Default configuration for r8a77995 on Draak board
drivers/clk/renesas/	renesas-cpg-mssr.h	Modify	Module Standby, Software Reset
	renesas-cpg-mssr.c	Modify	Module Standby, Software Reset
	r8a7795-cpg-mssr.c	Modify	Specific SoC setting
	r8a7796-cpg-mssr.c	Modify	Specific SoC setting
	r8a77965-cpg-mssr.c	Modify	Specific SoC setting
	r8a77990-cpg-mssr.c	Modify	Specific SoC setting
	r8a77995-cpg-mssr.c	Modify	Specific SoC setting
	rcar-gen3-cpg.h	Modify	Clock Pulse Generator driver
	rcar-gen3-cpg.c	Modify	Clock Pulse Generator driver
drivers/gpio/	gpio-rcar.c	Modify	GPIO driver
drivers/i2c/	rcar_iic.c	-	I2C-DVFS driver
	rcar_i2c.c	Modify	I2C driver
drivers/misc/	i2c_eeprom.c	Modify	I2C EEPROM framework
drivers/mmc/	tmio-common.c	Modify	SD/MMC devices handling
	tmio-common.h	Modify	SD/MMC devices handling
	mmc.c	Modify	SD/MMC devices handling
	renesas-sdhi.c	Modify	SDHI driver
drivers/net/	ravb.c	Modify	EthernetAVB driver
drivers/net/phy/	phy.c	Modify	Fixed PHY support
drivers/pinctrl/renesas/	sh_pfc.h	Modify	PFC header
	pfc.c	Modify	PFC driver
	pfc-r8a7795.c	Modify	Specific SoC setting
	pfc-r8a7796.c	Modify	Specific SoC setting
	pfc-r8a77965.c	Modify	Specific SoC setting
	pfc-r8a77990.c	Modify	Specific SoC setting
	pfc-r8a77995.c	Modify	Specific SoC setting
drivers/serial/	serial_sh.h	Modify	SCIF driver header
	serial_sh.c	Modify	SCIF driver
drivers/usb/host/	ehci-generic.c	-	Generic EHCI driver
	xhci-rcar.c	Modify	USB 3.0 Host Controller driver
include/configs/	rcar-gen3-common.h	Modify	Common configuration file
include/	init.h	Modify	Function prototype used during U-Boot initialization
include/dt-bindings/clock/	r8a7795-cpg-mssr.h	Modify	Header from Linux kernel
	r8a7796-cpg-mssr.h	Modify	Header from Linux kernel

	r8a77961-cpg-mssr.h	Add	Header from Linux kernel
	r8a77965-cpg-mssr.h	Modify	Header from Linux kernel
	r8a77990-cpg-mssr.h	Modify	Header from Linux kernel
	r8a77995-cpg-mssr.h	Modify	Header from Linux kernel
include/dt-bindings/power/	r8a7795-sysc.h	Modify	Header from Linux kernel
	r8a7796-sysc.h	Modify	Header from Linux kernel
	r8a77961-sysc.h	Add	Header from Linux kernel
	r8a77965-sysc.h	Modify	Header from Linux kernel
	r8a77990-sysc.h	Modify	Header from Linux kernel
	r8a77995-sysc.h	Modify	Header from Linux kernel

4.2 Definition of build

The definition of build that it is preparing for R-CarH3-SiP/M3-SiP/M3N-SiP/E3/D3 System Evaluation Board is shown in a Table 4-2. In addition, R-Car H3/M3/M3N/E3 supports several memory bank map configurations, which are based on prior-stage firmware DT fragment configuration. Each memory map definition is shown in Table 4-3 to Table 4-7**Error! Reference source not found.**

Table 4-2 Definition of build for R-CarH3-SiP/M3-SiP/M3N-SiP/E3/D3 System Evaluation Board

Target			Definition name	Definition file (under configs/ directory)
CPU	Board	Type		
r8a77951	Salvator-X	Memory 4GiB: 4bank x 1GiB	rcar3_salvator-x_defconfig	rcar3_salvator-x_defconfig
	Salvator-XS	Memory 4GiB: 4bank x 1GiB		
		Memory 4GiB: 2bank x 2GiB		
		Memory 8GiB: 4bank x 2GiB		
r8a77960	Salvator-X	Memory 4GiB: 2bank x 2GiB		
	Salvator-XS	Memory 4GiB: 2bank x 2GiB		
r8a77961	Salvator-XS	Memory 8GiB: 2bank x 4GiB		
r8a77965	Salvator-X	Memory 2GiB: 1bank x 2GiB		
	Salvator-XS	Memory 2GiB: 1bank x 2GiB		
r8a77990	Ebisu	Memory 1GiB: 1bank x 1GiB	r8a77990_ebisu_defconfig	r8a77990_ebisu_defconfig
	Ebisu-4D	Memory 2GiB: 1bank x 2GiB		
r8a77995	Draak	Memory 512MiB 1bank x 512MiB	r8a77995_draak_defconfig	r8a77995_draak_defconfig

! WARNING: R-Car R8A77950 (R-Car H3 Ver.1.x) is NOT supported!

Build example for R-CarH3-SiP DDR 8GiB (2GiB x 4ch)

```
$ make rcar3_salvator-x_defconfig
```

Table 4-3 Memory bank definition of build for R-CarH3-SiP System Evaluation Board

Memory bank type	meaning	
bank num: 4 bank size: 1 GiB	Bank #0: 0x048000000 - 0x07FFFFFFF, Bank #1: 0x500000000 - 0x53FFFFFFF, Bank #2: 0x600000000 - 0x63FFFFFFF, Bank #3: 0x700000000 - 0x73FFFFFFF,	896MiB 1GiB 1GiB 1GiB
bank num: 4 bank size: 2 GiB	Bank #0: 0x048000000 - 0x0BFFFFFFF, Bank #1: 0x500000000 - 0x57FFFFFFF, Bank #2: 0x600000000 - 0x67FFFFFFF, Bank #3: 0x700000000 - 0x77FFFFFFF,	1.9 GiB 2 GiB 2 GiB 2 GiB
bank num:2 bank size: 2GiB	Bank #0: 0x048000000 - 0x0BFFFFFFF, Bank #1: 0x500000000 - 0x57FFFFFFF,	1.9 GiB 2 GiB

Build example for R-CarM3-SiP DDR 8GiB (4GiB x 2ch)

```
$ make rcar3_salvator-x_defconfig
```

Table 4-4 Memory bank definition of build for R-CarM3-SiP System Evaluation Board

Memory bank type	meaning	
bank num: 2 bank size: 2 GiB	Bank #0: 0x048000000 - 0x0BFFFFFFF, Bank #1: 0x600000000 - 0x67FFFFFFF,	1.9 GiB 2 GiB
bank num: 2*1 bank size: 4 GiB	Bank #0: 0x048000000 - 0x0BFFFFFFF, Bank #1: 0x480000000 - 0x4FFFFFFFF Bank #2: 0x600000000 - 0x6FFFFFFFF,	1.9 GiB 2 GiB 4 GiB

*1: The number of memory banks is 2. The first memory bank is defined by dividing it into #0 and #1.

Build example for R-CarM3N-SiP DDR 2GiB (1GiB x 2ch)

```
$ make rcar3_salvator-x_defconfig
```

Table 4-5 Memory bank definition of build for R-CarM3N-SiP System Evaluation Board

Memory bank type	meaning	
bank num: 1 bank size: 2 GiB	Bank #0: 0x048000000 - 0x0BFFFFFFF,	1.9 GiB

Build example for R-CarE3 System Evaluation Board DDR 2GiB (2GiB x 1ch)

```
$ make r8a77990_ebisu_defconfig
```

Table 4-6 Memory bank definition of build for R-CarE3 System Evaluation Board

Memory bank type	meaning
bank num: 1 bank size: 1 GiB	Bank #0: 0x048000000 - 0x07FFFFFF, 896MiB
bank num: 1 bank size: 2 GiB	Bank #0: 0x048000000 - 0x0BFFFFFF, 1.9 GiB

Build example for R-CarD3 System Evaluation Board DDR 512MiB (512MiB x 1ch)

```
$ make r8a77995_draak _defconfig
```

Table 4-7 Memory bank definition of build for R-CarD3 System Evaluation Board

Memory bank type	meaning
bank num: 1 bank size: 512 MiB	Bank #0: 0x048000000 - 0x05FFFFFF, 384MiB

4.3 Boot image configuration

The boot image configuration of U-Boot is shown in Table 4-8. IPL copies the U-Boot image from HyperFlash to the specified area of LPDDR4/LPDDR4X/DDR3/DDR3L-SDRAM. U-Boot start at 0x50000000 on LPDDR4/LPDDR4X/DDR3/DDR3L-SDRAM.

Table 4-8 Boot image list

Image file name	Contents
u-boot-elf.srec	S-record format boot image file. It used at the time of writing by the serial transfer.
u-boot.srec	S-record format boot image file. * Do not use. Because not support Arm 64-bit architecture
u-boot.bin	RAM boot image

4.4 Memory map

4.4.1 Physical address memory map

Physical address memory map of R-Car H3 is shown by Figure 4-1 for DDR 4GiB (1GiB x 4ch). Accessible areas are:

- 0x00_4800_0000 to 0x00_7FFF_FFFF (896MiB), 0x05_0000_0000 to 0x05_3FFF_FFFF (1GiB)
  0x06_0000_0000 to 0x06_3FFF_FFFF (1GiB), 0x07_0000_0000 to 0x07_3FFF_FFFF (1GiB)

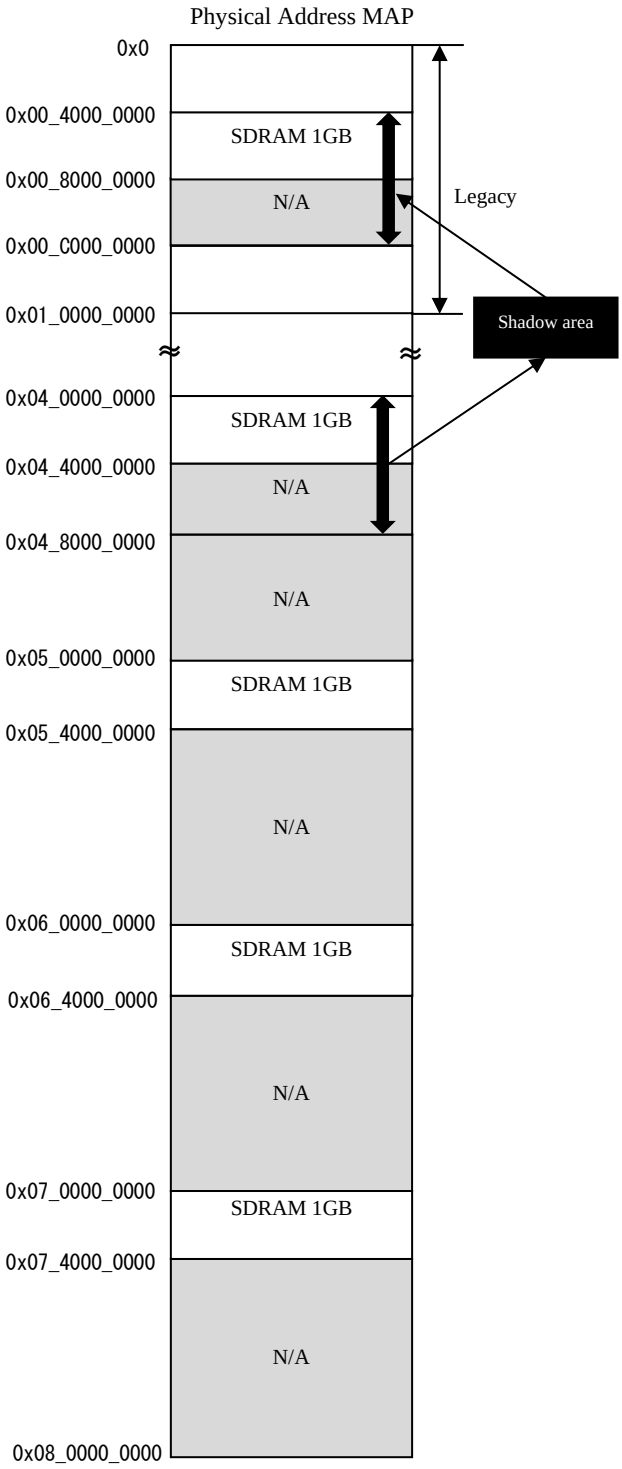


Figure 4-1 Physical memory map for R-Car H3 (4GiB:1GiB x 4ch)

Physical address memory map of R-Car H3 is shown by Figure 4-2 for DDR 8GiB (2GiB x 4ch) and DDR 4GiB (2GiB x 2ch). Accessible areas are:

- 4GiB: 2GiB x 2ch: 0x00_4800_0000 to 0x00_BFFF_FFFF (1920MiB)
  0x06_0000_0000 to 0x06_7FFF_FFFF (2GiB)
- 8GiB: 2GiB x 4ch: 0x00_4800_0000 to 0x00_BFFF_FFFF (1920MiB)
  0x04_8000_0000 to 0x04_FFFF_FFFF (2GiB)
  0x06_0000_0000 to 0x06_7FFF_FFFF (2GiB)
  0x07_0000_0000 to 0x07_7FFF_FFFF (2GiB)

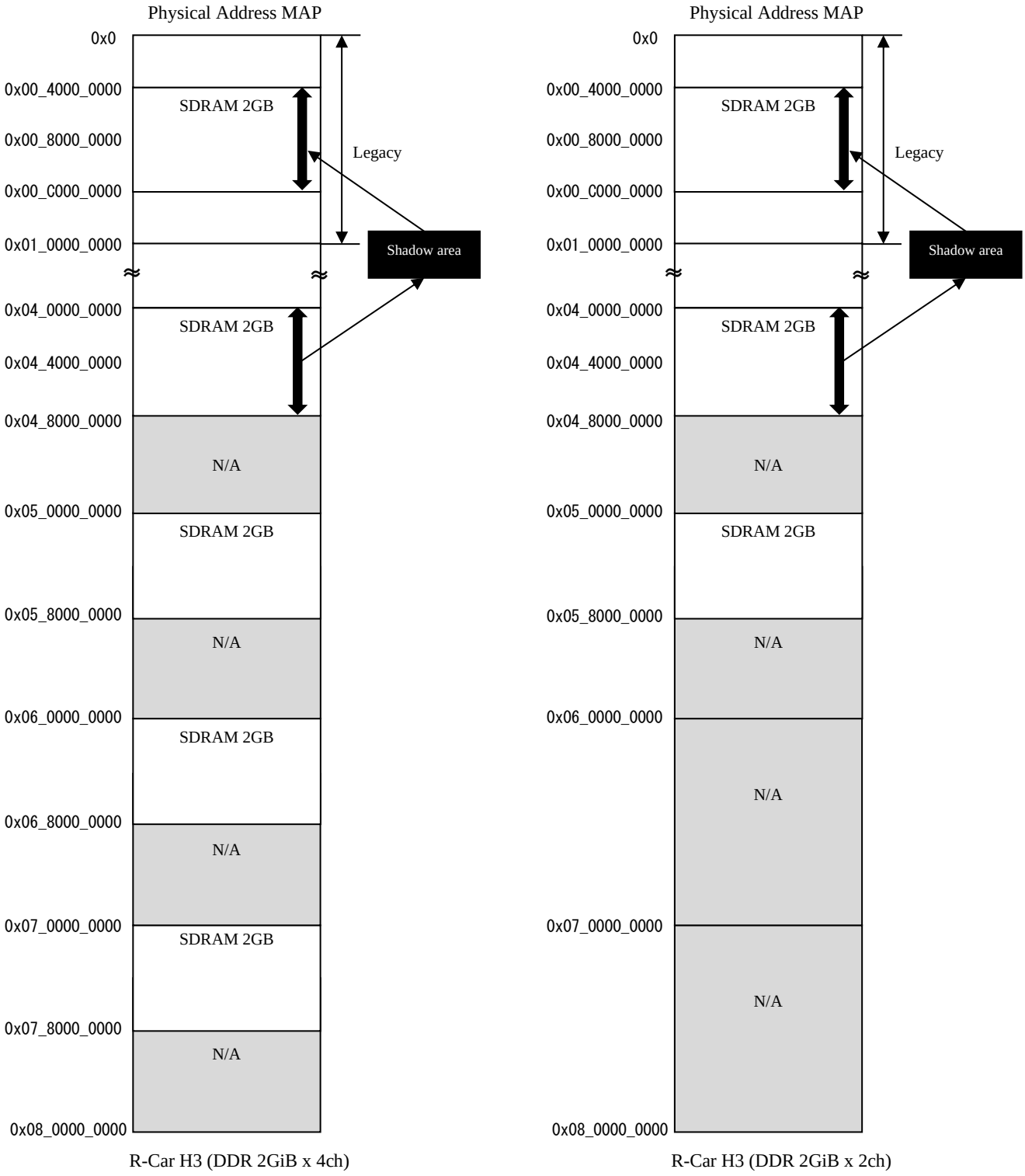


Figure 4-2 Physical memory map for R-Car H3 (8GiB:2GiB x 4ch / 4GiB:2GiB x 2ch)

Physical address memory map for R-Car M3 is shown by Figure 4-3. Accessible areas are:

- 4GiB: 2GiB x 2ch: 0x00_4800_0000 to 0x00_BFFF_FFFF (1920MiB)
  0x06_0000_0000 to 0x06_7FFF_FFFF (2GiB)
- 8GiB: 4GiB x 2ch: 0x00_4800_0000 to 0x00_BFFF_FFFF (1920MiB)
  0x04_8000_0000 to 0x04_FFFF_FFFF (2GiB)
  0x06_0000_0000 to 0x06_FFFF_FFFF (4GiB)

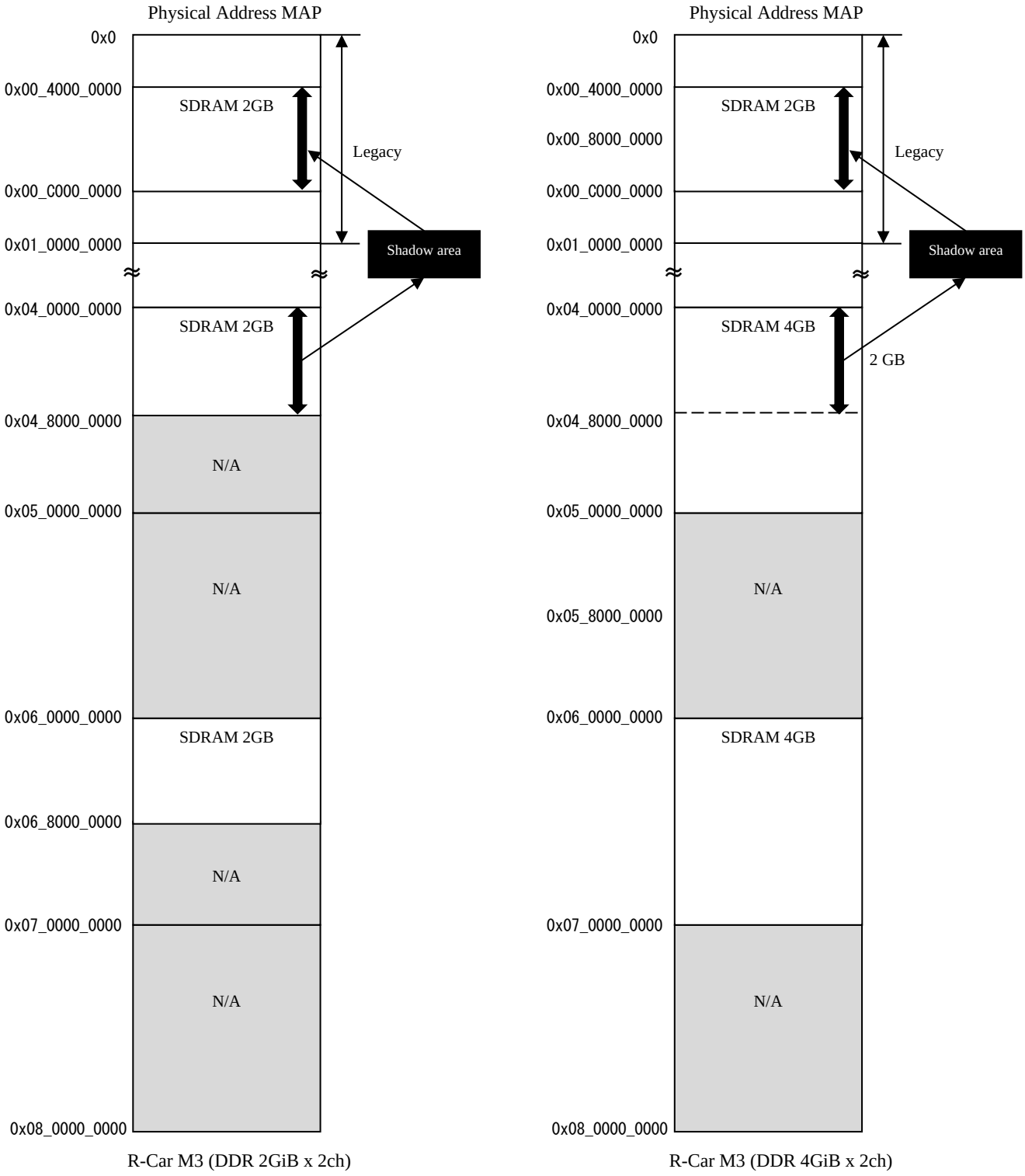


Figure 4-3 Physical memory map for R-Car M3 (4GiB:2GiB x 2ch / 8GiB:4GiB x 2ch)

Physical address memory map for R-Car M3N is shown by Figure 4-4. Accessible area is:

- 0x00_4800_0000 to 0x00_BFFF_FFFF (1920MiB).

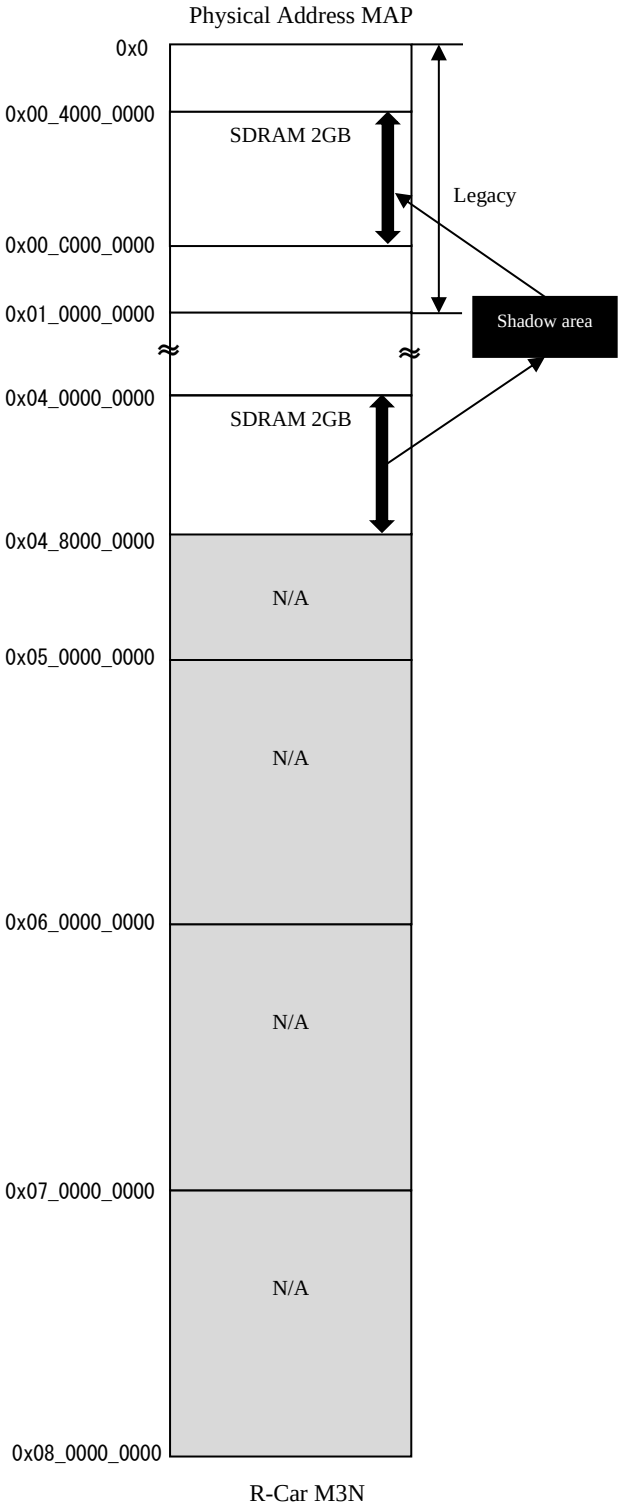


Figure 4-4 Physical memory map for R-Car M3N

Physical address memory map for R-Car E3 System Evaluation Board is shown by Figure 4-5. Accessible area is:

- DDR 1GiB: 0x00_4800_0000 to 0x00_7FFF_FFFF (896MiB)
- DDR 2GiB: 0x00_4800_0000 to 0x00_BFFF_FFFF (1920MiB)

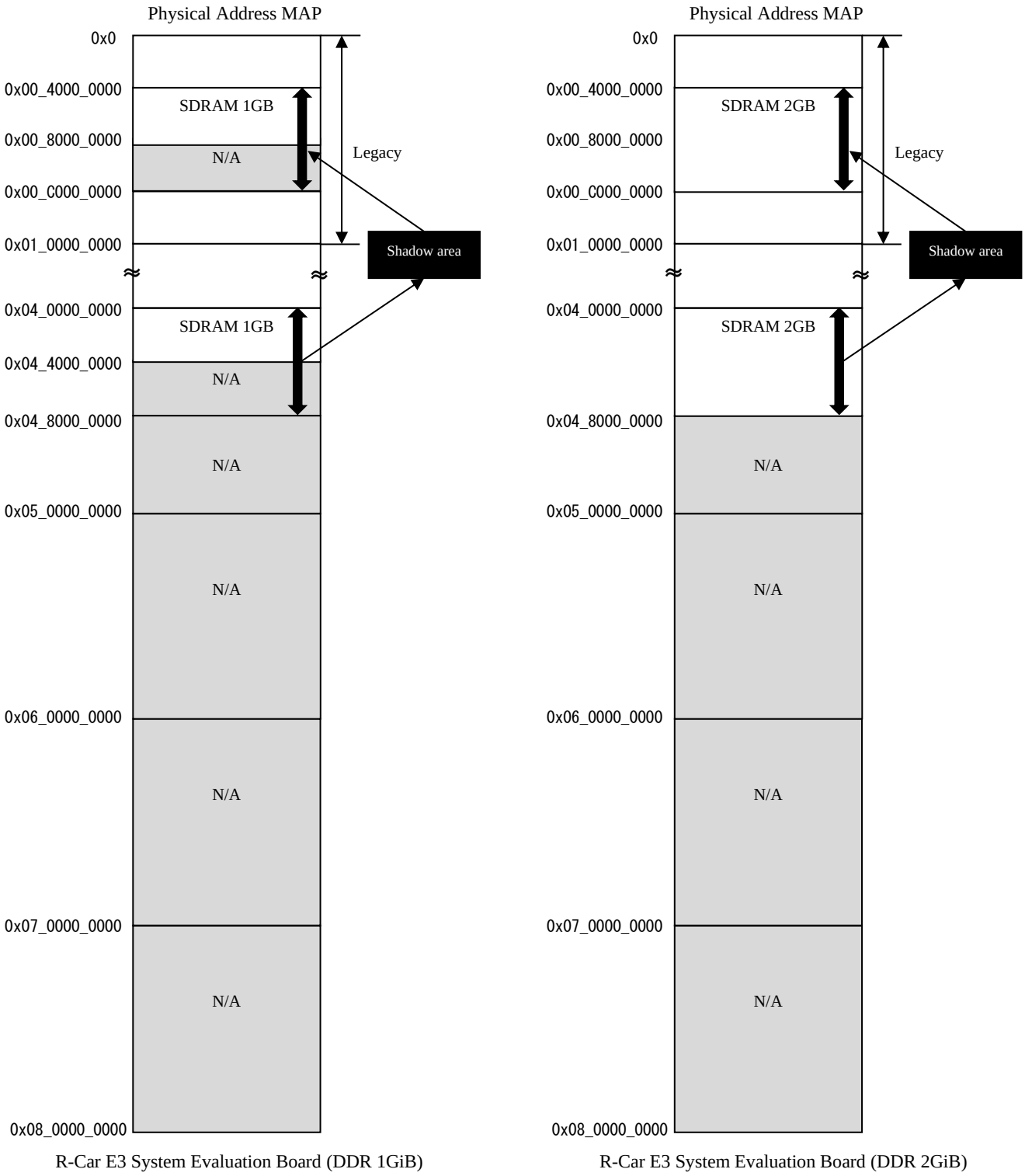
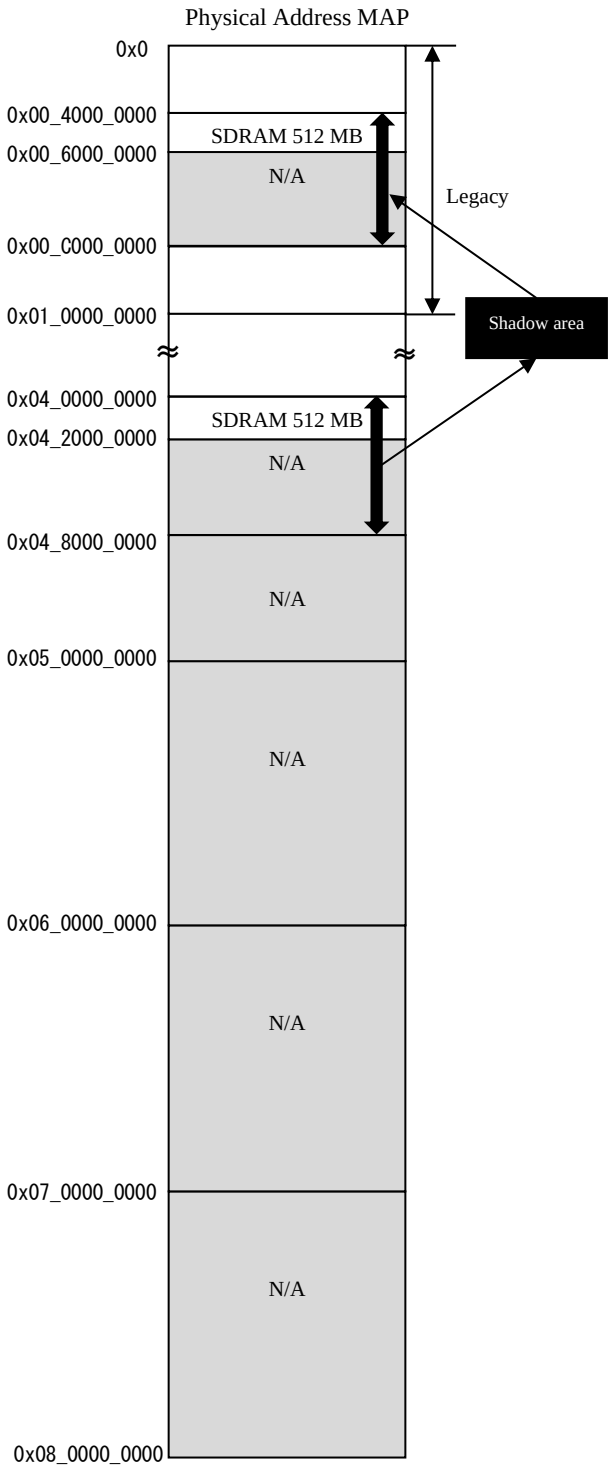


Figure 4-5 Physical memory map for R-Car E3 System Evaluation Board (1GiB / 2GiB)

Physical address memory map for R-Car D3 System Evaluation Board is shown by Figure 4-6. Accessible area is:

- DDR 512MiB: 0x00_4800_0000 to 0x00_5FFF_FFFF (384MiB)



R-Car D3 System Evaluation Board (DDR 512 MiB)

Figure 4-6 Physical memory map for R-Car D3 System Evaluation Board

4.4.2 Legacy memory map

Legacy memory map for R-Car H3 boot loader is shown by Figure 4-7. The maximum size of U-Boot is 1 MiB and uses sectors 25-28. U-Boot is not allowed to access secure region: 0x00_43F0_0000 to 0x00_47E0_0000 (63MiB) in SDRAM.

U-Boot image copy to LPDDR4-SDRAM from HyperFlash by IPL.

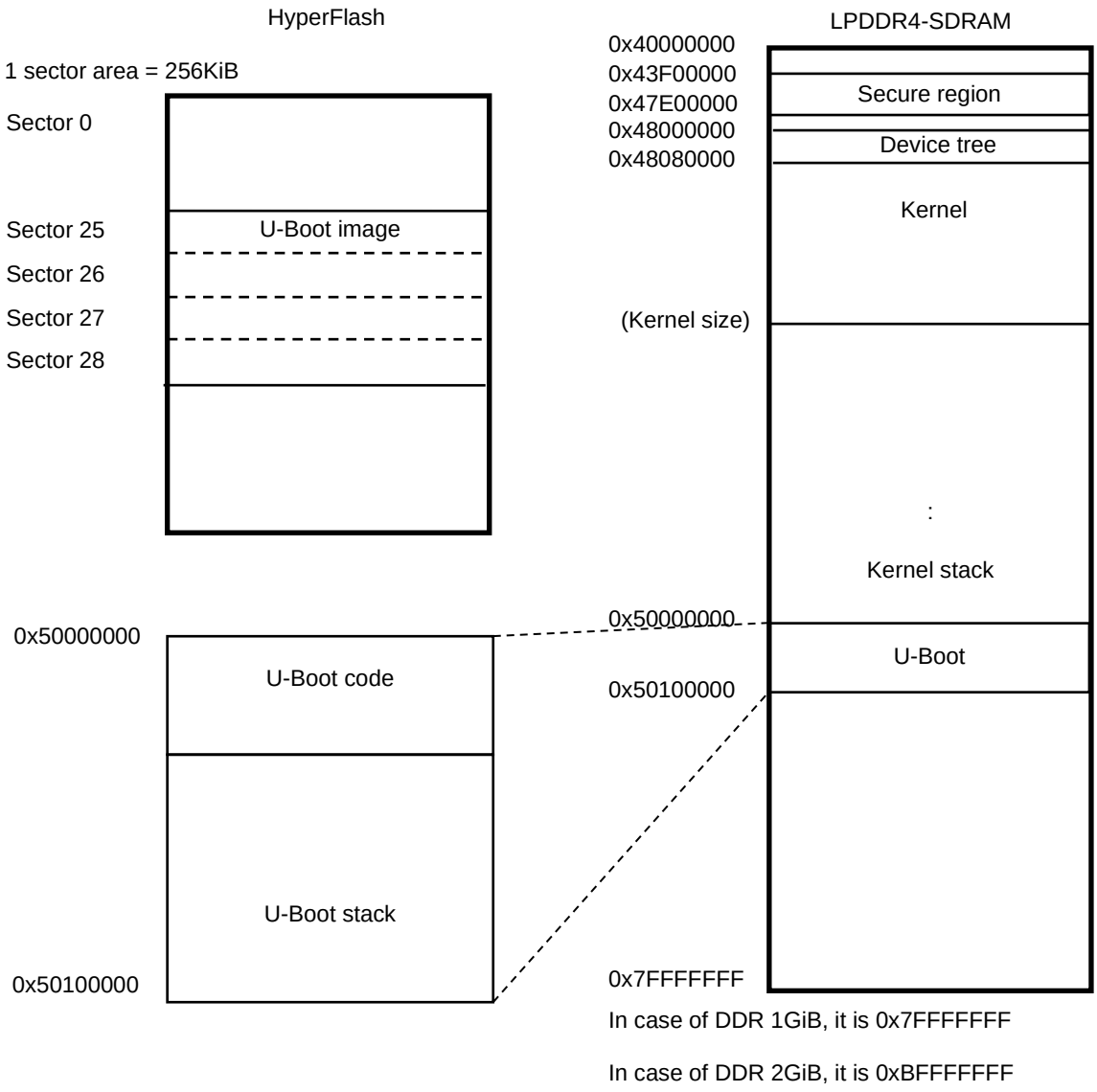


Figure 4-7 Memory map for R-Car H3 U-Boot

Legacy memory map for R-Car M3/M3N boot loader is shown by Figure 4-8. The maximum size of U-Boot is 1 MiB and uses sectors 25-28. U-Boot is not allowed to access secure region: 0x00_43F0_0000 to 0x00_47E0_0000 (63MiB) in SDRAM.

U-Boot image copy to LPDDR4-SDRAM from HyperFlash by IPL.

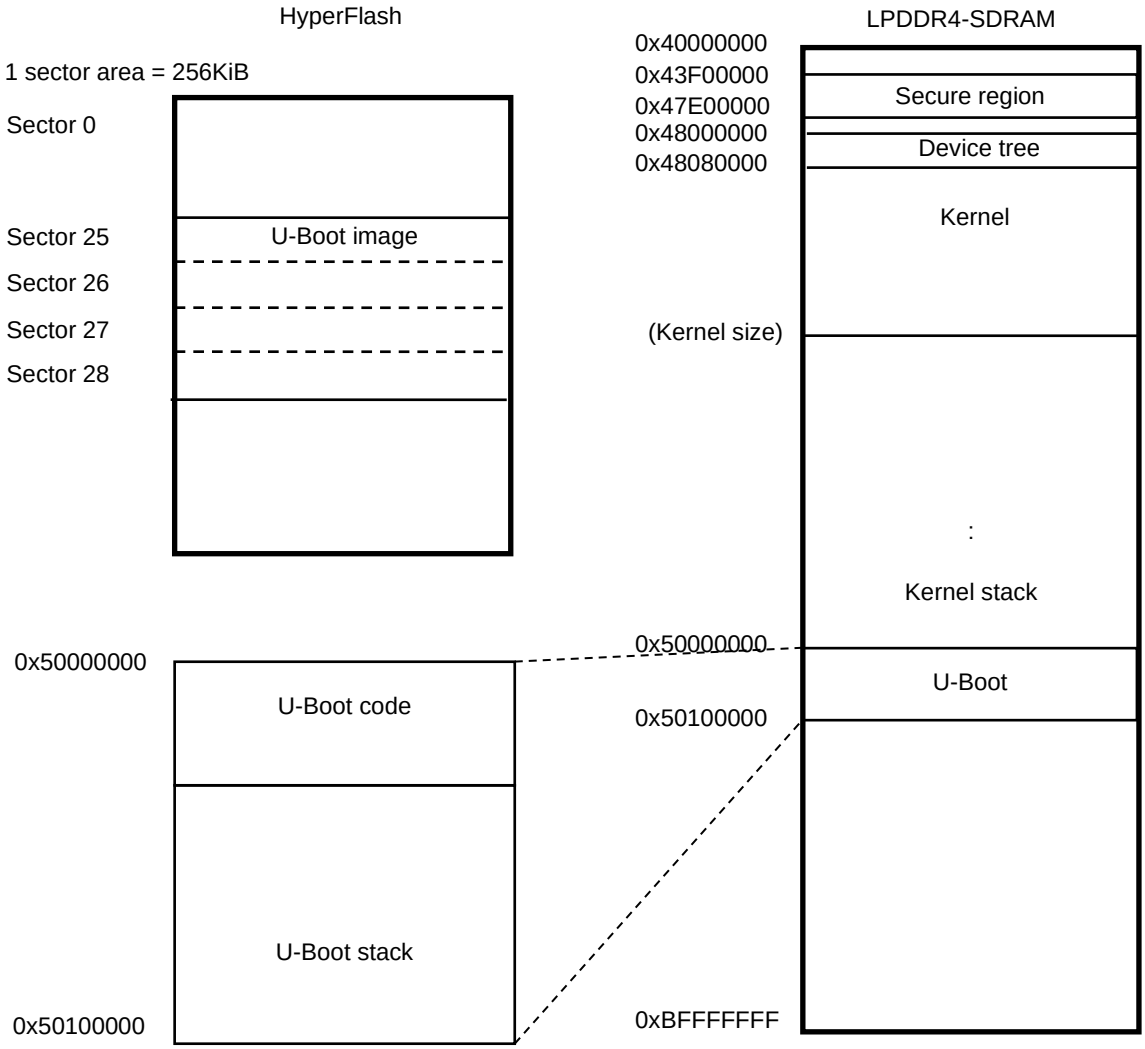


Figure 4-8 Memory map for R-Car M3/M3N U-Boot

Legacy memory map for R-Car E3 boot loader is shown by Figure 4-9. The maximum size of U-Boot is 1 MiB and uses sectors 25-28. U-Boot is not allowed to access secure region: 0x00_43F0_0000 to 0x00_47E0_0000 (63MiB) in SDRAM.

U-Boot image copy to DDR3/DDR3L-SDRAM from HyperFlash by IPL.

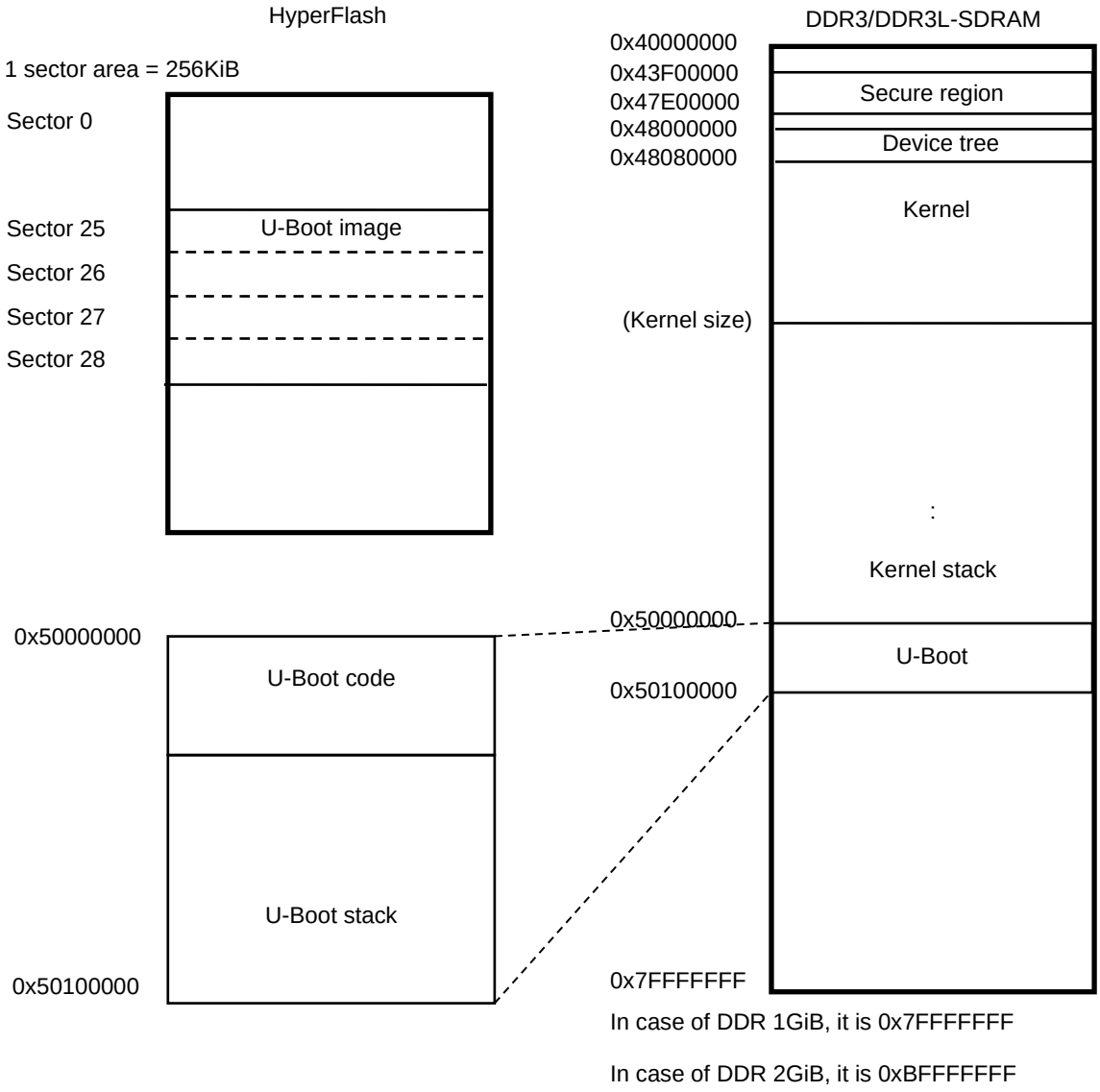


Figure 4-9 Memory map for R-Car E3 U-Boot

Legacy memory map for R-Car D3 boot loader is shown by Figure 4-10. The maximum size of U-Boot is 1 MiB and uses sectors 25-28. U-Boot is not allowed to access secure region: 0x00_43F0_0000 to 0x00_47E0_0000 (63MiB) in SDRAM.

U-Boot image copy to DDR3/DDR3L-SDRAM from HyperFlash by IPL.

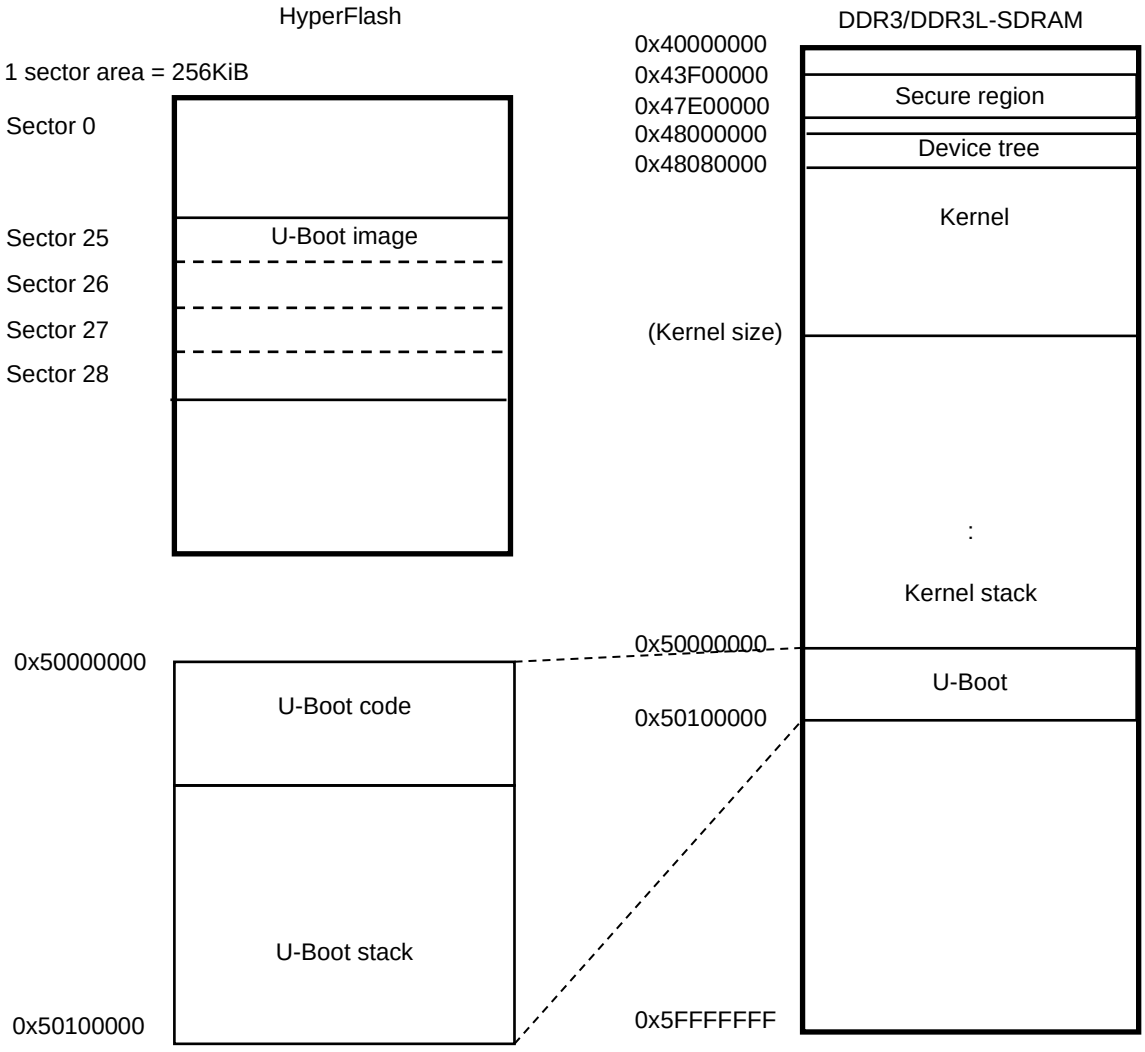


Figure 4-10 Memory map for R-Car D3 U-Boot

4.5 Boot sequence

Boot sequence shown in Figure 4-11.

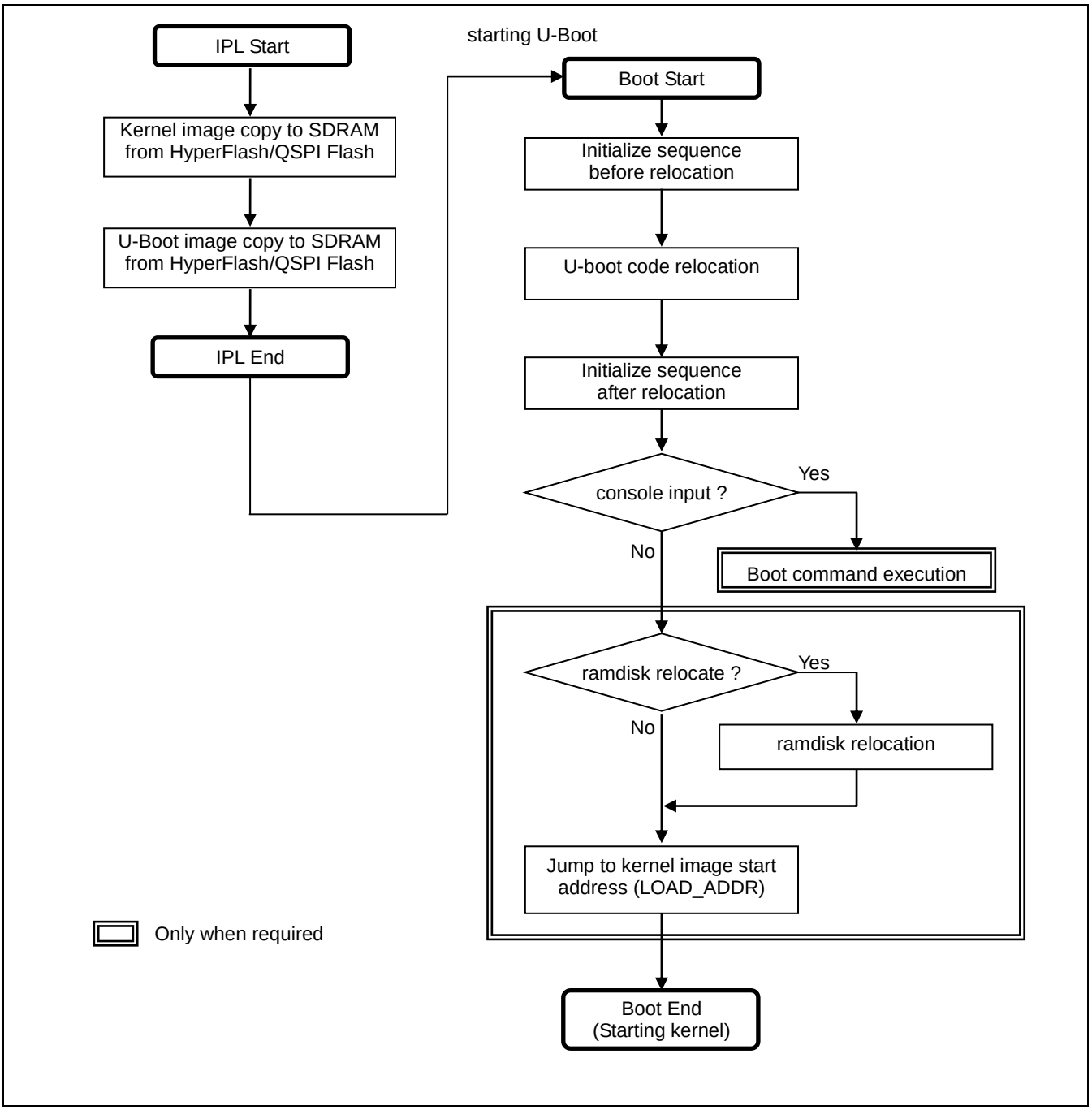


Figure 4-11 U-Boot processing flow

4.5.1 Boot Start

IPL copies the U-Boot image from HyperFlash to the specified area of LPDDR4/LPDDR4X/DDR3/DDR3L-SDRAM.

The IPL provides a Device Tree fragment that contains board information (Salvator-X/Salvator-XS/Ebisu/Draak) and memory bank map configurations (R-Car H3/M3/M3N/E3) to U-Boot via Arm General-purpose registers (X0 – X3).

IPL jumps to the start address of the U-Boot after completion of the processing of the image copy. U-Boot starts at 0x50000000 on LPDDR4/LPDDR4X/DDR3/DDR3L-SDRAM.

4.5.2 Initialize sequence before relocation

It is shown below in the initialization contents processing order of up to relocate itself in U-Boot. U-Boot starts operating on the SDRAM. This case does not relocate.

- U-Boot saves the Device Tree fragment from IPL (contains board information/memory configuration)
- U-Boot merges above Device Tree fragment into its own Device Tree
- Based on the information obtained from the IPL, the device tree to be used is determined by U-Boot and selected. This means that the same U-Boot image can be used for a specific system evaluation board. If it cannot be determined, control is started using the first device tree.
- In the default configuration for the R-CarH3-SiP/M3-SiP/M3N-SiP System Evaluation Board, CONFIG_OF_CONTROL, CONFIG_MULTI_DTB_FIT_LZ0, and CONFIG_OF_LIST are enabled, and it contains both Salvator-X and Salvator-XS device tree.
- The default configuration for the R-CarE3 System Evaluation Board contain Ebisu device tree only. The difference from Ebisu-4D device tree, from U-Boot viewpoint, is minor and U-Boot does not affect by the changes (unused feature)

Table 4-9 Device tree structure for r8a77951/r8a77960/r8a77961/r8a77965

CPU	Board	Type	default config	include device tree names
r8a779951	Salvator-X	Memory 4GiB: 4bank x 1GiB	rcar3_salvator-x_defconfig	r8a77951-salvator-x-u-boot*1
	Salvator-XS	Memory 4GiB: 4bank x 1GiB		r8a77951-salvator-xs-u-boot
		Memory 4GiB: 2bank x 2GiB		
		Memory 8GiB: 4bank x 2GiB		
r8a77960	Salvator-X	Memory 4GiB: 2bank x 2GiB		r8a77960-salvator-x-u-boot
	Salvator-XS	Memory 4GiB: 2bank x 2GiB		r8a77960-salvator-xs-u-boot
r8a77961	Salvator-XS	Memory 8GiB: 2bank x 4GiB		r8a77961-salvator-xs-u-boot
r8a77965	Salvator-X	Memory 4GiB: 2bank x 2GiB		r8a77965-salvator-x-u-boot
	Salvator-XS	Memory 4GiB: 2bank x 2GiB		r8a77965-salvator-xs-u-boot

*1: first device tree

Table 4-10 Device tree structure for r8a77990

Board	Type	default config	include device tree names
Ebisu	Memory 1GiB: 1bank x 1GiB	r8a77990_ebisu_defconfig	r8a77990-ebisu-u-boot*[1]
Ebisu-4D	Memory 2GiB: 1bank x 2GiB		

*1: first device tree

Table 4-11 Device tree structure for r8a77995

Board	Type	default config	include device tree names
Draak	Memory 512MiB: 1bank x 512MiB	r8a77995_draak_defconfig	r8a77995-draak-u-boot

- A setup of Arm Core and enable I-Cache
  - Arm Cortex-A57 MPCore [R-Car H3/M3/M3N]
  - Arm Cortex-A53 MPCore [R-Car E3/D3]
- System Evaluation Board early initialize, supply clocks for:
  - SCIF2, EthernetAVB, eMMC, SDHI0, SDHI3 [R-Car H3/M3/M3N]
  - SCIF2, EthernetAVB, eMMC, SDHI0, SDHI1 [R-Car E3]
  - SCIF2, EthernetAVB, eMMC [R-Car D3]
- Set default environment.

  These environment variables are defined in “include/configs/rcar-gen3-common.h”.

  The default environment variables are shown as follows:

  In case of r8a77951/r8a77960/r8a77961/r8a77965:

```
baudrate=115200
bootargs=root=/dev/nfs rw nfsroot=192.168.0.1:/export/rfs ip=192.168.0.20
bootcmd=tftp 0x48080000 Image; tftp 0x48000000 Image-r8a7795-salvator-x.dtb; booti 0x48080000 - 0x48000000
bootdelay=2
bootm_size=0x10000000
loadaddr=0x58000000
usb_pgood_delay=2000
```

  Note: The default DT is only a detail, as the actual DT to be used to configure U-Boot is detected automatically based on the CPU ID and Board information, hence the default DT is not meaningful.

In case of r8a77990:

```
baudrate=115200
bootargs=root=/dev/nfs rw nfsroot=192.168.0.1:/export/rfs ip=192.168.0.20
bootcmd=tftp 0x48080000 Image; tftp 0x48000000 Image-r8a77990-ebisu.dtb; booti 0x48080000 - 0x48000000
bootdelay=2
bootm_size=0x10000000
loadaddr=0x58000000
usb_pgood_delay=2000
```

In case of r8a77995:

```
baudrate=115200
bootargs=root=/dev/nfs rw nfsroot=192.168.0.1:/export/rfs ip=192.168.0.20
bootcmd=tftp 0x48080000 Image; tftp 0x48000000 Image-r8a77995-draak.dtb; booti 0x48080000 - 0x48000000
bootdelay=2
bootm_size=0x10000000
loadaddr=0x58000000
usb_pgood_delay=2000
```

- UART initialization processing and console input-and-output initialization processing by 115200 bps are performed. Please refer to 1.3.1 for the serial console setting.
- CPU information and board information output to console.

  Example) case of r8a77951 (R-Car H3 Ver.3.0)

  CPU information:

  ```
  CPU: Renesas Electronics R8A7795 rev 3.0
  ```

  Board information:

  ```
  Model: Renesas Salvator-X 2nd version board based on r8a77951
  ```

- Local memory settings for functions. These are MMU, debug trace, address relocation, memory allocation, board information, global data, fdt, and stack.
- If fdt_high not 0xFFFFFFFFFFFFFFFF then relocate fdt.
- Setup u-boot relocation.

4.5.3 U-Boot code relocation

This function is not used.

4.5.4 Initialize sequence after relocation

Initialize run a list of the following.

- Clear bbs data area.
- Enable D-Cache
  - Arm Cortex-A57 MPCore [R-Car H3/M3/M3N]
  - Arm Cortex-A53 MPCore [R-Car E3/D3]
- R-CarH3-SiP/M3-SiP/M3N-SiP/E3/D3 System Evaluation Board initialize.
- Setup serial device assign.
-
- Setup environment. Read the saved environment variables from eMMC Flash. If read environment variables are illegal then use the default environment. Please fefer to Figure 4-12 for eMMC memory map

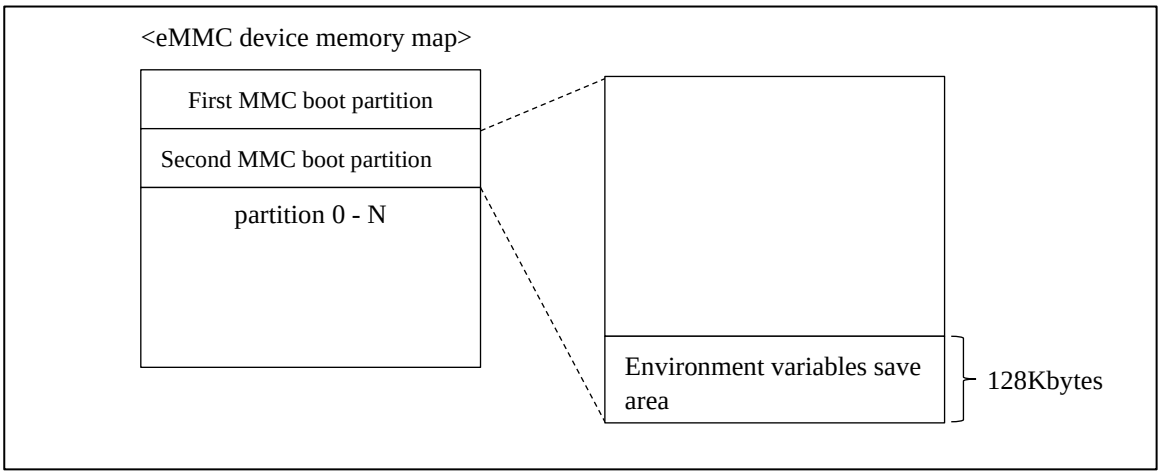


Figure 4-12 Memory map of eMMC (R-Car H3/M3/M3N/E3/D3)

Setup standard Input/Output devices.
[R-Car H3/M3/M3N/E3/D3]

```
In:    serial@e6e88000
Out:   serial@e6e88000
Err:   serial@e6e88000
```

- Setup network.

```
Net:   eth0: ethernet@e6800000
```

4.5.5 Boot command execution

If the input from a serial console is detected during U-Boot execution, it will change to a boot command run state (2 seconds detection waiting in initial setting).

When you use a boot command, please perform the writing of U-Boot, etc. from this state.

In addition, please refer to Table 5-2 for the boot command which can be used.

4.5.6 RAM disk relocation

When settings of the RAM disk is enabled, U-Boot executes the relocation sequence. U-Boot refers to the “initrd_high” environment variables. It does not relocate if the value of "initrd_high" is 0xFFFFFFFFFFFFFFFF.

4.5.7 Start and Stop the module clocks

U-Boot controls a clock of use modules. These controls are done using SMSTPCR. Clock supply of the used module will be stopped before the kernel is started. Refer to 4.7 for more detail

4.5.8 Jump to kernel image start address

Kernel image before the start of the U-Boot, will be copied to the LOAD_ADDR by IPL (initial setting: 0x48080000).

When loading Kernel DTB, U-Boot updates the Device Tree based on memory configuration from IPL before booting into the Operating System. The device tree is then passed to the OS.

U-Boot jump to kernel image start address.

4.5.9 Boot End

The kernel is started.

4.6 Pin Control

This boot loader sets pin function for itself. Following pins are available by controlling PFC module. Details are shown below.

Table 4-12 List of pins configuration for R-Car H3

Module name	Pins name
EthernetAVB	AVB_AVTP_CAPTURE_A, AVG_AVTP_MATCH_A, AVB_LINK, AVB_PHY_INT, AVB_MDC, AVB_AVTP_PPS, AVB_AVTP_MATCH_B, AVB_AVTP_CAPTURE_B, GP_2_10
SDHI0	SD0_DAT0, SD0_DAT1, SD0_DAT2, SD0_DAT3, SD0_CLK, SD0_CMD, SD0_CD, SD0_WP, GP_5_2, GP_5_1
SDHI1/SDHI2, eMMC	SD1_DAT0, SD1_DAT1, SD1_DAT2, SD1_DAT3, SD1_CLK, SD1_CMD, SD2_DAT0, SD2_DAT1, SD2_DAT2, SD2_DAT3, SD2_CLK, SD2_CMD, GP_5_3, GP_5_9
SDHI3	SD3_DAT0(GP_4_9), SD3_DAT1(GP_4_10), SD3_DAT2(GP_4_11), SD3_DAT3(GP_4_12), SD3_CLK(GP_4_7), SD3_CMD(GP_4_8), SD3_CD(GP4_15), SD3_WP(GP4_16), GP_3_15, GP_3_14
USB	USB30_PWEN, USB30_OVC, USB1_PWEN, USB1_OVC. USB2_PWEN(GP_6_14), USB2_OVC(GP_6_15)

Table 4-13 List of pins configuration for R-Car M3/M3N

Module name	Pins name
EthernetAVB	AVB_AVTP_CAPTURE_A, AVG_AVTP_MATCH_A, AVB_LINK, AVB_PHY_INT, AVB_MDC, AVB_AVTP_PPS, AVB_AVTP_MATCH_B, AVB_AVTP_CAPTURE_B, GP_2_10
SDHI0	SD0_DAT0, SD0_DAT1, SD0_DAT2, SD0_DAT3, SD0_CLK, SD0_CMD, SD0_CD, SD0_WP, GP_5_2, GP_5_1
SDHI1/SDHI2, eMMC	SD1_DAT0, SD1_DAT1, SD1_DAT2, SD1_DAT3, SD1_CLK, SD1_CMD, SD2_DAT0, SD2_DAT1, SD2_DAT2, SD2_DAT3, SD2_CLK, SD2_CMD, GP_5_3, GP_5_9
SDHI3	SD3_DAT0(GP_4_9), SD3_DAT1(GP_4_10), SD3_DAT2(GP_4_11), SD3_DAT3(GP_4_12), SD3_CLK(GP_4_7), SD3_CMD(GP_4_8), SD3_CD(GP4_15), SD3_WP(GP4_16), GP_3_15, GP_3_14
USB	USB30_PWEN, USB30_OVC, USB1_PWEN, USB1_OVC.

Table 4-14 List of pins configuration for R-Car E3

Module name	Pins name
EthernetAVB	AVB_RX_CTL, AVB_RXC, AVB_RD0, AVB_RD1, AVB_RD2, AVB_RD3, AVB_TXCREFCLK, AVB_MDIO, AVB_MDC, AVB_PHY_INT, GP_1_20
SDHI0-SD	SD0_CLK, SD0_CMD, SD0_DAT0, SD0_DAT1, SD0_DAT2, SD0_DAT3, SD0_CD, SD0_WP
SDHI1-microSD	SD1_CLK, SD1_CMD, SD1_DAT0, SD1_DAT1, SD1_DAT2, SD1_DAT3, SD1_CD, SD1_WP
SDHI3-eMMC	SD3_CLK, SD3_CMD, SD3_DAT0, SD3_DAT0, SD3_DAT1, SD3_DAT2, SD3_DAT3, SD3_DAT4, SD3_DAT5, SD3_DAT6, SD3_DAT7, SD3_DS
USB	USB30_PWEN, USB30_OVC, USB0_PWEN_B(GP_6_11), USB0_OVC_B(GP_6_12)

Table 4-15 List of pins configuration for R-Car D3

Module name	Pin name
EthernetAVB0	AVB0_LINK (avb0_link group); AVB0_MDC, AVB0_MDIO (avb0_mdio group); AVB0_TX_CTL, AVB0_TXC, AVB0_TD0, AVB0_TD1, AVB0_TD2, AVB0_TD3, AVB0_RX_CTL, AVB0_RXC, AVB0_RD0, AVB0_RD1, AVB0_RD2, AVB0_RD3, AVB0_TXCREFCLK (avb0_mii group)
SDHI/MMC	MMC_D0, MMC_D1, MMC_D2, MMC_D3, MMC_D4, MMC_D5, MMC_D6, MMC_D7 (mmc_data8 group); MMC_CLK, MMC_CMD (mmc_ctrl group);
SCIF0	RX2, TX2 (scif2_data group);
I2C	SCL0, SDA0 (i2c0 group)
USB	USB0_PWEN(GP_0_01), USB0_OVC(GP_0_00) (usb0 group);

4.7 Clock Control

Initial clock supply setting. Set the clock supply state at modules to the CPG (to supply module power)
Clock supply of the used module will be stopped before the kernel starts

Table 4-16 Control modules by U-Boot (R-Car H3/M3/M3N)

Modules	R-Car H3/M3/M3N	At Salvator-X board, and Salvator-XS board	Control registers bit assign
SCIF2	SCIF2	-	SMSTPCR3.10
EthernetAVB	EAVB-IF	-	SMSTPCR8.12
	-	GPIO2	SMSTPCR9.10
SDHI3	SD-IF3	-	SMSTPCR3.11
	-	GPIO3	SMSTPCR9.09
SDHI0/1/2	SD-IF0, SD-IF1, SD-IF2		SMSTPCR3.14, SMSTPCR3.13, SMSTPCR3.12
	-	GPIO5	SMSTPCR9.07
USB-Host	USB3.0-IF0, HS-USB-IF, EHCI/OHCI0, EHCI/OHCI1, EHCI/OHCI2, *[H3 only] EHCI/OHCI3 *[H3 only]	-	SMSTPCR3.28, SMSTPCR7.04, SMSTPCR7.03, SMSTPCR7.02, SMSTPCR7.01, *[H3 only] SMSTPCR7.00 *[H3 only]
	-	GPIO6	SMSTPCR9.06
IIC-DVFS	IIC-DVFS	-	SMSTPCR9.26

Table 4-17 Control modules by U-Boot (R-Car E3)

Modules	R-Car E3	At Ebisu board	Control registers bit assign
SCIF2	SCIF2	-	SMSTPCR3.10
EthernetAVB	EAVB_IF	-	SMSTPCR8.12
	-	GPIO1	SMSTPCR9.11
SDHI0,1	SD_IF0, SD_IF1	-	SMSTPCR3.14, SMSTPCR3.13
	-	GPIO5	SMSTPCR9.07
SDHI3	SD_IF3	-	SMSTPCR3.11
	-	GPIO3	SMSTPCR9.09
USB-Host	USB3.0-IF0, HS-USB-IF, EHCI/OHCI0,	-	SMSTPCR3.28, SMSTPCR7.04, SMSTPCR7.03
	-	GPIO6	SMSTPCR9.06
IIC-DVFS	IIC-DVFS	-	SMSTPCR9.26

Table 4-18 Control modules by U-Boot (R-Car D3)

Modules	R-Car D3	At Draak board	Control registers bit assign
SCIF2	SCIF2	-	SMSTPCR3.10
EthernetAVB	EAVB_IF	-	SMSTPCR8.12
SDHI	eMMC0	-	SMSTPCR3.12
USB-Host	EHCI/OHCI0	GPIO0	SMSTPCR7.03 SMSTPCR9.12
I2C	I2C0	-	SMSTPCR9.31

5. Interface specification

5.1 Environment variables

This boot loader supports environment variables in Table 5-1.

Table 5-1 U-Boot environment variables

Environment variables name	contents
baudrate	Baud rate between U-Boot console and Host PC. Example: => setenv baudrate 115200(Do not change)
bootargs	Kernel boot parameters. Default setting is “root=/dev/nfs rw nfsroot=192.168.0.1:/export/rfs ip=192.168.0.20”
bootcmd	Command line to boot Linux Kernel. This command will be executed after bootdelay seconds. If no definitions are held on this variable, u-boot use console mode. Default setting for r8a77951/r8a77960/r8a77961/r8a77965 is “tftp 0x48080000 Image; tftp 0x48000000 Image-r8a7795-salvator-x.dtb; booti 0x48080000 - 0x48000000” Default setting for r8a77990 is “tftp 0x48080000 Image; tftp 0x48000000 Image-r8a77990-ebisu.dtb; booti 0x48080000 - 0x48000000” Default setting for r8a77995 is “tftp 0x48080000 Image; tftp 0x48000000 Image-r8a77995-draak.dtb; booti 0x48080000 - 0x48000000”
bootdelay	Seconds to wait to execute bootcmd. Example: => setenv bootdelay 2
fdt_high	If set this restricts the maximum address that the flattened device tree will be copied into upon boot. If this is set to the special value 0xFFFFFFFFFFFFFFFF then the fdt will not be copied.
initrd_high	If this variable is not set, initrd images will be copied to the highest possible address in RAM. If this is set to the special value 0xFFFFFFFFFFFFFFFF then the initrd images will not be copied.
ipaddr	IP address of target System Evaluation Board Example: => setenv ipaddr 192.168.0.20
serverip	IP address of TFTP server. or IP address of NFS server on NFS. Example: => setenv serverip192.168.0.1
bootfile	Target filename to download via TFTP. Example: => setenv bootfile Image
stdin	Standard Input Example: => setenv stdin serial (Do not change)
stdout	Standard Output Example:

	=> setenv stdout serial (Do not change)
stderr	Standard Error Example: => setenv stderr serial (Do not change)
usb_pgood_delay	Wait for power to become stable (msec). If not defined, the default wait time is 100 ms. If it is set to a value less than 100, it will be processed with a waiting time of 100 ms. Example => setenv usb_pgood_delay 2000
ver	U-Boot version information.

Note) It means command prompt of the u-boot console as "=>" in each examples.

5.2 Supported command

Please confirm "help" command (help+ command name) about each command usage.

Table 5-2 Supported U-Boot commands

Command	Contents	Remarks
?	alias for 'help'	
base	print or set address offset	
bdinfo	print Board Info structure	
blkcache	block cache diagnostics and control	not supported
boot	boot default, i.e., run 'bootcmd'	not supported
bootd	boot default, i.e., run 'bootcmd'	not supported
bootefi	Boots an EFI payload from memory	not supported
bootelf	Boot from an ELF image in memory	not supported
booti	boot Linux kernel 'Image' format from memory	
bootm	boot application image from memory	
bootp	boot image via network using BOOTP/TFTP protocol	
bootvx	Boot vxWorks from an ELF image	not supported
bootz	boot Linux zImage image from memory	
cmp	memory compare	
coninfo	print console devices and information	
cp	memory copy	
crc32	checksum calculation	
dcache	enable or disable data cache	
dhcp	boot image via network using DHCP/TFTP protocol	
dm	Driver model low level access	
echo	echo args to console	
editenv	edit environment variable	
env	environment handling commands	
exit	exit script	
ext2load	load binary file from a Ext2 filesystem	
ext2ls	list files in a directory (default /)	
ext4load	load binary file from a Ext4 filesystem	
ext4ls	list files in a directory (default /)	
ext4size	determine a file's size	
ext4write	create a file in the root directory	not supported*1
false	do nothing, unsuccessfully	not supported

fatinfo	print information about filesystem	
fatload	load binary file from a dos filesystem	
fatls	list files in a directory (default /)	
fatsize	determine a file's size	
fatwrite	write file into a dos filesystem	
fdt	flattened device tree utility commands	
fstype	Look up a filesystem type	
fsuuid	Look up a filesystem UUID	not supported
go	start application at address 'addr'	
gpio	query and control gpio pins	
gpt	GUID Partition Table	not supported
guid	GUID - generate Globally Unique Identifier based on random UUID	not supported
gzwrite	unzip and write memory to block device	not supported
help	print command description/usage	
i2c	I2C sub-system	
icache	enable or disable instruction cache	
iminfo	print header information for application image	not supported
imxtract	extract a part of a multi-image	not supported
itest	return true/false on integer compare	not supported
ln	Create a symbolic link	
load	load binary file from a filesystem	not supported
loadb	load binary file over serial line (kermit mode)	not supported
loads	load S-Record file over serial line	
loadx	load binary file over serial line (xmodem mode)	not supported
loady	load binary file over serial line (ymodem mode)	not supported
loop	infinite loop on address range	
ls	list files in a directory (default /)	
lzmadec	lzma uncompress a memory region	not supported
md	memory display	
mdio	MDIO utility commands	
mii	MII utility commands	
mm	memory modify (auto-incrementing address)	
mmc	MMC sub system	
mmcinfo	display MMC info	
mw	memory write (fill)	
nfs	boot image via network using NFS protocol	
nm	memory modify (constant address)	
pci	list and access PCI Configuration Space	
ping	send ICMP ECHO_REQUEST to network host	
pinmux	show pin-controller muxing	
printenv	print environment variables	
random	fill memory with random pattern	
reset	Perform RESET of the CPU	
run	run commands in an environment variable	
save	save file to a filesystem	not supported
saveenv	save environment variables to persistent storage	
setenv	set environment variables	
sf	SPI flash sub-system	not supported
setexpr	set environment variable as the result of eval expression	not supported

showvar	print local hushshell variables	
size	determine a file's size	not supported
sleep	delay execution for some time	
source	run script from memory	not supported
string	display strings	not supported
sspi	SPI utility command	not supported
test	minimal test like /bin/sh	not supported
tftpboot	boot image via network using TFTP protocol	
true	do nothing, successfully	not supported
unzip	unzip a memory region	not supported
usb	USB sub-system	
usbboot	boot from USB device	not supported*2
uuid	UUID - generate random Universally Unique Identifier	not supported
version	print monitor, compiler and linker version	

*1. For writing, work memory of file system information is necessary, but work memory is not enough.

*2. not available Arm 64-bit architecture Boot image.

5.3 Examples of instructions

5.3.1 Example: booting image from SD Card

Following instructions shows booting images with reading Image from SD Card.
To check file list, listing directory by “fatls” command. Device number is 0 and partition 1 for example.

```
=> mmc dev 0
switch to partitions #0, OK
mmc0 is current device
=> fatls mmc 0:1
   <listing directory>
=>
```

Loading Image to memory address 0x48080000.

```
=> fatload mmc 0:1 0x48080000 Image
reading Image
2126504 bytes read in 201 ms (10.1 MiB/s)
=>
```

Loading dtb file to memory address 0x48000000.

```
=> fatload mmc 0:1 0x48000000 r8a77951-salvator-x.dtb
reading r8a77951-salvator-x.dtb
1518 bytes read in 16 ms (91.8 KiB/s)
=>
```

Boot with loaded Image.

```
=> booti 0x48080000 - 0x48000000
## Flattened Device Tree blob at 48000000 ...
```

5.3.2 Example: booting image from USB storage media

Following instructions shows booting images with reading Image from USB storage media

Connect storage media into USB connector of the System Evaluation Board
After that, enable USB interface in the U-Boot console.

Please confirm the dialogs that USB storage media recognized as Storage Device(s).

```
=> usb start
starting USB...
Bus usb@ee000000: Register 2000120 NbrPorts 2
Starting the controller
USB XHCI 1.00
Bus usb@ee080100: USB EHCI 1.10
Bus usb@ee0a0100: USB EHCI 1.10
Bus usb@ee0c0100: USB EHCI 1.10
scanning bus usb@ee000000 for devices... 1 USB Device(s) found
scanning bus usb@ee080100 for devices... 1 USB Device(s) found
scanning bus usb@ee0a0100 for devices... 2 USB Device(s) found
scanning bus usb@ee0c0100 for devices... 1 USB Device(s) found
       scanning usb for storage devices... 1 Storage Device(s) found
```

To check file list, listing directory by “fatls” command. Device number is 0 and partition 1 for example.

```
=> fatls usb 0:1
   <listing directory>
```

Loading Image to memory address 0x48080000.

```
=> fatload usb 0:1 0x48080000 Image
9863680 bytes read in 357 ms (26.3 MiB/s)
```

Loading dtb file to memory address 0x48f80000.

```
=> fatload usb 0:1 0x48000000 r8a77951-salvator-xs.dtb
28930 bytes read in 56 ms (503.9 KiB/s)
```

Stopping USB module.

```
=> usb stop
stopping USB..
```

Boot with loaded Image.

```
=> booti 0x48080000 - 0x48000000
## Flattened Device Tree blob at 48000000
```

5.3.3 Example: booting image via network connection over USB-Ethernet adapter

To enable USB-Ethernet function, make the following U-Boot Configuration (command "make menuconfig")

```
Device Drivers    --->
  [*] USB support    --->
    [*]     USB to Ethernet Controller Drivers    --->          /* CONFIG_USB_HOST_ETHER */
      [*]     ASIX AX8817X (USB 2.0) support                   /* CONFIG_USB_ETHER_ASIX */
      [*]     ASIX AX88179 (USB 3.0) support                   /* CONFIG_USB_ETHER_ASIX88179 */
```

NOTE: To use the USB-Ethernet adapter not listed in Table 5-3, you need to enable the appropriate USB to Ethernet Controller Drivers in the U-Boot Configuration (command "make menuconfig") then integrate the vendor ID and product ID into their u-boot source code (drivers/usb/eth/<The_Controller_Drivers>.c)

Below shows confirmed USB-Ethernet adapters.

Table 5-3 Confirmed USB-Ethernet adapters

Vendor	Name	Integrated Chip	VendorID	ProductID
Buffalo	LUA3-U2-ATX	ASIX AX88772	0x0b95	0x7720
Planex Communications	UE-100TX-G3	ASIX AX88772	0x0b95	0x7720
Unitek	Y-3461	ASIX AX88179	0x0b95	0x1790

Once the rebuilding and flashing U-Boot finishes, follow below instructions to boot images via network connection over USB-Ethernet adapter

At first, connect USB-Ethernet adapter into USB connector of the System Evaluation Board.
Then, enable USB interface in the U-Boot console.

```
=> usb start
starting USB...
Bus usb@ee000000: Register 2000120 NbrPorts 2
Starting the controller
USB XHCI 1.00
Bus usb@ee080100: USB EHCI 1.10
Bus usb@ee0a0100: USB EHCI 1.10
Bus usb@ee0c0100: USB EHCI 1.10
scanning bus usb@ee000000 for devices... 1 USB Device(s) found
scanning bus usb@ee080100 for devices... 1 USB Device(s) found
scanning bus usb@ee0a0100 for devices...
Warning: ax88179_eth using MAC address from ROM
2 USB Device(s) found
scanning bus usb@ee0c0100 for devices... 1 USB Device(s) found
       scanning usb for storage devices... 0 Storage Device(s) found
=>
```

Set default ethernet device to USB-Ethernet adapter, i.e. ax88179_eth

```
=> setenv ethact ax88179_eth
=> printenv ethact
ethact=ax88179_eth
=>
```

Prepare a working tftp setup. Configure MAC address and IP address

```
=> setenv ethaddr xx:xx:xx:xx:xx:xx
=> setenv ipaddr aa.bb.cc.dd
=> setenv serverip AA.BB.CC.DD
```

Confirm connection with the ping command.

```
=> ping AA.BB.CC.DD
Using ax88179_eth device
host AA.BB.CC.DD is alive
```

After the above initialization, it is able to boot images from tftpserver

```
=> tftp 0x48080000 Image; tftp 0x48000000 <Device_Tree>.dtb; booti 0x48080000 - 0x48000000
=>
```

5.3.4 Example: enable Ethernet fixed-link support

To enable Ethernet fixed-link support, make the following changes:

Firstly, specify CONFIG_PHY_FIXED in each board default configuration file.

For example: Update U-Boot <board>_defconfig: CONFIG_PHY_FIXED=y

Secondly, update U-Boot DTS to add fixed-link node, where 'fixed-link' is a sub-node of the Ethernet MAC device node, with the following properties:

- 'speed' (integer, mandatory), to indicate the link speed.
  Accepted values are 1000 (R-Car H3/M3/M3N) and 100 (R-Car E3/D3)
- 'full-duplex' (boolean, optional), to indicate that full duplex is used.

For example:

```
&avb {
    ...
    fixed-link {
          speed = <1000>;
          full-duplex;
    };
    ...
};
```

REVISION HISTORY	Linux Interface Specification U-Boot User's Manual: Software

Rev.	Date	Description	
		Page	Summary
0.1	Sep. 25, 2015	—	New creation.
0.2	Nov. 20, 2015	All	- Support SDHI(SD Card/eMMC)
		3	- Table 2.2 Hardware macro list Add support SDHI.
		4	- 3.2 Using resource Add about eMMC resource.
		6	- Table 3.2 list of pins configuration Change SDHI0, 1, 2, 3 and eMMC pins.
		8	- Table 5.1 file list for R-Car H3 Add SDHI driver files. Add modified files. These modified files are USB and network for support 64bit code.
		12	- 5.5.2 Add description about default environment, memory map of eMMC and output information of CPU and Board.
		16	- 6.2 Supported command Add command of 'mmc', 'mmcinfo' and 'saveenv'.
0.3	Apr. 15, 2016	All	- Add R-Car M3 support.
		2	- 1.3.2 Related documents Update Table 1.2 Related document (R-Car H3/M3).
		4	- 3.2 Using resource Add details of LPDDR4-SDRAM area.
		5	- 3.2 Using resource Change S3D4 clock (66.6MHz) at Table 3.1.
		6	- 3.2 Using resource Add Table 3.3 list of drvctrl settings for R-Car H3/M3.
		8	- 4. Control of device Remove GSX power on and clock supply.
		8	- 4.7 Ether AVB Add description about the support 1000Mbps transfer.
		9	- 5.1 File configuration Add/Update files for R-Car M3 at Table 5.1.
		11	- 5.4.1 Physical address memory map Add physical address memory map.
		12	- 5.4.2 Legacy area memory map Add R-Car M3 memory map.
		17	- 5.5.7 Start and Stop the module clocks Add this chapter.
0.4	Aug. 5, 2016	2	- 1.3.2 Related documents Update Table 1.2 Related document (R-Car H3/M3).
		4	- 3.1 Hardware Environment Update Table 3.1 Hardware specification (R-Car H3/M3).
		7	- 3.2 Using resource Add Note to the Table 3.4 List of drvctrl settings for R-Car H3/M3
		9	- 5.1 File configuration Update Table 5.1 file list for R-Car H3/M3
		17	- 5.5.7 Start and Stop the module clocks Change the RMSTPCR register setting specification.
		20	- 6.2 Supported command Change show the confirmed usb-ether device only.
0.5	Dec. 16, 2016	4	- 3.2 Using resource Change use of HyperFlash memory area.
		8	- 4 Control of device Remove the control of HyperFlash memory.

		10	- 5.3 Boot image configuration Change the description content.
		12,13	- 5.4.2 Legacy area memory map Change load address of U-Boot code.
		12,13	- 5.4.2 Legacy area memory map Change use of HyperFlash memory area.
		14	- 5.5.1 Boot start Change the start address to 0x50000000
		17	- 5.5.7 Start and Stop the module clocks Change the initial values of SMSTPCR0 (BIT16) and SMSTPCR8 (BIT0) of H3.
0.6	Mar. 15, 2017	2	- 1.3.2 Related Documents Update related documents.
		5	- 3.1 Hardware Environment Add R-CarH3-SiP/M3-SiP System Evaluation Board Salvator-XS.
		6	- 3.2 Using resource Remove the “list of drvctrl settings”.
		10	- 5.2 Definition of build Add support the R-CarH3-Sip/M3-Sip System Evaluation Board.
0.7	Jun. 14, 2017	2	- 1.3.2 Related Documents Update related documents (Refer Rev 0.54).
		7	- 3.2 Using resource Fix H/W revision notation from WS to Ver.
		17	- 5.5.7 Start and Stop the module clocks Change the clock controls.
1.00	Aug. 8, 2017	All	Update document format.
		9	- 5.1 File configuration Update the file list which modified or added.
1.01	Oct. 24, 2017	All	Add R-Car M3N support.
		3	- 1.5 Notice Added explanation about memory area which can be accessed directly from USB memory.
1.50	Jan. 29, 2018	2	- 1.3.2 Related Documents Update related documents (Refer Rev 0.80).
		3	- 1.5 Notice Update USB-Ethernet description for R8A77965.
		5	- 3.2 Using resource Update description about using Arm generic timer.
		22	- 6.2 Supported command Add command of ‘dcache’ and ‘icache’.
1.51	Mar. 28, 2018	All	Add R-Car E3 support.
		13	- 5.2 definition of Build Add a description about the build for the Salvator-X board.
1.52	Apr. 25, 2018	5	- 3.2 Using resource Add memory map of DDR 8GiB (2GiB x 4ch) and 4GiB (2GiB x 2ch) for R-Car H3
		13	- 5.2 definition of Build Add explanation about build of new memory map for R-Car H3
		16	- 5.4.1 Physical address memory map Add memory map of DDR 8GiB (2GiB x 4ch) and 4GiB (2GiB x 2ch) for R-Car H3
1.53	Jun. 27, 2018	All	Remove R-Car H3 Ver.1.0/1.1 support (supported Ver.2.0 or later).
		All	Add R-Car M3 Ver.1.1/1.2 support.
1.54	Sep. 26, 2018	2	- 1.3.2 Related Documents Update related documents (Refer Rev 1.00).
1.60	Oct. 29, 2018	1	- 1.1 Overview Update base code to v2018.09
		3	- 1.4 Restrictions Add SD and USB Restrictions.
		3	- 1.5 Notice Update USB Ethernet activation procedure.
		5	- 3.2 Using resource Add support for 2GiB memory map of R-Car E3.

		11,12	- 5.1 file configuration Update file list which modified or added (Talbe 5-1).
		13,14	- 5.2 Definition of Build Update default configuration files for Build (Table 5-2, 5-3).
		18	- 5.4.1 physical address memory map Add support for 2GiB physical memory map of R-Car E3 (Figure 5-4).
		23,24	- 5.5.2 Initialize sequence before relocation Update environment variables.
		25	- 5.5.4 Initialize sequence after relocation Update setup information.
		25	- 5.5.5 Boot command execution Change initial wait time to 2 sec.
		27,28	- 6.1 Environment variables Update the initial (default) setting of the “bootcmd” variable. Add description for “usb_pgood_delay” variable.
		28-30	- 6.2 Supported command Update to v2018.09 based command list.
		31-34	- 6.3 Examples of instructions Update to v2018.09 based description.
2.00	Dec. 25, 2018	-	- Update AddressList
		2	- 1.3.2 Related Documents Update related documents.
		5	- 3.1 Hardware Environment Add M3N-SiP System Evaluation Board Salvator-XS. Add R-CarE3 System Evaluation Board Ebisu-4D.
		11,12	- 5.1 file configuration Update the file list in Table 5-1 (arch/arm/dts/r8a77965-salvator-xs-u-boot.dts, arch/arm/dts/r8a77965-salvator-xs.dts, cmd/net.c, drivers/net/ravb.c).
		23,24	- 5.5.2 Initialize sequence before relocation Update default device tree file name and addition of “usb_pgood_delay” parameter.
		26	- 5.5.7 Start ans Stop the module clocks Update control target of GPIO and MSTPCR when using Ethernet module in Table 5-5.
		27	- 6.1 Environment variables Update bootcmd default parameters in Table 6-1.
2.01	Apr. 17, 2019	-	- Update AddressList
		All	- Update description about support M3N-SiP.
		2	- 1.3.2 Related Documents Update related documents (R-Car Series, 3rd Generation User’s Manual: Refer Rev 1.50).
		10	- 4.3 eMMC Add Table 4-1 eMMC function
		10	- 4.4 SD Add Table 4-2 SD function
		11	- 4.6 EtherAVB (Giga-bit Ethernet) Update description on R-Car E3 network transfer.
		20-22	- 5.4.2 Legacy area memory map Update the use memory area size of Hyper Flash to 1 Mbyte.
		24,25	- 5.5.2 Initialize sequence before relocation

			Add specification about detect board type.
2.02	Jun. 26, 2019	5	- 3.2 Using resource Add support for 8GiB(4GiB x 2ch) memory map of R-Car M3.
		13,14	- 5.1 File configuration Update the file list in Table 5-1 (arch/arm/dts/r8a7796-salvator-xs-2x4g-u-boot.dts, arch/arm/dts/r8a7796-salvator-xs-2x4g.dts, configs/r8a7796_salvator-x-2x4g_defconfig).
		15,16	- 5.2 definition of Build Add support for 8GiB(4GiB x 2ch) memory map of R-Car M3
		21	- 5.4.1 physical address memory map Add support for 8GiB(4GiB x 2ch) physical memory map of R-Car M3 (Figure 5-3).
		28	- 5.5.2 Initialize sequence before relocation Add support for 8GiB(4GiB x 2ch) memory map configurations of R-Car M3.
2.50	Apr. 24, 2020	-	- Update Address List
		All	- Update document format
		1 – 6	- Clarify U-Boot functions and supported modules
		7	- Update related standards and documents
		All	- Update for U-Boot v2020.01 supports: + Support USB 3.0 and revise Notice chapter + Support Write function on USB 2.0 and revise Notice chapter + Update file configuration and definition of builds + Update Initialize sequence before relocation + Update supported command
		All	- Update document chapters: + Previous “3.2. Using resource” and “4. Control of device” are moved to “1.2 Function”, “1.3 Supported module”, “4.6 Pin Control” and “4.7 Clock Control” + Move USB-Ethernet information to 6.3.3 and support more adapter
		16 – 25	- Update Memory Map: + Clarify accessible area + Clarify that U-Boot is not allowed to access secure region
		All	Add R-Car V3U support
2.51	Jun. 24, 2020	-	- Update Address List
		1 – 2	- Remove incorrect L2 cache information
		11 – 16 28 – 29	- Support same build configurations as previous software release
2.52	Dec. 1, 2020	1 – 7, 9, 14	- Clarify SPI Flash Memory device support on R-Car V3U + Update functions and supported modules table + Update file configuration table + Update Notice
		4	- Update R-Car V3U SCIF0 source clock
		6	- Clarify support status for microSD Card
		8	- Update related documents
		27	- Update legacy memory map for R-Car V3U to use QSPI Flash
		28	- Update U-Boot processing flow to clarify that the images can be loaded from HyperFlash/QSPI Flash
		33	- Update 4.5.4 Initialize sequence after relocation to describe the location to save environment variables on R-Car V3U
		36	- Update pin control table for R-Car V3U
		37	- Update clock control table for R-Car V3U
		41	- Describe SPI related commands, available on R-Car V3U

2.53	Jan. 29, 2021	All	Add R-Car V3H Ver.2.0 support
2.54	May 20, 2021	All	Add R-Car V3M Ver.2.0 support
2.55	Aug. 16, 2021	-	Support U-Boot v2020.10
		-	Add R-Car D3 support
		-	Remove R-Car V3U/V3H/V3M information since they are not available yet on U-Boot v2020.10
3.00	Dec. 10, 2021	-	Update Notice to v5.0.1

CONFIDENTIAL

Linux Interface Specification U-Boot
User's Manual: Software

Publication Date: Rev.0.1 Sep. 25, 2015
Rev.3.00 Dec. 10, 2021

Published by: Renesas Electronics Corporation

SALES OFFICES

Renesas Electronics Corporation

http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics Corporation
TOYOSU FORESIA, 3-2-24 Toyosu, Koto-ku, Tokyo 135-0061, Japan

Renesas Electronics America Inc. Milpitas Campus
1001 Murphy Ranch Road, Milpitas, CA 95035, U.S.A.
Tel: +1-408-432-8888, Fax: +1-408-434-5351

Renesas Electronics America Inc. San Jose Campus
6024 Silver Creek Valley Road, San Jose, CA 95138, USA
Tel: +1-408-284-8200, Fax: +1-408-284-2775

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 101-T01, Floor 1, Building 7, Yard No. 7, 8th Street, Shangdi, Haidian District, Beijing 100085, China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai 200333, China
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, #06-02 Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit No 3A-1 Level 3A Tower 8 UOA Business Park, No 1 Jalan Pengaturcara U1/51A, Seksyen U1, 40150 Shah Alam, Selangor, Malaysia
Tel: +60-3-5022-1288, Fax: +60-3-5022-1290

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL 2nd Stage, Indiranagar, Bangalore 560 038, India
Tel: +91-80-67208700

Renesas Electronics Korea Co., Ltd.
17F, KAMCO Yangjae Tower, 262, Gangnam-daero, Gangnam-gu, Seoul, 06265 Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5338

Colophon 7.0

ルネサス エレクトロニクス株式会社

■営業お問合せ窓口

http://www.renesas.com

※営業お問合せ窓口の住所は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス株式会社 〒135-0061 東京都江東区豊洲3-2-24（豊洲フォレシア）

■技術的なお問合せおよび資料のご請求は下記へどうぞ。
総合お問合せ窓口：https://www.renesas.com/contact/

Colophon 3.1

Linux Interface Specification
U-Boot

Renesas Electronics Corporation