RH850/U2B Group

R01AN6915EJ0100
Rev.1.00

VM Implementation by Virtualization Application Note

Summary

This application describes VM (virtual machine) implementation operation example by virtualization function of RH850/U2Bx.

Aim of this document and software is to provide supplemental information for the function on RH850/U2B. It is not intended to implement in the design for mass production.

There is no guarantee to update in this document and software to reflect the latest manual, errata, technical update and development environment. You are fully responsible for the incorporation or any other use of the information of this document in the design of your product or system, and please refer to latest manual, errata, technical update and development environment.

Target Device

- RH850/U2B Group

Target Integrated Development Environment

CS+(from RENESAS Electronics)

Version : E8.07.00g6

Device file : DR7F702Z21EDBA.DVF

Reference Document

RH850/U2B User's Manual Hardware

For function details and electrical characteristics, please refer to "User's Manual: Hardware".

This application note is based on the following manual.

・ RH850/U2B User's Manual (Rev.0.80): R01UH0923EJ0080

Items

1. Virtualization Function

1.1 Virtualization

It is the function to implement the multiple systems with different characteristics on one chip by sophistication of the requirements system.

This function facilities the application development and the maintenance by providing an environment in which multiple systems to be implemented can operate independently without mutual interference.

1.2 VM Allocation Mode

There are multiple systems for implementing an independent system VM (virtual machine) in virtualization depending on om the CPU implemented and VM implemented.

“Figure 1-1 VM Allocated Model” shows the system example of each combination.

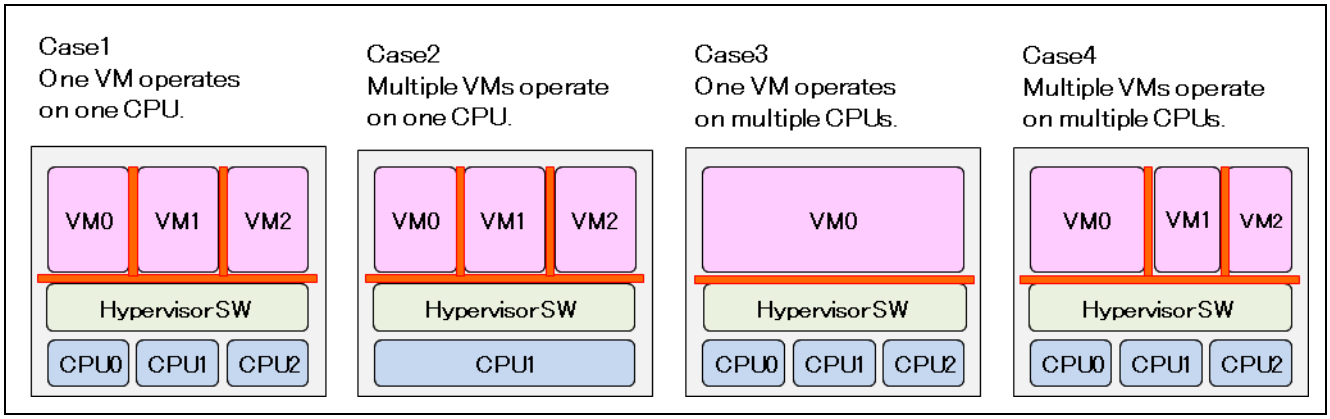

Figure 1-1 VM Allocation Model

This APN describes an implementation example when multiple VM are operated on one CPU in Case 2.

1.3 VM Time Division Operation

Each VM operation in this APN is performed by timer division by timer function and interrupt.

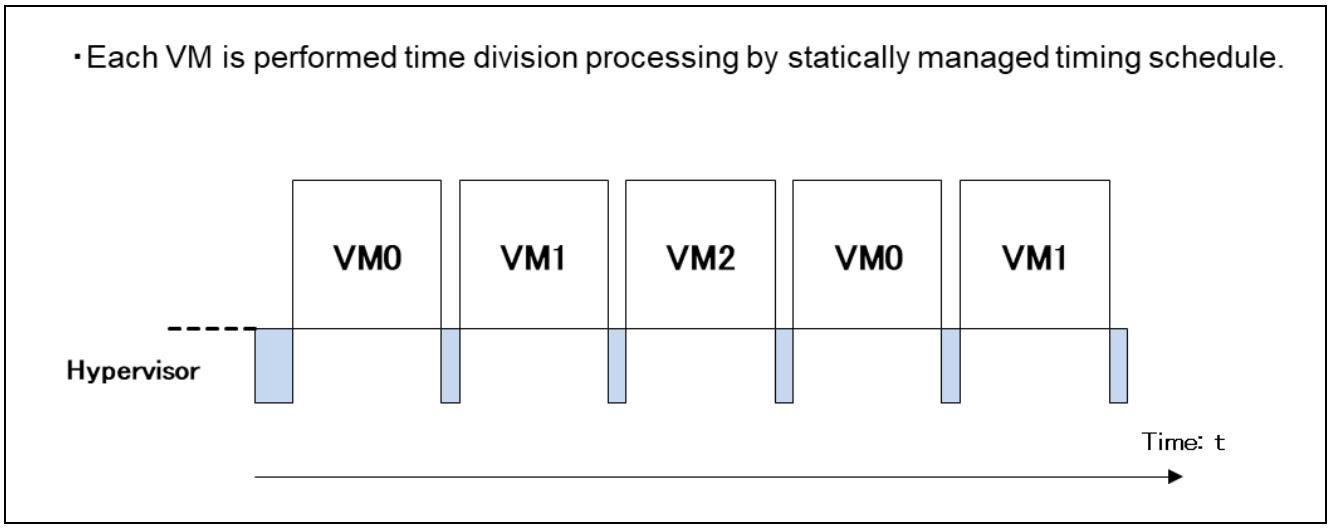

Figure 1-2 Operation Image of Time Division Processing

1.4 Operation Mode and Authority

The system transits into the virtualization mode after booting. In the virtualized state, it is implemented as the host mode with the hypervisor authority and the guest mode with the supervisor authority, and the state transitions between the two modes.

"Figure 1-3 State Transition" shows each stats.

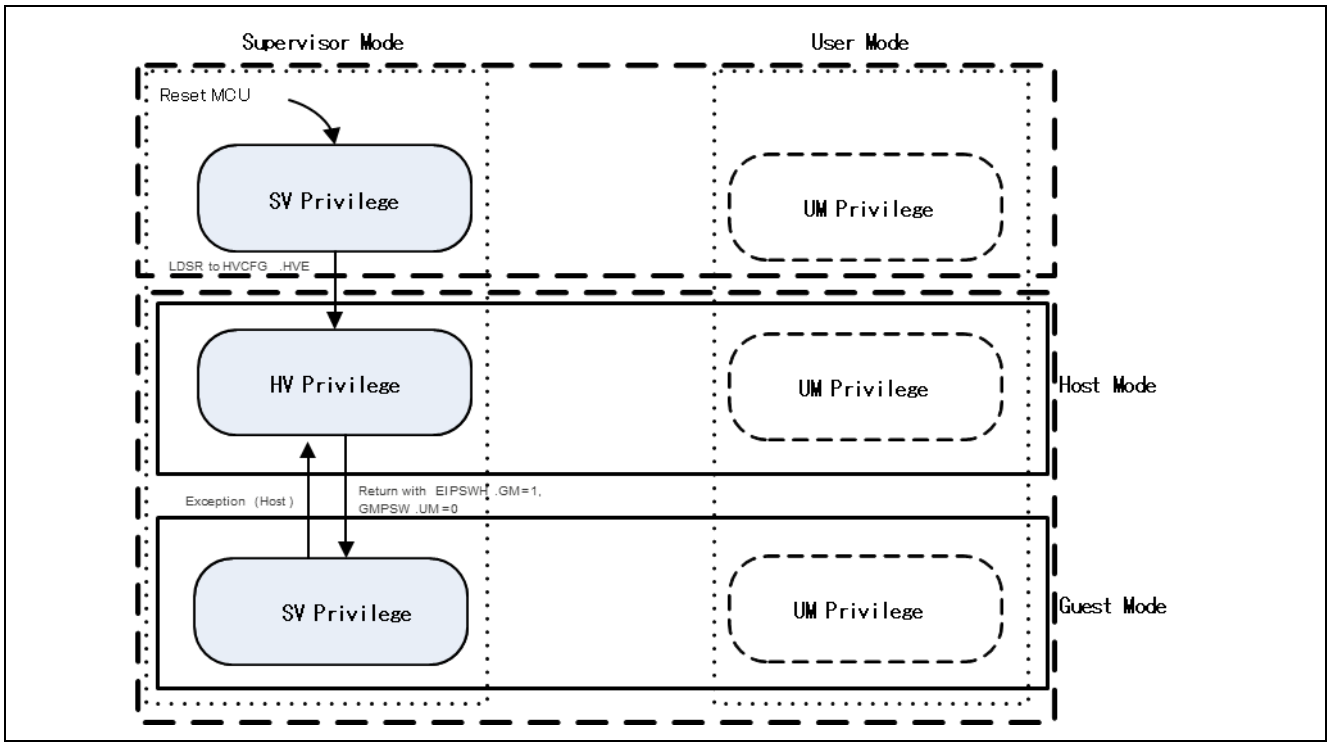

Figure 1-3 State Transition

2. Implementation of Virtualization Function

2.1 System Configuration

Figure 2-1 shows the overall configuration diagram. Figure 2-2 shows the startup and switching of VM.

- CPU：Use CPU0 and CPU1 of U2A.CPU0 starts CPU1.
- Hypervisor Software：Software of host mode operation with HV authority.
- VM Software： Two software of guest mode and operation with SV authority.
- VM Switching (Time Division control)：VM switching is performed by TPTM0 (VM0/1) interrupts, and the context is switched during the interrupt processing. Switching period is TPTM0 (VM0/1)：5s, TPTM1 (VM2/3)：3s.
- VM0/2 performs LED display. VM1 performs CAN communication. VM3 performs SPI communication.
- Use TRQ5 (VM0) and IRQ17 (VM2) as the external interrupt.

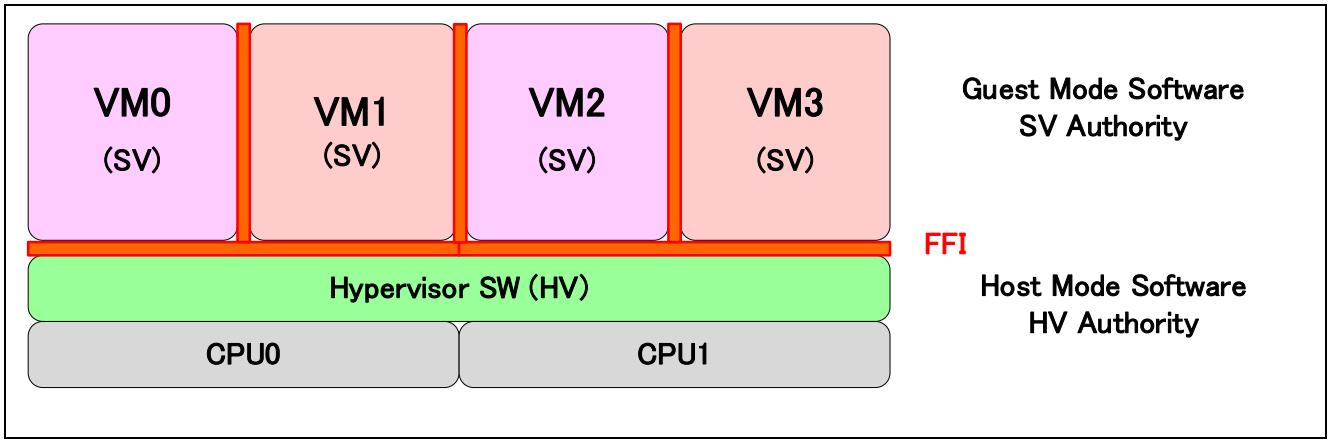

Figure 2-1 Overall Configuration Diagram

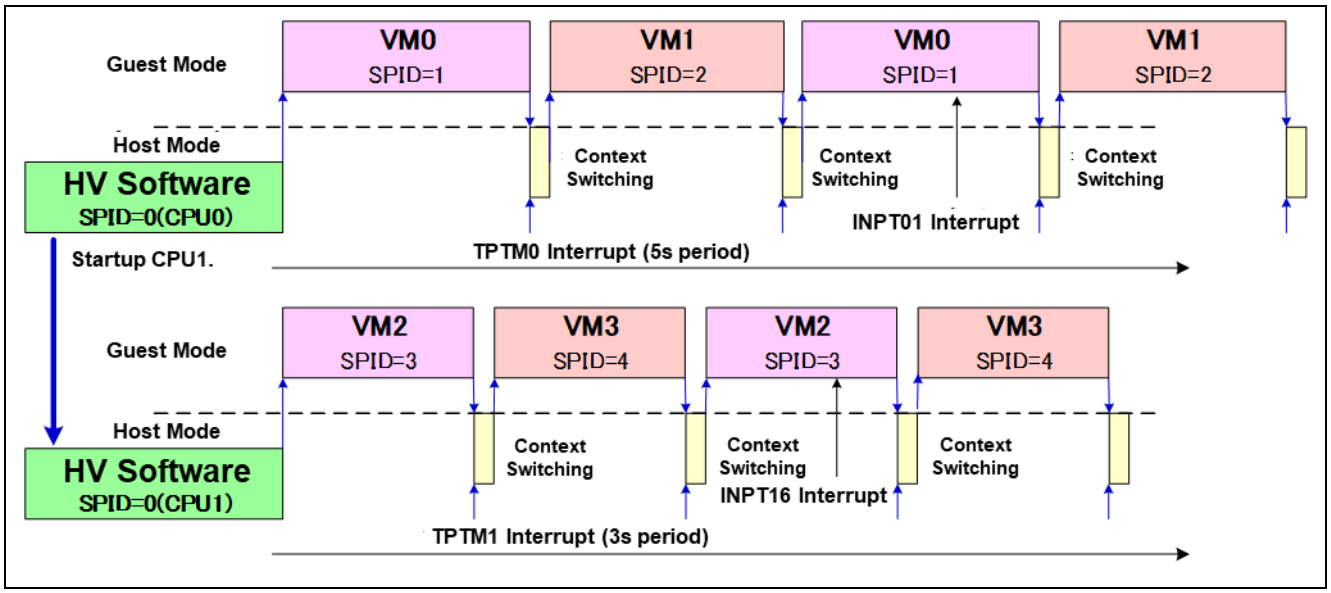

Figure 2-2 VM Startup and Switching

2.1.1 Authority Administration by SPID

The effective SPID is set to each layer by higher authority software. In this APU, the host mode is set to SPID=0, the guest mode VM0 is set to SPID=1, and VM1 is set to SPID=0.

Figure 2-3 shows the SPID setting image diagram.

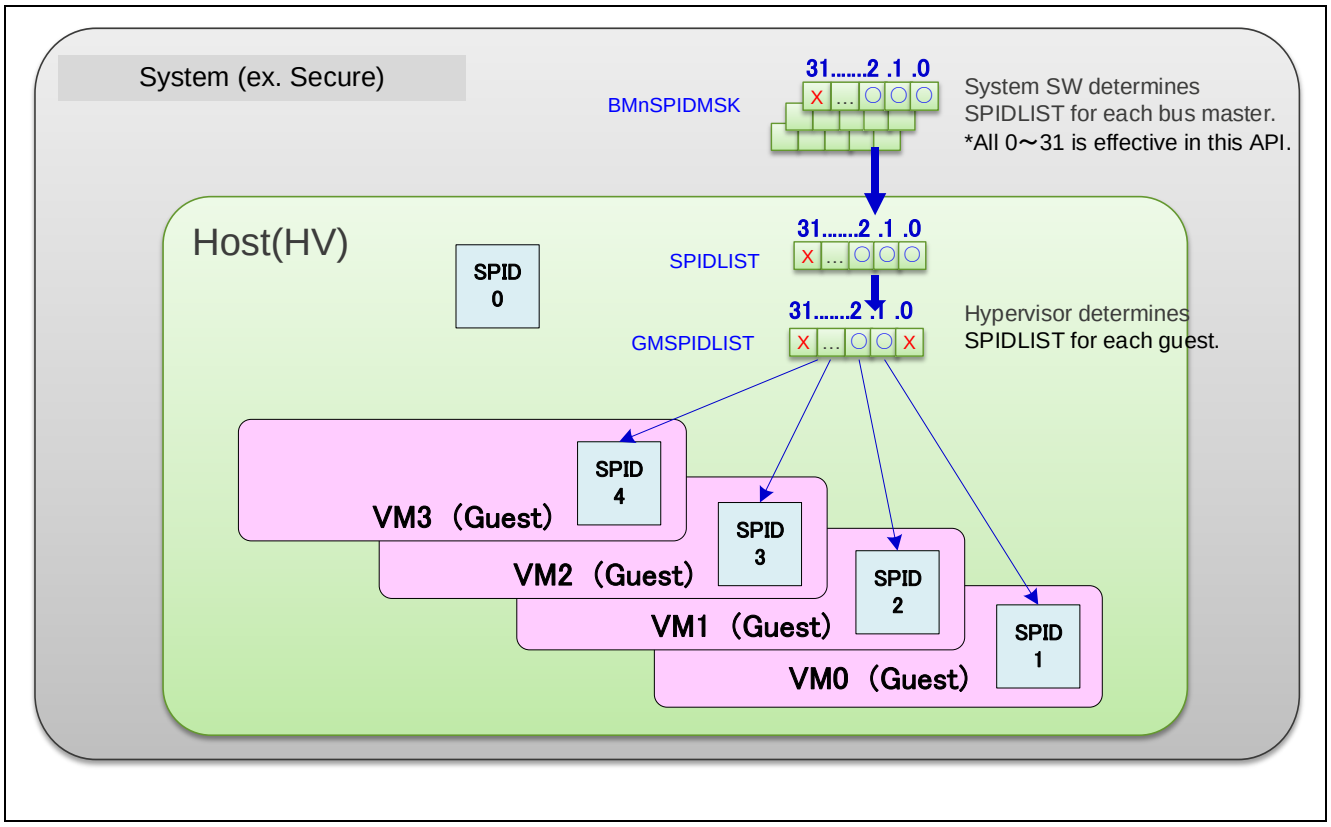

Figure 2-3 SPID Setting

2.1.2 Data Sharing between VMs

There are two ways to share data between VMs in virtualization as shown in Figure 2-4.

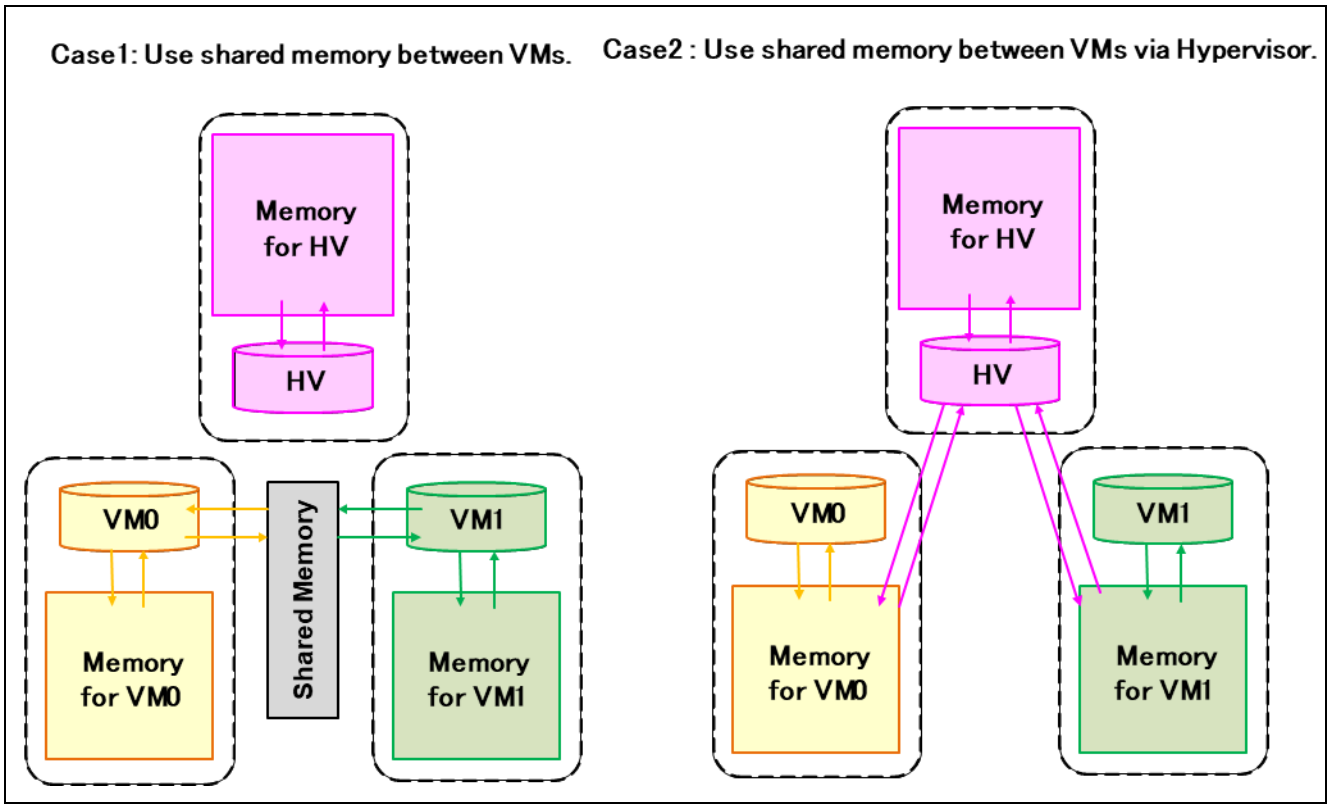

Figure 2-4 Data Sharing between VMs

Case1：Use the shared memory between VMs.

Set up shared memory that can be accessed directly from VMs and share data between VMs.

Merit：

- Low overhead.（Memory access directly from VM.）

Demerit：

- Access conflicts between VMs need to be considered.
- If the data is erroneously written, it may affect other VMs.。

Case2：Data sharing between VMs via hypervisor.

Data between VMs is performed via hypervisor without setting shared memory.。

Merit：

- Robustness is high because FFI between VMs is established.
- Easy to avoid lock holder re-emption problem.

Demerit：

- Since the data between VMs is sent via the Hypervisor, the overhead is large.

Case1 is consisted in this APN.

2.1.3 MPU Memory Management Function

The MPU function manages the usage range of HV / VM0 – VM3 by SPID. Table 2-1 shows the MPU setting example by SPID management.

Table 2-1 MPU Setting Example by SPID Management

Memory		MPU SETTING		
		HV(SPID=0)	VM0/2(SPID=1/3)	VM1/3(SPID=2/4)
		L2 MPU	L1/L2 MPU	L1/L2 MPU
Code Flash	HV	RO		
	VM0/2	RO	SPID=1/3 RO	
	VM1/3	RO		SPID=2/4 RO
Cluster RAM	HV	R/W		
	Shared (VM1/VM3)	R/W		SPID=2/4 R/W
Local RAM	HV	R/W		
	VM0/2	R/W	SPID=1/3 R/W	
	VM1/3	R/W		SPID=2/4 R/W
Number of MPU Entry		**5**	4	4

L1 MPU：MPU is managed in guest mode.
L2 MPU：MPU is managed in host mode.

（1）MPU Setting Sequence (For CPU0)

① Set the HV's own L2 MPU when the HV is started.

② HV sets L1 MPU and L2 MPU of VM before starting VM.

③ HV switches L1 MPU every time the boot VM is switched.

2.1.4 Slave Side Guard Function

SPID manages the operation authority of peripheral functions and protects them from unintended operations.

Table 2-2 Slave Side Gard by SPID

Peripheral Function	Guard Management
INTC2	SPID/MPID
sDMAC	SPID/MPID
DTS	SPID/MPID
DFP	SPID/MPID
I-Bus	SPID
H-Bus	SPID
P-Bus	SPID

Manage the following in this APN.

・INTC2 Function

・I-Bus Function：IPIR、TPTM

・P-Bus Function：IBG、Port、MSPI、RSCANFD、SPIDCTL

2.2 Interrupt Bind

Interrupt allocation can be set for each VM / HV.

Table 2-3 Interrupt Bind

Interrupt Input	CPU Function Mode	Interrupt Bind		Background Processing	Interrupt Request to CPU
		Interrupt Mode	GPID		
EIINT / FEINT	HOST	HOST			EIINT / EINT
		GUIST			None
	GUEST	HOST			EIINT / FEINT
		GEST	Match		GMEIINT / GMFEINT
			Not Match	Permitted	BGEIINT / BGFEINT
				Not Permitted	None

The guest mode interrupt during guest mode operation is determined by the GPID, and when the same GPID is used, the guest mode interrupt is executed. When the interrupts have different GPIDs and the background is enabled, the interrupts are held until the same GPID processing is performed, but if the background is not permitted, the interrupts are not executed.

2.2.1 Interrupt Management by GPID

When the guest channel interrupt occurs during VM (Guest) operation, it will operate in two ways depending on the GPID .

- The processing VM matches the interrupt GPID：The interrupts are allowed and executed.
- When the GPID of the processing VM and the interrupt do not match.：The interrupt is held and the interrupt is executed when the VM is switched to the VM with the same GPID.

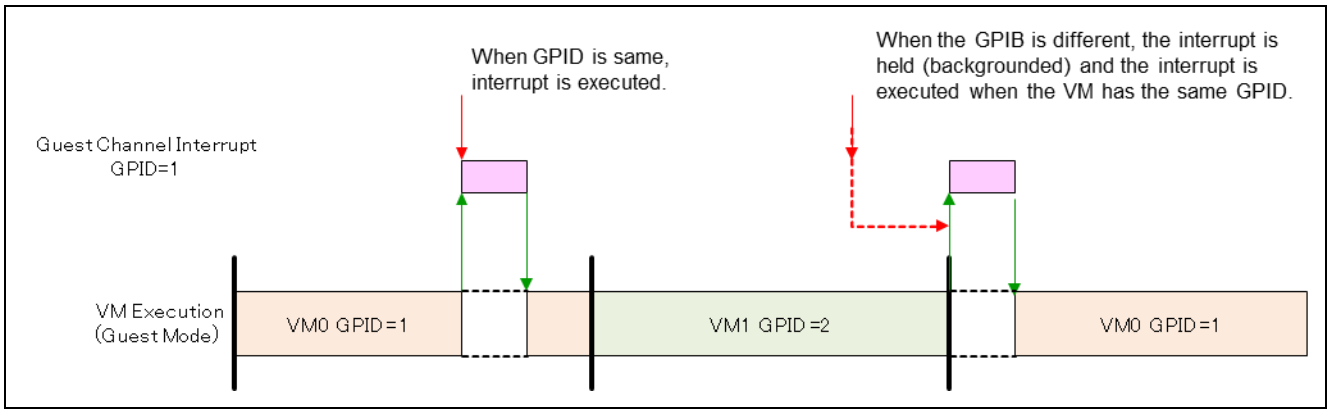

Figure 2-5 Interrupt Management by GPID

2.3 Independent Exception Interrupt Handler

As shown in Figure 2-6, it is possible to define different exception/interrupt handlers for HV (Host) and VM (guest).

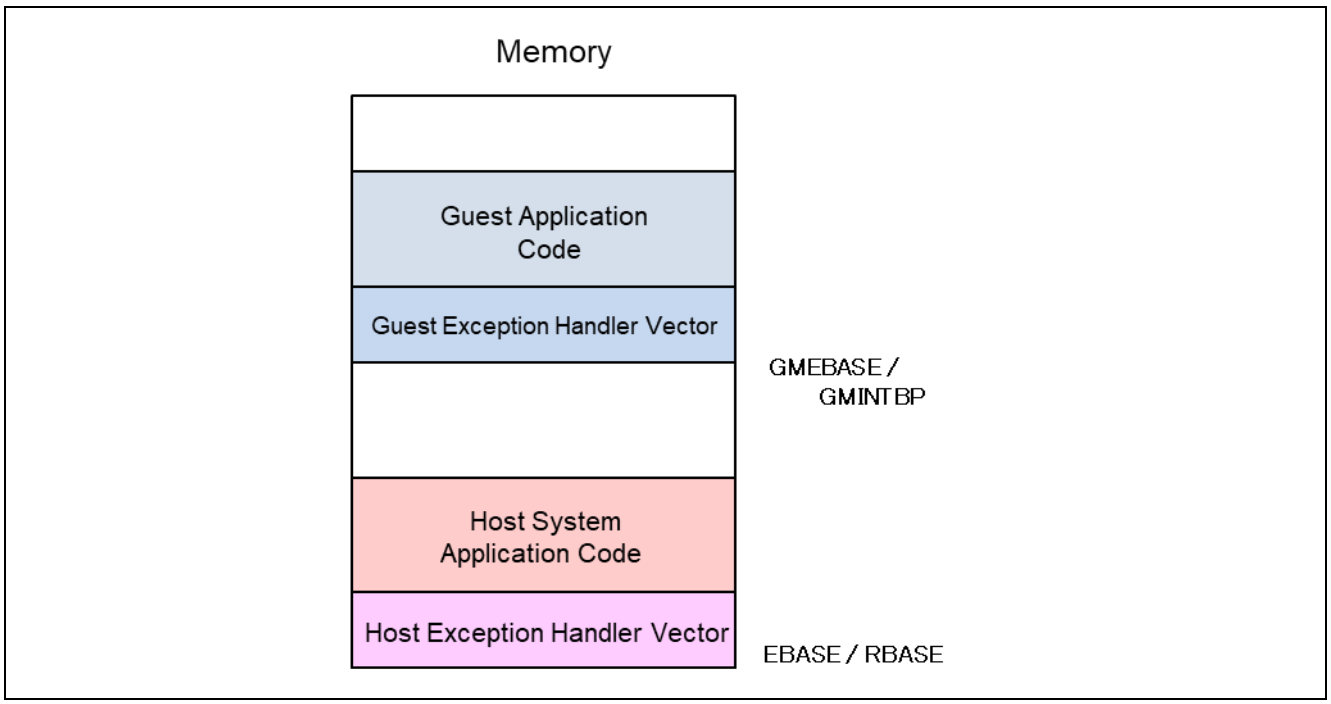

Figure 2-6 Exception Handler Definition

3. Operation Example of Virtualization Function

3.1 VM Startup and Switching Processing

3.1.1 Startup and Switching

After starting the HV, set the context for each VM and start VM0 by the EIRET instruction.

3.1.2 Save and Recovery of Context

Set up context storage memory for each VM and switch context.

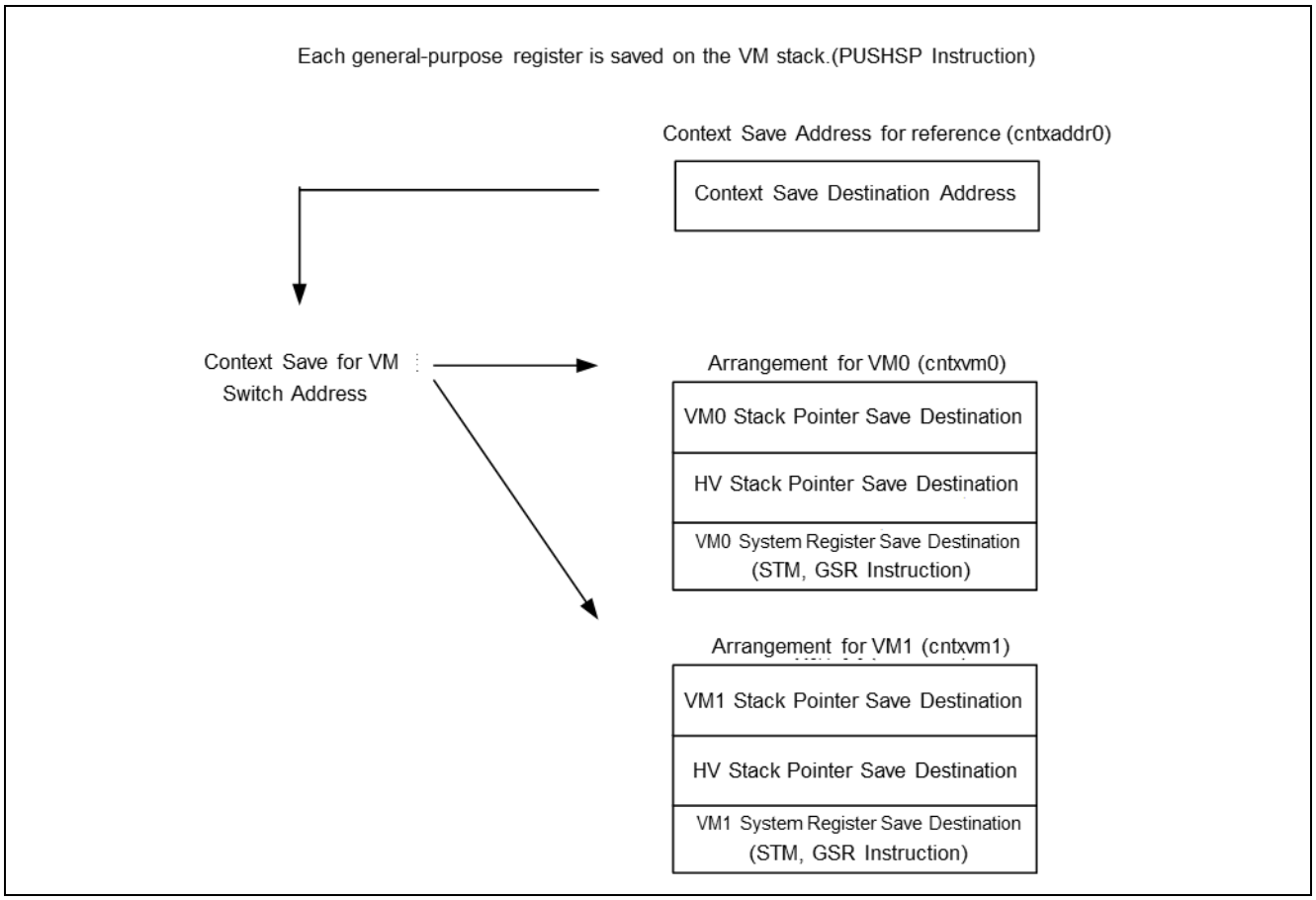

Figure 3-1 Context Management (for CPU0)

3.1.3 Memory Management by MPU

The MPU function manages the usage range of HV / VM0 to VM3 by SPID. In this APN, the memory range is allocated as shown in Figure 3-2.

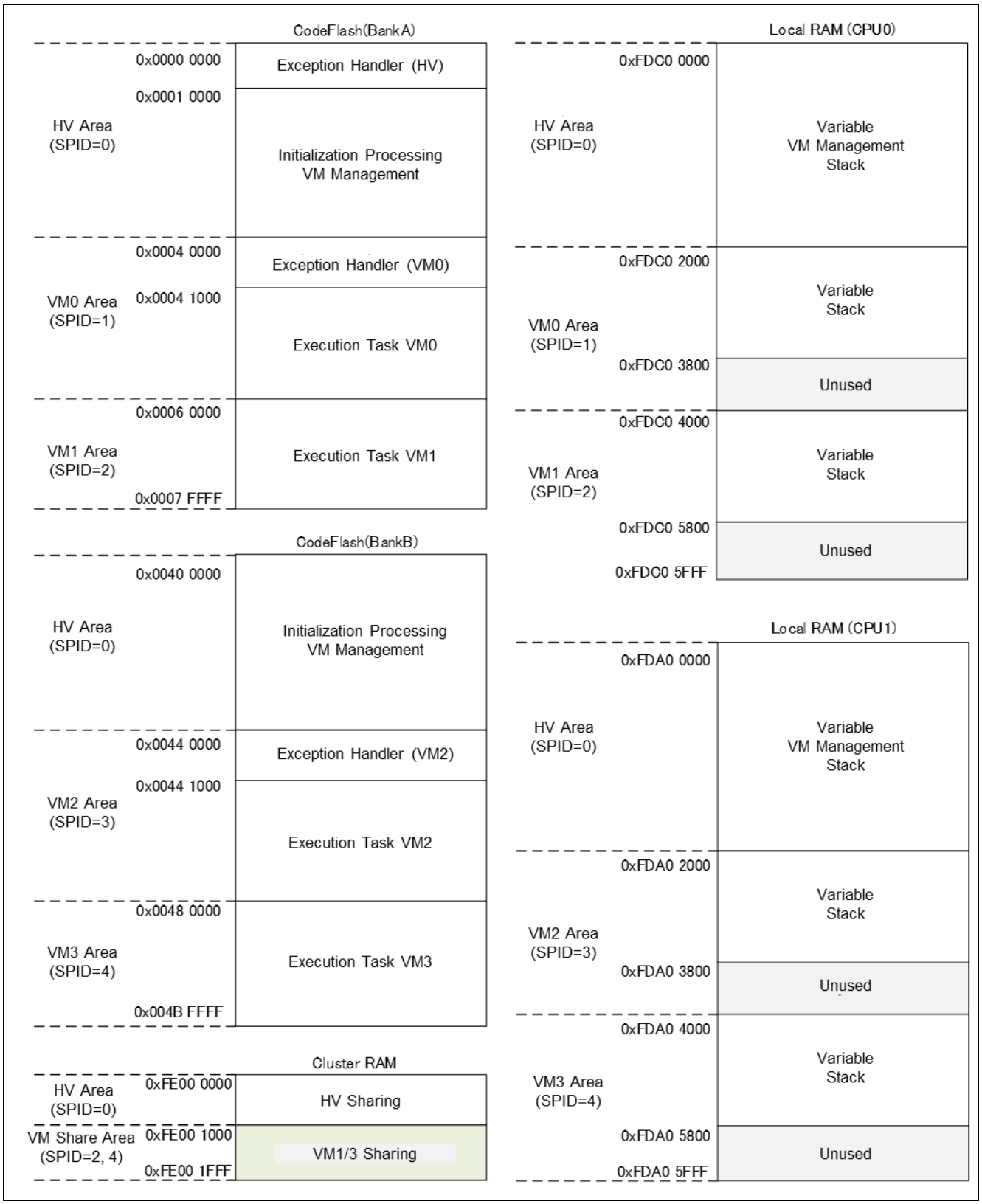

Figure 3-2 MPU Memory Management

3.1.4 MPU Entry

The number of MPU entries is used as follows.

- CPU0：6 in Host Mode, 4 in Geust Mode（Switch between two with VM0/1.）
- CPU1：7 in Hos tMode, 4 in Geust Mode（Swicth between two with VM2/3.）

Table 3-3 shows the entry configuration, and Table 3-1 and Table 3-2 show the entry allocation.

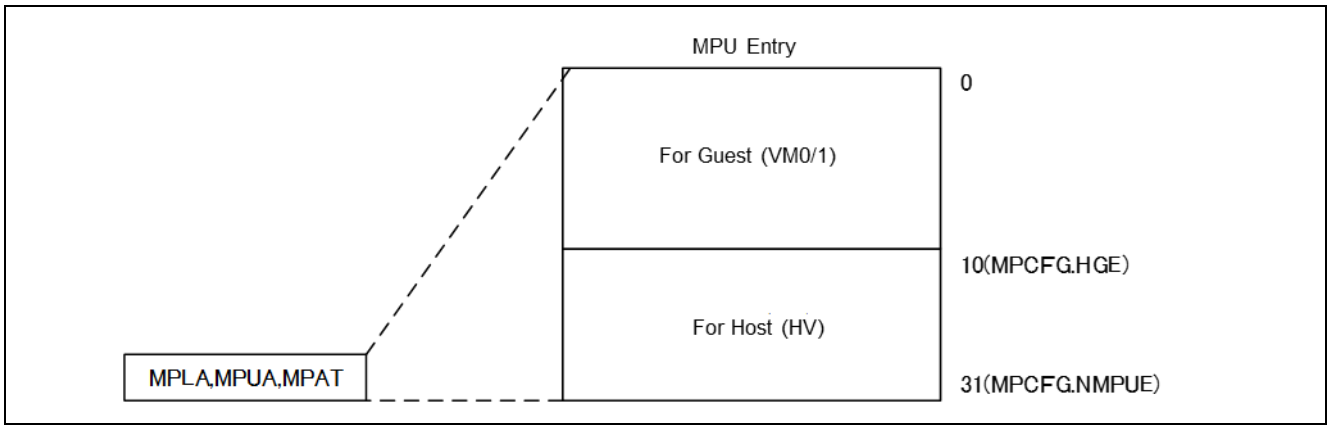

Figure 3-3 Configuration of MPU Entry

Table 3-1 Entry Allocation (CPU0)

Entry No.	Allocation	SPID	Mode
0	Internal I/O(VM0/VM1)	1、2	Guest
1	Cluster RAM(VM1/VM3)	2	
2	Code Fash(Switch between VM0 and VM1.)	1、2	
3	Local RAM(Switch between VM0 and VM1.)	1、2	
10	Code Fash(HV)	0	Host
11	Code Fash(VM0/VM1)	0、1、2	
12	Local RAM(HV)	0	
13	Local RAM(VM0/VM1)	0、1、2	
14	Internal I/O(VM0/VM1)	0、1、2	
15	Cluster RAM(HV)	0、2	

Table 3-2 Entry Allocation (CPU1)

Entry No.	Allocation Space	SPID	Mode
0	Entry I/O(VM2/VM3)	3、4	Guest
1	Cluster RAM(VM1/VM3)	4	
2	Code Fash(Switch between VM2 and VM3)	3、4	
3	Local RAM(Switch between VM2 and VM3.)	3、4	
10	Code Fash(HV)	0	Host
11	Code Fash(VM2/VM3)	0、3、4	
12	Local RAM(HV)	0	
13	Local RAM(VM2/VM3)	0、3、4	
14	Internal I/O(VM2/VM3)	0、3、4	
15	Cluster RAM(HV)	0、4	
16	Code Fash (HV)	0、3、4	

3.2 Monitor Function

Manage the VM function by PMC.

- (1) Execution Time Management

  Measure the execution Sycle of VM0 to VM3.
- (2) Event Management
- Measure the number of interrupts of TPTM0/1 for rewriting.

3.3 Task for Each VM

3.3.1 LED Display (VM0/2)

Displayed on the LED to check the operating status of VM0/2. Table 3-3 shows the LED port interrupt. Three LEDs are used for VM0/2 operation, and two LEDs are used for IRQ5/17 interrupt generation. Figure 3-4 shows the display pattern of LED.

Table 3-3 LED Port Interrupt

LED Number	Interrupt Port	Function
2	P11_0	VM0 occurrence
3	P11_1	
4	P11_2	
5	P11_3	IRQ5 occurrence
6	P22_11	IRQ17 occurrence
7	P11_5	VM1 operation
8	P11_6	
9	P11_7	

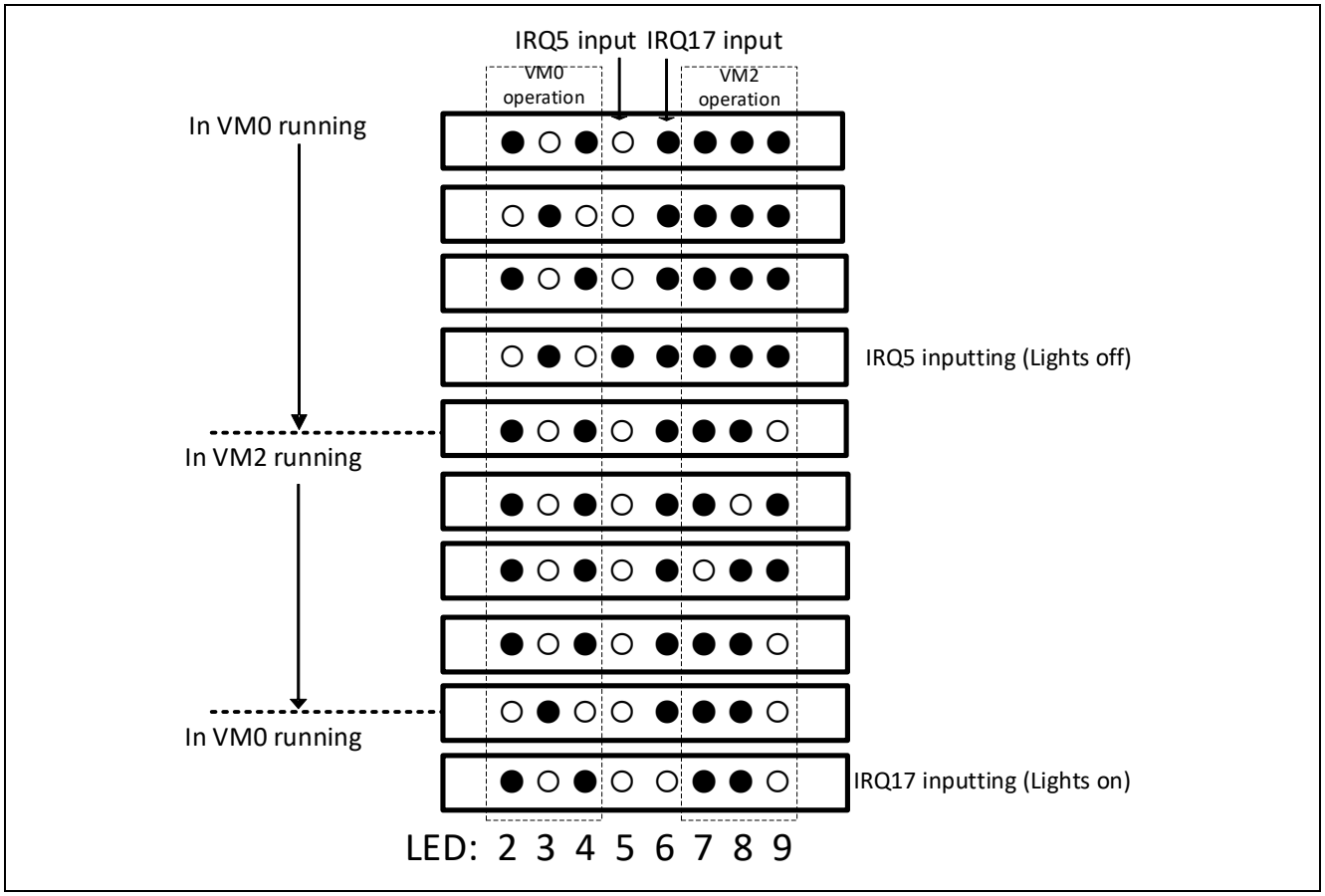

Figure 3-4 LED Display Pattern

3.3.2 CAN/SPI Communication (VM1/3)

VM1/3 performs CAN/SPI communication (internal loopback). The transmission data is allocated in the cluster RAM shared by VM1 / 3 and exclusively controlled by IVC, and VM1 becomes the ownership and sends to VM3. Figure 3-5 shows the CAN/SPI communication.

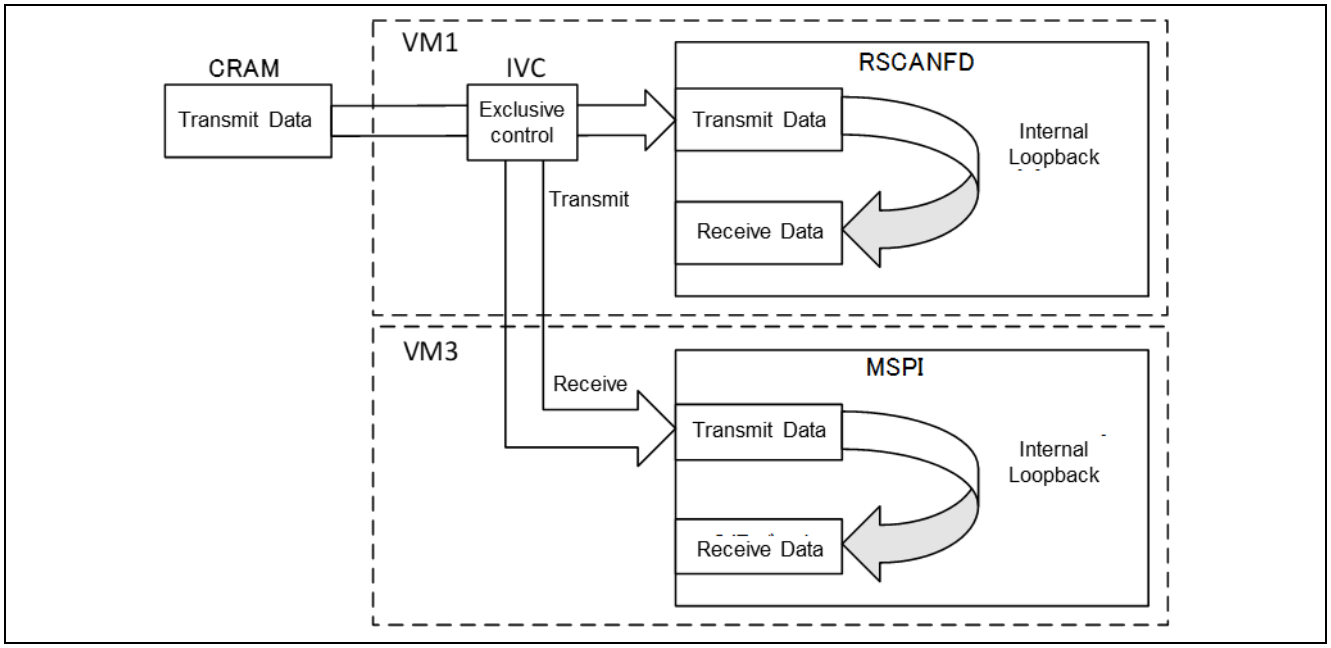

Figure 3-5 CAN/SPI Communication

4. Software Explanation

4.1 Module Explanation

Table 4-1 Module List

Module Name	Label Name	Function
Main Ruthin (CPU0)	main_pm0	Processing body, Various setting.
Main Ruthin (CPU1)	main_pm1	Processing body, Various setting.
Interrupt Function of Direct Vector Method	intprg	Register the function to be executed by the direct vector method interrupt.
Interrupt Vector Table	eiint_vecttbl	Vector table of EIINT interrupt.
Host Mode Vector Table	vecttbl	Interrupt vector table used in host mode.
Guest Mode Vector Table	vecttbl_vm	Interrupt vector table used in guest mode.

*Describe only related module with this APN.

4.2 Register Setting

（1） Virtualization Setting

Register Name	Setting Value	Function
HVCFG	0x00000001	Virtualization Function Activation (HVE=1)

（2） IVC Setting

Register Name	Setting Value	Function
CRGBAD0	0x00001000	Base address = 0x1000
CRGADV0	0x03FFC000	Valid bits
CRGSPID0	0x00000004	Ownership : SPID = 4
CRGIVCSPID0	0x00000014	Access enable : SPID = 2/4

（3） TPTM0/1 Setting

Register Name	Setting Value	Function
TPTM0IUSTR	0x00000000	IUSTR0 Clear
TPTM0IIEN	0x00000001	Interrupt Enable
TPTM0IDIV	0x00000027	clk_cpu / 40 = 10MHz 0.1us
TPTM0ILD0	50000000	Comparison Value Setting of Counter : 10MHz:5s
TPTM0IRUN	0x00000001	Counter Start
TPTM1IUSTR	0x00000000	IUSTR0 Clear
TPTM1IIEN	0x00000001	Enable Interrupt
TPTM1IDIV	0x00000027	clk_cpu / 40 = 10MHz 0.1uS
TPTM1ILD0	50000000	Comparison Value Setting of Counter : 10MHz:3s
TPTM1IRUN	0x00000001	Counter Start
TPTMSEL	0x00000001	EIINT interrupt.

（4） Monitor Function (PMC) Setting

・CPU0

Register Name	Setting Value	Function
PMCTRL0	0x01020001	Count operation in PMC0 setting HM and GPID=1.
PMCTRL1	0x01040001	Count operation in PMC1 setting HM and GPID=2.
PMCTRL2	0x01012001	Count operation in OSTM interrupt issue of PMC2 setting HM.

・CPU1

Register Name	Setting Value	Function
PMCTRL0	0x01080001	Count operation in PMC0 setting HM and GPID=3.
PMCTRL1	0x01100001	Count operation in PMC1 setting HM and GPID=4.
PMCTRL2	0x01012001	Count operation in TPTM1 interrupt issue of PMC2 setting HM.

（5） PBG Setting

Register Name	Setting Value	Function
PBG00.PBGPROT1_0	0x00000001	IBG Protection
PBG00.PBGPROT1_2	0x04000001	BOOTCTL Protection
PBG21.PBGPROT1_8	0x0000000B	port CategoryB Protection
PBG21.PBGPROT1_15	0x00000009	port CategoryF Protection
PBG40.PBGPROT1_10	0x00000010	MSPI0 Protection
PBG100.PBGPROT1_2	0x00000004	RSCFD0 Protection
PBG100.PBGPROT1_3	0x00000004	SPIDCTL Protection
PBG100.PBGPROT1_4	0x00000004	IBG Protection
PBG90.PBGPROT1_5	0x00000001	BOOTCTL Protection

（6） IBG Setting

Register Name	Setting Value	Function
IPIGPROT0_4	0x00000100	IPIR Guard Enable
IPIGPROT1_4	0x00000014	IPIR ProtectionSPID=2/4
TPTGPROT0_0	0x00000100	TPTM0 Guard Enable
TPTGPROT0_1	0x00000100	TPTM1 Guard Enable
TPTGPROT1_0	0x00000001	TPTM0 Protection
TPTGPROT1_1	0x00000001	TPTM1 Protection

（7） INTC2_GUARD Setting

Register Name	Setting Value	Function
INTC2GPROT_GR	0x00010100	EI Level Interrupt Mask Register Protection
INTC2GPROT_IMR	0x00010100	EI level Interrupt Bind Register Protection
INTC2GPROT_770	0x00010100	Extended EI level Interrupt Control Register Protection
INTC2GPROT_782	0x00010100	Extended EI level Interrupt Control Register Protection

（8） MPU Setting (CPU0)

Register Name	Setting Value	Function
MPID0	0x00000000	Enable SPID = 0 access.
MPIDX	0x0000000A	Set area 10.
MPLA	0x00000000	Lower limit address of protected area.
MPUA	0x0003FFFC	Upper limit address of protected area.
MPAT	0x010100A8	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Enable instruction execution with supervisor mode.
		Disable write with supervisor mode.
		Enable read with supervisor mode..
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPIDX	0x0000000B	Set area 11.
MPLA	0x00040000	Lower limit address of protected area.
MPUA	0x000BFFFC	Upper limit address of protected area.
MPAT	0x070700A8	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Enable instruction execution with supervisor mode.
		Disable write with supervisor mode.
		Enable read with supervisor mode..
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPIDX	0x0000000C	Set area 12.
MPLA	0xFDC00000	Lower limit address of protected area.
MPUA	0xFDC01FFC	Upper limit address of protected area.
MPAT	0x01010098	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Enable instruction execution with supervisor mode.
		Disable write with supervisor mode.
		Enable read with supervisor mode..
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPIDX	0x0000000D	Set area 13.
MPLA	0xFDC02000	Lower limit address of protected area.
MPUA	0xFDC05FFC	Upper limit address of protected area.
MPAT	0x07070098	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.

Register Name	Setting Value	Function
		Area is effective.
		Enable instruction execution with supervisor mode.
		Enable write with supervisor mode.
		Enable read with supervisor mode.
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPIDX	0x0000000E	Set area 14.
MPLA	0xFF980000	Lower limit address of protected area.
MPUA	0xFFFFFFFC	Upper limit address of protected area.
MPAT	0x07070098	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Enable instruction execution with supervisor mode.
		Enable write with supervisor mode.
		Enable read with supervisor mode..
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPIDX	0x0000000F	Set area15.
MPLA	0xFE000000	Lower limit address of protected area.
MPUA	0xFE001FFC	Upper limit address of protected area.
MPAT	0x05050098	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Enable instruction execution with supervisor mode.
		Enable write with supervisor mode.
		Enable read with supervisor mode..
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPM	0x00000003	Enable MPU function.

・VM0/1 Common

MPIDX	0x00000000	Lower limit address of protected area.
MPLA	0xFF980000	Upper limit address of protected area.
MPUA	0xFFFFFFFC	Enable SPID write access set with MPID0.
MPAT	0x06060098	Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Enable instruction execution with supervisor mode.
		Enable write with supervisor mode.
		Enable read with supervisor mode..
		Disable instruction execution with user mode.
		Disable write with user mode.

		Disable read with user mode.
		Set area 1.
MPIDX	0x00000001	Lower limit address of protected area.
MPLA	0xFE000000	Upper limit address of protected area.
MPUA	0xFE001FFC	Enable SPID write access set with MPID0.
MPAT	0x04040098	Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Enable instruction execution with supervisor mode.
		Enable write with supervisor mode.
		Enable read with supervisor mode..
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
		Enable MPU function.
GMMPM	0x00000003	Lower limit address of protected area.

・In VM0

MPIDX	0x00000002	Set area 2.
MPLA	0x00040000	Lower limit address of protected area.
MPUA	0x0007FFFC	Upper limit address of protected area.
MPAT	0x020200A8	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Enable instruction execution with supervisor mode.
		Disable write with supervisor mode.
		Enable read with supervisor mode..
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPIDX	0x00000003	Set area 3.
MPLA	0xFDC02000	Lower limit address of protected area.
MPUA	0xFDC03FFC	Upper limit address of protected area.
MPAT	0x02020098	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Enable instruction execution with supervisor mode.
		Enable write with supervisor mode.
		Enable read with supervisor mode..
		Set area 2.
		Lower limit address of protected area.
		Upper limit address of protected area.

・In VM1

MPIDX	0x00000002	Set area 2.
MPLA	0x00080000	Lower limit address of protected area.
MPUA	0x000BFFFC	Upper limit address of protected area.
MPAT	0x040400A8	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Enable instruction execution with supervisor mode.
		Disable write with supervisor mode.
		Enable read with supervisor mode..
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPIDX	0x00000003	Set area 3.
MPLA	0xFDC04000	Lower limit address of protected area.
MPUA	0xFDC05FFC	Upper limit address of protected area.
MPAT	0x04040098	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Enable instruction execution with supervisor mode.
		Enable write with supervisor mode.
		Enable read with supervisor mode..
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.

（9） MPU Setting (CPU1)

Register Name	Setting Value	Function
MPID0	0x00000000	Enable access of SPID = 0.
MPIDX	0x0000000A	Set area 10.
MPLA	0x00400000	Lower limit address of protected area.
MPUA	0x0043FFFC	Upper limit address of protected area.
MPAT	0x010100A8	Enable SPID read access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Enable instruction execution with supervisor mode.
		Disable write with supervisor mode.
		Enable read with supervisor mode.
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPIDX	0x0000000B	Set area11.
MPLA	0x00440000	Lower limit address of protected area.
MPUA	0x004BFFFC	Upper limit address of protected area.
MPAT	0x070700A8	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Enable instruction execution with supervisor mode.
		Disable write with supervisor mode.
		Enable read with supervisor mode.
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPIDX	0x0000000C	Set area 12.
MPLA	0xFDA00000	Lower limit address of protected area.
MPUA	0xFDA01FFC	Upper limit address of protected area.
MPAT	0x01010098	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Disable instruction execution with supervisor mode.
		Enable read with supervisor mode.
		Enable read with supervisor mode.
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPIDX	0x0000000D	Set area13.
MPLA	0xFDA02000	Lower limit address of protected area.
MPUA	0xFDA05FFC	Upper limit address of protected area.
MPAT	0x07070098	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.

Register Name	Setting Value	Function
		Area is effective.
		Disable instruction execution with supervisor mode.
		Enable read with supervisor mode.
		Enable read with supervisor mode.
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPIDX	0x0000000E	Set area 14.
MPLA	0xFF980000	Lower limit address of protected area.
MPUA	0xFFFFFFFC	Upper limit address of protected area.
MPAT	0x07070098	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Disable instruction execution with supervisor mode.
		Enable read with supervisor mode.
		Enable read with supervisor mode.
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPIDX	0x0000000F	Set area 15.
MPLA	0xFE000000	Lower limit address of protected area.
MPUA	0xFE001FFC	Upper limit address of protected area.
MPAT	0x05050098	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Disable instruction execution with supervisor mode.
		Enable read with supervisor mode.
		Enable read with supervisor mode.
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPIDX	0x00000010	Set area 16.
MPLA	0x00000000	Lower limit address of protected area.
MPUA	0x0000FFFC	Upper limit address of protected area.
MPAT	0x07070098	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Disable instruction execution with supervisor mode.
		Enable read with supervisor mode.
		Enable read with supervisor mode.
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPM	0x00000003	Enable MPU operation.

・VM2/3 Common

MPIDX	0x00000000	Set area 0.
MPLA	0xFF980000	Lower limit address of protected area.
MPUA	0xFFFFFFFC	Upper limit address of protected area.
MPAT	0x06060098	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Disable instruction execution with supervisor mode.
		Enable read with supervisor mode.
		Enable read with supervisor mode.
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPIDX	0x00000001	Set area 1.
MPLA	0xFE000000	Lower limit address of protected area.
MPUA	0xFE001FFC	Upper limit address of protected area.
MPAT	0x04040098	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Disable instruction execution with supervisor mode.
		Enable read with supervisor mode.
		Enable read with supervisor mode.
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
GMMPM	0x00000003	Enable MPU operation.

・In VM2

MPIDX	0x00000002	Set area 2.
MPLA	0x00440000	Lower limit address of protected area.
MPUA	0x0047FFFC	Upper limit address of protected area.
MPAT	0x020200A8	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Enable instruction execution with supervisor mode.
		Disable write with supervisor mode.
		Enable read with supervisor mode.
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPIDX	0x00000003	Set area 3.
MPLA	0xFDA02000	Lower limit address of protected area.
MPUA	0xFDA03FFC	Upper limit address of protected area.
MPAT	0x02020098	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.

		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Disable instruction execution with supervisor mode.
		Enable read with supervisor mode.
		Enable read with supervisor mode.
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.

・In VM3

MPIDX	0x00000002	Set area 2.
MPLA	0x00480000	Lower limit address of protected area.
MPUA	0x004BFFFC	Upper limit address of protected area.
MPAT	0x040400A8	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Enable instruction execution with supervisor mode.
		Disable write with supervisor mode.
		Enable read with supervisor mode.
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.
MPIDX	0x00000003	Set area 3.
MPLA	0xFDA04000	Lower limit address of protected area.
MPUA	0xFDA05FFC	Upper limit address of protected area.
MPAT	0x04040098	Enable SPID write access set with MPID0.
		Enable SPID read access set with MPID0.
		Disable write with any SPID.
		Disable execution and read with any SPID.
		Area is effective.
		Disable instruction execution with supervisor mode.
		Enable read with supervisor mode.
		Enable read with supervisor mode.
		Disable instruction execution with user mode.
		Disable write with user mode.
		Disable read with user mode.

（１０）INTC1/2 Setting

Register Name	Setting Value	Function
PE0_EIBD31	0x00000000	Host、GPID=0、PEID=0
PE0_EIC31	0x000C	EI level 12
PE1_EIBD31	0x00000001	Host、GPID=0、PEID=1
PE1_EIC31	0x000D	EI level 13
EIBD770	0x00008100	Guest,、GPID=1、PEID=0
EIC770	0x000E	EI level 14
EEIC770	0x0000000E	Extended EI level 14
EIBD782	0x00008301	Guest,、GPID=3、PEID=1

EIC782	0x000F	EI level 15
EEIC782	0x0000000F	Extended EI level 15

4.3 Function Configuration

4.3.1 Main() Function

Table 4-2 “main_pe0/1() Function”

Function Name	Overview
main_pe0()	Wait for interrupt after Initialize Function (TPTM0, INTC2, MPU, PBG PORT), CPU1Startup, System Register Initialize Setting (Host/Geust), Variable Initialization, and Interrupt Enable.
main_pe1()	Wait for interrupt after Initialize Function (TPTM1、MPU), System Register Initialize Setting (Host/Geust), Variable Initialization, and Interrupt Enable.

4.3.2 TPTM0/1_init () Function

Table 4-3 ”TPTM0/1_init () Function”

Function Name	Overview
TPTM0/1_init ()	TPTM0/1 Initialization (Interval Timer Mode, Interrupt Enable, Period)

4.3.3 PBG_Init_hv () Function

Table 4-4 ”PBG_Init_hv () Function”

Function Name	Overview
PBG_Init_hv ()	PBG Initialization (IBG, port CategoryB/F, MSPI0, RSCFD0 ,SPIDCTL Access Enable)

4.3.4 IBG_Init_hv () Function

Table 4-5 ”IBG_Init_hv () Function”

Function Name	Overview
IBG_Init_hv ()	IBG Initialization (IPIR , TPTM Access Enable)

4.3.5 INTC2_GUARD() Function

Table 4-6 ”INTC2_GUARD () Function”

Function Name	Overview
INTC2_GUARD ()	INTC2 Initialization (INTC2 Access Disable)

4.3.6 MPU_Init_hv () Function

Table 4-7 ”MPU_Init_hv () Function”

Function Name	Overview
MPU_Init_hv ()	MPU Initialization (HV Area, VM0/1 Common Area Protection)

4.3.7 MPU_Init_ vm0 () Function

Table 4-8 ”MPU_Init_ vm0 () Function”

Function Name	Overview
MPU_Init_ vm0 ()	MPU Initialization (VM0 Unique Area Protection)

4.3.8 MPU_Init_ vm1 () Function

Table 4-9 ”MPU_Init_ vm1 () Function”

Function Name	Overview
MPU_Init_ vm1 ()	MPU Initialization (VM1 Unique Area Protection)

4.3.9 MPU_Init_hv2 () Function

Table 4-10 ”MPU_Init_hv2 () Function”

Function Name	Overview
MPU_Init_hv2 ()	MPU Initialization (HV Area, VM2/3 Common Area Protection)

4.3.10 MPU_Init_ vm2 () Function

Table 4-11 ”MPU_Init_ vm2 () Function”

Function Name	Overview
MPU_Init_ vm2 ()	MPU Initialization (VM2 Unique Area Protection)

4.3.11 MPU_Init_ vm3 () Function

Table 4-12 ”MPU_Init_ vm3 () Function”

Function Name	Overview
MPU_Init_ vm3 ()	MPU Initialization (VM3 Unique Area Protection)

4.3.12 port_Init () Function

Table 4-13 ”port_Init() Function”

Function Name	Overview
port_Init()	PORT Initialization (External Interrupt INTP01/16, PORT11)

4.3.13 can () Function

Table 4-14 ”can() Function”

Function Name	Overview
can()	Exclusive Control of Transmit Data by IVC, Interrupt Request to CPU1, CAN Transmit/Receive

4.3.14 R_CAN_Init () Function

Table 4-15 ”R_CAN_Init () Function”

Function Name	Overview
R_CAN_Init ()	RSCANFD Initialization (Mode Switching, Baudrate Setting, Loopback Enable)

4.3.15 R_CAN_Global_TestStart () Function

Table 4-16 ”R_CAN_Global_TestStart () Function”

Function Name	Overview
R_CAN_Global_TestStart ()	CAN Loopback Transmit/Receive, Transmit/Receive Data Comparison

4.3.16 mspi_Init () Function

Table 4-17 ”mspi_Init() Function”

Function Name	Overview
mspi_Init()	MSPI Initialization (Muster Mode, Loopback, MSPI Enable), Transmit/Receive Enable, Priority：8, MSB First, None-Parity, 32 bit Length, 10MHz

4.3.17 mspi0_ch0_activate () Function

Table 4-18 ”mspi0_ch0_activate () Function”

Function Name	Overview
mspi0_ch0_activate ()	Channel Enable, Soft Trigger

4.3.18 mspi_main () Function

Table 4-19 ”mspi_main () Function”

Function Name	Overview
mspi_main ()	Loopback Transmit/Receive, Transmit/Receive Data Comparison

4.3.19 mspi0_ch0_communicate () Function

Table 4-20 ”mspi0_ch0_communicate () Function”

Function Name	Overview
mspi0_ch0_communicate ()	Loopback transmit/Receive

Figure 4-1 to Figure 4-12 show the flow chart.

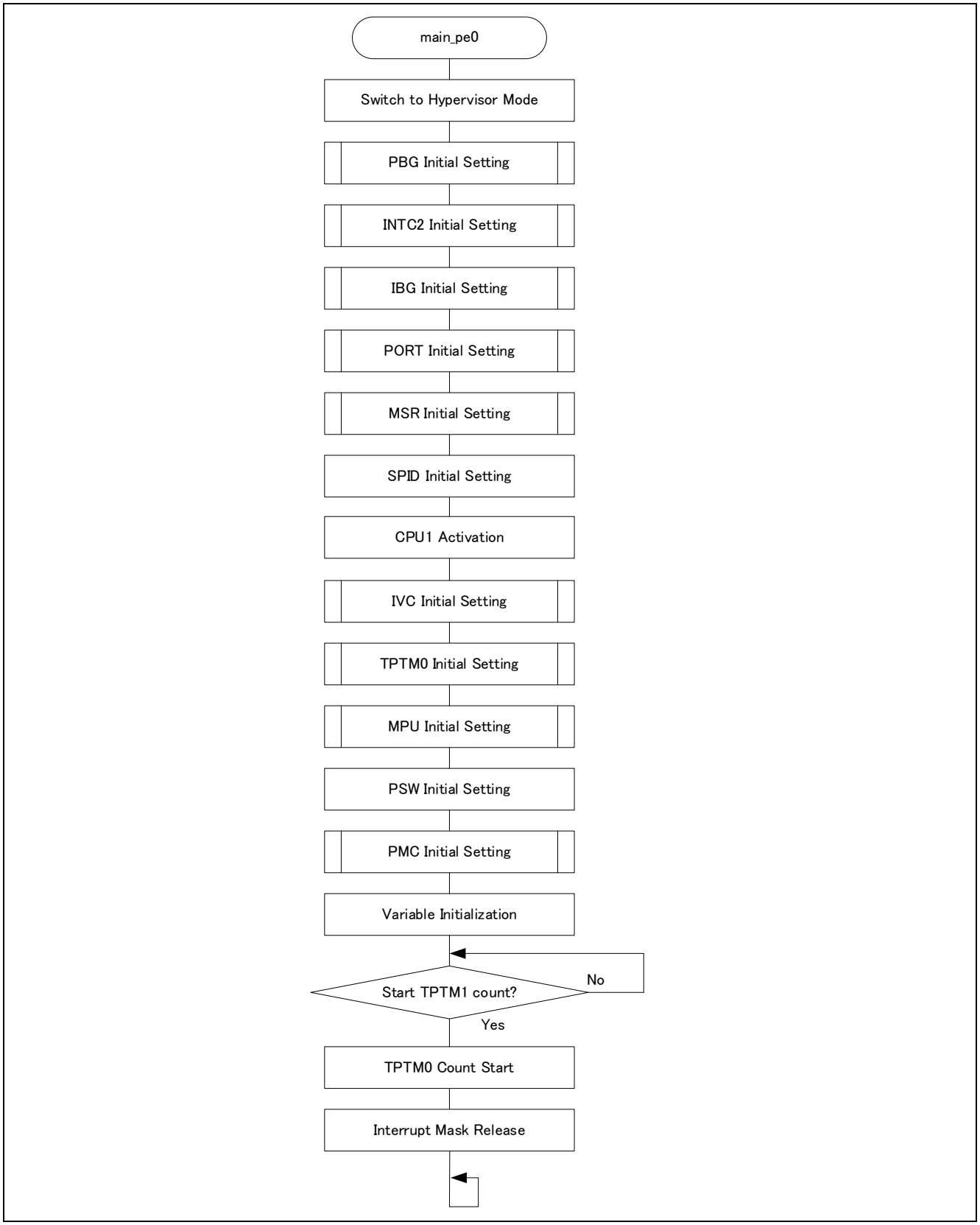

Figure 4-1 Main (CPU0)

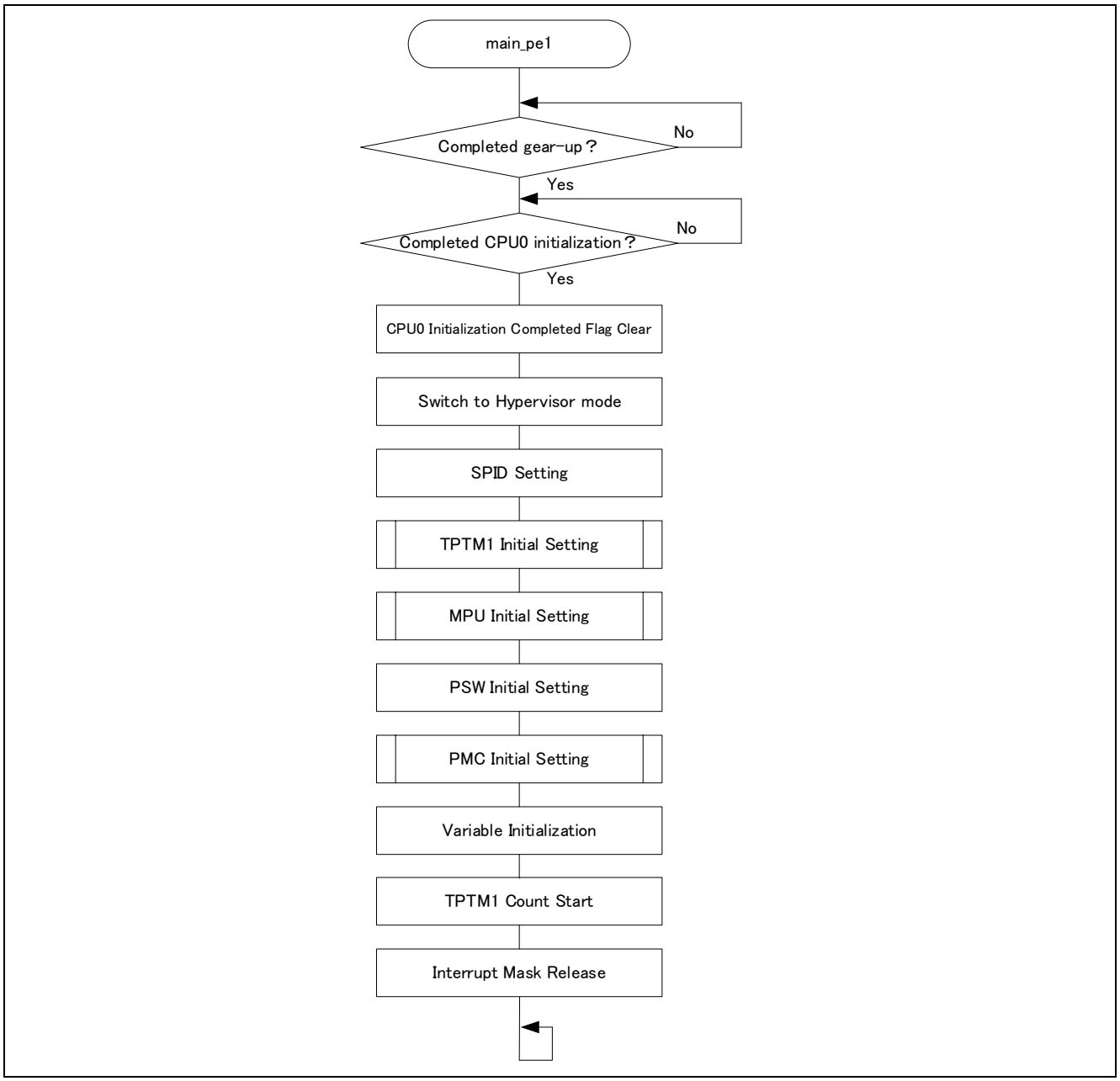

Figure 4-2 Main (CPU1)

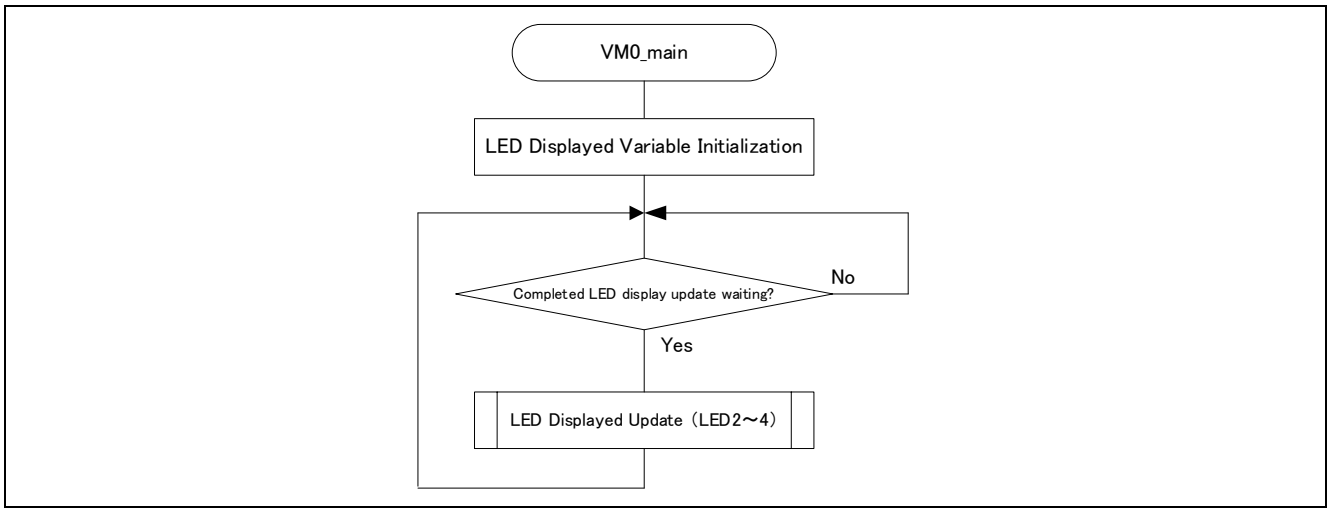

Figure 4-3 VM0 Main

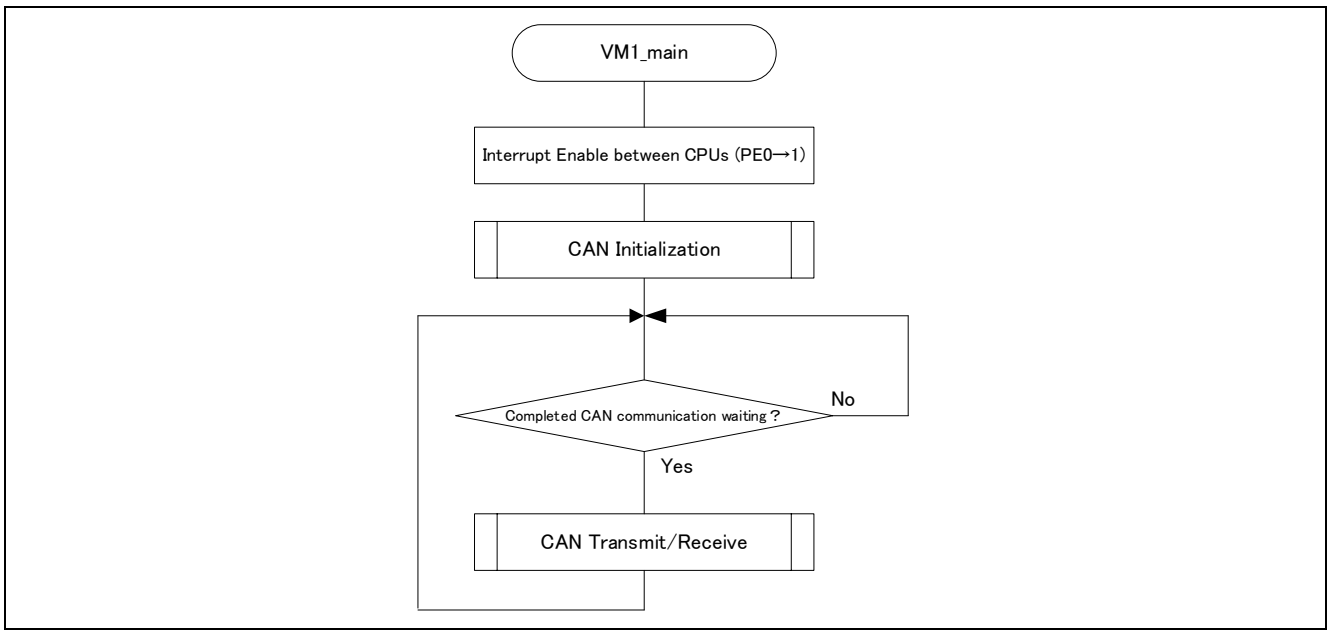

Figure 4-4 VM1 Main

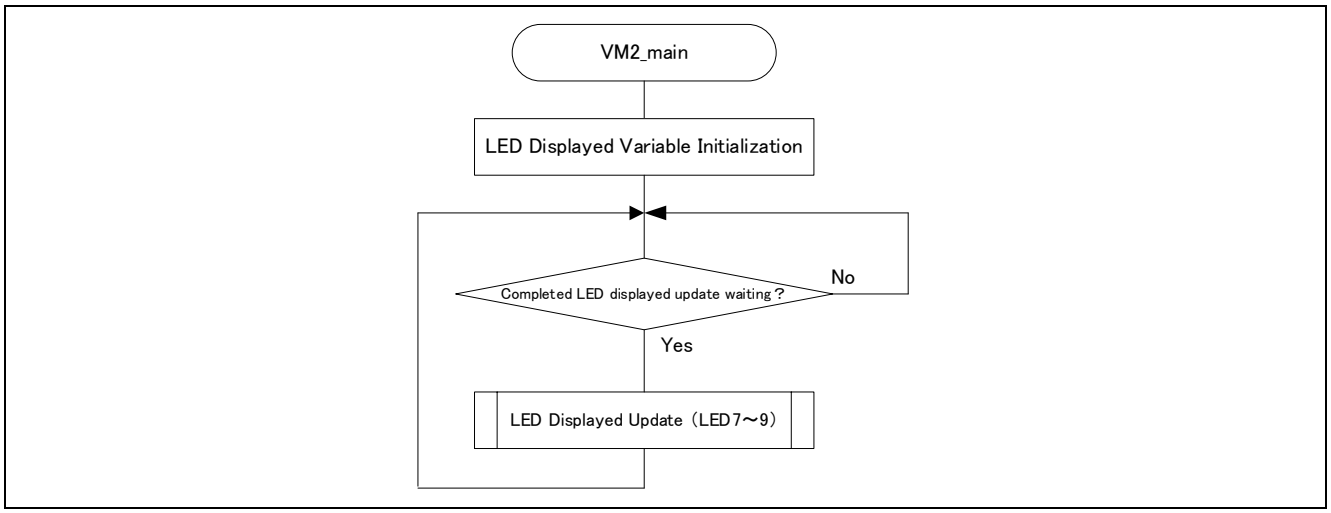

Figure 4-5 VM2 Main

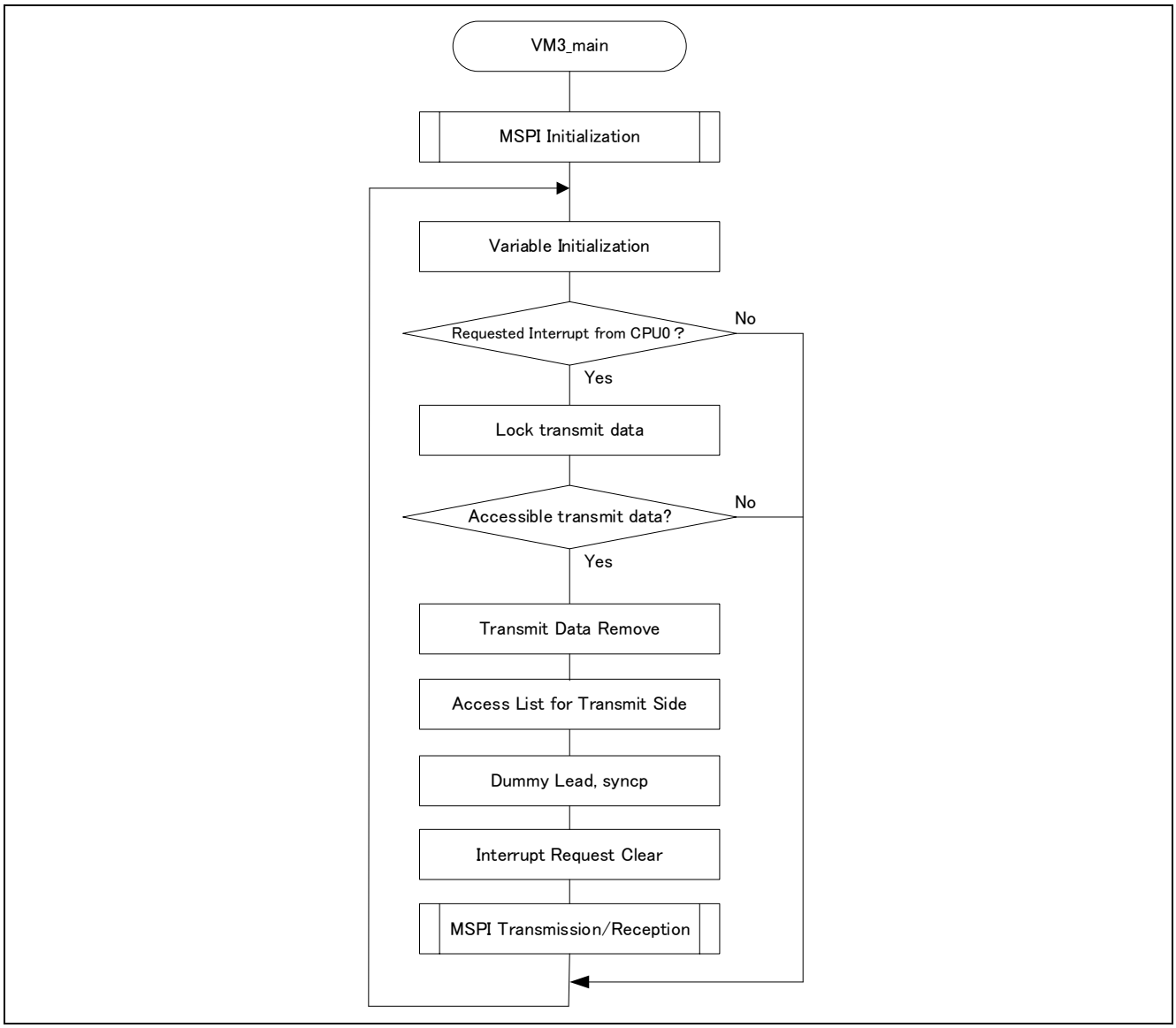

Figure 4-6 VM3 Main

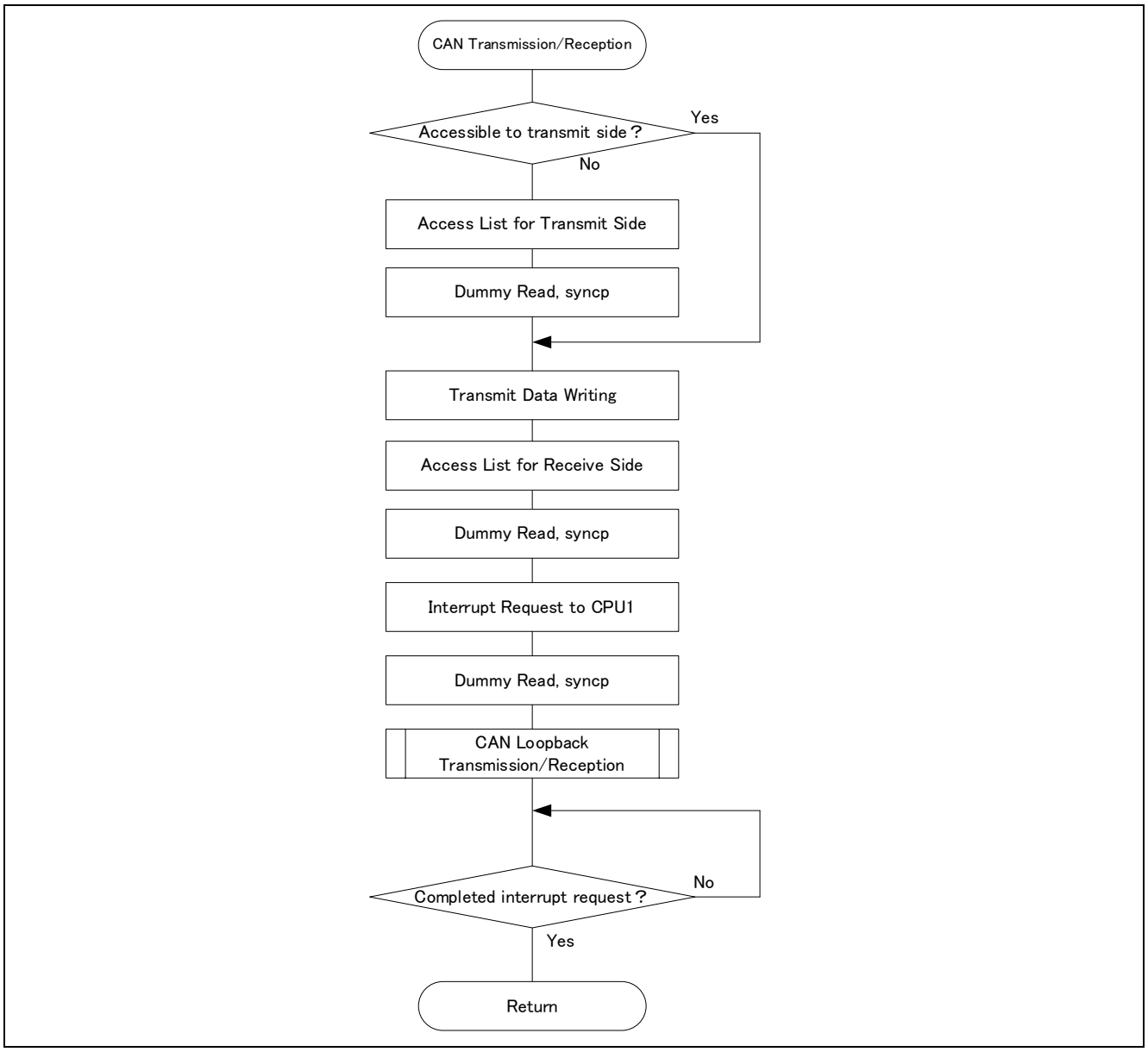

Figure 4-7 CAN Transmission/Reception

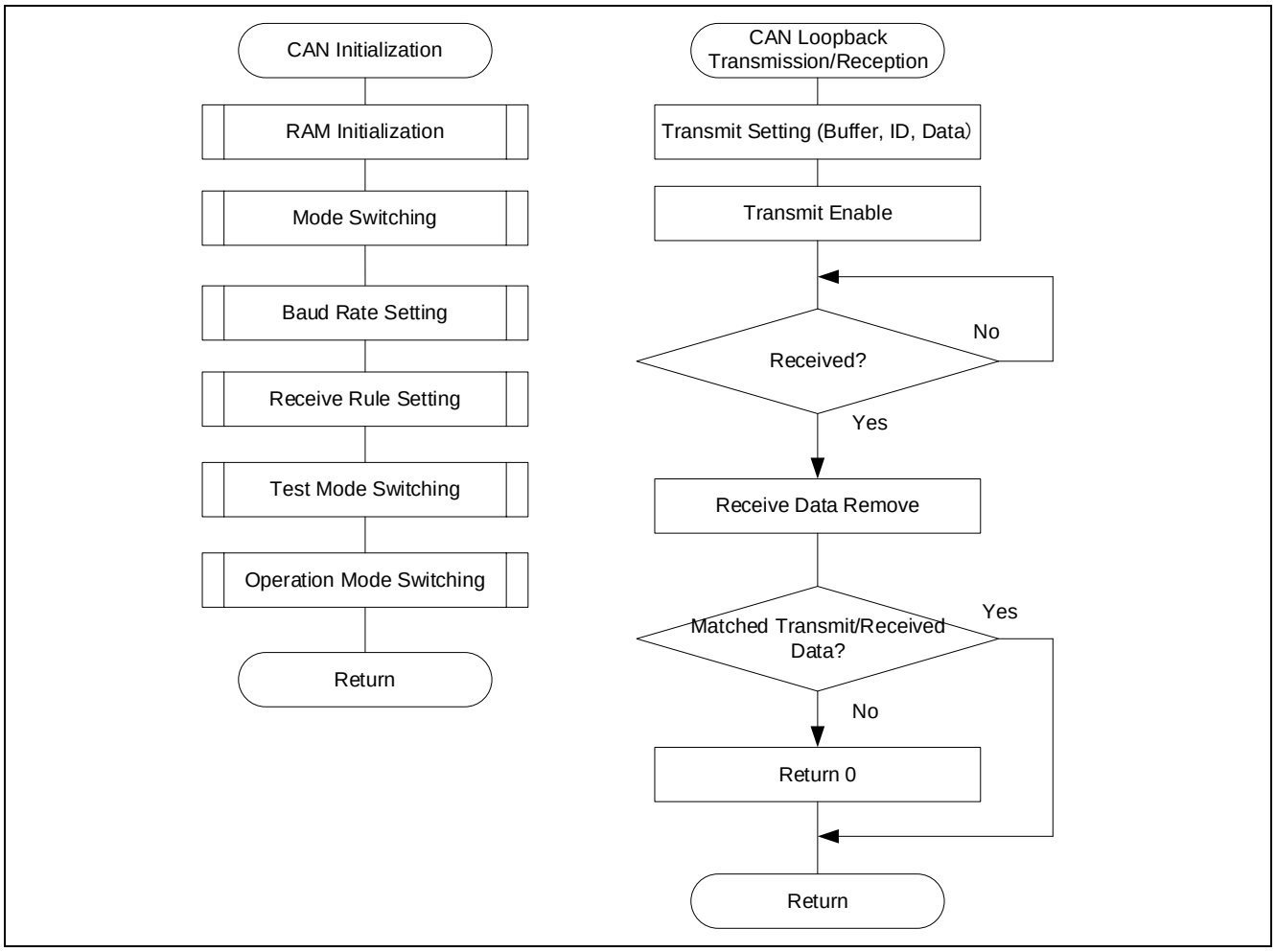

Figure 4-8 CAN Initialization/Loopback Transmission/Reception

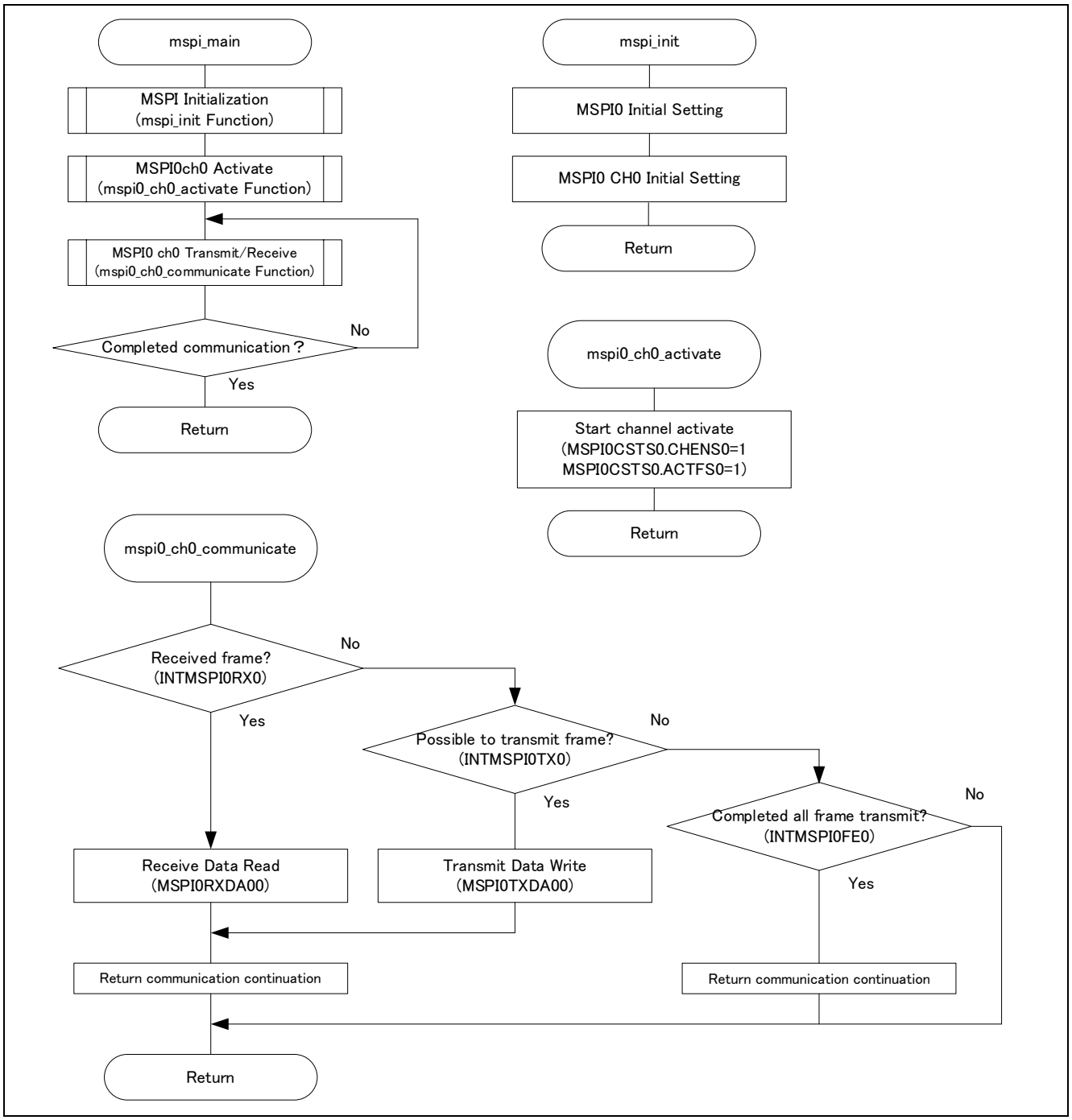

Figure 4-9 MSPI Transmission/Reception

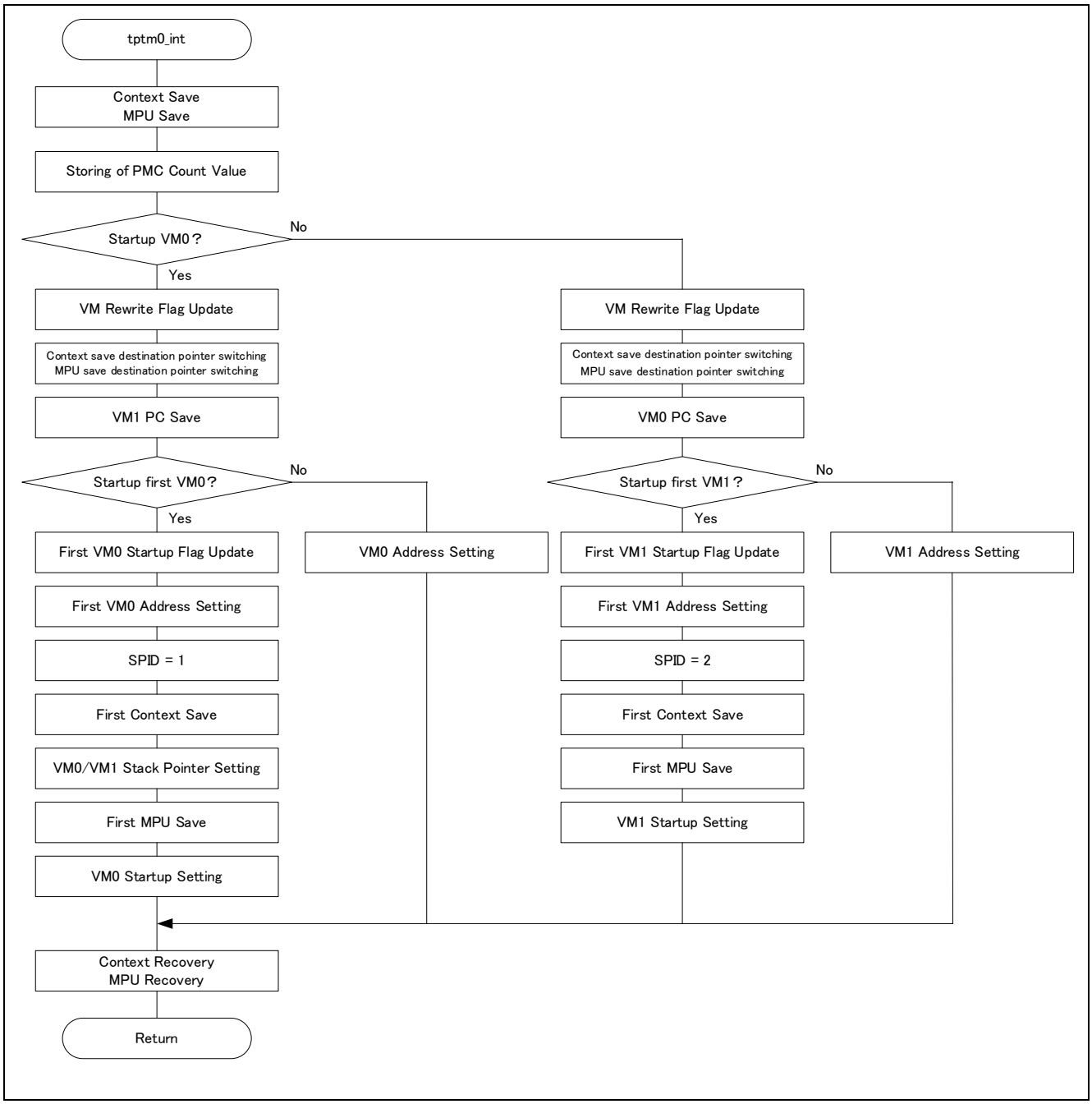

Figure 4-10 TPTM0 Interrupt (HV : CPU0)

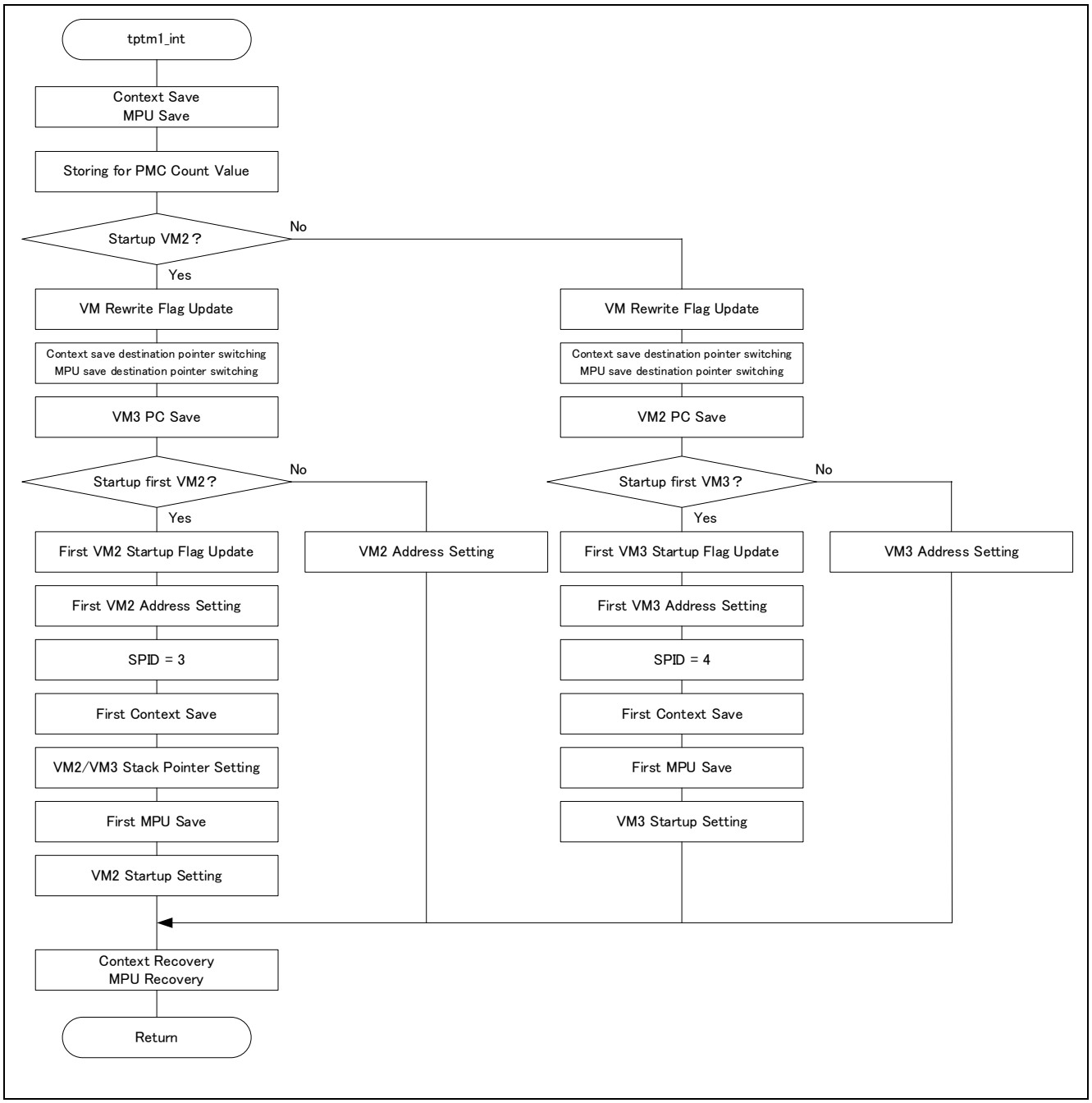

Figure 4-11 TPTM1 Interrupt (HV : CPU1)

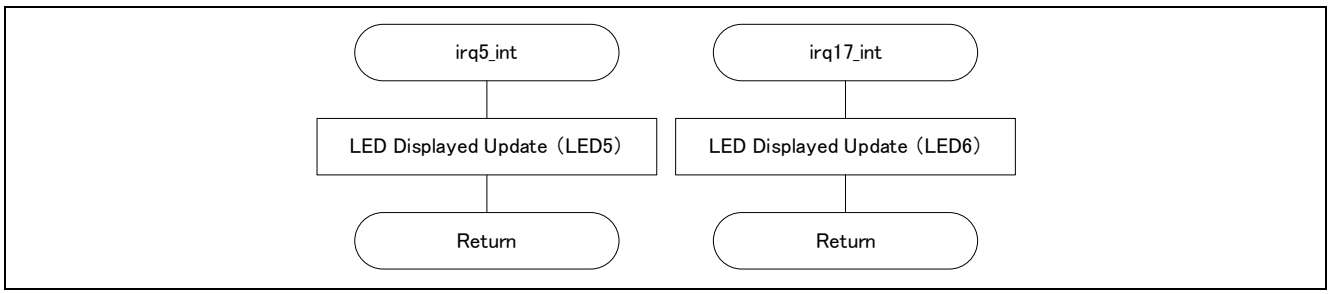

Figure 4-12 IRQ5/17 Interrupt (VM0/2)

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	2023.12.15	—	Initial edition

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.

2. Processing at power-on

   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.

3. Input of signal during power-off state

   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.

4. Handling of unused pins

   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.

5. Clock signals

   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.

6. Voltage application waveform at input pin

   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).

7. Prohibition of access to reserved addresses

   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.

8. Differences between products

   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

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

Trademarks

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

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/.