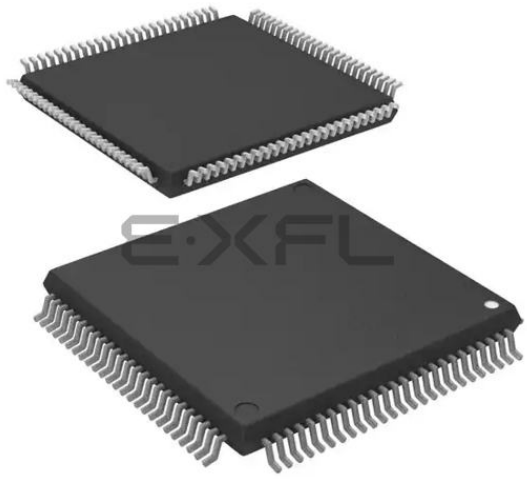
Renesas Electronics America Inc - UPD70F3770GC-UEU-AX Datasheet



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	V850ES
Core Size	32-Bit Single-Core
Speed	48MHz
Connectivity	CANbus, CSI, EBI/EMI, I²C, UART/USART, USB
Peripherals	DMA, LVD, PWM, WDT
Number of I/O	77
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	40K x 8
Voltage - Supply (Vcc/Vdd)	2.85V ~ 3.6V
Data Converters	A/D 12x10b; D/A 2x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/upd70f3770gc-ueu-ax

(2/2)

Peripheral Function	Register Name	Access	k
I^2C00 to I^2C02	IICS0 to IICS2	Read	1
CRC	CRCD	Write	1
CAN controller (m = 0 to 31, a = 1 to 4)	C0GMABT, C0GMABTD, C0MASKaL, C0MASKaH, C0LEC, C0INFO, C0ERC, C0IE, C0INTS, C0BRP, C0BTR, C0TS	Read/Write	$f_{XX}/f_{CANMOD} + 1) / (2 + j)$ (MIN.)[Note] $(2 \times f_{XX}/f_{CANMOD} + 1) / (2 + j)$ (MAX.)[Note]
	C0GMCTRL, C0GMCS, C0CTRL	Read/Write	$(f_{XX}/f_{CAN} + 1) / (2 + j)$ (MIN.)[Note] $(2 \times f_{XX}/f_{CAN} + 1) / (2 + j)$ (MAX.)[Note]
	C0RGPT, C0TGPT	Write	$(f_{XX}/f_{CANMOD} + 1) / (2 + j)$ (MIN.)[Note] $(2 \times f_{XX}/f_{CANMOD} + 1) / (2 + j)$ (MAX.)[Note]
		Read	$(3 \times f_{XX}/f_{CANMOD} + 1) / (2 + j)$ (MIN.)[Note] $(4 \times f_{XX}/f_{CANMOD} + 1) / (2 + j)$ (MAX.)[Note]
	C0LIPT, C0LOPT	Read	$(3 \times f_{XX}/f_{CANMOD} + 1) / (2 + j)$ (MIN.)[Note] $(4 \times f_{XX}/f_{CANMOD} + 1) / (2 + j)$ (MAX.)[Note]
	C0MCTRLm	Write	$(4 \times f_{XX}/f_{CAN} + 1) / (2 + j)$ (MIN.)[Note] $(5 \times f_{XX}/f_{CAN} + 1) / (2 + j)$ (MAX.)[Note]
		Read	$(3 \times f_{XX}/f_{CAN} + 1) / (2 + j)$ (MIN.)[Note] $(4 \times f_{XX}/f_{CAN} + 1) / (2 + j)$ (MAX.)[Note]
	C0MDATA01m, C0MDATA0m, C0MDATA1m, C0MDATA23m, C0MDATA2m, C0MDATA3m, C0MDATA45m, C0MDATA4m, C0MDATA5m, C0MDATA67m, C0MDATA6m, C0MDATA7m, C0MDLCm, C0MCONFm, C0MIDLm, C0MIDHm	Write (8 bits)	$(4 \times f_{XX}/f_{CANMOD} + 1) / (2 + j)$ (MIN.)[Note] $(5 \times f_{XX}/f_{CANMOD} + 1) / (2 + j)$ (MAX.)[Note]
		Write (16 bits)	$(2 \times f_{XX}/f_{CANMOD} + 1) / (2 + j)$ (MIN.)[Note] $(3 \times f_{XX}/f_{CANMOD} + 1) / (2 + j)$ (MAX.)[Note]
		Read (8/16 bits)	$(3 \times f_{XX}/f_{CANMOD} + 1) / (2 + j)$ (MIN.)[Note] $(4 \times f_{XX}/f_{CANMOD} + 1) / (2 + j)$ (MAX.)[Note]

Number of clocks necessary for access = $3 + i + j + (2 + j) \times k$

Note Digits below the decimal point are rounded up.

Caution Accessing the above registers is prohibited in the following statuses. If a wait cycle is generated, it can only be cleared by a reset.

- **When the CPU operates with the subclock and the main clock oscillation is stopped**
- **When the CPU operates with the internal oscillation clock**

Remark i: Values (0) of higher 4 bits of VSWC register
j: Values (0 or 1) of lower 4 bits of VSWC register

PFCE34	PFC34	Specification of P34 pin alternate function
0	0	TIAA10 input
0	1	TOAA10 output
1	0	TOAA1OFF input/INTP09 input[Note]
1	1	Setting prohibited

Note TOAA1OFF and INTP09 are alternate functions. When using the pin as the TOAA1OFF pin, disable INTP09 pin edge detection, which is the alternate function. Also, when using the pin as the INTP09 pin, stop the high-impedance output controller.

PFCE33	PFC33	Specification of P33 pin alternate function
0	0	TIAA01 iput
0	1	TOAA01 output
1	0	RTCDIV output
1	1	RTCCL output

PFCE32	PFC32	Specification of P32 pin alternate function
0	0	ASCKC0 input
0	1	$\overline{\text{SCKF4}}$ I/O
1	0	TIAA00 input
1	1	TOAA00 output

PFCE31	PFC31	Specification of P31 pin alternate function
0	0	RXDC0 input
0	1	SIF4 input
1	0	INTP08 input
1	1	Setting prohibited

PFCE30	PFC30	Specification of P30 pin alternate function
0	0	TXDC0 output
0	1	SOF4 output
1	0	INTP07 input
1	1	Setting prohibited

4.3.9 Port 9

Port 9 is a 16-bit port for which I/O settings can be controlled in 1-bit units.

Port 9 includes the following alternate-function pins.

Table 4-14. Port 9 Alternate-Function Pins

Pin Name	Pin No.		Alternate-Function Pin Name	I/O	Remark
	V850ES/JG3-H	V850ES/JH3-H			
P90	42	54	KR6/TXDC1/SDA02/A0Note	I/O	Selectable as N-ch open-drain output
P91	43	55	KR7/RXDC1/SCL02/A1Note	I/O	
P92	44	56	TENC01/TIT01/TOT01/A2Note	I/O	–
P93	45	57	TECR0/TIT00/TOT00/A3Note	I/O	
P94	46	58	TIAA31/TOAA31/TENC00 /EVTT0/A4Note	I/O	
P95	47	59	TIAA30/TOAA30/A5Note	I/O	
P96	48	62	TIAA21/TOAA21/INTP11/A6Note	I/O	
P97	49	63	SIF1/TIAA20/TOAA20/A7Note	I/O	
P98	50	64	SOF1/INTP12/A8Note	I/O	
P99	51	65	$\overline{\text{SCKF1}}$/INTP13/A9Note	I/O	
P910	52	66	SIF3/TXDC2/INTP14/A10Note	I/O	
P911	53	67	SOF3/RXDC2/INTP15/A11Note	I/O	
P912	54	68	$\overline{\text{SCKF3}}$/A12Note	I/O	
P913	55	69	TOAB1OFF/INTP16/A13Note	I/O	
P914	56	70	TIAA51/TOAA51/INTP17/A14Note	I/O	
P915	57	71	TIAA50/TOAA50/INTP18/A15Note	I/O	

Note V850ES/JH3-H only

Caution The P90 to P915 pins have hysteresis characteristics in the input mode of the alternate-function pin, but do not have the hysteresis characteristics in the port mode.

Table 4-20. Using Port Pin as Alternate-Function Pin (9/10)

Pin Name	Alternate Function		Pnx Bit of Pn Register	PMnx Bit of PMn Register	PMCnx Bit of PMCn Register	PFCEnx Bit of PFCEn Register	PFCnx Bit of PFCn Register	Other Bits (Registers)
	Name	I/O						
PCM0	$\overline{\text{WAIT}}$Note	Input	PCM0 = Setting not required	PMCM0 = Setting not required	PMCCM0 = 1	–	–	
PCM1	CLKOUT	Output	PCM1 = Setting not required	PMCM1 = Setting not required	PMCCM1 = 1	–	–	
PCM2	$\overline{\text{HLDAK}}$Note	Output	PCM2 = Setting not required	PMCM2 = Setting not required	PMCCM2 = 1	–	–	
PCM3	$\overline{\text{HLDRQ}}$Note	Input	PCM3 = Setting not required	PMCM3 = Setting not required	PMCCM3 = 1	–	–	
PCS0	$\overline{\text{CS0}}$Note	Output	PCS0 = Setting not required	PMCS0 = Setting not required	PMCCS0 = 1	–	–	
PCS2	$\overline{\text{CS2}}$Note	Output	PCS2 = Setting not required	PMCS2 = Setting not required	PMCCS2 = 1	–	–	
PCS3	$\overline{\text{CS3}}$Note	Output	PCS3 = Setting not required	PMCS3 = Setting not required	PMCCS3 = 1	–	–	
PCT0	$\overline{\text{WR0}}$	Output	PCT0 = Setting not required	PMCT0 = Setting not required	PMCCT0 = 1	–	–	
PCT1	$\overline{\text{WR1}}$	Output	PCT1 = Setting not required	PMCT1 = Setting not required	PMCCT1 = 1	–	–	
PCT4	$\overline{\text{RD}}$Note	Output	PCT4 = Setting not required	PMCT4 = Setting not required	PMCCT4 = 1	–	–	
PCT6	ASTBNote	Output	PCT6 = Setting not required	PMCT6 = Setting not required	PMCCT6 = 1	–	–	
PDH0	A16	Output	PDH0 = Setting not required	PMDH0 = Setting not required	PMCDH0 = 1	–	–	
PDH1	A17	Output	PDH1 = Setting not required	PMDH1 = Setting not required	PMCDH1 = 1	–	–	
PDH2	A18	Output	PDH2 = Setting not required	PMDH2 = Setting not required	PMCDH2 = 1	–	–	
PDH3	A19	Output	PDH3 = Setting not required	PMDH3 = Setting not required	PMCDH3 = 1	–	–	
PDH4	A20	Output	PDH4 = Setting not required	PMDH4 = Setting not required	PMCDH4 = 1	–	–	
PDH5	A21	Output	PDH5 = Setting not required	PMDH5 = Setting not required	PMCDH5 = 1	–	–	
PDL0	AD0	I/O	PDL0 = Setting not required	PMDL0 = Setting not required	PMCDL0 = 1	–	–	
PDL1	AD1	I/O	PDL1 = Setting not required	PMDL1 = Setting not required	PMCDL1 = 1	–	–	
PDL2	AD2	I/O	PDL2 = Setting not required	PMDL2 = Setting not required	PMCDL2 = 1	–	–	
PDL3	AD3	I/O	PDL3 = Setting not required	PMDL3 = Setting not required	PMCDL3 = 1	–	–	
PDL4	AD4	I/O	PDL4 = Setting not required	PMDL4 = Setting not required	PMCDL4 = 1	–	–	

Note V850ES/JH3-H only

(1) TAAn control register 0 (TAAnCTL0)

The TAAnCTL0 register is an 8-bit register that controls the operation of TAAn.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

The same value can always be written to the TAAnCTL0 register by software.

After reset: 00H R/W Address: TAA0CTL0 FFFFF630H, TAA1CTL0 FFFFF640H, TAA2CTL0 FFFFF650H, TAA3CTL0 FFFFF660H, TAA4CTL0 FFFFF670H, TAA5CTL0 FFFFF680H

	7	6	5	4	3	2	1	0
TAAnCTL0 (n = 0 to 5)	TAAnCE	0	0	0	0	TAAnCKS2	TAAnCKS1	TAAnCKS0

TAAnCE	TAAn operation control
0	TAAn operation disabled (TAAn reset asynchronously[Note]).
1	TAAn operation enabled. TAAn operation started.

TAAnCKS2	TAAnCKS1	TAAnCKS0	Internal count clock selection	
			n = 0, 1, 4	n = 2, 3, 5
0	0	0	fxx (20.8 ns)	fxx/2 (41.7 ns)
0	0	1	fxx/2 (41.7 ns)	fxx/4 (83.3 ns)
0	1	0	fxx/4 (83.3 ns)	fxx/8 (166.7 ns)
0	1	1	fxx/8 (166.7 ns)	fxx/16 (333.3 ns)
1	0	0	fxx/16 (333.3 ns)	fxx/64 (1.3333 μs)
1	0	1	fxx/32 (666.7 ns)	fxx/256 (5.3333 μs)
1	1	0	fxx/64 (1.3333 μs)	fxx/512 (10.6667 μs)
1	1	1	fxx/128 (2.6667 μs)	fxx/1024 (21.3333 μs)

Note TAAnOPT0.TAAnOVF bit, 16-bit counter, timer output (TOAAn0, TOAAn1 pins)

Cautions 1. Set the TAAnCKS2 to TAAnCKS0 bits when the TAAnCE bit = 0. When the value of the TAAnCE bit is changed from 0 to 1, the TAAnCKS2 to TAAnCKS0 bits can be set simultaneously.

2. Be sure to set bits 3 to 6 to "0".

Remark fxx: Main clock frequency

The values in parentheses indicate the cycles when fxx = 48 MHz.

CHAPTER 12 REAL-TIME COUNTER

12.1 Functions

The real-time counter (RTC) has the following features.

- Counting up to 99 years using year, month, day-of-week, day, hour, minute, and second sub-counters provided
- Year, month, day-of-week, day, hour, minute, and second counter display using BCD codes[Note 1]
- Alarm interrupt function
- Constant-period interrupt function (period: 1 month to 0.5 second)
- Interval interrupt function (period: 1.95 ms to 125 ms)
- Pin output function of 1 Hz
- Pin output function of 32.768 kHz
- Pin output function of 512 Hz or 16.384 kHz
- Watch error correction function
- Subclock operation or main clock operation[Note 2] selectable

Notes 1. A BCD (binary coded decimal) code expresses each digit of a decimal number in 4-bit binary format.

2. Use the baud rate generator dedicated to the real-time counter to divide the main clock frequency to 32.768 kHz for use.

17.3.5 Mode switching between UARTC4, CSIF0, and I²C01

In the V850ES/JG3-H and V850ES/JH3-H, UARTC4, CSIF0, and I²C01 share the same pin and therefore cannot be used simultaneously. Set UARTC4 in advance, using the PMC4, PFC4, and PMCE4 registers, before use.

Caution The transmit/receive operation of UARTC4, CSIF0, and I²C01 is not guaranteed if these functions are switched during transmission or reception. Be sure to disable the one that is not used.

Figure 17-6. UARTC4, CSIF0 and I²C01 Mode Switch Settings

After reset: 00H R/W Address: FFFFF448H

	7	6	5	4	3	2	1	0
PMC4	0	0	0	0	0	PMC42	PMC41	PMC40

After reset: 00H R/W Address: FFFFF468H

	7	6	5	4	3	2	1	0
PFC4	0	0	0	0	0	PFC42	PFC41	PFC40

After reset: 00H R/W Address: FFFFF708H

	7	6	5	4	3	2	1	0
PFCE4	0	0	0	0	0	0	PFCE41	PFCE40

PMC41	PFCE41	PFC41	Operation mode
0	×	×	Port I/O mode
1	0	0	SOF0 (CSIF0)
1	0	1	RXDC4 (UARTC4)
1	1	0	SCL01 (I²C01)

PMC40	PFCE40	PFC40	Operation mode
0	×	×	Port I/O mode
1	0	0	SIF0 (CSIF0)
1	0	1	TXDC4 (UARTC4)
1	1	0	SDA01 (I²C01)

Remark × = don't care

(2) UARTCn control register 1 (UCnCTL1)

The UCnCTL1 register is an 8-bit register that selects the UARTCn base clock.

This register can be read or written in 8-bit units.

Reset sets this register to 00H.

Caution Clear the UCnCTL0.UCnPWR bit to 0 before rewriting the UCnCTL1 register.

After reset: 00H R/W Address: UC0CTL1 FFFFFA01H, UC1CTL1 FFFFFA11H, UC2CTL1 FFFFFA21H, UC3CTL1 FFFFFA31H, UC4CTL1 FFFFFA41H

	7	6	5	4	3	2	1	0
UCnCTL1 (n = 0 to 4)	0	0	0	0	UCnCKS3	UCnCKS2	UCnCKS1	UCnCKS0

UCnCKS3	UCnCKS2	UCnCKS1	UCnCKS0	Base clock (f_{UCLK}) selection
0	0	0	0	fxx/2
0	0	0	1	fxx/4
0	0	1	0	fxx/8
0	0	1	1	fxx/16
0	1	0	0	fxx/32
0	1	0	1	fxx/64
0	1	1	0	fxx/128
0	1	1	1	fxx/256
1	0	0	0	fxx/512
1	0	0	1	fxx/1,024
1	0	1	0	fxx/2,048
1	0	1	1	External clock[Note] (ASCKC0 pin)
Other than above				Setting prohibited

Note Only UARTC0 is valid; setting UARTC1 to UARTC4 is prohibited.

Remark fxx: Main clock frequency

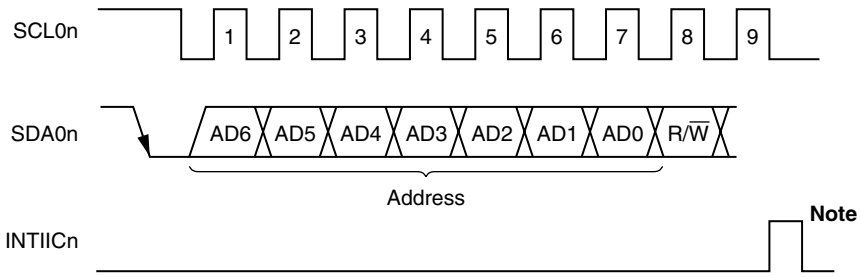
19.6.2 Addresses

The 7 bits of data that follow the start condition are defined as an address.

An address is a 7-bit data segment that is output so that the master device can select one of the slave devices that are connected to the bus lines. Therefore, each slave device connected via the bus lines must have a unique address.

The slave devices detect via hardware the start condition and check whether or not the 7-bit address data matches the data values stored in the SVAn register. If the address data matches the values of the SVAn register, the slave device is selected and communicates with the master device until the master device generates a start condition or stop condition (n = 0 to 2).

Figure 19-9. Address



Note The interrupt request signal (INTIICn) is generated if a local address or extension code is received during slave device operation.

Remark n = 0 to 2

An address is output when the slave address and the transfer direction described in **19.6.3 Transfer direction specification** are written together to the IICn registers as eight bits of data. Received addresses are written to the IICn register (n = 0 to 2).

The slave address is assigned to the higher 7 bits of the IICn register.

Table 19-6. Wait Periods

Clock Selection	CLXn	SMCn	CLn1	CLn0	Wait Period
fxx/6 (OCKSm = 11H)	0	0	0	0	156 clocks
fxx/8 (OCKSm = 12H)	0	0	0	0	208 clocks
fxx/10 (OCKSm = 13H)	0	0	0	0	260 clocks
fxx/4 (OCKSm = 10H)	0	0	0	1	188 clocks
fxx/6 (OCKSm = 11H)	0	0	0	1	282 clocks
fxx/8 (OCKSm = 12H)	0	0	0	1	376 clocks
fxx/10 (OCKSm = 13H)	0	0	0	1	470 clocks
fxx/4 (OCKSm = 10H)	0	0	1	1	148 clocks
fxx/6 (OCKSm = 11H)	0	0	1	1	222 clocks
fxx/4 (OCKSm = 10H)	0	1	0	×	64 clocks
fxx/6 (OCKSm = 11H)	0	1	0	×	96 clocks
fxx/8 (OCKSm = 12H)	0	1	0	×	128 clocks
fxx/10 (OCKSm = 13H)	0	1	0	×	160 clocks
fxx/4 (OCKSm = 10H)	0	1	1	1	52 clocks
fxx/6 (OCKSm = 11H)	0	1	1	1	78 clocks
fxx/6 (OCKSm = 11H)	1	1	0	×	60 clocks
fxx/8 (OCKSm = 12H)	1	1	0	×	80 clocks
fxx/10 (OCKSm = 13H)	1	1	0	×	100 clocks

Remarks 1. m = 0 and 1
n = 0 to 2

2. × = Don't care

The communication reservation timing is shown below.

Table 20-18. CAN Module Register Bit Configuration (1/2)

Address	Symbol	Bit 7/15	Bit 6/14	Bit 5/13	Bit 4/12	Bit 3/11	Bit 2/10	Bit 1/9	Bit 0/8
03FEC040H	C0MASK1L	CMID7 to CMID0							
03FEC041H		CMID15 to CMID8							
03FEC042H	C0MASK1H	CMID23 to CMID16							
03FEC043H		0	0	0	CMID28 to CMID24				
03FEC044H	C0MASK2L	CMID7 to CMID0							
03FEC045H		CMID15 to CMID8							
03FEC046H	C0MASK2H	CMID23 to CMID16							
03FEC047H		0	0	0	CMID28 to CMID24				
03FEC048H	C0MASK3L	CMID7 to CMID0							
03FEC049H		CMID15 to CMID8							
03FEC04AH	C0MASK3H	CMID23 to CMID16							
03FEC04BH		0	0	0	CMID28 to CMID24				
03FEC04CH	C0MASK4L	CMID7 to CMID0							
03FEC04DH		CMID15 to CMID8							
03FEC04EH	C0MASK4H	CMID23 to CMID16							
03FEC04FH		0	0	0	CMID28 to CMID24				
03FEC050H	C0CTRL (W)	0	Clear AL	Clear VALID	Clear PSMODE1	Clear PSMODE0	Clear OPMODE2	Clear OPMODE1	Clear OPMODE0
03FEC051H		Set CCERC	Set AL	0	Set PSMODE1	Set PSMODE0	Set OPMODE2	Set OPMODE1	Set OPMODE0
03FEC050H	C0CTRL (R)	CCERC	AL	VALID	PS MODE1	PS MODE0	OP MODE2	OP MODE1	OP MODE0
03FEC051H		0	0	0	0	0	0	RSTAT	TSTAT
03FEC052H	C0LEC (W)	0	0	0	0	0	0	0	0
03FEC052H	C0LEC (R)	0	0	0	0	0	LEC2	LEC1	LEC0
03FEC053H	C0INFO	0	0	0	BOFF	TECS1	TECS0	RECS1	RECS0
03FEC054H	C0ERC	TEC7 to TEC0							
03FEC055H		REC7 to REC0							
03FEC056H	C0IE (W)	0	0	Clear CIE5	Clear CIE4	Clear CIE3	Clear CIE2	Clear CIE1	Clear CIE0
03FEC057H		0	0	Set CIE5	Set CIE4	Set CIE3	Set CIE2	Set CIE1	Set CIE0
03FEC056H	C0IE (R)	0	0	CIE5	CIE4	CIE3	CIE2	CIE1	CIE0
03FEC057H		0	0	0	0	0	0	0	0
03FEC058H	C0INTS (W)	0	0	Clear CINTS5	Clear CINTS4	Clear CINTS3	Clear CINTS2	Clear CINTS1	Clear CINTS0
03FEC059H		0	0	0	0	0	0	0	0
03FEC058H	C0INTS (R)	0	0	CINTS5	CINTS4	CINTS3	CINTS2	CINTS1	CINTS0
03FEC059H		0	0	0	0	0	0	0	0

(5) CAN0 module mask control register (C0MASKaL, C0MASKaH) (a = 1, 2, 3, or 4)

The C0MASKaL and C0MASKaH registers are used to extend the number of receivable messages in the same message buffer by masking part of the identifier (ID) of a message and invalidating the ID comparison of the masked part.

(1/2)

- CAN0 module mask 1 register (C0MASK1L, C0MASK1H)

After reset: Undefined R/W Address: C0MASK1L 03FEC040H, C0MASK1H 03FEC042H

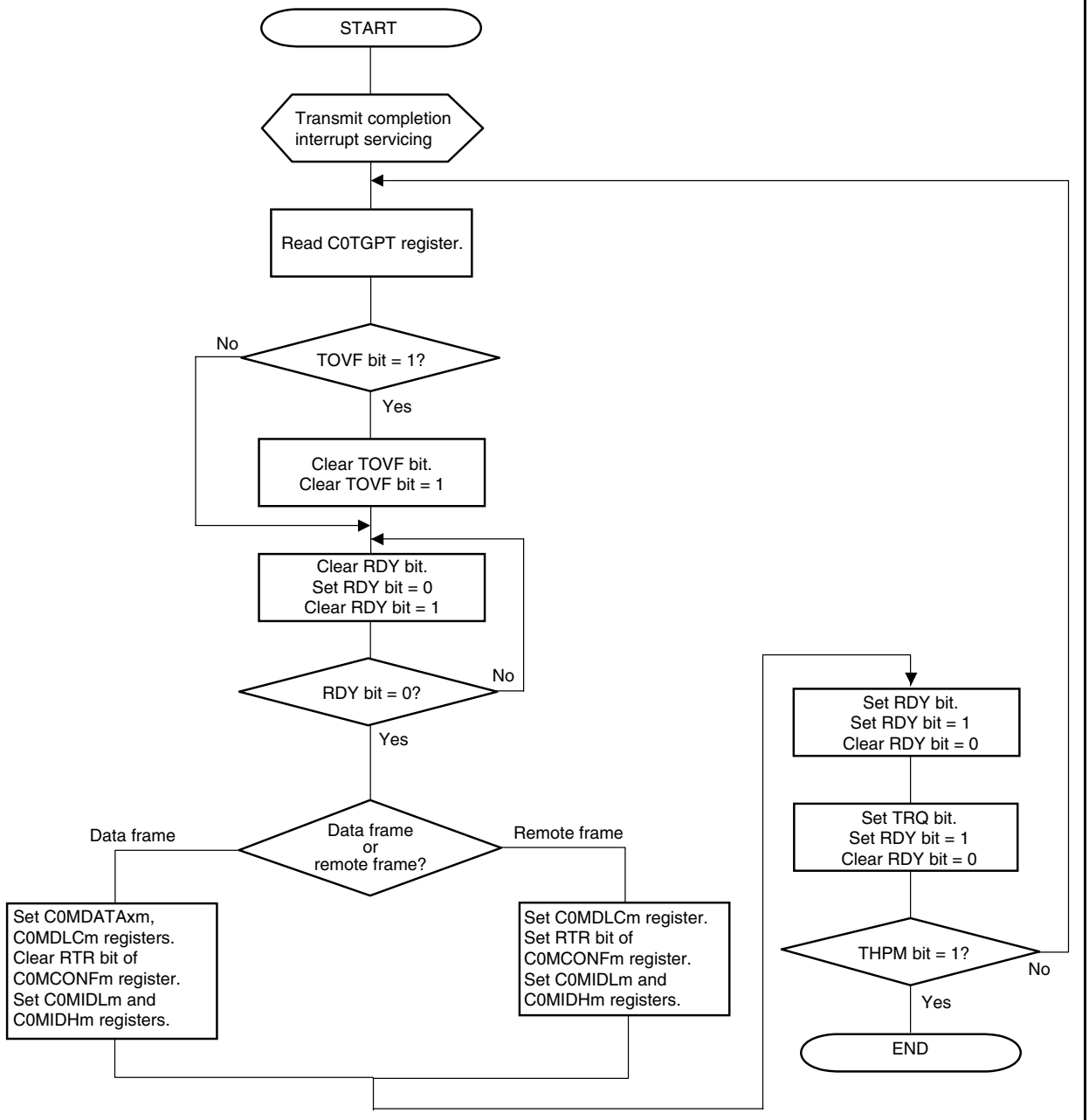
	15	14	13	12	11	10	9	8
C0MASK1L	CMID15	CMID14	CMID13	CMID12	CMID11	CMID10	CMID9	CMID8
	7	6	5	4	3	2	1	0
	CMID7	CMID6	CMID5	CMID4	CMID3	CMID2	CMID1	CMID0
	15	14	13	12	11	10	9	8
C0MASK1H	0	0	0	CMID28	CMID27	CMID26	CMID25	CMID24
	7	6	5	4	3	2	1	0
	CMID23	CMID22	CMID21	CMID20	CMID19	CMID18	CMID17	CMID16

- CAN0 module mask 2 register (C0MASK2L, C0MASK2H)

After reset: Undefined R/W Address: C0MASK2L 03FEC044H, C0MASK2H 03FEC046H

	15	14	13	12	11	10	9	8
C0MASK2L	CMID15	CMID14	CMID13	CMID12	CMID11	CMID10	CMID9	CMID8
	7	6	5	4	3	2	1	0
	CMID7	CMID6	CMID5	CMID4	CMID3	CMID2	CMID1	CMID0
	15	14	13	12	11	10	9	8
C0MASK2H	0	0	0	CMID28	CMID27	CMID26	CMID25	CMID24
	7	6	5	4	3	2	1	0
	CMID23	CMID22	CMID21	CMID20	CMID19	CMID18	CMID17	CMID16

Figure 20-44. Transmission via Interrupt (Using C0TGPT Register)



Cautions 1. The TRQ bit should be set after the RDY bit is set.

2. The RDY bit and TRQ bit should not be set at the same time.

Remarks 1. Check the MBON bit at the start and end of the interrupt routine to see if the message buffer and transmit history register can be accessed, because a CAN sleep mode transition request which has been held pending may be under execution. If the MBON bit is cleared (0), stop the processing under execution. Re-execute the processing after the MBON bit is set (1) again. It is therefore recommended to cancel the CAN sleep mode transition request before executing transmission interrupt servicing.

2. If the TOVF bit is set (1) again, the transmit history list contradicts. Therefore, scan all the transmit message buffers that have completed transmission.

(24) UF0 INT clear 3 register (UF0IC3)

This register controls clearing the interrupt sources indicated by the UF0IS3 register.

This register is write-only, in 8-bit units. If this register is read, the value FFH is read.

FW can clear an interrupt source by writing 0 to the corresponding bit of this register. Even a bit that is automatically cleared to 0 by hardware can be cleared by FW before it is cleared by hardware. Writing 0 to a bit of this register automatically sets the bit to 1. Writing 1 is invalid.

The related bits are invalid if each endpoint is not supported by the setting of the UF0EnIM register (n = 2, 4) and the current setting of the interface.

	7	6	5	4	3	2	1	0	Address	After reset
UF0IC3	BKO2 FLC	BKO2 NLC	BKO2 NAKC	BKO2 DTC	BKO1 FLC	BKO1 NLC	BKO1 NAKC	BKO1 DTC	00200042H	FFH

Bit position	Bit name	Function
7, 3	BKOnFLC	These bits clear the BLKOnFL interrupt. 0: Clear
6, 2	BKOnNLC	These bits clear the BLKOnNL interrupt. 0: Clear
5, 1	BKOnNAKC	These bits clear the BLKOnNK interrupt. 0: Clear
4, 0	BKOnDTC	These bits clear the BLKOnDT interrupt. 0: Clear

Remark n = 1, 2

(38) UF0 endpoint 1 interface mapping register (UF0E1IM)

This register specifies for which Interface and Alternative Setting Endpoint1 is valid.

This register can be read or written in 8-bit units.

The setting of this register and the Alternative Setting selected by the SET_INTERFACE request indicate whether Endpoint1 is currently valid, and the hardware determines how the GET_STATUS/CLEAR_FEATURE/SET_FEATURE Endpoint1 request and the IN transaction to Endpoint1 are responded to, and whether the related bits are valid or invalid.

	7	6	5	4	3	2	1	0	Address	After reset
UF0E1IM	E1EN2	E1EN1	E1EN0	E12AL1	E15AL4	E15AL3	E15AL2	E15AL1	00200086H	00H

Bit position	Bit name	Function
7 to 5	E1EN2 to E1EN0	These bits set a link between the Interface of Endpoint1 and the two-/five-series Alternative Setting. The endpoint is linked with Alternative Setting 0. The endpoint linked with Alternative Setting 0 cannot be excluded from Alternative Setting 1 to 4.

E1EN2	E1EN1	E1EN0	Link status
1	1	1	Not linked with Interface
1	1	0	
1	0	1	Linked with Interface 4 and Alternative Setting 0
1	0	0	Linked with Interface 3 and Alternative Setting 0
0	1	1	Linked with Interface 2 and Alternative Setting 0
0	1	0	Linked with Interface 1 and Alternative Setting 0
0	0	1	Linked with Interface 0 and Alternative Setting 0
0	0	0	Not linked with Interface (default value)

When these bits are set to 110 or 111, they are invalid even if the E12AL1 bit is cleared to 0.

If the endpoint is linked, setting of the CONF bit of the UF0MODS register to 1 indicates that Endpoint1 is valid.

4	E12AL1	This bit validates Endpoint1 when the two-series Alternative Setting and the Alternative Setting of the linked Interface are set to 1. 1: Validate the endpoint when Alternative Setting 1 is set with CONF bit = 1. 0: Do not validate the endpoint even when Alternative Setting 1 is set with CONF bit = 1 (default value). This bit is valid when the E15AL4 to E15AL1 bits are 0000.
3 to 0	E15ALn	These bits validate Endpoint1 when the five-series Alternative Setting and the Alternative Setting of the linked Interface are set to n. 1: Validate the endpoint when Alternative Setting n is set with CONF bit = 1. 0: Do not validate the endpoint even when Alternative Setting n is set with CONF bit = 1 (default value).

Remark n = 1 to 4

23.3.4 Interrupt control register (xxICn)

The xxICn register is assigned to each interrupt request signal (maskable interrupt) and sets the control conditions for each maskable interrupt request.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 47H.

Caution Disable interrupts (DI) or mask the interrupt to read the xxICn.xxIFn bit. If the xxIFn bit is read while interrupts are enabled (EI) or while the interrupt is unmasked, the correct value may not be read when acknowledging an interrupt and reading the bit conflict.

After reset: 47H R/W Address: FFFFF112H to FFFFF184H

	<7>	<6>	5	4	3	2	1	0
xxICn	xxIFn	xxMKn	0	0	0	xxPRn2	xxPRn1	xxPRn0

xxIFn	Interrupt request flag[Note]
0	Interrupt request not issued
1	Interrupt request issued

xxMKn	Interrupt mask flag
0	Interrupt servicing enabled
1	Interrupt servicing disabled (pending)

xxPRn2	xxPRn1	xxPRn0	Interrupt priority specification bit
0	0	0	Specifies level 0 (highest).
0	0	1	Specifies level 1.
0	1	0	Specifies level 2.
0	1	1	Specifies level 3.
1	0	0	Specifies level 4.
1	0	1	Specifies level 5.
1	1	0	Specifies level 6.
1	1	1	Specifies level 7 (lowest).

Note The flag xxIFn is reset automatically by the hardware if an interrupt request signal is acknowledged.

Remark xx: Identification name of each peripheral unit (see **Table 23-4 Interrupt Control Register (xxICn)**)
n: Peripheral unit number (see **Table 23-4 Interrupt Control Register (xxICn)**).

The addresses and bits of the interrupt control registers are as follows.

32.2.3 Securement of user resources

The user must prepare the following to perform communication between MINICUBE2 and the target device and implement each debug function. These items need to be set in the user program or using the compiler options.

(1) Securement of memory space

The shaded portions in Figure 32-4 are the areas reserved for placing the debug monitor program, so user programs and data cannot be allocated in these spaces. These spaces must be secured so as not to be used by the user program.

(2) Security ID setting

The ID code must be embedded in the area between 0000070H and 0000079H in Figure 32-4, to prevent the memory from being read by an unauthorized person. For details, refer to **32.3 ROM Security Function**.

(11/37)

Symbol	Name	Unit	Page
C0MDATA518	CAN0 message data byte 5 register 18	CAN	969
C0MDATA519	CAN0 message data byte 5 register 19	CAN	969
C0MDATA520	CAN0 message data byte 5 register 20	CAN	969
C0MDATA521	CAN0 message data byte 5 register 21	CAN	969
C0MDATA522	CAN0 message data byte 5 register 22	CAN	969
C0MDATA523	CAN0 message data byte 5 register 23	CAN	969
C0MDATA524	CAN0 message data byte 5 register 24	CAN	969
C0MDATA525	CAN0 message data byte 5 register 25	CAN	969
C0MDATA526	CAN0 message data byte 5 register 26	CAN	969
C0MDATA527	CAN0 message data byte 5 register 27	CAN	969
C0MDATA528	CAN0 message data byte 5 register 28	CAN	969
C0MDATA529	CAN0 message data byte 5 register 29	CAN	969
C0MDATA530	CAN0 message data byte 5 register 30	CAN	969
C0MDATA531	CAN0 message data byte 5 register 31	CAN	969
C0MDATA600	CAN0 message data byte 6 register 00	CAN	969
C0MDATA601	CAN0 message data byte 6 register 01	CAN	969
C0MDATA602	CAN0 message data byte 6 register 02	CAN	969
C0MDATA603	CAN0 message data byte 6 register 03	CAN	969
C0MDATA604	CAN0 message data byte 6 register 04	CAN	969
C0MDATA605	CAN0 message data byte 6 register 05	CAN	969
C0MDATA606	CAN0 message data byte 6 register 06	CAN	969
C0MDATA607	CAN0 message data byte 6 register 07	CAN	969
C0MDATA608	CAN0 message data byte 6 register 08	CAN	969
C0MDATA609	CAN0 message data byte 6 register 09	CAN	969
C0MDATA610	CAN0 message data byte 6 register 10	CAN	969
C0MDATA611	CAN0 message data byte 6 register 11	CAN	969
C0MDATA612	CAN0 message data byte 6 register 12	CAN	969
C0MDATA613	CAN0 message data byte 6 register 13	CAN	969
C0MDATA614	CAN0 message data byte 6 register 14	CAN	969
C0MDATA615	CAN0 message data byte 6 register 15	CAN	969
C0MDATA616	CAN0 message data byte 6 register 16	CAN	969
C0MDATA617	CAN0 message data byte 6 register 17	CAN	969
C0MDATA618	CAN0 message data byte 6 register 18	CAN	969
C0MDATA619	CAN0 message data byte 6 register 19	CAN	969
C0MDATA620	CAN0 message data byte 6 register 20	CAN	969
C0MDATA621	CAN0 message data byte 6 register 21	CAN	969
C0MDATA622	CAN0 message data byte 6 register 22	CAN	969
C0MDATA623	CAN0 message data byte 6 register 23	CAN	969
C0MDATA624	CAN0 message data byte 6 register 24	CAN	969
C0MDATA625	CAN0 message data byte 6 register 25	CAN	969

(4/4)

Edition	Description	Chapter
3rd	Modification of **4.3.3 (6)Port 2 alternate function specifications**	**CHAPTER 4 PORT FUNCTION**
	Modification of **4.3.9 Port 9 function control register (PFC9)**	
	Modification of **4.3.9 Port 9 function control expansion register (PFCE9)**	
	Modification of **Table 4-20. Using Port Pin as Alternate-Function Pin**	
	Modification of **Figure 12-1. Block Diagram of Real-Time Counter**	**CHAPTER 12 REAL-TIME COUNTER**
	Modification of **12.2.2 (3)**	
	Modification of **Figure 12-10. Watch Error Correction Example**	
	Modification of **18.4 (2) CSIFn control register 1 (CFnCTL1) Note**	**CHAPTER 18 3-WIRE VARIABLE-LENGTH SERIAL I/O (CSIF)**
	Modification of **Figure 21-30. Example of Suspend/Resume Processing**	**CHAPTER 21 USB FUNCTION CONTROLLER (USBF)**
	Addition of Caution to **(6) DMA trigger factor registers 0 to 3 (DTFR0 to DTFR3)**	**CHAPTER 22 DMA FUNCTION (DMA CONTROLLER)**
	Modification of **Table 31-2. Basic Functions**	**CHAPTER 31 FLASH MEMORY**
	Modification of **Table 31-8. Flash Memory Control Commands**	
	Modification of **33.9 (1) Basic characteristics**	**CHAPTER 33 ELECTRICAL SPECIFICATIONS**