Renesas Electronics America Inc - M306S0FAGP#U3 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	Last Time Buy
Core Processor	M16C/60
Core Size	16-Bit
Speed	15MHz
Connectivity	I²C, SIO, UART/USART
Peripherals	DMA, WDT
Number of I/O	20
Program Memory Size	96KB (96K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	24K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	64-LFQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/m306s0fagp-u3

Pin Configuration

Figures 1.1.4 show the pin configurations (top view).

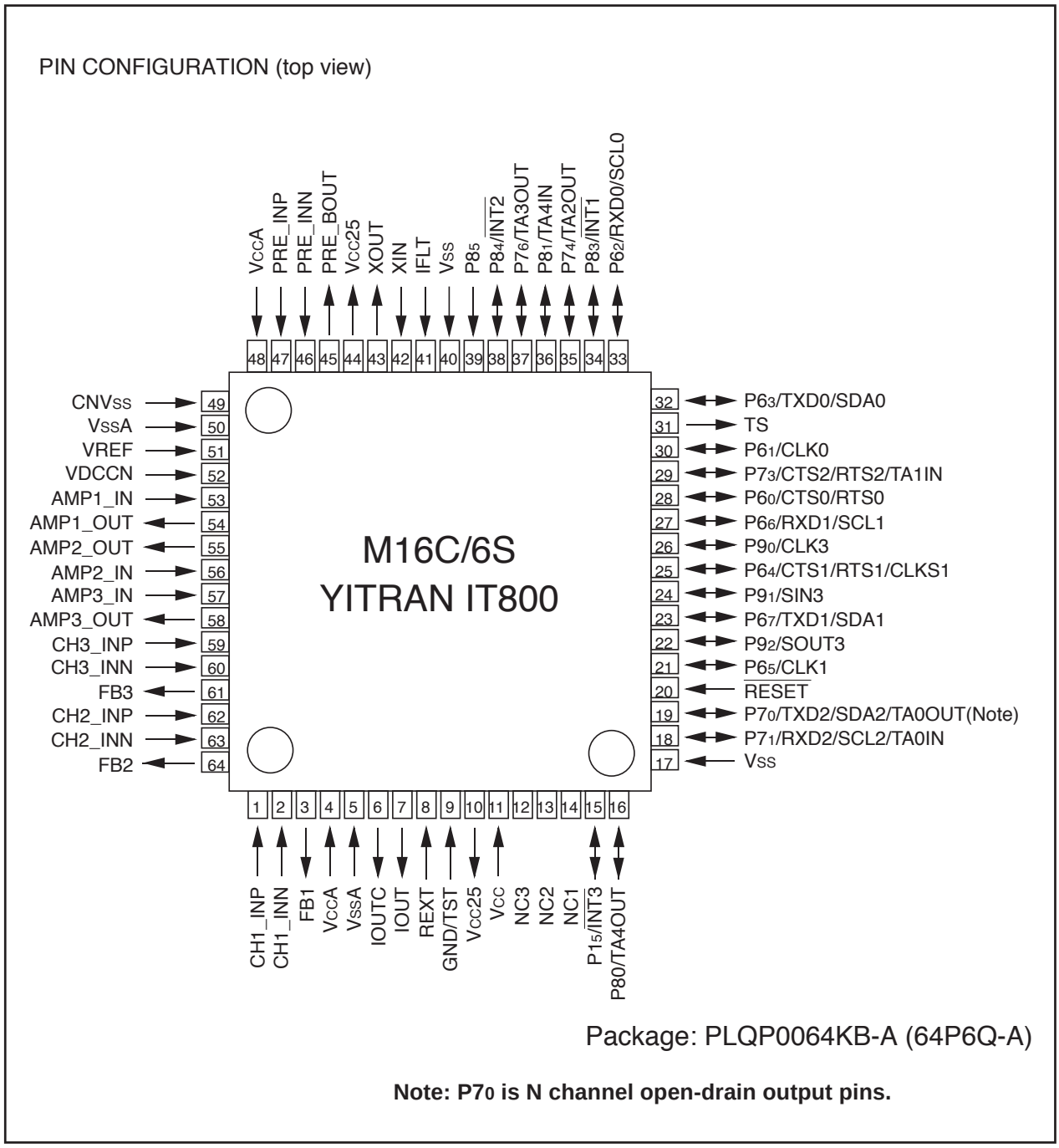


Figure 1.1.4. Pin Configuration (Top View)

Table 1.1.8 Pin Description (2) (Analog pin)

Pin name	I/O type	Function
PRE-BOUT	Output	This is a pre-amp buffer output.
PRE-INN	Input	This is a pre-amp differential signal input.
PRE-INP	Input	This is a pre-amp differential signal input.
VREF	Input	This is the reference voltage input of amplifier common to channels 1, 2, and 3.
VDCCN	Input	This is a pin for a test. Usually, please carry out a pull-up.
AMP1-IN	Input	This is a channel 1 amplifier input.
AMP1-OUT	Output	This is a channel 1 amplifier output.
AMP2-IN	Input	This is a channel 2 amplifier input.
AMP2-OUT	Output	This is a channel 2 amplifier output.
AMP3-IN	Input	This is a channel 3 amplifier input.
AMP3-OUT	Output	This is a channel 3 amplifier output.
CHI-INP	Input	This is a channel 1 comparator differential input.
CHI-INN	Input	This is a channel 1 comparator differential input.
FB1	Output	This is a channel 1 comparator feedback output.
CH2-INP	Input	This is a channel 2 comparator differential input.
CH2-INN	Input	This is a channel 2 comparator differential input.
FB2	Output	This is a channel 2 comparator feedback output.
CH3-INP	Input	This is a channel 3 comparator differential input.
CH3-INN	Input	This is a channel 3 comparator differential input.
FB3	Output	This is a channel 3 comparator feedback output.
IOUTC	Output	This is the differential current output of DAC.
IOUT	Output	This is the differential current output of DAC.
REXT	Input	This is for external reference resistor of DAC.
IFLT	Input	This is the pin for low path filters of PLL.

Address	Register	Symbol	After reset
008016			
008116			
008216			
008316			
008416			
008516			
008616			
≈			≈
01B016			
01B116			
01B216			
01B316			
01B416			
01B516	Flash memory control register 1	FMR1	0?00??0?2
01B616			
01B716	Flash memory control register 0	FMR0	??0000012
01B816 01B916 01BA16	Address match interrupt register 2	RMAD2	0016 0016 X016
01BB16	Address match interrupt enable register 2	AIER2	XXXXXX002
01BC16 01BD16 01BE16	Address match interrupt register 3	RMAD3	0016 0016 X016
01BF16			
≈			≈
025016			
025116			
025216			
025316			
025416			
025516			
025616			
025716			
025816			
025916			
025A16			
025B16			
025C16			
025D16			
025E16	Peripheral clock select register	PCLKR	000000112
025F16			
≈			≈
033016			
033116			
033216			
033316			
033416			
033516			
033616			
033716			
033816			
033916			
033A16			
033B16			
033C16			
033D16			
033E16			
033F16			

Note 1: The blank areas are reserved and cannot be used by users.

X : Nothing is mapped to this bit
? : Undefined

(2) Setting PLC Mode

PLC mode is simply set by putting P15 High level during RESET.

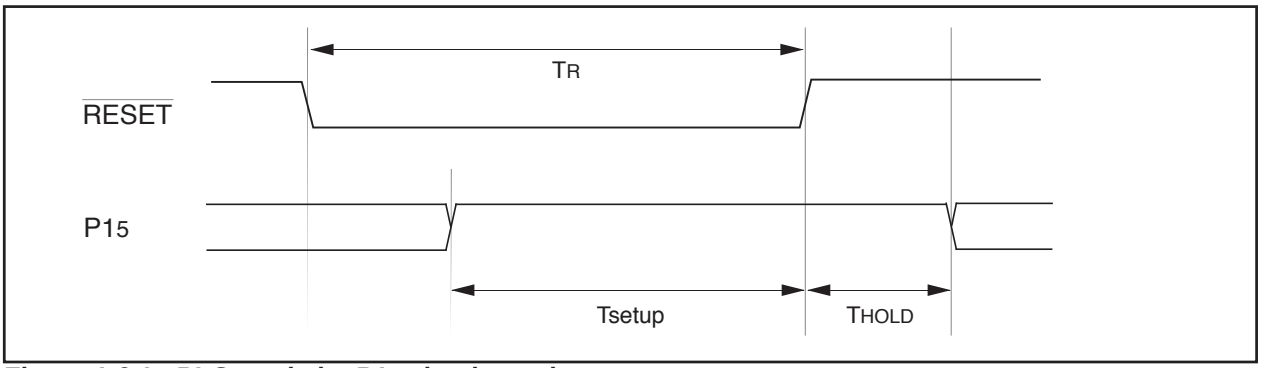


Figure 1.6.1. PLC mode by P15 simply setting

Table 1.6.3. RESET and P15 Input

	min	typ	max
TR	40us	—	—
Tset up	5us	—	—
THOLD	5us	—	—

Table 1.7.3. Pin Status During Wait Mode

Pin	Status
I/O ports	Retains status before wait mode

Table 1.7.4. Interrupts to Exit Wait Mode

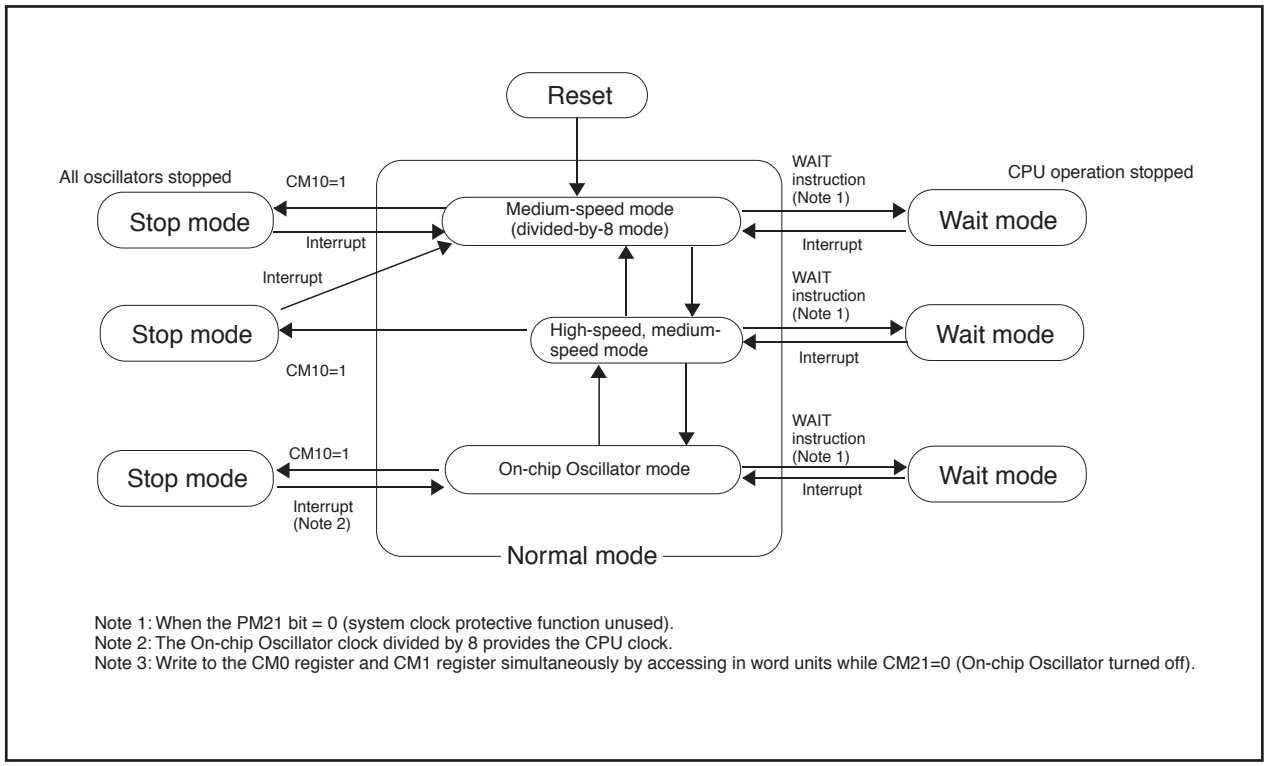
Interrupt	CM02=0	CM02=1
Serial I/O interrupt	Can be used when operating with internal or external clock	Can be used when operating with external clock
Timer A interrupt	Can be used in all modes	Can be used in event counter mode
$\overline{\text{INT}}$ interrupt	Can be used	Can be used

If the microcomputer is to be moved out of wait mode by a peripheral function interrupt, set up the following before executing the WAIT instruction.

1. In the ILVL2 to ILVL0 bits of interrupt control register, set the interrupt priority level of the peripheral function interrupt to be used to exit wait mode.
   Also, for all of the peripheral function interrupts not used to exit wait mode, set the ILVL2 to ILVL0 bits to “000_2” (interrupt disable).
2. Set the I flag to “1”.
3. Enable the peripheral function whose interrupt is to be used to exit wait mode.
   In this case, when an interrupt request is generated and the CPU clock is thereby turned on, an interrupt routine is executed.

The CPU clock turned on when exiting wait mode by a peripheral function interrupt is the same CPU clock that was on when the WAIT instruction was executed.

Figure 1.7.7 shows the state transition from normal operation mode to stop mode and wait mode.



Note 1: When the PM21 bit = 0 (system clock protective function unused).
Note 2: The On-chip Oscillator clock divided by 8 provides the CPU clock.
Note 3: Write to the CM0 register and CM1 register simultaneously by accessing in word units while CM21=0 (On-chip Oscillator turned off).

Figure 1.7.7. State Transition to Stop Mode and Wait Mode

Interrupts and Interrupt Vector

One interrupt vector consists of 4 bytes. Set the start address of each interrupt routine in the respective interrupt vectors. When an interrupt request is accepted, the CPU branches to the address set in the corresponding interrupt vector. Figure 1.9.2 shows the interrupt vector.

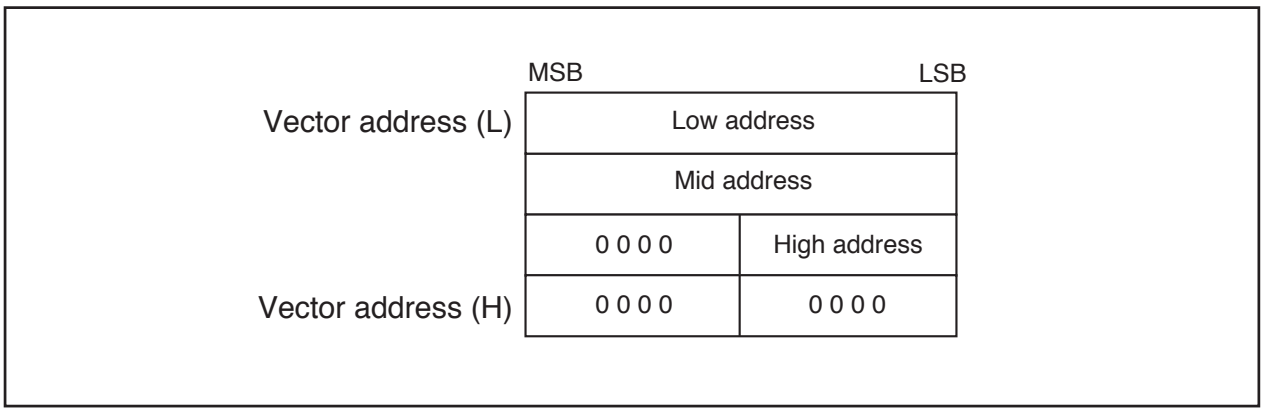


Figure 1.9.2. Interrupt Vector

- **Fixed Vector Tables**

The fixed vector tables are allocated to the addresses from FFFDC16 to FFFFF16. Table 1.9.1 lists the fixed vector tables. In the flash memory version of microcomputer, the vector addresses (H) of fixed vectors are used by the ID code check function. For details, refer to the section "flash memory rewrite disabling function".

Table 1.9.1. Fixed Vector Tables

Interrupt source	Vector table addresses Address (L) to address (H)	Remarks	Reference
Undefined instruction	FFFDC16 to FFFDF16	Interrupt on UND instruction	M16C/60, M16C/20 series software manual
Overflow	FFFE016 to FFFE316	Interrupt on INTO instruction	
BRK instruction	FFFE416 to FFFE716	If the contents of address FFFE716 is FF16, program execution starts from the address shown by the vector in the relocatable vector table.	
Address match	FFFE816 to FFFEB16		Address match interrupt
Single step (Note)	FFFEC16 to FFFEF16		
Watchdog timer Oscillation stop and re-oscillation detection	FFFF016 to FFFF316		Watchdog timer Clock generating circuit
$\overline{\text{DBC}}$ (Note)	FFFF416 to FFFF716		
—— (Reserved)	FFFF816 to FFFFB16		
Reset	FFFFC16 to FFFFF16		Reset

Note: Do not normally use this interrupt because it is provided exclusively for use by development support tools.

Timer Ai mode register (i=0 to 4)
(When not using two-phase pulse signal processing)

b7	b6	b5	b4	b3	b2	b1	b0
		0				0	1

Symbol	Address	After reset
TA0MR to TA4MR	0396_{16} to $039A_{16}$	00_{16}

Bit symbol	Bit name	Function	RW
TMOD0	Operation mode select bit	b1 b0 0 1 : Event counter mode (Note 1)	RW
TMOD1			RW
MR0	Pulse output function select bit	0 : Pulse is not output (TAiOUT pin functions as I/O port) 1 : Pulse is output (Note 2) (TAiOUT pin functions as pulse output pin)	RW
MR1	Count polarity select bit (Note 3)	0 : Counts external signal's falling edge 1 : Counts external signal's rising edge	RW
MR2	Up/down switching cause select bit	0 : UDF register 1 : Input signal to TAiOUT pin (Note 4)	RW
MR3	Must be set to “0” in event counter mode		RW
TCK0	Count operation type select bit	0 : Reload type 1 : Free-run type	RW
TCK1	Can be “0” or “1” when not using two-phase pulse signal processing		RW

Note 1: During event counter mode, the count source can be selected using the ONSF and TRGSR registers.
Note 2: TA0OUT pin is N-channel open drain output.
Note 3: Effective when the TAiGH and TAiGL bits of ONSF or TRGSR register are ‘00_2’ (TAiIN pin input).
Note 4: Count down when input on TAiOUT pin is low or count up when input on that pin is high. The port direction bit for TAiOUT pin must be set to “0” (= input mode).

Figure 1.12.7. TAiMR Register in Event Counter Mode (when not using two-phase pulse signal processing)

Counter Measure for Communication Error Occurs

If a communication error occurs while transmitting or receiving in clock synchronous serial I/O mode, follow the procedures below.

• Resetting the UiRB register (i=0 to 2)

(1) Set the RE bit in the UiC1 register to “0” (reception disabled)

(2) Set the SMD2 to SMD0 bits in the UiMR register to “000b” (Serial I/O disabled)

(3) Set the SMD2 to SMD0 bits in the UiMR register to “001b” (Clock synchronous serial I/O mode)

(4) Set the RE bit in the UiC1 register to “1” (reception enabled)

• Resetting the UiTB register (i=0 to 2)

(1) Set the SMD2 to SMD0 bits in the UiMR register “000b” (Serial I/O disabled)

(2) Set the SMD2 to SMD0 bits in the UiMR register “001b” (Clock synchronous serial I/O mode)

(3) “1” is written to RE bit in the UiC1 register (reception enabled), regardless of the TE bit in the UiCi register

(a) CLK Polarity Select Function

Use the UiC0 register (i = 0 to 1)’s CKPOL bit to select the transfer clock polarity. Figure 1.14.2 shows the polarity of the transfer clock.

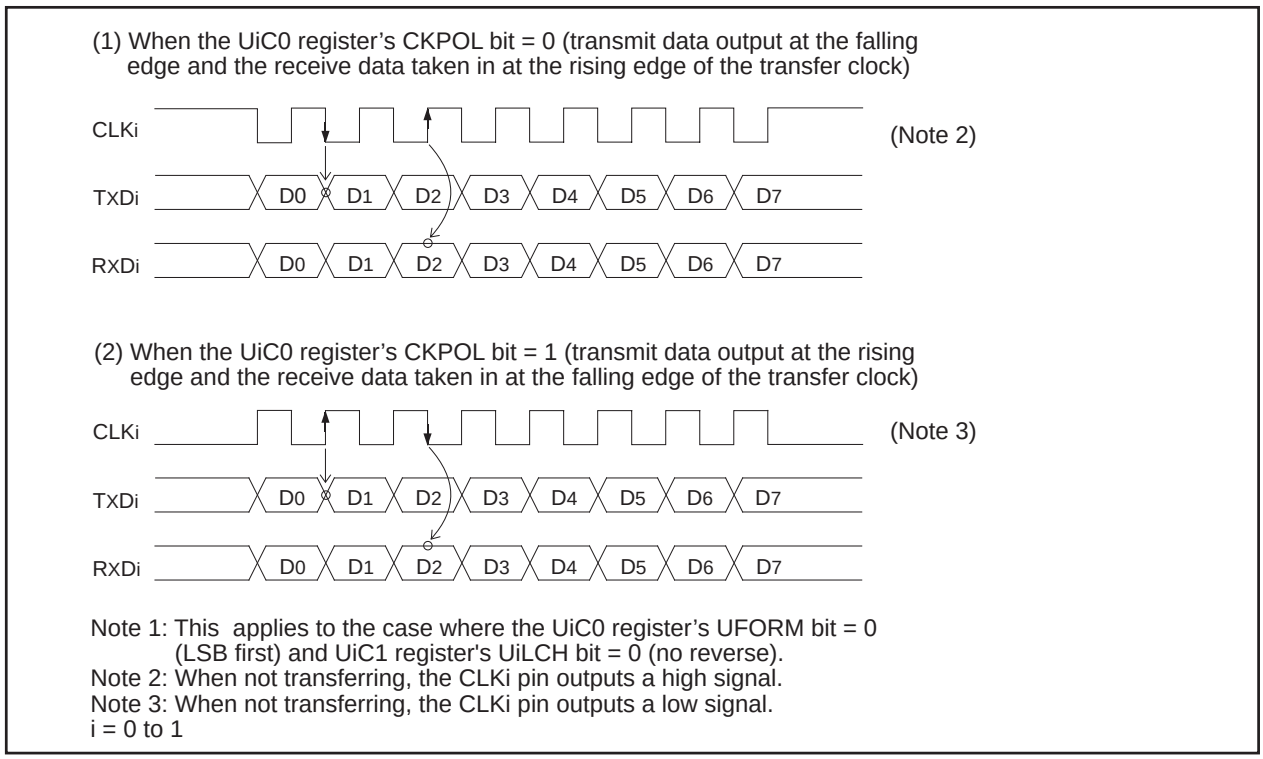


Figure 1.14.2. Transfer Clock Polarity

Table 1.15.3 lists the functions of the input/output pins during UART mode. Table 1.15.4 lists the $P6_4$ pin functions during UART mode. Note that for a period from when the UARTi operation mode is selected to when transfer starts, the TxDi pin outputs an “H”. (If the N-channel open-drain output is selected, this pin is in a high-impedance state.)

Table 1.15.3. I/O Pin Functions

Pin name	Function	Method of selection
TxDi (i = 0 to 2) ($P6_3$, $P6_7$, $P7_0$)	Serial data output	(Outputs dummy data when performing reception only)
RxDi ($P6_2$, $P6_6$, $P7_1$)	Serial data input	PD6 register’s PD6_2 bit=0, PD6_6 bit=0, PD7 register’s PD7_1 bit=0 (Can be used as an input port when performing transmission only)
CLKi ($P6_1$, $P6_5$)	Input/output port	UiMR register’s CKDIR bit=0
	Transfer clock input	UiMR register’s CKDIR bit=1 PD6 register’s PD6_1 bit=0, PD6_5 bit=0, PD7 register’s PD7_2 bit=0
$\overline{CTS}i/\overline{RTS}i$ ($P6_0$, $P6_4$, $P7_3$)	$\overline{CTS}$ input	UiC0 register’s CRD bit=0 UiC0 register’s CRS bit=0 PD6 register’s PD6_0 bit=0, PD6_4 bit=0, PD7 register’s PD7_3 bit=0
	$\overline{RTS}$ output	UiC0 register’s CRD bit=0 UiC0 register’s CRS bit=1
	Input/output port	UiC0 register’s CRD bit=1

Table 1.15.4. $P6_4$ Pin Functions

Pin function	Bit set value				
	U1C0 register		UCON register		PD6 register
	CRD	CRS	RCSP	CLKMD1	PD6_4
$P6_4$	1	—	0	0	Input: 0, Output: 1
$\overline{CTS}_1$	0	0	0	0	0
$\overline{RTS}_1$	0	1	0	0	—
$\overline{CTS}_0$ (Note)	0	0	1	0	0

Note: In addition to this, set the U0C0 register’s CRD bit to “0” ($\overline{CTS}_0/\overline{RTS}_0$ enabled) and the U0C0 register’s CRS bit to “1” ($\overline{RTS}_0$ selected).

• **Detection of Start and Stop Condition**

Whether a start or a stop condition has been detected is determined.

A start condition-detected interrupt request is generated when the SDAi pin changes state from high to low while the SCLi pin is in the high state. A stop condition-detected interrupt request is generated when the SDAi pin changes state from low to high while the SCLi pin is in the high state.

Because the start and stop condition-detected interrupts share the interrupt control register and vector, check the UiSMR register's BBS bit to determine which interrupt source is requesting the interrupt.

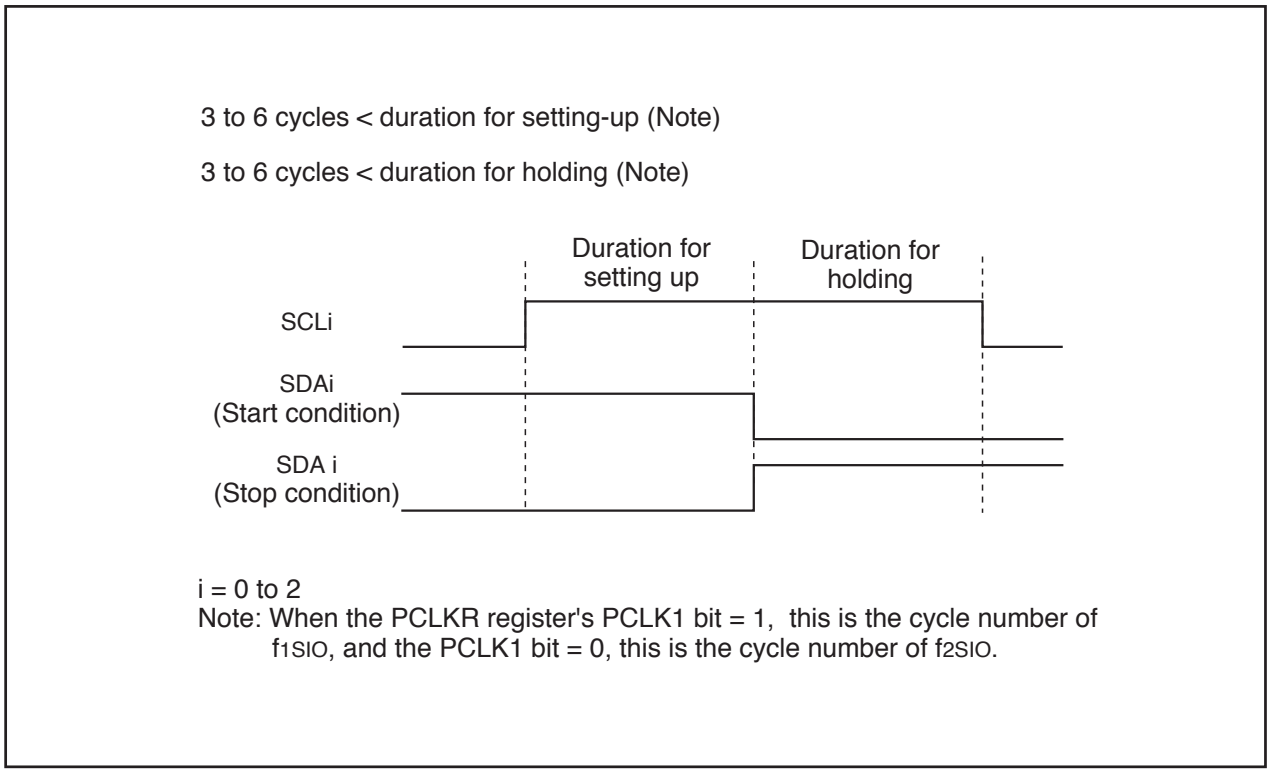


Figure 1.16.3. Detection of Start and Stop Condition

• **Output of Start and Stop Condition**

A start condition is generated by setting the STAREQ bit in the UiSMR4 register (i = 0 to 2) to "1" (start).

A restart condition is generated by setting the RSTAREQ bit in the UiSMR4 register to "1" (start).

A stop condition is generated by setting the STPREQ bit in the UiSMR4 register to "1" (start).

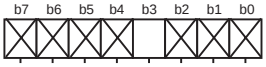
The output procedure is described below.

(1) Set the STAREQ bit, RSTAREQ bit or STPREQ bit to "1" (start).

(2) Set the STSPSEL bit in the UiSMR4 register to "1" (output).

The function of the STSPSEL bit is shown in Table 1.16.5 and Figure 1.16.4.

Pull-up control register 0 (Note 1)

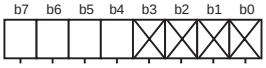


Symbol	Address	After reset
PUR0	03FC16	0016

Bit symbol	Bit name	Function	RW
(b2-b0)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—
PU03	P14 to P17 pull-up	0 : Not pulled high 1 : Pulled high (Note 2)	RW
(b7-b4)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

Note 1: During memory extension and microprocessor modes, the pins are not pulled high although their corresponding register contents can be modified.
Note 2: The pin for which this bit is "1" (pulled high) and the direction bit is "0" (input mode) is pulled high.

Pull-up control register 1

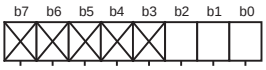


Symbol	Address	After reset(Note 5)
PUR1	03FD16	000000002 000000102

Bit symbol	Bit name	Function	RW
(b3-b0)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—
PU14	P60 to P63 pull-up	0 : Not pulled high 1 : Pulled high (Note 3)	RW
PU15	P64 to P67 pull-up		RW
PU16	P72 to P73 pull-up (Note 1)		RW
PU17	P74 to P77 pull-up		RW

Note 1: The P70 and P71 pins do not have pull-ups.
Note 2: During memory extension and microprocessor modes, the pins are not pulled high although the contents of these bits can be modified.
Note 3: The pin for which this bit is "1" (pulled high) and the direction bit is "0" (input mode) is pulled high.
Note 4: If the PM01 to PM00 bits are set to "012" (memory expansion mode) or "112" (microprocessor mode) in a program during single-chip mode, the PU11 bit becomes "1".
Note 5: The values after hardware reset 1 and 2 are as follows:
• 000000002 when input on CNVss pin is "L"
• 000000102 when input on CNVss pin is "H"
The values after software reset, watchdog timer reset and oscillation stop detection reset are as follows:
• 000000002 when PM 01 to PM00 bits of PM0 register are "002" (single-chip mode)
• 000000102 when PM 01 to PM00 bits of PM0 register are "012" (memory expansion mode) or "112" (microprocessor mode)

Pull-up control register 2



Symbol	Address	After reset
PUR2	03FE16	0016

Bit symbol	Bit name	Function	RW
PU20	P80 to P83 pull-up	0 : Not pulled high 1 : Pulled high (Note 1)	RW
PU21	P84 to P87 pull-up (Note 2)		RW
PU22	P90 to P93 pull-up		RW
(b7-b3)	Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be "0".		—

Note 1: The pin for which this bit is "1" (pulled high) and the direction bit is "0" (input mode) is pulled high.
Note 2: The P85 pin does not have pull-up.

Figure 1.18.8. PUR0 to PUR2 Registers

Table 1.18.1. Unassigned Pin Handling in Single-chip Mode (Excluding Analog Pins)

Pin name	Connection
Ports P1, P6 to P9 (excluding P8$_5$)	After setting for input mode, connect every pin to V_{SS} via a resistor(pull-down); or after setting for output mode, leave these pins open. (2, 3, 4)
X_{OUT} (Note 1)	Open
P8$_5$	Connect via resistor to V_{CC} (pull-up)
$V_{CC}A$	Connect to V_{CC}
$V_{SS}A$	Connect to V_{SS}

NOTES:
1. With external clock input to X_{IN} pin.
2. When setting the port for output mode and leave it open, be aware that the port remains in input mode until it is switched to output mode in a program after reset. For this reason, the voltage level on the pin becomes indeterminate, causing the power supply current to increase while the port remains in input mode. Furthermore, by considering a possibility that the contents of the direction registers could be changed by noise or noise-induced runaway, it is recommended that the contents of the direction registers be periodically reset in software, for the increased reliability of the program.
3. Make sure the unused pins are processed with the shortest possible wiring from the microcomputer pins (within 2 cm).
4. When the port P7_0 is set for output mode, make sure a low-level signal is output from the pin. The port P7_0 is N-channel open-drain output.

1. Memory Map

The ROM in the flash memory version is separated between a user ROM area and a boot ROM area. Figures 1.20.1 and 1.20.2 show the block diagram of flash memory.

The user ROM area is divided into several blocks, so that memory can be erased one block at a time. The user ROM area can be rewritten in all of CPU rewrite, standard serial input/output, and parallel input/output modes.

The boot ROM area is reserved. The rewrite control program for standard serial I/O mode is stored in this area before shipment, so that the area cannot be rewritten.

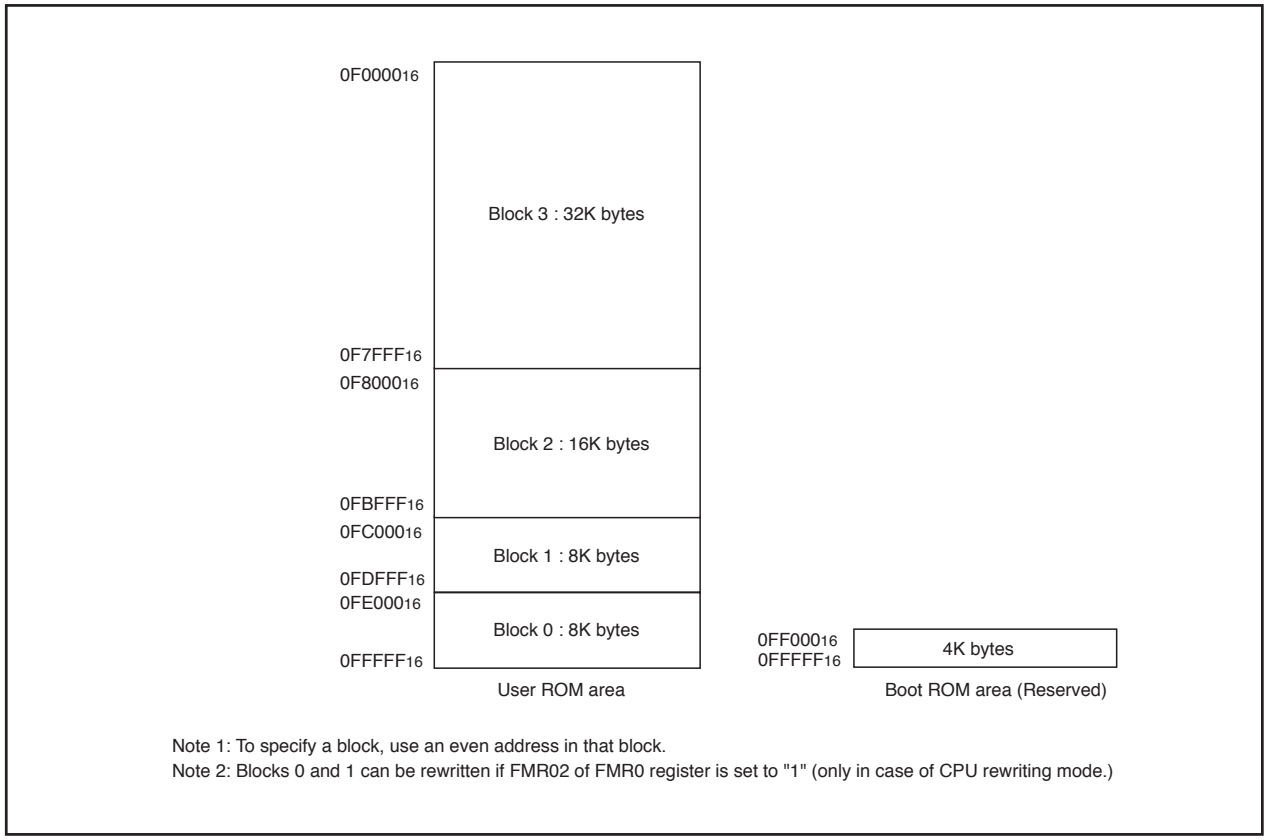


Figure 1.20.1. Flash Memory Block Diagram (ROM Capacity 64K bytes)

CPU Rewrite Mode

In CPU rewrite mode, the user ROM area can be rewritten by executing software commands from the CPU. Therefore, the user ROM area can be rewritten directly while the microcomputer is mounted on-board without having to use a ROM programmer, etc.

Make sure the Program and the Block Erase commands are executed only on each block in the user ROM area.

During CPU rewrite mode, the user ROM area be operated on in either Erase Write 0 (EW0) mode or Erase Write 1 (EW1) mode. Table 1.21.1 lists the differences between Erase Write 0 (EW0) and Erase Write 1 (EW1) modes.

Table 1.21.1. EW0 Mode and EW1 Mode

Item	EW0 mode	EW1 mode
Operation mode	• Single chip mode	Single chip mode
Areas in which a rewrite control program can be located	• User ROM area	User ROM area
Areas in which a rewrite control program can be executed	Must be transferred to any area other than the flash memory (e.g., RAM) before being executed	Can be executed directly in the user ROM area
Areas which can be rewritten	User ROM area	User ROM area However, this does not include the area in which a rewrite control program exists
Software command limitations	None	• Program, Block Erase command Cannot be executed on any block in which a rewrite control program exists • Read Status Register command Cannot be executed
Modes after Program or Erase	Read Status Register mode	Read Array mode
CPU status during Auto Write and Auto Erase	Operating	Hold state (I/O ports retain the state in which they were before the command was executed)(Note)
Flash memory status detection	• Read the FMR0 register's FMR00, FMR06, and FMR07 bits in a program • Execute the Read Status Register command to read the status register's SR7, SR5, and SR4 flags.	Read the FMR0 register's FMR00, FMR06, and FMR07 bits in a program

Note: Make sure no interrupts (except $\overline{\text{NMI}}$ and watchdog timer interrupts) and DMA transfers will occur.

- **EW0 Mode**

  The microcomputer is placed in CPU rewrite mode by setting the FMR0 register's FMR01 bit to "1" (CPU rewrite mode enabled), ready to accept commands. In this case, because the FMR1 register's FMR11 bit = 0, EW0 mode is selected. The FMR01 bit can be set to "1" by writing "0" and then "1" in succession.
  Use software commands to control program and erase operations. Read the FMR0 register or status register to check the status of program or erase operation at completion.

- **EW1 Mode**

  EW1 mode is selected by setting FMR11 bit to "1" (by writing "0" and then "1" in succession) after setting the FMR01 bit to "1" (by writing "0" and then "1" in succession).
  Read the FMR0 register to check the status of program or erase operation at completion. The status register cannot be read during EW1 mode.
  When an erase/program operation is initiated the CPU halts all program execution until the operation is completed.

Software Commands

Software commands are described below. The command code and data must be read and written in 16-bit units, to and from even addresses in the user ROM area. When writing command code, the 8 high-order bits (D_{1t}–D_8) are ignored.

Table 1.21.2. Software Commands

Command	First bus cycle			Second bus cycle		
	Mode	Address	Data (D_0 to D_7)	Mode	Address	Data (D_0 to D_7)
Read array	Write	X	xxFF16			
Read status register	Write	X	xx7016	Read	X	SRD
Clear status register	Write	X	xx5016			
Program	Write	WA	xx4016	Write	WA	WD
Block erase	Write	X	xx2016	Write	BA	xxD016

SRD: Status register data (D_7 to D_0)
WA: Write address (Make sure the address value specified in the first bus cycle is the same even address as the write address specified in the second bus cycle.)
WD: Write data (16 bits)
BA: Uppermost block address (even address, however)
X: Any even address in the user ROM area
x: High-order 8 bits of command code (ignored)

Read Array Command (FF16)

This command reads the flash memory.

Writing 'xxFF16' in the first bus cycle places the microcomputer in read array mode. Enter the read address in the next or subsequent bus cycles, and the content of the specified address can be read in 16-bit units.

Because the microcomputer remains in read array mode until another command is written, the contents of multiple addresses can be read in succession.

However, when you use Read array command immediately after a program command, please read data in the following procedure.

(1) FF16, FF16, FF16, and FF16 are written to 4 arbitrary continuous addresses.
(2) The head address of (1) is specified in Read array mode.
(3) (2) is repeated until the read value and FFFF16 are in agreement.
(4) The head address +2 of (1) is specified.
(5) (4) is repeated until the read value and FFFF16 are in agreement.
(6) Arbitrary addresses are specified.

Read Status Register Command (7016)

This command reads the status register.

Write 'xx7016' in the first bus cycle, and the status register can be read in the second bus cycle. (Refer to "Status Register.") When reading the status register too, specify an even address in the user ROM area.

Do not execute this command in EW1 mode.

(b)ADC (Analog to Digital Converter)

The output of the channel filter (maximum of three) connected outside is connected to 1 bit ADC which consists of an operational amplifier and a comparator.

M16C/6S build in ADC of three equivalent performances. The circuit of one ADC has composition shown in the following figure.

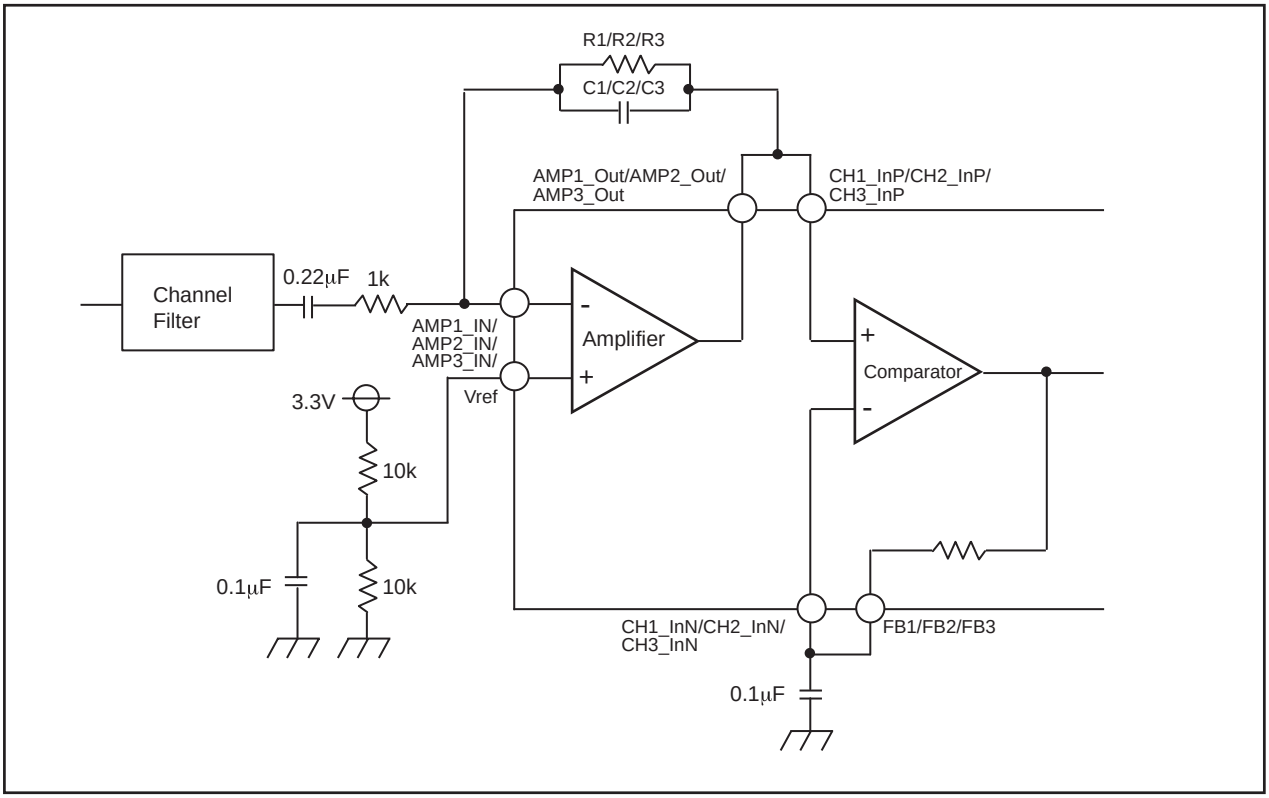


Figure 1.23.4 ADC block diagram

Three ADC use all three by the signal zone to be used. The constant of a filter is set up as shown in the following table.

Table 1.23.3 The example of an ADC part constant setting

Parts	Application characteristic	C1	C2	C3	Unit
Capacitor	FCC/ARIB	33	22	10	pF
	CECLEC-B	33	—	—	pF
Parts	Application characteristic	R1	R2	R3	Unit
Register	FCC/ARIB	10	10	12	kΩ
	CECLEC-B	10	—	—	kΩ

Serial I/O

Clock-Synchronous Serial I/O

Transmission/reception

With an external clock selected, and choosing the $\overline{\text{RTS}}$ function, the output level of the $\overline{\text{RTSi}}$ pin goes to "L" when the data-receivable status becomes ready, which informs the transmission side that the reception has become ready. The output level of the $\overline{\text{RTSi}}$ pin goes to "H" when reception starts. So if the $\overline{\text{RTSi}}$ pin is connected to the $\overline{\text{CTSi}}$ pin on the transmission side, the circuit can transmission and reception data with consistent timing. With the internal clock, the $\overline{\text{RTS}}$ function has no effect.

Transmission

When an external clock is selected, the conditions must be met while if the CKPOL bit in the UiC0 register is set to 0 (transmit data output at the falling edge and the receive data taken in at the rising edge of the transfer clock), the external clock is in the high state; if the CKPOL bit in the UiC0 register is set to 1 (transmit data output at the rising edge and the receive data taken in at the falling edge of the transfer clock), the external clock is in the low state.

- The TE bit in UiC1 register is set to 1 (transmission enabled)
- The TI bit in UiC1 register is set to 0 (data present in UiTB register)
- If $\overline{\text{CTS}}$ function is selected, input on the $\overline{\text{CTSi}}$ pin is set to "L"

Reception

In operating the clock-synchronous serial I/O, operating a transmitter generates a shift clock. Fix settings for transmission even when using the device only for reception. Dummy data is output to the outside from the TxDi pin when receiving data.

When an internal clock is selected, set the TE bit in the UiC1 register (i = 0 to 2) to 1 (transmission enabled) and write dummy data to the UiTB register, and the shift clock will thereby be generated. When an external clock is selected, set the TE bit in the UiC1 register (i = 0 to 2) to 1 and write dummy data to the UiTB register, and the shift clock will be generated when the external clock is fed to the CLKi input pin.

When successively receiving data, if all bits of the next receive data are prepared in the UARTi receive register while the RE bit in the UiC1 register (i = 0 to 2) is set to 1 (data present in the UiRB register), an overrun error occurs and the UiRB register OER bit is set to 1 (overrun error occurred). In this case, because the content of the UiRB register is undefined, a corrective measure must be taken by programs on the transmit and receive sides so that the valid data before the overrun error occurred will be retransmitted. Note that when an overrun error occurred, the SiRIC register IR bit does not change state.

To receive data in succession, set dummy data in the lower-order byte of the UiTB register every time reception is made.

When an external clock is selected, make sure the external clock is in high state if the CKPOL bit is set to 0, and in low state if the CKPOL bit is set to 1 before the following conditions are met:

- The RE bit in the UiC1 register is set to 1 (reception enabled)
- The TE bit in the UiC1 register is set to 1 (transmission enabled)
- The TI bit in the UiC1 register= 0 (data present in the UiTB register)

Flash Memory Version

Functions to Inhibit Rewriting Flash Memory Rewrite

ID codes are stored in addresses 0FFFDF16, 0FFFE316, 0FFFEB16, 0FFFEF16, 0FFFF316, 0FFFF716, and 0FFFFB16. If wrong data are written to theses addresses, the flash memory cannot be read or written in standard serial I/O mode.

The ROMCP register is mapped in address 0FFFFF16. If wrong data is written to this address, the flash memory cannot be read or written in parallel I/O mode.

In the flash memory version of MCU, these addresses are allocated to the vector addresses (“H”) of fixed vectors. The b3 to b0 in address 0FFFFF16 are reserved bits. Set these bits to 11112.

Regarding Programming/Erasure Times and Execution Time

As the number of programming/erasure times increases, so does the execution time for software commands (Program, and Block Erase).

The software commands are aborted by hardware reset 1, brown-out detection reset (hardware reset 2), and watchdog timer interrupt. If a software command is aborted by such reset or interrupt, the affected block must be erased before reexecuting the aborted command.

Definition of Programming/Erasure Times

“Number of programs and erasure” refers to the number of erasure per block.

If the number of program and erasure is n (n=100) each block can be erased n times. For example, if a 8K byte block A is erased after writing 1 word data 4096 times, each to a different address, this is counted as one program and erasure. However, data cannot be written to the same address more than once without erasing the block. (Rewrite prohibited)

Boot Mode

An undefined value is sometimes output in the I/O port until the internal power supply becomes stable when “H” is applied to the CNVSS pin and “L” is applied to the $\overline{\text{RESET}}$ pin.

When setting the CNVSS pin to “H”, the following procedure is required:

(1) Apply an “L” signal to the $\overline{\text{RESET}}$ pin and the CNVSS pin.
(2) Bring VCC to more than 2.7V, and wait at least 2 msec. (Internal power supply stable waiting time)
(3) Apply an “H” signal to the CNVSS pin.
(4) Apply an “H” signal to the $\overline{\text{RESET}}$ pin.

When the CNVSS pin is “H” and $\overline{\text{RESET}}$ pin is “L”, P67 pin is connected to the pull-up resister.