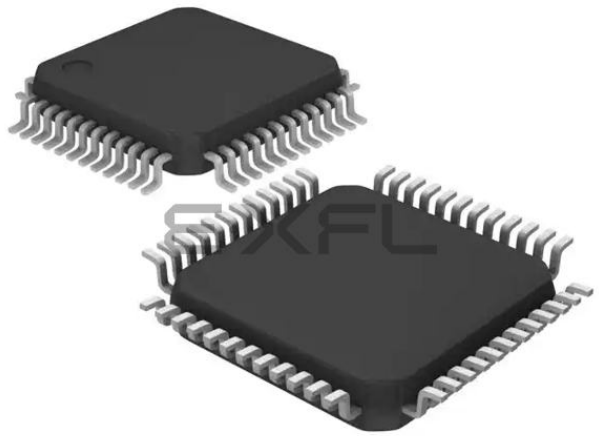


Welcome to E-XFL.COM

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	Not For New Designs
Core Processor	78K/0
Core Size	8-Bit
Speed	10MHz
Connectivity	LINbus, UART/USART
Peripherals	LCD, LVD, POR, PWM, WDT
Number of I/O	30
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 6x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/upd78f0413ga-gam-ax

(4) Port function register 1 (PF1)

This register sets the pin functions of P13/TxD0/KR4/<TxD6> pins.

PF1 is set using a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets PF1 to 00H.

Remark The functions within arrowheads (< >) can be assigned by setting the input switch control register (ISC).

Figure 4-19. Format of Port Function Register 1 (PF1)

Address: FF20H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
PF1	0	0	0	0	PF13	0	0	0

PF13	Port (P13), key interrupt (KR4), UART0, and UART6 output specification
0	Used as P13 or KR4
1	Used as TxD0 or TxD6

(5) Port function register 2 (PF2)

This register sets whether to use pins P20 to P25 as port pins (other than segment output pins) or segment output pins.

PF2 is set using a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets PF2 to 00H.

Figure 4-20. Format of Port Function Register 2 (PF2)

Address: FFB5H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
PF2	0	0	PF25	PF24	PF23	PF22	PF21	PF20

PF2n	Port/segment output specification
0	Used as port (other than segment output)
1	Used as segment output

Remark n = 0 to 5

(7) A/D port configuration register 0 (ADPC0) (μPD78F041x only)

This register switches the P20/ANI0 to P25/ANI5 pins to analog input of A/D converter or digital I/O of port.

ADPC0 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 08H.

Caution Set the values shown in Figure 4-22 after the reset is released.

Figure 4-22. Format of A/D Port Configuration Register 0 (ADPC0)

Address: FF8FH After reset: 08H R/W

Symbol	7	6	5	4	3	2	1	0
ADPC0	0	0	0	0	0	ADPC02	ADPC01	ADPC00

ADPC02	ADPC01	ADPC00	Digital I/O (D)/analog input (A) switching					
			P25 /ANI5	P24 /ANI4	P23 /ANI3	P22 /AN2	P21 /ANI1	P20 /ANI0
0	0	0	A	A	A	A	A	A
0	0	1	A	A	A	A	A	D
0	1	0	A	A	A	A	D	D
0	1	1	A	A	A	D	D	D
1	0	0	A	A	D	D	D	D
1	0	1	A	D	D	D	D	D
1	1	0	D	D	D	D	D	D
Other than above			Setting prohibited					

Cautions 1. Set the channel used for A/D conversion to the input mode by using port mode register 2 (PM2).

2. The pin to be set as a digital I/O via ADPC, must not be set via ADS, ADDS1 or ADDS0.

3. If data is written to ADPC0, a wait cycle is generated. Do not write data to ADPC0 when the CPU is operating on the subsystem clock and the peripheral hardware clock is stopped. For details, see CHAPTER 29 CAUTIONS FOR WAIT.

4. If pins ANI0/P20/SEG21 to ANI5/P25/SEG16 are set to segment output via the PF2 register, output is set to segment output, regardless of the ADPC0 setting (for μPD78F041x only).

(5) Input switch control register (ISC)

The input source to TI000 becomes the input signal from the P33/TI000 pin, by setting ISC1 to 0.

ISC can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets ISC to 00H.

Figure 6-10. Format of Input Switch Control Register (ISC)

Address: FF4FH After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
ISC	0	0	ICS5	ICS4	ICS3	ICS2	ICS1	ICS0

ICS5	ICS4	TxD6, RxD6 input source selection
0	0	TxD6:P112, RxD6: P113
1	0	TxD6:P13, RxD6: P12
Other than above		Setting prohibited

ISC3	RxD6/P113 input enabled/disabled
0	RxD6/P113 input disabled
1	RxD6/P113 input enabled

ISC2	TI52 input source control
0	No enable control of TI52 input (P34)
1	Enable controlled of TI52 input (P34)Note 1

ISC1	TI000 input source selection
0	TI000 (P33)
1	RxD6 (P12 or P113Note 2)

ISC0	INTP0 input source selection
0	INTP0 (P120)
1	RxD6 (P12 or P113Note 2)

Notes 1. TI52 input is controlled by TOH2 output signal.

2. TI000 and INTP0 inputs are selected by ISC5 and ISC4.

CHAPTER 7 8-BIT TIMER/EVENT COUNTERS 50, 51, AND 52

7.1 Functions of 8-Bit Timer/Event Counters 50, 51, and 52

8-bit timer/event counters 50, 51 and 52 have the following functions.

- Interval timer
- External event counter[Note]

Note TM52 only. TM52 and TM00 can be connected in cascade to be used as an external 24-bit event counter. Also, the external event input of TM52 can be input enable-controlled via TMH2. For detail, see **CHAPTER 6 16-BIT TIMER/EVENT COUNTER 00.**

7.2 Configuration of 8-Bit Timer/Event Counters 50, 51, and 52

8-bit timer/event counters 50, 51, and 52 include the following hardware.

Table 7-1. Configuration of 8-Bit Timer/Event Counters 50, 51, and 52

Item	Configuration
Timer register	8-bit timer counter 5n (TM5n)
Register	8-bit timer compare register 5n (CR5n)
Timer input	TI5n
Control registers	Timer clock selection register 5n (TCL5n) 8-bit timer mode control register 5n (TMC5n) Input switch control register (ISC) Port mode register 3 (PM3) Port register 3 (P3)

Remark n = 0 to 2

Figures 7-1 to 7-3 show the block diagrams of 8-bit timer/event counters 50, 51, and 52.

(2) 8-bit timer mode control register 5n (TMC5n)

TMC5n is a register that controls the count operation of 8-bit timer counter 5n (TM5n).

TMC5n can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets this register to 00H.

Remark n = 0 to 2

Figure 7-9. Format of 8-Bit Timer Mode Control Register 50 (TMC50)

Address: FF6BH After reset: 00H R/W[Note]

Symbol	<7>	6	5	4	<3>	<2>	1	0
TMC50	TCE50	0	0	0	LVS50	LVR50	TMC501	0

TCE50	TM50 count operation control
0	After clearing to 0, count operation disabled (counter stopped)
1	Count operation start

LVS50	LVR50	Timer output F/F status setting
0	0	No change
0	1	Timer output F/F clear (0) (default value of TM50 output: low level)
1	0	Timer output F/F set (1) (default value of TM50 output: high level)
1	1	Setting prohibited

TMC501	Timer F/F control
0	Inversion operation disabled
1	Inversion operation enabled

Note Bits 2 and 3 are write-only.

Cautions 1. Be sure to clear bits 0, and 4 to 6 to 0.

2. Perform <1> to <3> below in the following order, not at the same time.

<1> Set TMC501: Operation mode setting

<2> Set LVS50, LVR50: Timer F/F setting

<4> Set TCE50

Remark If LVS50 and LVR50 are read, the value is 0.

(4) Port mode registers 3 (PM3)

These registers set port 3 input/output in 1-bit units.

When using the P34/TI52/TI010/TO00/RTC1HZ/INTP1 pins for timer input, set PM34 to 1. The output latch of PM34 at this time may be 0 or 1.

PM3 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets these registers to FFH.

Figure 7-13. Format of Port Mode Register 3 (PM3)

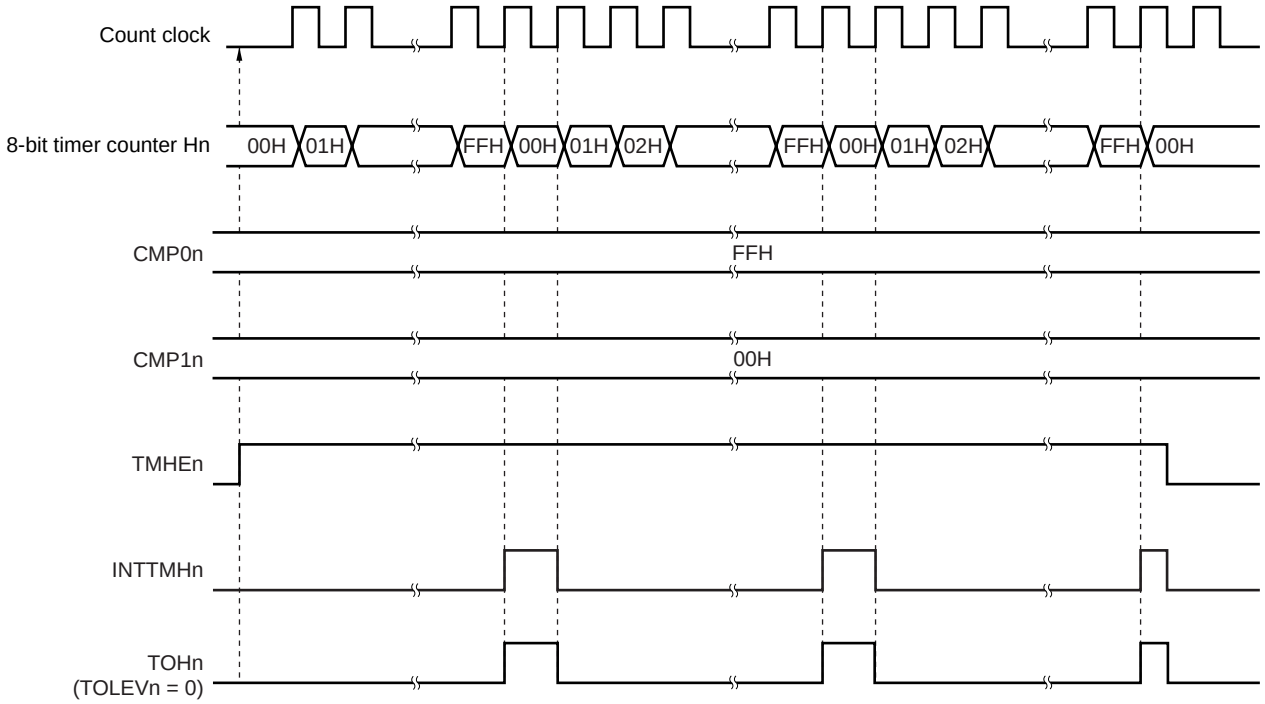
Address: FF23H After reset: FFH R/W

Symbol	7	6	5	4	3	2	1	0
PM3	1	1	1	PM34	PM33	PM32	PM31	1

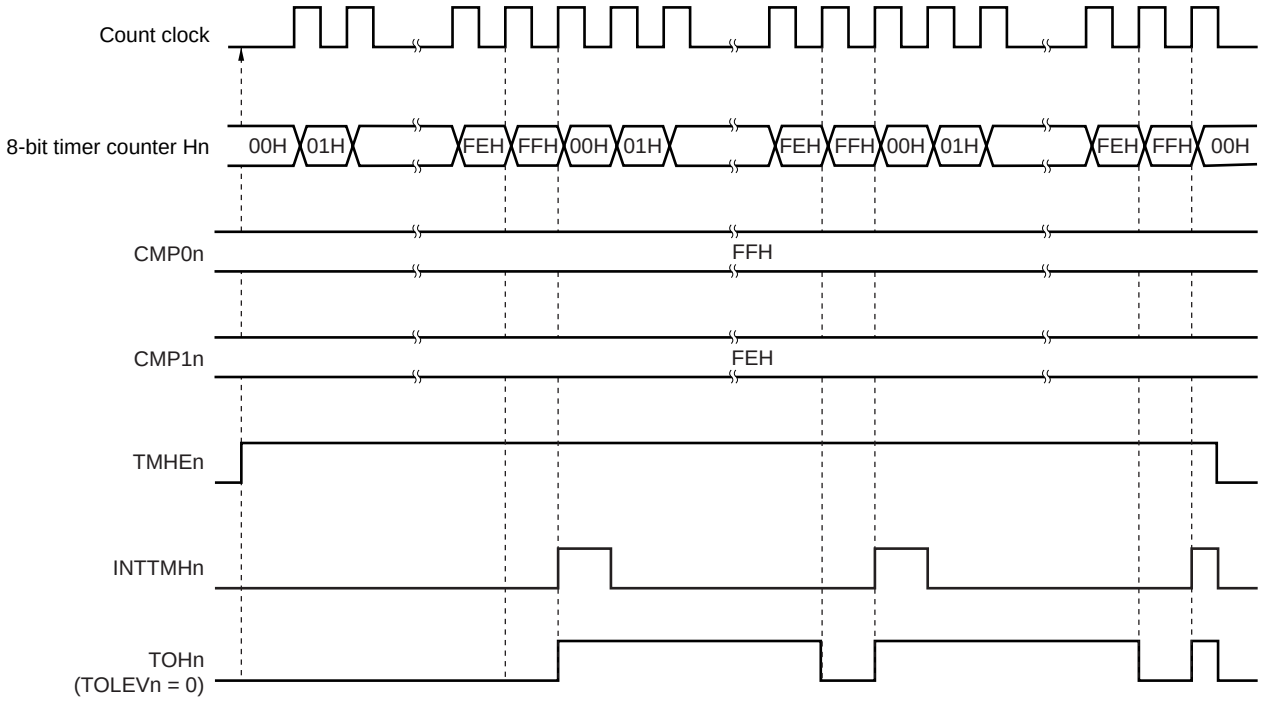
PM3n	P1n pin I/O mode selection (n = 1 to 4)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

Figure 8-14. Operation Timing in PWM Output Mode (2/4)

(b) Operation when CMP0n = FFH, CMP1n = 00H

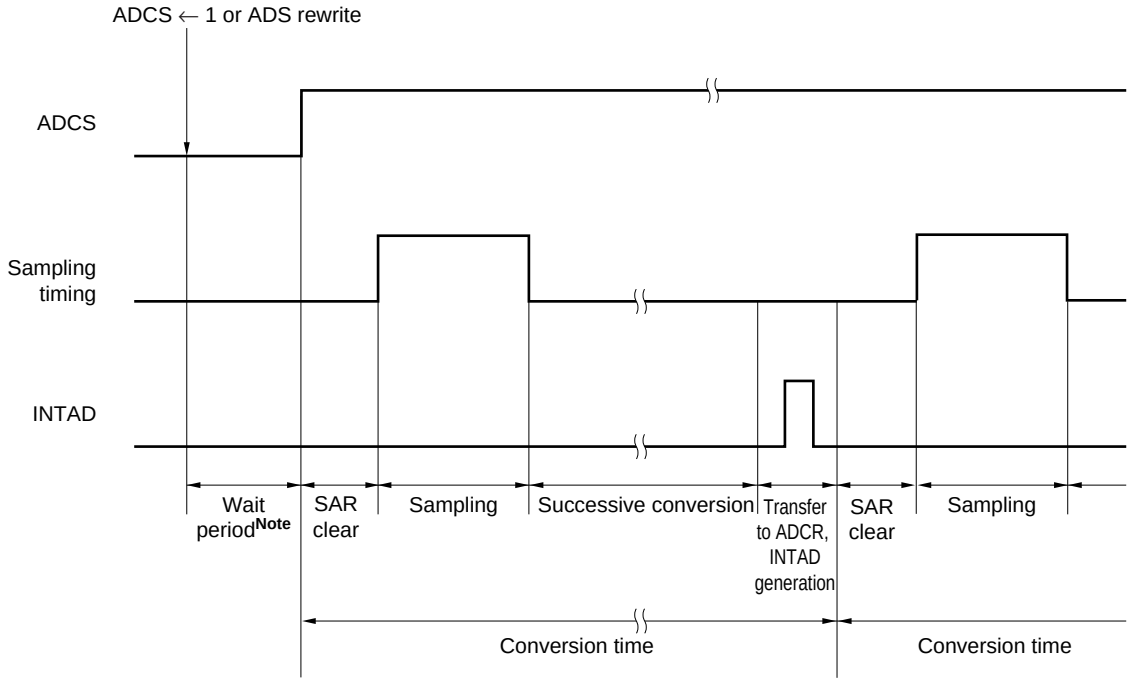


(c) Operation when CMP0n = FFH, CMP1n = FEH



Remark n = 0 to 2, however, TOH0 and TOH1 only for TOHn

Figure 12-5. A/D Converter Sampling and A/D Conversion Timing



Note For details of wait period, see **CHAPTER 29 CAUTIONS FOR WAIT**.

(2) 10-bit A/D conversion result register (ADCR)

This register is a 16-bit register that stores the A/D conversion result. The lower 6 bits are fixed to 0. Each time A/D conversion ends, the conversion result is loaded from the successive approximation register. The higher 8 bits of the conversion result are stored in FF07H and the lower 2 bits are stored in the higher 2 bits of FF06H.

ADCR can be read by a 16-bit memory manipulation instruction.

Reset signal generation clears this register to 0000H.

Figure 12-6. Format of 10-Bit A/D Conversion Result Register (ADCR)

Address: FF06H, FF07H After reset: 0000H R

Symbol	FF07H								FF06H							
ADCR											0	0	0	0	0	0

Cautions 1. When writing to the A/D converter mode register (ADM), analog input channel specification register (ADS), and A/D port configuration register 0 (ADPC0), the contents of ADCR may become undefined. Read the conversion result following conversion completion before writing to ADM, ADS, and ADPC0. Using timing other than the above may cause an incorrect conversion result to be read.

2. If data is read from ADCR, a wait cycle is generated. Do not read data from ADCR when the CPU is operating on the subsystem clock and the peripheral hardware clock is stopped. For details, see CHAPTER 29 CAUTIONS FOR WAIT.

(6) Port mode register 2 (PM2)

When using the ANI0/P20 to ANI5/P25 pins for analog input port, set PM20 to PM25 to 1. The output latches of P20 to P25 at this time may be 0 or 1.

If PM20 to PM25 are set to 0, they cannot be used as analog input port pins.

PM2 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets this register to FFH.

Figure 12-10. Format of Port Mode Register 2 (PM2)

Address: FF8FH After reset: 08H R/W

Symbol	7	6	5	4	3	2	1	0
PM2	1	1	PM25	PM24	PM23	PM22	PM21	PM20

PM2n	P2n pin I/O mode selection (n = 0 to 5)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

ANI0/P20 to ANI5/P25 pins are as shown below depending on the settings of PF2, ADPC0, PM2, and ADS.

Table 12-3. Setting Functions of P20/ANI0 to P25/ANI5 Pins

PF2	ADPC0	PM2	ADS	P20/SEG21/ANI0 to P25/SEG16/ANI5 Pins
Digital/Analog selection	Analog input selection	Input mode	Does not select ANI.	Analog input (not to be converted)
			Selects ANI.	Analog input (to be converted by successive approximation type A/D converter)
		Output mode	–	Setting prohibited
	Digital I/O selection	Input mode	–	Digital input
		Output mode	–	Digital output
SEG output selection	–	–	–	Segment output

13.4.3 Dedicated baud rate generator

The dedicated baud rate generator consists of a source clock selector and a 5-bit programmable counter, and generates a serial clock for transmission/reception of UART0.

Separate 5-bit counters are provided for transmission and reception.

(1) Configuration of baud rate generator

- Base clock

  The clock selected by bits 7 and 6 (TPS01 and TPS00) of baud rate generator control register 0 (BRGC0) is supplied to each module when bit 7 (POWER0) of asynchronous serial interface operation mode register 0 (ASIM0) is 1. This clock is called the base clock and its frequency is called f_{XCLK0}. The base clock is fixed to low level when POWER0 = 0.

- Transmission counter

  This counter stops operation, cleared to 0, when bit 7 (POWER0) or bit 6 (TXE0) of asynchronous serial interface operation mode register 0 (ASIM0) is 0.

  It starts counting when POWER0 = 1 and TXE0 = 1.

  The counter is cleared to 0 when the first data transmitted is written to transmit shift register 0 (TXS0).

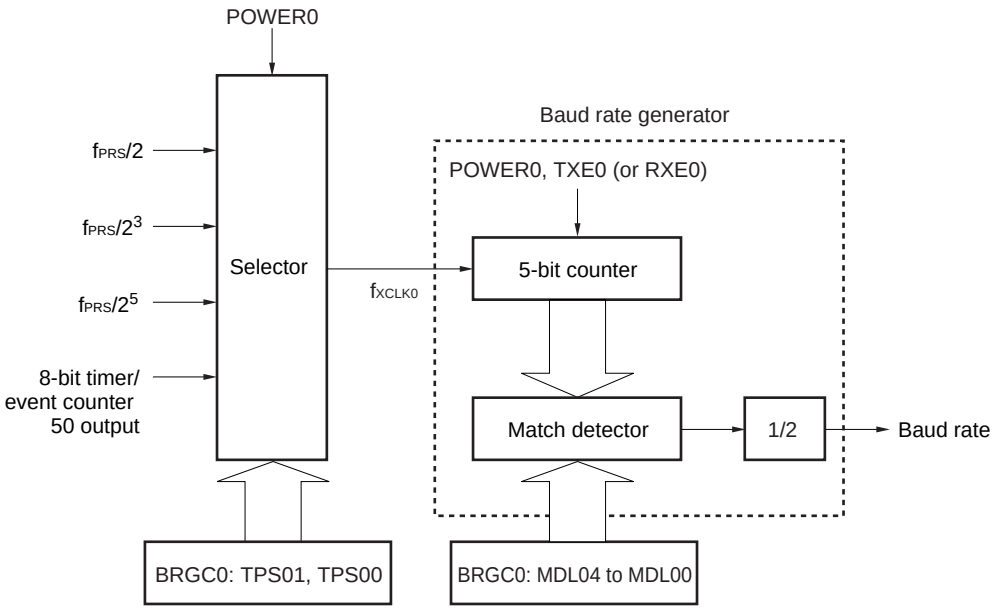
- Reception counter

  This counter stops operation, cleared to 0, when bit 7 (POWER0) or bit 5 (RXE0) of asynchronous serial interface operation mode register 0 (ASIM0) is 0.

  It starts counting when the start bit has been detected.

  The counter stops operation after one frame has been received, until the next start bit is detected.

Figure 13-12. Configuration of Baud Rate Generator



Remark POWER0: Bit 7 of asynchronous serial interface operation mode register 0 (ASIM0)
TXE0: Bit 6 of ASIM0
RXE0: Bit 5 of ASIM0
BRGC0: Baud rate generator control register 0

CHAPTER 15 LCD CONTROLLER/DRIVER

15.1 Functions of LCD Controller/Driver

The functions of the LCD controller/driver in the 78K0/LC3 are as follows.

(1) The LCD driver voltage generator can switch external resistance division and internal resistance division.
(2) Automatic output of segment and common signals based on automatic display data memory read
(3) Six different display modes:
- Static
- 1/2 duty (1/2 bias)
- 1/3 duty (1/2 bias)
- 1/3 duty (1/3 bias)
- 1/4 duty (1/3 bias)
- 1/8 duty (1/4 bias)

(4) Six different frame frequencies, selectable in each display mode
(5) Segment signal outputs: 22[Note] (SEG0 to SEG21), Common signal outputs: 8 [Note] (COM0 to COM7)

Note The four segment signal outputs (SEG0 to SEG3) and four common signal outputs (COM4 to COM7) are alternate-function pins. COM4 to COM7 can be used only when eight-time-slice mode is selected by the setting of the LCD display mode register (LCDM).

Figure 15-1. Block Diagram of LCD Controller/Driver

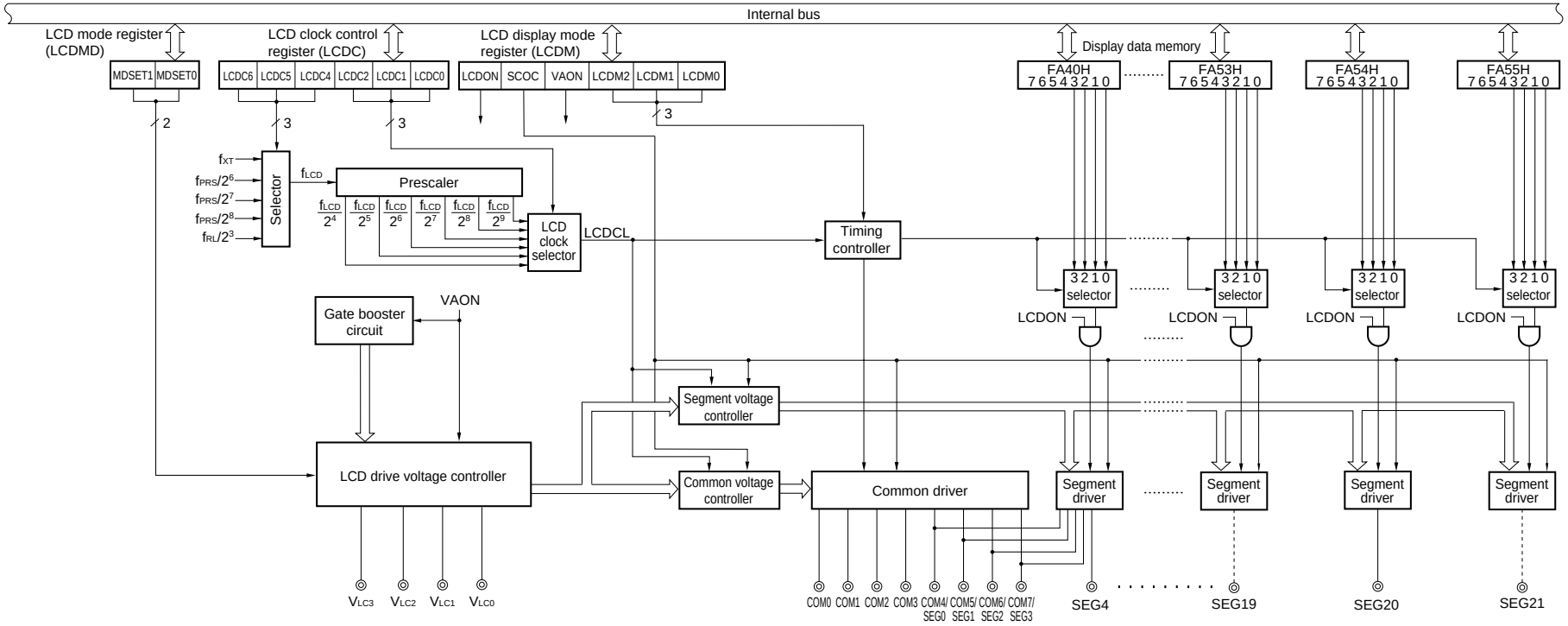
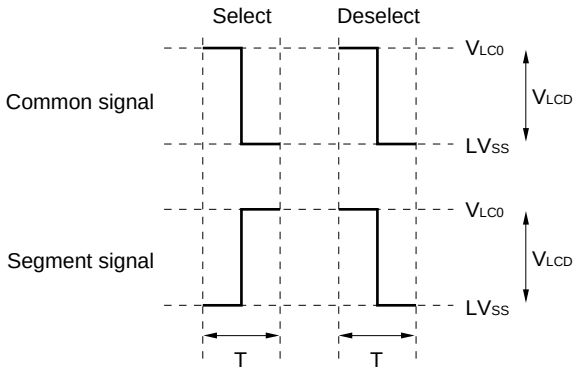


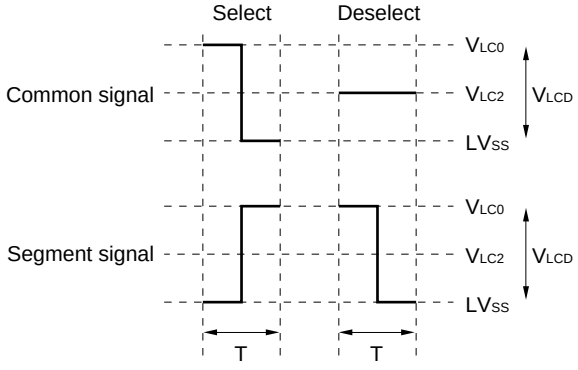
Figure 15-9. Voltages and Phases of Common and Segment Signals

(a) Static display mode



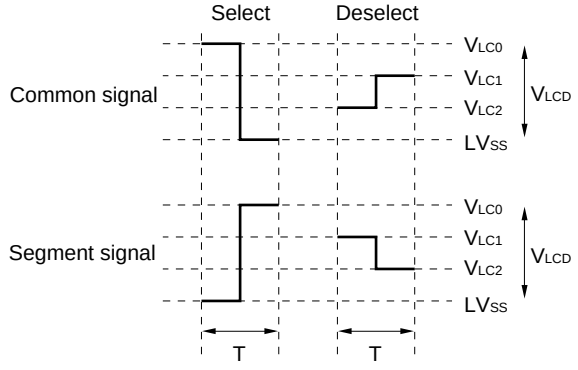
T: One LCD clock period

(b) 1/2 bias method



T: One LCD clock period

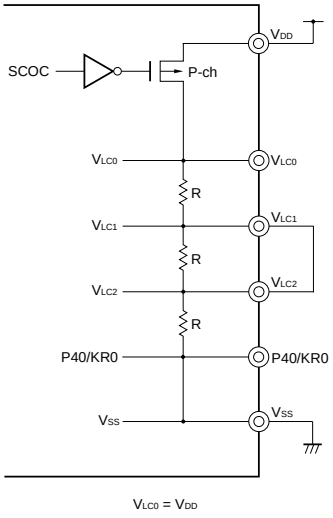
(c) 1/3 bias method



T: One LCD clock period

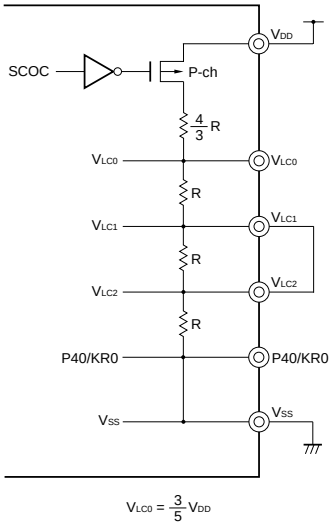
Figure 15-23. Examples of LCD Drive Power Connections (Internal Resistance Division Method) (2/2)

(c) 1/2 bias method
(MDSET1, MDSET0 = 0, 1)
(example of V_{DD} = 5 V, V_{LC0} = 5 V)



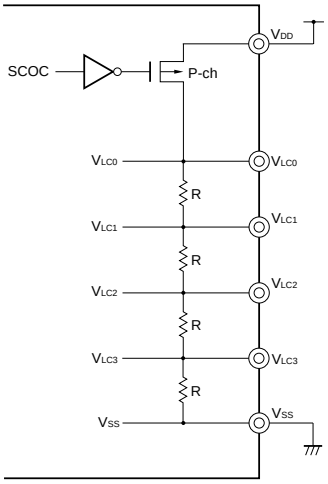
$V_{LC0} = V_{DD}$

(d) 1/4 bias method
(MDSET1, MDSET0 = 1, 1)
(example of V_{DD} = 5 V, V_{LC0} = 3 V)



$V_{LC0} = \frac{3}{5} V_{DD}$

(e) 1/4 bias method
(MDSET1, MDSET0 = 0, 1)
(example of V_{DD} = 5 V, V_{LC0} = 5 V)



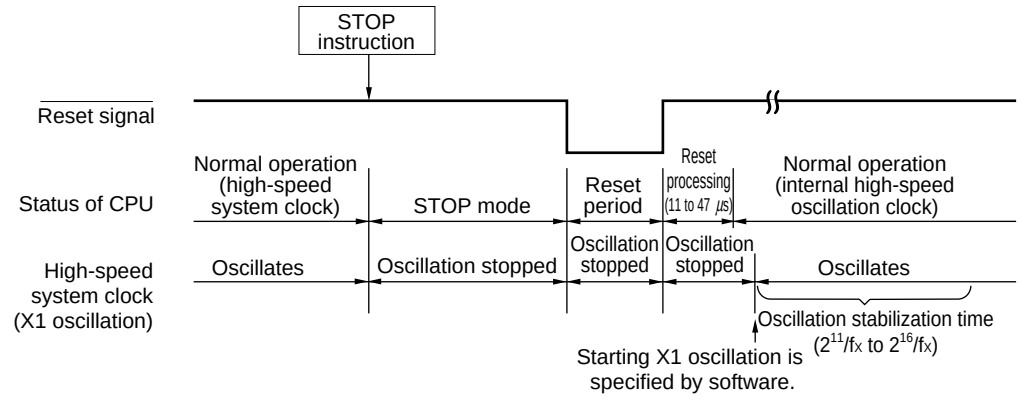
$V_{LC0} = V_{DD}$

(b) Release by reset signal generation

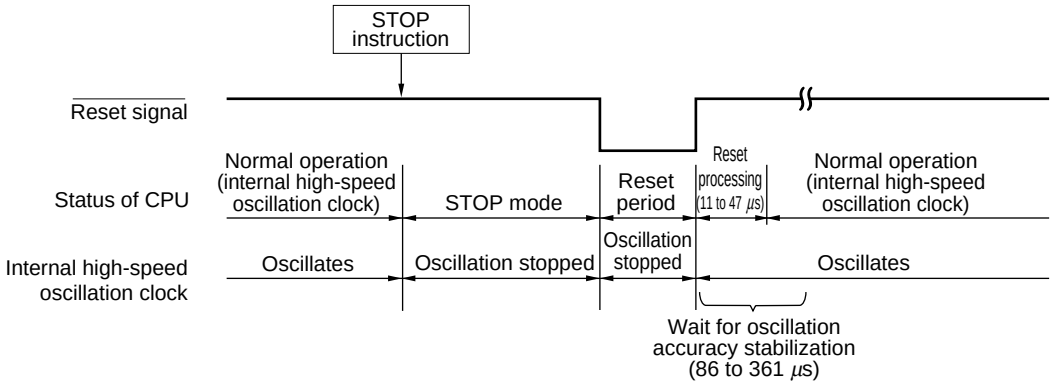
When the reset signal is generated, STOP mode is released, and then, as in the case with a normal reset operation, the program is executed after branching to the reset vector address.

Figure 19-7. STOP Mode Release by Reset

(1) When high-speed system clock is used as CPU clock



(2) When internal high-speed oscillation clock is used as CPU clock



Remark fx: X1 clock oscillation frequency

Table 19-4. Operation in Response to Interrupt Request in STOP Mode

Release Source	MK××	PR××	IE	ISP	Operation
Maskable interrupt request	0	0	0	×	Next address instruction execution
	0	0	1	×	Interrupt servicing execution
	0	1	0	1	Next address instruction execution
	0	1	×	0	
	0	1	1	1	Interrupt servicing execution
	1	×	×	×	STOP mode held
Reset	–	–	×	×	Reset processing

×: don't care

24.5 Connection of Pins on Board

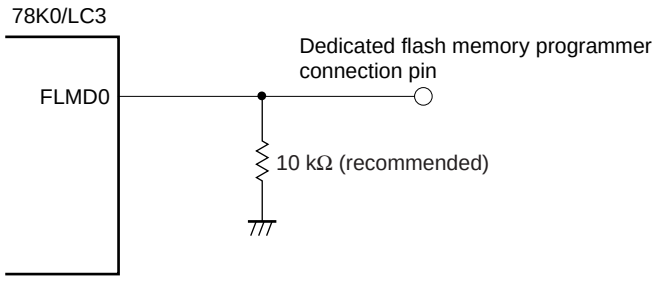
To write the flash memory on-board, connectors that connect the dedicated flash memory programmer must be provided on the target system. First provide a function that selects the normal operation mode or flash memory programming mode on the board.

When the flash memory programming mode is set, all the pins not used for programming the flash memory are in the same status as immediately after reset. Therefore, if the external device does not recognize the state immediately after reset, the pins must be handled as described below.

24.5.1 FLMD0 pin

In the normal operation mode, 0 V is input to the FLMD0 pin. In the flash memory programming mode, the V_{DD} write voltage is supplied to the FLMD0 pin. An FLMD0 pin connection example is shown below.

Figure 24-5. FLMD0 Pin Connection Example



24.5.2 Serial interface pins

The pins used by each serial interface are listed below.

Table 24-4. Pins Used by Each Serial Interface

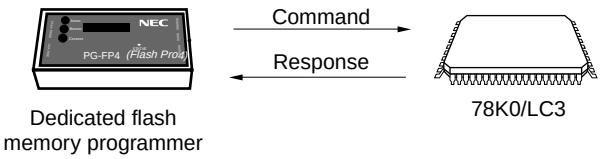
Serial Interface	Pins Used
UART6	TxD6, RxD6

To connect the dedicated flash memory programmer to the pins of a serial interface that is connected to another device on the board, care must be exercised so that signals do not collide or that the other device does not malfunction.

24.6.4 Communication commands

The 78K0/LC3 communicates with the dedicated flash memory programmer by using commands. The signals sent from the flash memory programmer to the 78K0/LC3 are called commands, and the signals sent from the 78K0/LC3 to the dedicated flash memory programmer are called response.

Figure 24-11. Communication Commands



The flash memory control commands of the 78K0/LC3 are listed in the table below. All these commands are issued from the programmer and the 78K0/LC3 perform processing corresponding to the respective commands.

Table 24-7. Flash Memory Control Commands

Classification	Command Name	Function
Verify	Verify	Compares the contents of a specified area of the flash memory with data transmitted from the programmer.
Erase	Chip Erase	Erases the entire flash memory.
	Block Erase	Erases a specified area in the flash memory.
Blank check	Block Blank Check	Checks if a specified block in the flash memory has been correctly erased.
Write	Programming	Writes data to a specified area in the flash memory.
Getting information	Status	Gets the current operating status (status data).
	Silicon Signature	Gets 78K0/Lx3 information (such as the part number and flash memory configuration).
	Version Get	Gets the 78K0/Lx3 version and firmware version.
	Checksum	Gets the checksum data for a specified area.
Security	Security Set	Sets security information.
Others	Reset	Used to detect synchronization status of communication.
	Oscillating Frequency Set	Specifies an oscillation frequency.

The 78K0/LC3 return a response for the command issued by the dedicated flash memory programmer. The response names sent from the 78K0/LC3 are listed below.

Table 24-8. Response Names

Response Name	Function
ACK	Acknowledges command/data.
NAK	Acknowledges illegal command/data.

Table 24-9. Relationship Between Enabling Security Function and Command

(1) During on-board/off-board programming

Valid Security	Executed Command		
	Batch Erase (Chip Erase)	Block Erase	Write
Prohibition of batch erase (chip erase)	Cannot be erased in batch	Blocks cannot be erased.	Can be performed[Note].
Prohibition of block erase	Can be erased in batch.		Can be performed.
Prohibition of writing			Cannot be performed.
Prohibition of rewriting boot cluster 0	Cannot be erased in batch	Boot cluster 0 cannot be erased.	Boot cluster 0 cannot be written.

Note Confirm that no data has been written to the write area. Because data cannot be erased after batch erase (chip erase) is prohibited, do not write data if the data has not been erased.

(2) During self programming

Valid Security	Executed Command	
	Block Erase	Write
Prohibition of batch erase (chip erase)	Blocks can be erased.	Can be performed.
Prohibition of block erase		
Prohibition of writing		
Prohibition of rewriting boot cluster 0	Boot cluster 0 cannot be erased.	Boot cluster 0 cannot be written.

Table 24-10 shows how to perform security settings in each programming mode.

Table 24-10. Setting Security in Each Programming Mode

(1) On-board/off-board programming

Security	Security Setting	How to Disable Security Setting
Prohibition of batch erase (chip erase)	Set via GUI of dedicated flash memory programmer, etc.	Cannot be disabled after set.
Prohibition of block erase		Execute batch erase (chip erase) command
Prohibition of writing		
Prohibition of rewriting boot cluster 0		Cannot be disabled after set.

(2) Self programming

Security	Security Setting	How to Disable Security Setting
Prohibition of batch erase (chip erase)	Set by using information library.	Cannot be disabled after set.
Prohibition of block erase		Execute batch erase (chip erase) command during on-board/off-board programming (cannot be disabled during self programming)
Prohibition of writing		
Prohibition of rewriting boot cluster 0		Cannot be disabled after set.

24.8.1 Boot swap function

If rewriting the boot area has failed during self-programming due to a power failure or some other cause, the data in the boot area may be lost and the program may not be restarted by resetting.

The boot swap function is used to avoid this problem.

Before erasing boot cluster 0[Note], which is a boot program area, by self-programming, write a new boot program to boot cluster 1 in advance. When the program has been correctly written to boot cluster 1, swap this boot cluster 1 and boot cluster 0 by using the set information function of the firmware of the 78K0/LC3, so that boot cluster 1 is used as a boot area. After that, erase or write the original boot program area, boot cluster 0.

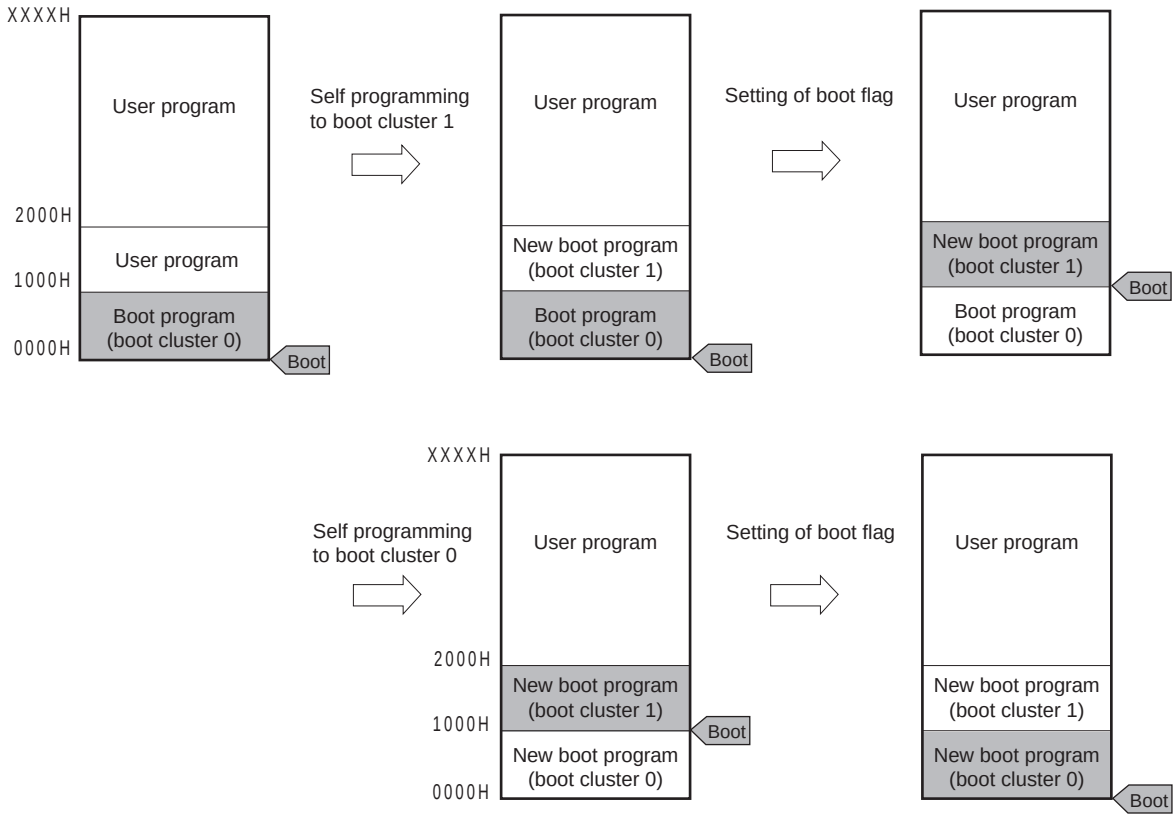
As a result, even if a power failure occurs while the boot programming area is being rewritten, the program is executed correctly because it is booted from boot cluster 1 to be swapped when the program is reset and started next.

If the program has been correctly written to boot cluster 0, restore the original boot area by using the set information function of the firmware of the 78K0/LC3.

Note A boot cluster is a 4 KB area and boot clusters 0 and 1 are swapped by the boot swap function.

Boot cluster 0 (0000H to 0FFFH): Original boot program area
Boot cluster 1 (1000H to 1FFFH): Area subject to boot swap function

Figure 24-13. Boot Swap Function



Remark Boot cluster 1 becomes 0000H to 0FFFH when a reset is generated after the boot flag has been set.