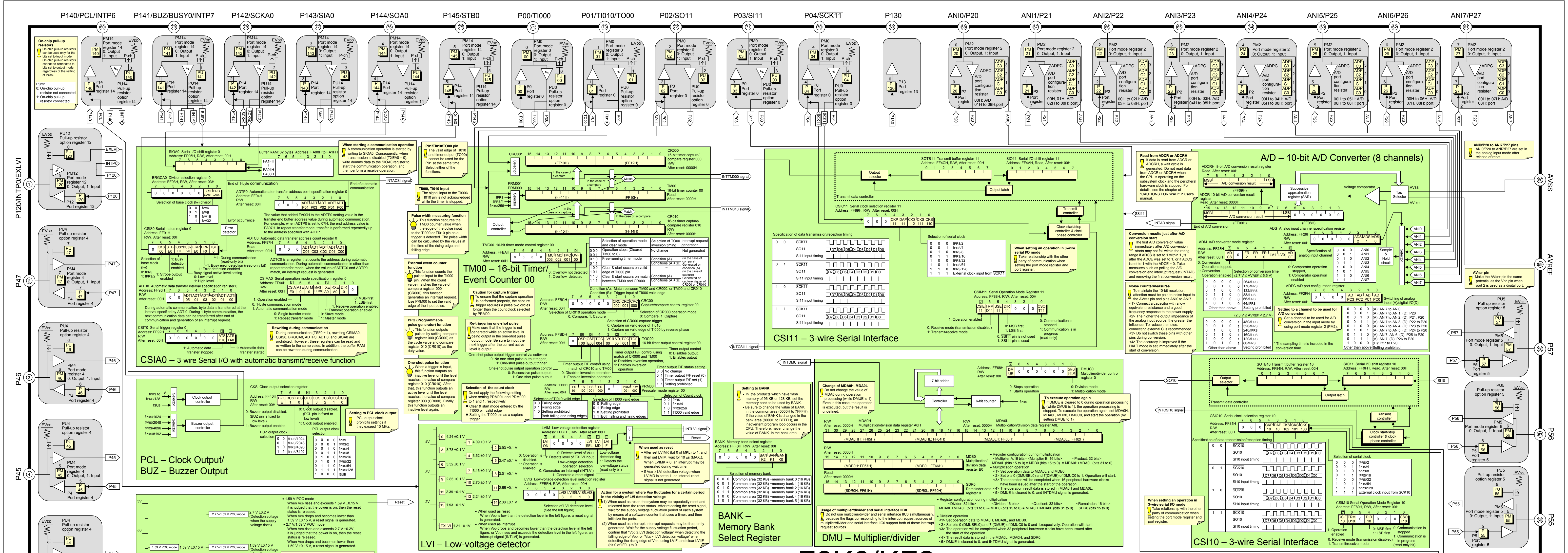
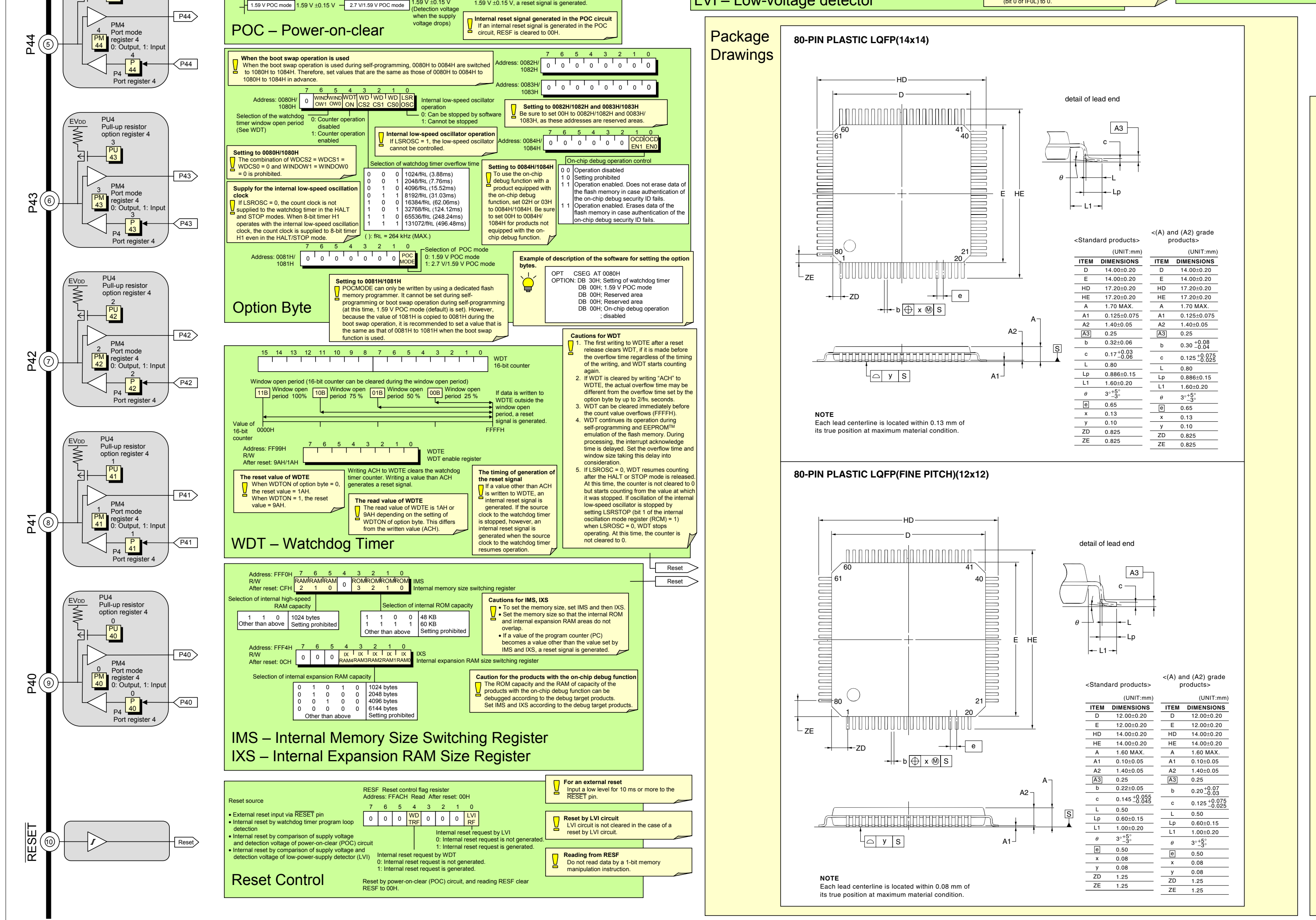




78K0/KF2

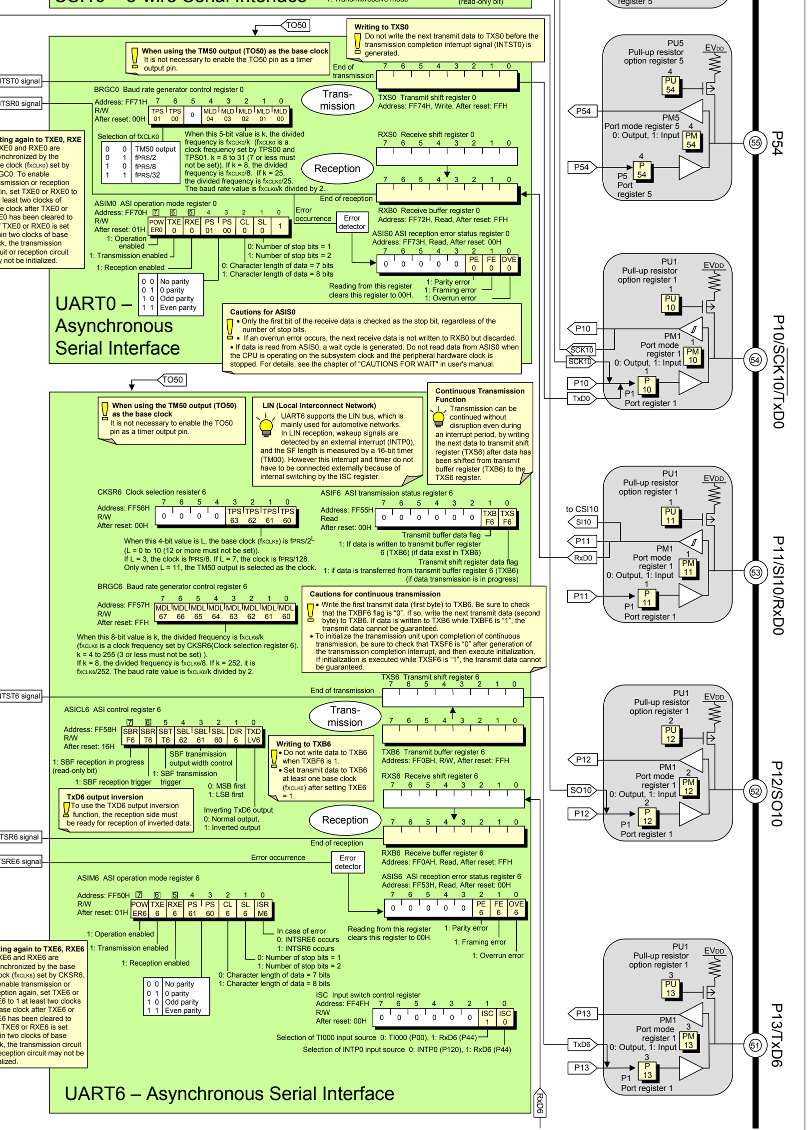
One-sheet Manual
Compact and Easily Readable

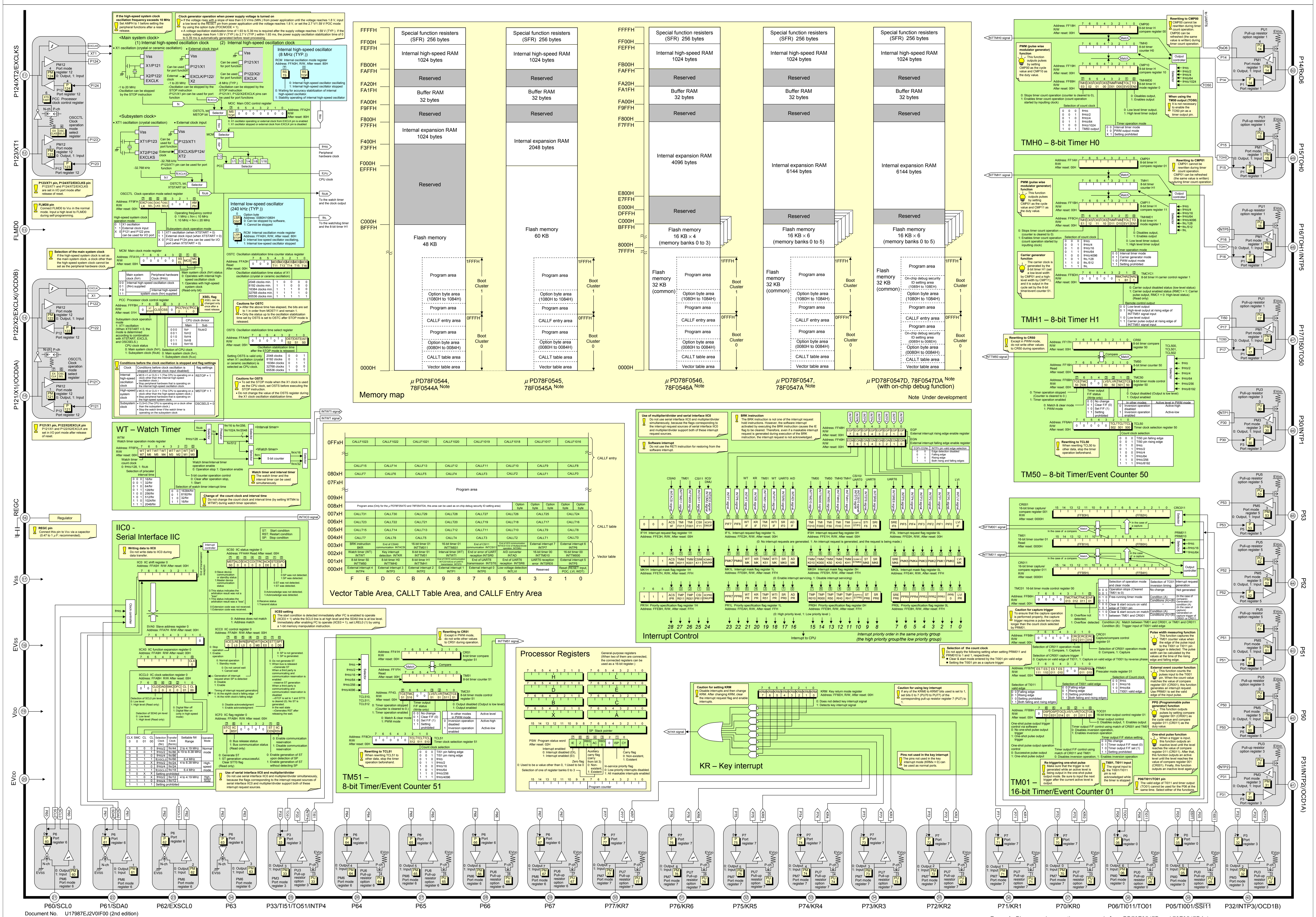
- μPD78F0544, 78F0545, 78F0546, 78F0547, 78F0547D, 78F0544A[Note], 78F0545A[Note], 78F0546A[Note], 78F0547A[Note], 78F0547DA[Note] - Note Under development

Outline of Functions

Item			μPD78F0544, 78F0544A[Note]	μPD78F0545, 78F0545A[Note]	μPD78F0546, 78F0546A[Note]	μPD78F0547, 78F0547A[Note]	μPD78F0547D, 78F0547DA[Note]
Internal memory	Flash memory (self-programming supported)		48 KB	60 KB	96 KB	128 KB	
		Memory bank	–	–	4 banks	6 banks	
	High-speed RAM		1 KB				
	Expansion RAM		1 KB	2 KB	4 KB	6 KB	
	Buffer RAM (can be used for transfer in CSIA0 with automatic transmit/receive function)		32 KB				
Memory space			64 KB				
Main system clock	High-speed system clock		X1(crystal/ceramic) oscillation, external system clock input (EXCLK) 1 to 20 MHz: V_{DD} = 4.0 to 5.5 V, 1 to 10 MHz: V_{DD} = 2.7 to 5.5 V, 1 to 5 MHz: V_{DD} = 1.8 to 5.5 V				
	Internal high-speed oscillation clock		Internal oscillation 8 MHz (TYP.) : V_{DD} = 1.8 to 5.5 V				
Subsystem clock			XT1(crystal) oscillation, external subsystem clock input (EXCLKS) 32.768 kHz (TYP.) : V_{DD} = 1.8 to 5.5 V				
Internal low-speed oscillation closk			Internal oscillation 240 kHz (TYP.) : V_{DD} = 1.8 to 5.5 V				
General-purpose registers			8 bits x 32 registers (8 bits x 8 registers x 4 banks)				
Minimum instruction execution time			0.1 μs (high-speed system clock: @ f_{XH} = 20 MHz operation)				
			0.25 μs (internal high-speed oscillation clock: @ f_{XH} = 8 MHz (TYP.) operation)				
			122 μs (subsystem clock: @ f_{SUB} = 32.768 kHz operation)				
Instruction set			• 8-bit operation, 16-bit operation • Multiply/divide (8 bits x 8 bits, 16 bits ÷ 8 bits) • Bit manipulate (set, reset, test, and Boolean operation) • BCD adjust, etc.				
I/O ports			Total: 71 • CMOS I/O: 66 • CMOS output: 1 • N-ch open-drain I/O (6.0 V tolerance) : 4				
Timers			• 16-bit timer/event counter: 2 channels • 8-bit timer/event counter: 2 channels • 8-bit timer: 2 channels • Watch timer: 1 channel • Watchdog timer: 1 channel				
	Timer outputs		6 (PWM output: 4, PPG output: 2)				
Clock output			• 156.25 kHz, 312.5 kHz, 625 kHz, 1.25 MHz, 2.5 MHz, 5 MHz, 10 MHz (peripheral hardware clock: @ f_{PRS} = 20 MHz operation) • 32.768 kHz (subsysterm clock: @ f_{SUB} = 32.768 kHz operation)				
Buzzer output			2.44 kHz, 4.88 kHz, 9.77 kHz, 19.54 kHz (peripheral hardware clock: @ f_{PRS} = 20 MHz operation)				
A/D converter			10-bit resolution x 8 channels (AV_{REF} = 2.3 to 5.5 V)				
Serial interface			• UART supporing LIN-bus: 1 channel • 3-wire serial I/O/UART: 1 channel • 3-wire serial I/O: 1 channel • 3-wire serial I/O with automatic transmit/receive function: 1 channel • I^2C bus				
Mutiplier/divider			• 16 bits x 16 bits = 32 bits (multiplication) • 32 bits ÷ 16 bits = 32 bits remainder of 16 bits (division)				
Vectored internal sources		Internal	20				
		External	9				
Key interrupt			Key interrupt (INTKR) occurs by detecting falling edge of key input pin (KR0 to KR7)				
Reset			• Reset using $\overline{RESET}$ pin • Internal reset by watchdog timer • Internal reset by power-on-clear • Internal reset by low-voltage detector				
On-chip debug function			None				Provided
Power supply voltage			V_{DD} = 1.8 to 5.5 V				
Operating ambient temperature			T_A = –40 to 85°C				
Package			• 80-pin plastic LQFP (14x14) • 80-pin plastic LQFP (fine pitch) (12x12)				

Note Under development





Document No. U17988EJ2V0IF00 (2nd edition)
Date Published February 2007 NS CP(K)

Remark Pin names in parentheses are only for μPD78F0547D and 78F0547DA (products with on-chip debug function).

Development Tools (1/5)

Remark For details about development tools, see the site for development tools at NEC Electronics Website.
NEC Electronics Website: http://www.necel.com/

(1) Software Tools

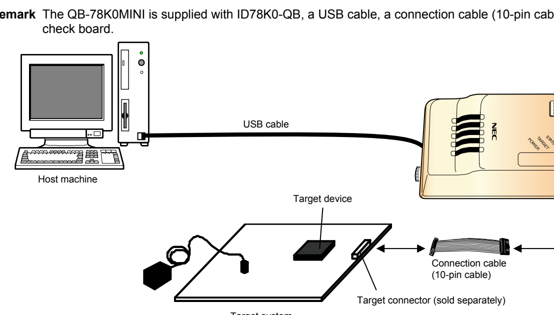
Host Machine	Software Tools
IBM PC/AT™ compatibles, PC98-NX series	Software package SP78K0
	Assembler package RA78K0
	C compiler CC78K0
	C library source file CC78K0-L
	Integrated debugger ID78K0-QB
	System simulator SM+ for 78K0/Kx2
	Device file DF780547

(2) Hardware Tools (1/3)

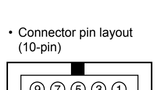
<1> On-chip debug emulator QB-78K0MINI (MINICUBE®)

On-chip Debug Emulator	Target Connector Specifications
QB-78K0MINI	10-pin general-purpose connector (2.54 mm pitch)

Remark The QB-78K0MINI is supplied with ID78K0-QB, a USB cable, a connection cable (10-pin cable) and a self-check board.



• Connector pin layout (10-pin)



(Top View)

• Connector pin configuration (10-pin)

Pin No.	Pin Name Note 1	IN/OUT Note 2	Description
1	RESET_IN	IN	Pin used to input reset signal from the target system
2	RESET_OUT	OUT	Pin used to output reset signal to the target device
3	FLMD0	OUT	Output pin used to control on-chip debugging functions
4	VDD	IN	Input pin for when using power supply of the target system
5	X2/DATA	IN/OUT	Pin used to input/output for data communication during debugging
6	GND	–	Connected to GND.
7	X1/CLK	OUT	Pin used to output clock signal to the target device
8	GND	–	Connected to GND.
9	RESERVED	–	Open
10	RESERVED	–	Open

Notes 1. Signal names in MINICUBE
2. As seen from MINICUBE.

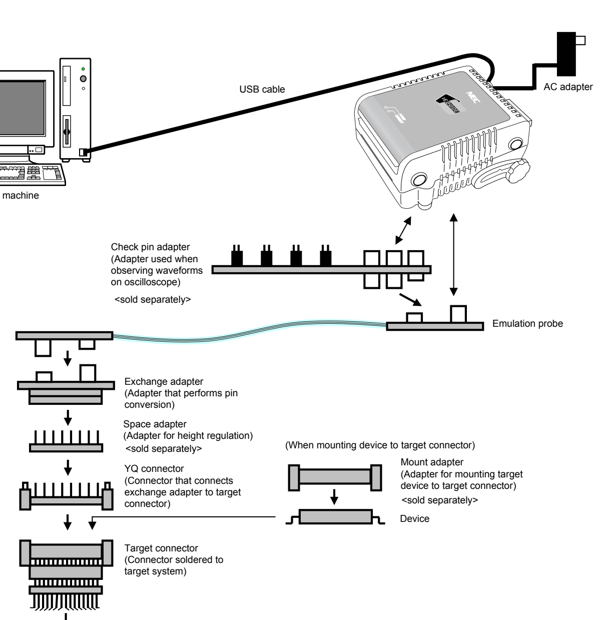
Development Tools (2/5)

(2) Hardware Tools (2/3)

<2> In-circuit emulator QB-78K0KX2 (IECUBE®)

In-Circuit Emulator	Package	Check Pin Adapter	Emulation Probe	Exchange Adapter	Space Adapter	YQ Connector	Mount Adapter	Target connector
QB-78K0KX2	80-pin plastic LQFP (14x14)	QB-144-CA-01	QB-80-EP-01T	QB-80GC-EA-01T	QB-80GC-YS-01T	QB-80GC-YQ-01T	QB-80GC-HQ-01T	QB-80GC-NQ-01T
	80-pin plastic LQFP (fine pitch) (12x12)			QB-80GK-EA-01T	QB-80GK-YS-01T	QB-80GK-YQ-01T	QB-80GK-HQ-01T	QB-80GK-NQ-01T

Remark The QB-78K0KX2 is supplied with ID78K0-QB, a USB cable, a power supply unit, QB-MINI2, connection cables (10-pin cable and 16-pin cable) and the 78K0-OCD board.



Development Tools (3/5)

(2) Hardware Tools (3/3)

<3> On-chip debug emulator with programming function QB-MINI2 (MINICUBE2) : for on-chip debugging

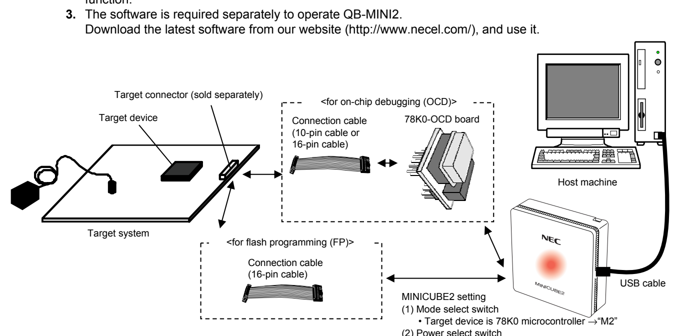
On-Chip Debug Emulator with Programming Function	Target Connector Specifications	
QB-MINI2	10-pin general-purpose connector (2.54 mm pitch)	When using 10-pin cable
	16-pin general-purpose connector (2.54 mm pitch)	When using 16-pin cable

(3) Flash Memory Write Tools (1/3)

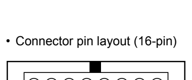
<1> On-chip debug emulator with programming function QB-MINI2 (MINICUBE2) : for flash programming

On-Chip Debug Emulator with Programming Function	Target Connector Specifications	
QB-MINI2	16-pin general-purpose connector (2.54 mm pitch)	When using 16-pin cable

Remarks 1. The QB-MINI2 is supplied with a USB cable, connection cables (10-pin cable and 16-pin cable) and the 78K0-OCD board.
2. A connection cable (10-pin cable) and the 78K0-OCD board are used only when using the on-chip debug function.
3. The software is required separately to operate QB-MINI2.
Download the latest software from our website (http://www.necel.com/), and use it.



• Connector pin layout (16-pin)



(Top View)

• Connector pin configuration (16-pin)

Pin No.	Pin Name Note 1	IN/OUT Note 2	Description
1	GND	–	Connected to GND.
2	RESET_OUT	OUT	Pin used to output reset signal to the target device
3	RESERVED (during OCD)	–	Open
	RXD (during FP)	IN	Pin used to input for command/data communication from the target device
4	VDD	IN	Input pin for when using power supply of the target system
5	RESERVED (during OCD)	–	Open
	TXD (during FP)	OUT	Pin used to output for command/data communication to the target device
6-8	RESERVED	–	Open
9	CLK	OUT	Pin used to output clock signal to the target device
10-12	RESERVED	–	Open
13	DATA (during OCD)	IN/OUT	Pin used to input/output for data communication during debugging
	RESERVED (during FP)	–	Open
14	FLMD0	OUT	Pin used to set to debug mode or programming mode
15	RESET_IN (during OCD)	IN	Pin used to input reset signal from the target system
	RESERVED (during FP)	–	Open
16	RESERVED	–	Open

Notes 1. Signal names in MINICUBE2
2. As seen from MINICUBE2.

Remark The 10-pin target connector is the same as it of MINICUBE. See the description of the target connector of MINICUBE.

Development Tools (4/5)

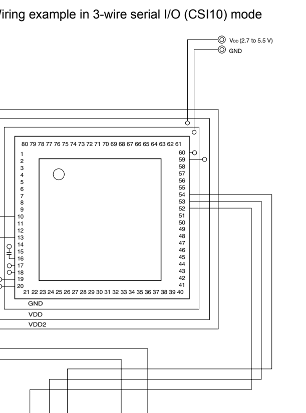
(3) Flash Memory Write Tools (2/3)

<2> Flash memory programmer PG-FP4, FL-PR4, PG-FPL3, FP-LITE3 (1/2)

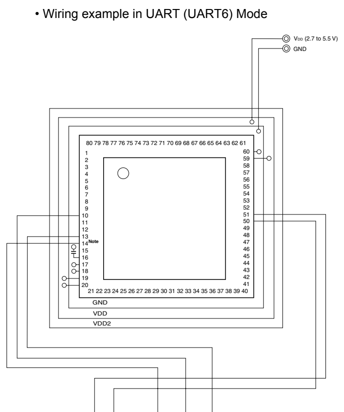
Flash Memory Programmer	Flash Memory Write Adapter	
Flash memory programmer PG-FP4, FL-PR4 Simple flash memory programmer PG-FPL3, FP-LITE3	FA-80GC-8BT-A	80-pin plastic LQFP (14x14)
	FA-78F0547GC-UBT-MX	
	FA-80GK-9EU-A	80-pin plastic LQFP (fine pitch) (12x12)
	FA-78F0547GK-8EU-MX	

Remarks 1. FL-PR4, FP-LITE3, FA-80GC-8BT-A, FA-78F0547GC-UBT-MX, FA-80GK-9EU-A, and FA-78F0547GK-8EU-MX are products of Naito Densei Machida Mfg. Co., Ltd.
TEL: +81-42-750-4172 Naito Densei Machida Mfg. Co., Ltd.
2. Use the latest version of the flash memory programming adapter.

• Wiring example in 3-wire serial I/O (CSI10) mode

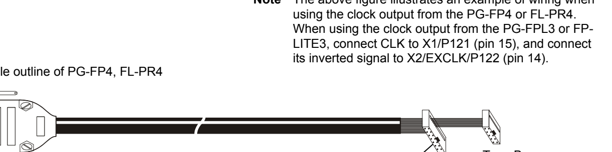


• Wiring example in UART (UART6) Mode



Note The above figure illustrates an example of wiring when using the clock output from the PG-FP4 or FL-PR4. When using the clock output from the PG-FPL3 or FP-LITE3, connect CLK to X1/P121 (pin 15), and connect its inverted signal to X2/EXCLK/P122 (pin 14).

• Target cable outline of PG-FP4, FL-PR4



Note Type B is not used to connect with 78K0/KF2 because 78K0/KF2 incorporates the single-power-supply flash memory.

Development Tools (5/5)

(3) Flash Memory Write Tools (3/3)

<2> Flash memory programmer PG-FP4, FL-PR4, PG-FPL3, FP-LITE3 (2/2)

• Connector pin layout of PG-FP4 and FL-PR4 (view from socket side)

1	3	5	7	9	11	13	15
2	4	6	8	10	12	14	16

Type A (16-pin)

• Connector pin configuration of PG-FP4 and FL-PR4

Signal Name of PG-FP4	Target Connector Type A: Signal (16-Pin)
GND	1
RESET	2
SI/RxD	3
VDD	4
SO/TxD	5
(VPP) Note	6
SCK	7
(H/S) Note	8
CLK	9
(VDE) Note	10
(VDD2) Note	11
(FLMD1) Note	12
(RFU-1) Note	13
FLMD0	14
(Not used) Note	15, 16

Note Signals in parentheses and the corresponding pins are not used with 78K0/KF2.

Operation List (1/6)

• Operand identifiers and specification methods

Identifier	Specification Method
r rp sfr sfrp	X (R0), A (R1), C (R2), B (R3), E (R4), D (R5), L (R6), H (R7) AX (RP0), BC (RP1), DE (RP2), HL (RP3) Special function register symbol Note Special function register symbol (16-bit manipulatable register even addresses only) Note
saddr saddrp	FE20H to FF1FH Immediate data or labels FE20H to FF1FH Immediate data or labels (even address only)
addr16 addr11 addr5	0000H to FFFFH Immediate data or labels (Only even addresses for 16-bit data transfer instructions) 0800H to 0FFFH Immediate data or labels 0040H to 007FH Immediate data or labels (even address only)
word byte bit	16-bit immediate data or label 8-bit immediate data or label 3-bit immediate data or label
RBn	RB0 to RB3

Note Addresses from FFD0H to FFDFH cannot be accessed with these operands.

Operation List (2/6)

• Description of operation column

A: A register (8-bit accumulator), X: X register, B: B register, C: C register, D: D register, E: E register, H: H register, L: L register, AX: AX register pair (16-bit accumulator), BC: BC register pair, DE: DE register pair, HL: HL register pair, PC: Program counter, SP: Stack pointer, PSW: Program status word, CY: Carry flag, AC: Auxiliary carry flag, Z: Zero flag, RBS: Register bank select flag, IE: Interrupt request enable flag,
( ): Memory contents indicated by address or register contents in parentheses,
X_H, X_L: Higher 8 bits and lower 8 bits of 16-bit register, ∧: Logical product (AND), ∨: Logical sum (OR),
⊻: Exclusive logical sum (exclusive OR), ¯¯: Inverted data, addr16: 16-bit immediate data or label,
jdisp8: Signed 8-bit data (displacement value)

• Description of flag operation column

(Blank): Not affected, 0: Cleared to 0, 1: Set to 1, ×: Set/cleared according to the result, R: Previously saved value is restored

Instruction Group	Mnemonic	Operands	Bytes	Clocks Note 1	Clocks Note 2	Operation	Z	AC	CY
8-bit data transfer	MOV	r, #byte	2	4	–	r ← byte			
		saddr, #byte	3	6	7	(saddr) ← byte			
		sfr, #byte	3	–	7	sfr ← byte			
		A, r Note 3	1	2	–	A ← r			
		r, A Note 3	1	2	–	r ← A			
		A, saddr	2	4	5	A ← (saddr)			
		saddr, A	2	4	5	(saddr) ← A			
		A, sfr	2	–	5	A ← sfr			
		sfr, A	2	–	5	sfr ← A			
		A, !addr16	3	8	9	A ← (addr16)			
		!addr16, A	3	8	9	(addr16) ← A			
		PSW, #byte	3	–	7	PSW ← byte	×	×	×
		A, PSW	2	–	5	A ← PSW			
		PSW, A	2	–	5	PSW ← A	×	×	×
		A, [DE]	1	4	5	A ← (DE)			
		[DE], A	1	4	5	(DE) ← A			
		A, [HL]	1	4	5	A ← (HL)			
		[HL], A	1	4	5	(HL) ← A			
		A, [HL+byte]	2	8	9	A ← (HL+byte)			
		[HL+byte], A	2	8	9	(HL+byte) ← A			
		A, [HL+B]	1	6	7	A ← (HL+B)			
		[HL+B], A	1	6	7	(HL+B) ← A			
		A, [HL+C]	1	6	7	A ← (HL+C)			
		[HL+C], A	1	6	7	(HL+C) ← A			
	XCH	A, r Note 3	1	2	–	A ↔ r			
		A, saddr	2	4	6	A ↔ (saddr)			
		A, sfr	2	–	6	A ↔ sfr			
		A, !addr16	3	8	10	A ↔ (addr16)			
		A, [DE]	1	4	6	A ↔ (DE)			
		A, [HL]	1	4	6	A ↔ (HL)			
		A, [HL+byte]	2	8	10	A ↔ (HL+byte)			
		A, [HL+B]	2	8	10	A ↔ (HL+B)			
		A, [HL+C]	2	8	10	A ↔ (HL+C)			
16-bit data transfer	MOVW	rp, #word	3	6	–	rp ← word			
		saddrp, #word	4	8	10	(saddrp) ← word			
		sfrp, #word	4	–	10	sfrp ← word			
		AX, saddrp	2	6	8	AX ← (saddrp)			
		saddrp, AX	2	6	8	(saddrp) ← AX			
		AX, sfrp	2	–	8	AX ← sfrp			
		sfrp, AX	2	–	8	sfrp ← AX			

Notes 1. When the internal high-speed RAM area is accessed or for an instruction with no data access
2. When an area except the internal high-speed RAM area is accessed
3. Except "r = A"

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC).
2. This clock cycle applies to the internal ROM program.

Operation List (3/6)

Instruction Group	Mnemonic	Operands	Bytes	Clocks Note 1	Clocks Note 2	Operation	Z	AC	CY
16-bit data transfer	MOVW	AX, rp Note 3	1	4	–	AX ← rp			
		rp, AX Note 3	1	4	–	rp ← AX			
		AX, !addr16	3	10	12	AX ← (addr16)			
		!addr16, AX	3	10	12	(addr16) ← AX			
	XCHW	AX, rp Note 3	1	4	–	AX ↔ rp			
8-bit operation	ADD	A, #byte	2	4	–	A, CY ← A+byte	×	×	×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) +byte	×	×	×
		A, r Note 4	2	4	–	A, CY ← A+r	×	×	×
		r, A	2	4	–	r, CY ← r+A	×	×	×
		A, saddr	2	4	5	A, CY ← A+ (saddr)	×	×	×
		A, !addr16	3	8	9	A, CY ← A+ (addr16)	×	×	×
		A, [HL]	1	4	5	A, CY ← A+ (HL)	×	×	×
		A, [HL+byte]	2	8	9	A, CY ← A+ (HL+byte)	×	×	×
		A, [HL+B]	2	8	9	A, CY ← A+ (HL+B)	×	×	×
		A, [HL+C]	2	8	9	A, CY ← A+ (HL+C)	×	×	×
	ADDC	A, #byte	2	4	–	A, CY ← A+byte+CY	×	×	×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) +byte+CY	×	×	×
		A, r Note 4	2	4	–	A, CY ← A+r+CY	×	×	×
		r, A	2	4	–	r, CY ← r+A+CY	×	×	×
		A, saddr	2	4	5	A, CY ← A+ (saddr) +CY	×	×	×
		A, !addr16	3	8	9	A, CY ← A+ (addr16) +CY	×	×	×
		A, [HL]	1	4	5	A, CY ← A+ (HL) +CY	×	×	×
		A, [HL+byte]	2	8	9	A, CY ← A+ (HL+byte) +CY	×	×	×
		A, [HL+B]	2	8	9	A, CY ← A+ (HL+B) +CY	×	×	×
		A, [HL+C]	2	8	9	A, CY ← A+ (HL+C) +CY	×	×	×
	SUB	A, #byte	2	4	–	A, CY ← A – byte	×	×	×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) – byte	×	×	×
		A, r Note 4	2	4	–	A, CY ← A – r	×	×	×
		r, A	2	4	–	r, CY ← r – A	×	×	×
		A, saddr	2	4	5	A, CY ← A – (saddr)	×	×	×
		A, !addr16	3	8	9	A, CY ← A – (addr16)	×	×	×
		A, [HL]	1	4	5	A, CY ← A – (HL)	×	×	×
		A, [HL+byte]	2	8	9	A, CY ← A – (HL+byte)	×	×	×
		A, [HL+B]	2	8	9	A, CY ← A – (HL+B)	×	×	×
		A, [HL+C]	2	8	9	A, CY ← A – (HL+C)	×	×	×
	SUBC	A, #byte	2	4	–	A, CY ← A – byte – CY	×	×	×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) – byte – CY	×	×	×
		A, r Note 4	2	4	–	A, CY ← A – r – CY	×	×	×
		r, A	2	4	–	r, CY ← r – A – CY	×	×	×
		A, saddr	2	4	5	A, CY ← A – (saddr) – CY	×	×	×
		A, !addr16	3	8	9	A, CY ← A – (addr16) – CY	×	×	×
		A, [HL]	1	4	5	A, CY ← A – (HL) – CY	×	×	×
		A, [HL+byte]	2	8	9	A, CY ← A – (HL+byte) – CY	×	×	×
		A, [HL+B]	2	8	9	A, CY ← A – (HL+B) – CY	×	×	×
		A, [HL+C]	2	8	9	A, CY ← A – (HL+C) – CY	×	×	×
	AND	A, #byte	2	4	–	A ← A ∧ byte	×		
		saddr, #byte	3	6	8	(saddr) ← (saddr) ∧ byte	×		
		A, r Note 4	2	4	–	A ← A ∧ r	×		
		r, A	2	4	–	r ← r ∧ A	×		
		A, saddr	2	4	5	A ← A ∧ (saddr)	×		
		A, !addr16	3	8	9	A ← A ∧ (addr16)	×		
		A, [HL]	1	4	5	A ← A ∧ (HL)	×		
		A, [HL+byte]	2	8	9	A ← A ∧ (HL+byte)	×		
		A, [HL+B]	2	8	9	A ← A ∧ (HL+B)	×		
		A, [HL+C]	2	8	9	A ← A ∧ (HL+C)	×		

Notes 1. When the internal high-speed RAM area is accessed or for an instruction with no data access
2. When an area except the internal high-speed RAM area is accessed
3. Only when rp = BC, DE or HL
4. Except "r = A"

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC).
2. This clock cycle applies to the internal ROM program.

Operation List (4/6)

Instruction Group	Mnemonic	Operands	Bytes	Clocks Note 1	Clocks Note 2	Operation	Z	AC	CY
8-bit operation	OR	A, #byte	2	4	–	A ← A ∨ byte	×		
		saddr, #byte	3	6	8	(saddr) ← (saddr) ∨ byte	×		
		A, r Note 3	2	4	–	A ← A ∨ r	×		
		r, A	2	4	–	r ← r ∨ A	×		
		A, saddr	2	4	5	A ← A ∨ (saddr)	×		
		A, !addr16	3	8	9	A ← A ∨ (addr16)	×		
		A, [HL]	1	4	5	A ← A ∨ (HL)	×		
		A, [HL+byte]	2	8	9	A ← A ∨ (HL+byte)	×		
		A, [HL+B]	2	8	9	A ← A ∨ (HL+B)	×		
		A, [HL+C]	2	8	9	A ← A ∨ (HL+C)	×		
	XOR	A, #byte	2	4	–	A ← A ⊻ byte	×		
		saddr, #byte	3	6	8	(saddr) ← (saddr) ⊻ byte	×		
		A, r Note 3	2	4	–	A ← A ⊻ r	×		
		r, A	2	4	–	r ← r ⊻ A	×		
		A, saddr	2	4	5	A ← A ⊻ (saddr)	×		
		A, !addr16	3	8	9	A ← A ⊻ (addr16)	×		
		A, [HL]	1	4	5	A ← A ⊻ (HL)	×		
		A, [HL+byte]	2	8	9	A ← A ⊻ (HL+byte)	×		
		A, [HL+B]	2	8	9	A ← A ⊻ (HL+B)	×		
		A, [HL+C]	2	8	9	A ← A ⊻ (HL+C)	×		
	CMP	A, #byte	2	4	–	A – byte	×	×	×
		saddr, #byte	3	6	8	(saddr) – byte	×	×	×
		A, r Note 3	2	4	–	A – r	×	×	×
		r, A	2	4	–	r – A	×	×	×
		A, saddr	2	4	5	A – (saddr)	×	×	×
		A, !addr16	3	8	9	A – (addr16)	×	×	×
		A, [HL]	1	4	5	A – (HL)	×	×	×
		A, [HL+byte]	2	8	9	A – (HL+byte)	×	×	×
		A, [HL+B]	2	8	9	A – (HL+B)	×	×	×
		A, [HL+C]	2	8	9	A – (HL+C)	×	×	×
16-bit operation	ADDW	AX, #word	3	6	–	AX, CY ← AX + word	×	×	×
	SUBW	AX, #word	3	6	–	AX, CY ← AX – word	×	×	×
	CMPW	AX, #word	3	6	–	AX – word	×	×	×
Multiply/divide	MULU	X	2	16	–	AX ← A × X			
	DIVUW	C	2	25	–	AX (Quotient), C (Remainder) ← AX ÷ C			
Increment/decrement	INC	r	1	2	–	r ← r + 1	×	×	
		saddr	2	4	6	(saddr) ← (saddr) + 1	×	×	
	DEC	r	1	2	–	r ← r – 1	×	×	
		saddr	2	4	6	(saddr) ← (saddr) – 1	×	×	
	INCW	rp	1	4	–	rp ← rp+1			
	DECW	rp	1	4	–	rp ← rp – 1			
Rotate	ROR	A, 1	1	2	–	(CY, A_7 ← A_0, A_{m-1} ← A_m) × 1 time			×
	ROL	A, 1	1	2	–	(CY, A_0 ← A_7, A_{m+1} ← A_m) × 1 time			×
	RORC	A, 1	1	2	–	(CY ← A_0, A_7 ← CY, A_{m-1} ← A_m) × 1 time			×
	ROLC	A, 1	1	2	–	(CY ← A_7, A_0 ← CY, A_{m+1} ← A_m) × 1 time			×
	ROR4	[HL]	2	10	12	A_{3-0} ← $(HL)_{3-0}$, $(HL)_{7-4}$ ← A_{3-0}, $(HL)_{3-0}$ ← $(HL)_{7-4}$			
	ROL4	[HL]	2	10	12	A_{3-0} ← $(HL)_{7-4}$, $(HL)_{3-0}$ ← A_{3-0}, $(HL)_{7-4}$ ← $(HL)_{3-0}$			

Notes 1. When the internal high-speed RAM area is accessed or for an instruction with no data access
2. When an area except the internal high-speed RAM area is accessed
3. Except "r = A"

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC).
2. This clock cycle applies to the internal ROM program.

Operation List (5/6)

Instruction Group	Mnemonic	Operands	Bytes	Clocks Note 1	Clocks Note 2	Operation	Z	AC	CY
BCD adjustment	ADJBA		2	4	–	Decimal Adjust Accumulator after Addition	×	×	×
	ADJBS		2	4	–	Decimal Adjust Accumulator after Subtract	×	×	×
Bit manipulate	MOV1	CY, saddr.bit	3	6	7	CY ← (saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← sfr.bit			×
		CY, A.bit	2	4	–	CY ← A.bit			×
		CY, PSW.bit	3	–	7	CY ← PSW.bit			×
		CY, [HL].bit	2	6	7	CY ← (HL).bit			×
		saddr.bit, CY	3	6	8	(saddr.bit) ← CY			
		sfr.bit, CY	3	–	8	sfr.bit ← CY			
		A.bit, CY	2	4	–	A.bit ← CY			
		PSW.bit, CY	3	–	8	PSW.bit ← CY	×	×	
		[HL]. bit, CY	2	6	8	(HL).bit ← CY			
	AND1	CY, saddr.bit	3	6	7	CY ← CY ∧ (saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← CY ∧ sfr.bit			×
		CY, A.bit	2	4	–	CY ← CY ∧ A.bit			×
		CY, PSW.bit	3	–	7	CY ← CY ∧ PSW.bit			×
		CY, [HL].bit	2	6	7	CY ← CY ∧ (HL).bit			×
	OR1	CY, saddr.bit	3	6	7	CY ← CY ∨ (saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← CY ∨ sfr.bit			×
		CY, A.bit	2	4	–	CY ← CY ∨ A.bit			×
		CY, PSW.bit	3	–	7	CY ← CY ∨ PSW.bit			×
		CY, [HL].bit	2	6	7	CY ← CY ∨ (HL).bit			×
	XOR1	CY, saddr.bit	3	6	7	CY ← CY ⊻ (saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← CY ⊻ sfr.bit			×
		CY, A.bit	2	4	–	CY ← CY ⊻ A.bit			×
		CY, PSW.bit	3	–	7	CY ← CY ⊻ PSW.bit			×
		CY, [HL].bit	2	6	7	CY ← CY ⊻ (HL).bit			×
	SET1	saddr.bit	2	4	6	(saddr.bit) ← 1			
		sfr.bit	3	–	8	sfr.bit ← 1			
		A.bit	2	4	–	A.bit ← 1			
		PSW.bit	2	–	6	PSW.bit ← 1	×	×	×
		[HL].bit	2	6	8	(HL).bit ← 1			
	CLR1	saddr.bit	2	4	6	(saddr.bit) ← 0			
		sfr.bit	3	–	8	sfr.bit ← 0			
		A.bit	2	4	–	A.bit ← 0			
		PSW.bit	2	–	6	PSW.bit ← 0	×	×	×
		[HL].bit	2	6	8	(HL).bit ← 0			
	SET1	CY	1	2	–	CY ← 1			1
	CLR1	CY	1	2	–	CY ← 0			0
	NOT1	CY	1	2	–	CY ← $\overline{CY}$			×
Call/return	CALL	!addr16	3	7	–	(SP – 1) ← $(PC+3)_H$, (SP – 2) ← $(PC+3)_L$, PC ← addr16, SP ← SP – 2			
	CALLF	!addr11	2	5	–	(SP – 1) ← $(PC+2)_H$, (SP – 2) ← $(PC+2)_L$, PC_{15-11} ← 00001, PC_{10-0} ← addr11, SP ← SP – 2			
	CALLT	[addr5]	1	6	–	(SP – 1) ← $(PC+1)_H$, (SP – 2) ← $(PC+1)_L$, PC_H ← (00000000, addr5+1), PC_L ← (00000000, addr5), SP ← SP – 2			
	BRK		1	6	–	(SP – 1) ← PSW, (SP – 2) ← $(PC+1)_H$, (SP – 3) ← $(PC+1)_L$, PC_H ← (003FH), PC_L ← (003EH), SP ← SP – 3, IE ← 0			

Notes 1. When the internal high-speed RAM area is accessed or for an instruction with no data access
2. When an area except the internal high-speed RAM area is accessed

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC).
2. This clock cycle applies to the internal ROM program.

Operation List (6/6)

Instruction Group	Mnemonic	Operands	Bytes	Clocks Note 1	Clocks Note 2	Operation	Z	AC	CY
Call/return	RET		1	6	–	PC_H ← (SP+1), PC_L ← (SP), SP ← SP+2			
	RETI		1	6	–	PC_H ← (SP+1), PC_L ← (SP), PSW ← (SP+2), SP ← SP+3	R	R	R
	RETB		1	6	–	PC_H ← (SP+1), PC_L ← (SP), PSW ← (SP+2), SP ← SP+3	R	R	R
Stack manipulate	PUSH	PSW	1	2	–	(SP – 1) ← PSW, SP ← SP – 1			
		rp	1	4	–	(SP – 1) ← rp_H, (SP – 2) ← rp_L, SP ← SP – 2			
	POP	PSW	1	2	–	PSW ← (SP), SP ← SP+1	R	R	R
		rp	1	4	–	rp_H ← (SP – 1), rp_L ← (SP), SP ← SP+2			
	MOVW	SP, #word	4	–	10	SP ← word			
		SP, AX	2	–	8	SP ← AX			
		AX, SP	2	–	8	AX ← SP			
Unconditional branch	BR	!addr16	3	6	–	PC ← addr16			
		$addr16	2	6	–	PC ← PC + 2 + jdisp8			
		AX	2	8	–	PC_H ← A, PC_L ← X			
Conditional branch	BC	$addr16	2	6	–	PC ← PC + 2 + jdisp8 if CY = 1			
	BNC	$addr16	2	6	–	PC ← PC + 2 + jdisp8 if CY = 0			
	BZ	$addr16	2	6	–	PC ← PC + 2 + jdisp8 if Z = 1			
	BNZ	$addr16	2	6	–	PC ← PC + 2 + jdisp8 if Z = 0			
	BT	saddr.bit, $addr16	3	8	9	PC ← PC + 3 + jdisp8 if (saddr.bit) = 1			
		sfr.bit, $addr16	4	–	11	PC ← PC + 4 + jdisp8 if sfr.bit = 1			
		A.bit, $addr16	3	8	–	PC ← PC + 3 + jdisp8 if A.bit = 1			
		PSW.bit, $addr16	3	–	9	PC ← PC + 3 + jdisp8 if PSW.bit = 1			
		[HL].bit, $addr16	3	10	11	PC ← PC + 3 + jdisp8 if (HL).bit = 1			
	BF	saddr.bit, $addr16	4	10	11	PC ← PC + 4 + jdisp8 if (saddr.bit) = 0			
		sfr.bit, $addr16	4	–	11	PC ← PC + 4 + jdisp8 if sfr.bit = 0			
		A.bit, $addr16	3	8	–	PC ← PC + 3 + jdisp8 if A.bit = 0			
		PSW.bit, $addr16	4	–	11	PC ← PC + 4 + jdisp8 if PSW.bit = 0			
		[HL].bit, $addr16	3	10	11	PC ← PC + 3 + jdisp8 if (HL).bit = 0			
	BTCLR	saddr.bit, $addr16	4	10	12	PC ← PC + 4 + jdisp8 if (saddr.bit) = 1 then reset (saddr.bit)			
		sfr.bit, $addr16	4	–	12	PC ← PC + 4 + jdisp8 if sfr.bit = 1 then reset sfr.bit			
		A.bit, $addr16	3	8	–	PC ← PC + 3 + jdisp8 if A.bit = 1 then reset A.bit			
		PSW.bit, $addr16	4	–	12	PC ← PC + 4 + jdisp8 if PSW.bit = 1 then reset PSW.bit	×	×	×
		[HL].bit, $addr16	3	10	12	PC ← PC + 3 + jdisp8 if (HL).bit = 1 then reset (HL).bit			
	DBNZ	B, $addr16	2	6	–	B ← B – 1, then PC ← PC + 2 + jdisp8 if B ≠ 0			
		C, $addr16	2	6	–	C ← C – 1, then PC ← PC + 2 + jdisp8 if C ≠ 0			
		saddr, $addr16	3	8	10	(saddr) ← (saddr) – 1, then PC ← PC + 3 + jdisp8 if (saddr) ≠ 0			
CPU control	SEL	RBn	2	4	–	RBS1, 0 ← n			
	NOP		1	2	–	No Operation			
	EI		2	–	6	IE ← 1 (Enable Interrupt)			
	DI		2	–	8	IE ← 0 (Disable Interrupt)			
	HALT		2	6	–	Set HALT Mode			
	STOP		2	6	–	Set STOP Mode			

Notes 1. When the internal high-speed RAM area is accessed or for an instruction with no data access
2. When an area except the internal high-speed RAM area is accessed

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC).
2. This clock cycle applies to the internal ROM program.

Special Function Register (SFR) List (1/4)

Address	Special Function Register (SFR) Name	Symbol	R/W	1 Bit	8 Bits	16 Bits	After Reset
FF00H	Port register 0	P0	R/W	√	√	–	00H
FF01H	Port register 1	P1	R/W	√	√	–	00H
FF02H	Port register 2	P2	R/W	√	√	–	00H
FF03H	Port register 3	P3	R/W	√	√	–	00H
FF04H	Port register 4	P4	R/W	√	√	–	00H
FF05H	Port register 5	P5	R/W	√	√	–	00H
FF06H	Port register 6	P6	R/W	√	√	–	00H
FF07H	Port register 7	P7	R/W	√	√	–	00H
FF08H	10-bit A/D conversion result register	ADCR	R	–	–	√	0000H
FF09H	8-bit A/D conversion result register	ADCRH	R	–	√	–	00H
FF0AH	Receive buffer register 6	RXB6	R	–	√	–	FFH
FF0BH	Transmit buffer register 6	TXB6	R/W	–	√	–	FFH
FF0CH	Port register 12	P12	R/W	√	√	–	00H
FF0DH	Port register 13	P13	R/W	√	√	–	00H
FF0EH	Port register 14	P14	R/W	√	√	–	00H
FF0FH	Serial I/O shift register 10	SIO10	R	–	√	–	00H
FF10H FF11H	16-bit timer counter 00	TM00	R	–	–	√	0000H
FF12H FF13H	16-bit timer capture/compare register 000	CR000	R/W	–	–	√	0000H
FF14H FF15H	16-bit timer capture/compare register 010	CR010	R/W	–	–	√	0000H
FF16H	8-bit timer counter 50	TM50	R	–	√	–	00H
FF17H	8-bit timer compare register 50	CR50	R/W	–	√	–	00H
FF18H	8-bit timer H compare register 00	CMP00	R/W	–	√	–	00H
FF19H	8-bit timer H compare register 10	CMP10	R/W	–	√	–	00H
FF1AH	8-bit timer H compare register 01	CMP01	R/W	–	√	–	00H
FF1BH	8-bit timer H compare register 11	CMP11	R/W	–	√	–	00H
FF1FH	8-bit timer counter 51	TM51	R	–	√	–	00H
FF20H	Port mode register 0	PM0	R/W	√	√	–	FFH
FF21H	Port mode register 1	PM1	R/W	√	√	–	FFH
FF22H	Port mode register 2	PM2	R/W	√	√	–	FFH
FF23H	Port mode register 3	PM3	R/W	√	√	–	FFH
FF24H	Port mode register 4	PM4	R/W	√	√	–	FFH
FF25H	Port mode register 5	PM5	R/W	√	√	–	FFH
FF26H	Port mode register 6	PM6	R/W	√	√	–	FFH
FF27H	Port mode register 7	PM7	R/W	√	√	–	FFH
FF28H	A/D converter mode register	ADM	R/W	√	√	–	00H
FF29H	Analog input channel specification register	ADS	R/W	√	√	–	00H
FF2CH	Port mode register 12	PM12	R/W	√	√	–	FFH
FF2EH	Port mode register 14	PM14	R/W	√	√	–	FFH
FF2FH	A/D port configuration register	ADPC	R/W	√	√	–	00H
FF30H	Pull-up resistor option register 0	PU0	R/W	√	√	–	00H
FF31H	Pull-up resistor option register 1	PU1	R/W	√	√	–	00H
FF33H	Pull-up resistor option register 3	PU3	R/W	√	√	–	00H

Special Function Register (SFR) List (2/4)

Address	Special Function Register (SFR) Name	Symbol	R/W	1 Bit	8 Bits	16 Bits	After Reset
FF34H	Pull-up resistor option register 4	PU4	R/W	√	√	–	00H
FF35H	Pull-up resistor option register 5	PU5	R/W	√	√	–	00H
FF36H	Pull-up resistor option register 6	PU6	R/W	√	√	–	00H
FF37H	Pull-up resistor option register 7	PU7	R/W	√	√	–	00H
FF3CH	Pull-up resistor option register 12	PU12	R/W	√	√	–	00H
FF3EH	Pull-up resistor option register 14	PU14	R/W	√	√	–	00H
FF40H	Clock output selection register	CKS	R/W	√	√	–	00H
FF41H	8-bit timer compare register 51	CR51	R/W	–	√	–	00H
FF43H	8-bit timer mode control register 51	TMC51	R/W	√	√	–	00H
FF48H	External interrupt rising edge enable register	EGP	R/W	√	√	–	00H
FF49H	External interrupt falling edge enable register	EGN	R/W	√	√	–	00H
FF4AH	Serial I/O shift register 11	SIO11	R	–	√	–	00H
FF4CH	Transmit buffer register 11	SOTB11	R/W	–	√	–	00H
FF4FH	Input switch control register	ISC	R/W	√	√	–	00H
FF50H	Asynchronous serial interface operation mode register 6	ASIM6	R/W	√	√	–	01H
FF53H	Asynchronous serial interface reception error status register 6	ASIS6	R	–	√	–	00H
FF55H	Asynchronous serial interface transmission status register 6	ASIF6	R	–	√	–	00H
FF56H	Clock selection register 6	CKSR6	R/W	–	√	–	00H
FF57H	Baud rate generator control register 6	BRGC6	R/W	–	√	–	FFH
FF58H	Asynchronous serial interface control register 6	ASICL6	R/W	√	√	–	16H
FF60H	Remainder data register 0	SDR0: SDR0L	R	–	√	√	00H
FF61H		SDR0: SDR0H		–	√		00H
FF62H	Multiplication/division data register A0	MDA0L: MDA0LL	R/W	–	√	√	00H
FF63H		MDA0L: MDA0LH	R/W	–	√		00H
FF64H		MDA0H: MDA0HL	R/W	–	√	√	00H
FF65H		MDA0H: MDA0HH	R/W	–	√		00H
FF66H	Multiplication/division data register B0	MDB0: MDB0L	R/W	–	√	√	00H
FF67H		MDB0: MDB0H	R/W	–	√		00H
FF68H	Multiplier/divider control register 0	DMUC0	R/W	√	√	–	00H
FF69H	8-bit timer H mode register 0	TMHMD0	R/W	√	√	–	00H
FF6AH	Timer clock selection register 50	TCL50	R/W	√	√	–	00H
FF6BH	8-bit timer mode control register 50	TMC50	R/W	√	√	–	00H
FF6CH	8-bit timer H mode register 1	TMHMD1	R/W	√	√	–	00H
FF6DH	8-bit timer H carrier control register 1	TMCYC1	R/W	√	√	–	00H
FF6EH	Key return mode register	KRM	R/W	√	√	–	00H
FF6FH	Watch timer operation mode register	WTM	R/W	√	√	–	00H
FF70H	Asynchronous serial interface operation mode register 0	ASIM0	R/W	√	√	–	01H
FF71H	Baud rate generator control register 0	BRGC0	R/W	–	√	–	1FH
FF72H	Receive buffer register 0	RXB0	R	–	√	–	FFH

Special Function Register (SFR) List (3/4)

Address	Special Function Register (SFR) Name	Symbol	R/W	1 Bit	8 Bits	16 Bits	After Reset
FF73H	Asynchronous serial interface reception error status register 0	ASIS0	R	–	√	–	00H
FF74H	Transmit shift register 0	TXS0	W	–	√	–	FFH
FF80H	Serial operation mode register 10	CSIM10	R/W	√	√	–	00H
FF81H	Serial clock selection register 10	CSIC10	R/W	√	√	–	00H
FF84H	Transmit buffer register 10	SOTB10	R/W	–	√	–	00H
FF88H	Serial operation mode register 11	CSIM11	R/W	√	√	–	00H
FF89H	Serial clock selection register 11	CSIC11	R/W	√	√	–	00H
FF8CH	Timer clock selection register 51	TCL51	R/W	√	√	–	00H
FF90H	Serial operation mode specification register 0	CSIMA0	R/W	√	√	–	00H
FF91H	Serial status register 0	CSIS0	R/W	√	√	–	00H
FF92H	Serial trigger register 0	CSIT0	R/W	√	√	–	00H
FF93H	Division value selection register 0	BRGCA0	R/W	–	√	–	03H
FF94H	Automatic data transfer address point specification register 0	ADTP0	R/W	–	√	–	00H
FF95H	Automatic data transfer interval specification register 0	ADTI0	R/W	–	√	–	00H
FF96H	Serial I/O shift register 0	SIOA0	R/W	–	√	–	00H
FF97H	Automatic data transfer address count register 0	ADTC0	R	–	√	–	00H
FF99H	Watchdog timer enable register	WDTE	R/W	–	√	–	1AH/9AH Note 1
FF9FH	Clock operation mode select register	OSCCTL	R/W	√	√	–	00H
FFA0H	Internal oscillation mode register	RCM	R/W	√	√	–	80H Note 2
FFA1H	Main clock mode register	MCM	R/W	√	√	–	00H
FFA2H	Main OSC control register	MOC	R/W	√	√	–	80H
FFA3H	Oscillation stabilization time counter status register	OSTC	R	√	√	–	00H
FFA4H	Oscillation stabilization time select register	OSTS	R/W	–	√	–	05H
FFA5H	IIC shift register 0	IIC0	R/W	–	√	–	00H
FFA6H	IIC control register 0	IICC0	R/W	√	√	–	00H
FFA7H	Slave address register 0	SVA0	R/W	–	√	–	00H
FFA8H	IIC clock selection register 0	IICCL0	R/W	√	√	–	00H
FFA9H	IIC function expansion register 0	IICX0	R/W	√	√	–	00H
FFAAH	IIC status register 0	IICS0	R	√	√	–	00H
FFABH	IIC flag register 0	IICF0	R/W	√	√	–	00H
FFACH	Reset control flag register	RESF	R	–	√	–	00H Note 3
FFB0H FFB1H	16-bit timer counter 01	TM01	R	–	–	√	0000H
FFB2H FFB3H	16-bit timer capture/compare register 001	CR001	R/W	–	–	√	0000H
FFB4H FFB5H	16-bit timer capture/compare register 011	CR011	R/W	–	–	√	0000H
FFB6H	16-bit timer mode control register 01	TMC01	R/W	√	√	–	00H
FFB7H	Prescaler mode register 01	PRM01	R/W	√	√	–	00H
FFB8H	Capture/compare control register 01	CRC01	R/W	√	√	–	00H
FFB9H	16-bit timer output control register 01	TOC01	R/W	√	√	–	00H
FFBAH	16-bit timer mode control register 00	TMC00	R/W	√	√	–	00H

Notes 1. The reset value of WDTE is determined by setting of option byte.
2. The value of this register is 00H immediately after a reset release but automatically changes to 80H after oscillation accuracy stabilization of high-speed internal oscillator has been waited.
3. The reset value of RESF varies depending on the reset source.

Special Function Register (SFR) List (4/4)

Address	Special Function Register (SFR) Name	Symbol	R/W	1 Bit	8 Bits	16 Bits	After Reset
FFBBH	Prescaler mode register 00	PRM00	R/W	√	√	–	00H
FFBCH	Capture/compare control register 00	CRC00	R/W	√	√	–	00H
FFBDH	16-bit timer output control register 00	TOC00	R/W	√	√	–	00H
FFBEH	Low-voltage detection register	LVIM	R/W	√	√	–	00H Note 1
FFBFH	Low-voltage detection level selection register	LVIS	R/W	√	√	–	00H Note 1
FFE0H	Interrupt request flag register 0L	IF0: IF0L	R/W	√	√	√	00H
FFE1H	Interrupt request flag register 0H	IF0: IF0H	R/W	√	√		00H
FFE2H	Interrupt request flag register 1L	IF1: IF1L	R/W	√	√	√	00H
FFE3H	Interrupt request flag register 1H	IF1: IF1H	R/W	√	√		00H
FFE4H	Interrupt mask flag register 0L	MK0: MK0L	R/W	√	√	√	FFH
FFE5H	Interrupt mask flag register 0H	MK0: MK0H	R/W	√	√		FFH
FFE6H	Interrupt mask flag register 1L	MK1: MK1L	R/W	√	√	√	FFH
FFE7H	Interrupt mask flag register 1H	MK1: MK1H	R/W	√	√		FFH
FFE8H	Priority specification flag register 0L	PR0: PR0L	R/W	√	√	√	FFH
FFE9H	Priority specification flag register 0H	PR0: PR0H	R/W	√	√		FFH
FFEAH	Priority specification flag register 1L	PR1: PR1L	R/W	√	√	√	FFH
FFEBH	Priority specification flag register 1H	PR1: PR1H	R/W	√	√		FFH
FFF0H	Internal memory size switching register Note 2	IMS	R/W	–	√	–	CFH
FFF3H	Memory bank select register	BANK	R/W	–	√	–	00H
FFF4H	Internal expansion RAM size switching register Note 2	IXS	R/W	–	√	–	0CH
FFFBH	Processor clock control register	PCC	R/W	√	√	–	01H

Notes 1. The reset values of LVIM and LVIS vary depending on the reset source.
2. Regardless of the internal memory capacity, the initial values of the internal memory size switching register (IMS) and internal expansion RAM size switching register (IXS) of all products in the 78K0/KF2 are fixed (IMS = CFH, IXS = 0CH). Therefore, set the value corresponding to each product as indicated below.

78K0/KF2	IMS	IXS	ROM Capacity	Internal High-Speed RAM Capacity	Internal Expansion RAM Capacity
μPD78F0544, 78F0544A Note 3	CCH	0AH	48 KB	1 KB	1 KB
μPD78F0545, 78F0545A Note 3	CFH	08H	60 KB		2 KB
μPD78F0546, 78F0546A Note 3	CCH	04H	96 KB		4 KB
μPD78F0547, 78F0547A Note 3 78F0547D Note 4, 78F0547DA Note 3, 4	CCH	00H	128 KB		6 KB

3. Under development
4. The ROM and RAM capacities of the products with the on-chip debug function can be debugged according to the debug target products. Set IMS and IXS according to the debug target products.

Caution: This product uses SuperFlash® technology licensed from Silicon Storage Technology, Inc.

• This document is a reference.

For further information, please contact:

NEC Electronics Corporation
1753, Shimonumabe, Nakahara-ku,
Kawasaki, Kanagawa 211-8668,
Japan
Tel: 044-435-5111
http://www.necel.com/

[America]

NEC Electronics America, Inc.
2880 Scott Blvd.
Santa Clara, CA 95050-2554, U.S.A.
Tel: 408-588-6000
800-366-9782
http://www.am.necel.com/

[Europe]

NEC Electronics (Europe) GmbH
Arcadiastrasse 10
40472 Düsseldorf, Germany
Tel: 0211-65030
http://www.eu.necel.com/

Hanover Office
Podbielskistrasse 166 B
30177 Hannover
Tel: 0 511 33 40 2-0

Munich Office
Werner-Eckert-Strasse 9
81829 München
Tel: 0 89 92 10 03-0

Stuttgart Office
Industriestrasse 3
70565 Stuttgart
Tel: 0 711 99 01 0-0

United Kingdom Branch
Cygnus House, Sunrise Parkway
Linford Wood, Milton Keynes
MK14 6NP, U.K.
Tel: 01908-691-133

Succursale Française
9, rue Paul Dautier, B.P. 52
78142 Velizy-Villacoublay Cédex
France
Tel: 01-3067-5800

Sucursal en España
Juan Esplandiu, 15
28007 Madrid, Spain
Tel: 091-504-2787

Tyskland Filial
Täby Centrum
Entrance S (7th floor)
18322 Täby, Sweden
Tel: 08 638 72 00

Filiale Italiana
Via Fabio Filzi, 25/A
20124 Milano, Italy
Tel: 02-667541

Branch The Netherlands
Steijgerweg 6
5616 HS Eindhoven
The Netherlands
Tel: 040 265 40 10

[Asia & Oceania]

NEC Electronics (China) Co., Ltd
7th Floor, Quantum Plaza, No. 27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: 010-8235-1155
http://www.cn.necel.com/

NEC Electronics Shanghai Ltd.
Room 2511-2512, Bank of China Tower,
200 Yincheng Road Central,
Pudong New Area, Shanghai P.R. China P.C:200120
Tel: 021-5888-5400
http://www.cn.necel.com/

NEC Electronics Hong Kong Ltd.
Unit 1601-1613, 16/F., Tower 2, Grand Century Place,
193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: 2886-9318
http://www.hk.necel.com/

NEC Electronics Taiwan Ltd.
7F, No. 363 Fu Shing North Road
Taipei, Taiwan, R. O. C.
Tel: 02-8175-9600
http://www.tw.necel.com/

NEC Electronics Singapore Pte. Ltd.
238A Thomson Road,
#12-08 Novena Square,
Singapore 307684
Tel: 6253-8311
http://www.sg.necel.com/

NEC Electronics Korea Ltd.
11F., Samik Lavied'or Bldg., 720-2,
Yeoksam-Dong, Kangnam-Ku,
Seoul, 135-080, Korea
Tel: 02-558-3737
http://www.kr.necel.com/

G07.1A