To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: http://www.renesas.com

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (http://www.renesas.com)

Send any inquiries to http://www.renesas.com/inquiry.

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: “Standard”, “High Quality”, and “Specific”. The recommended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as “Specific” without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as “Specific” or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is “Standard” unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

   “Standard”: Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.

   “High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.

   “Specific”: Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

USER'S MANUAL

RENESAS

Phase-out/Discontinued

AS17216 DEVICE FILE

VERSION V1

PC-9800 SERIES (MS-DOS™) BASED
IBM PC/AT™ (PC DOS) BASED

Document No. EEU-1453
(O. D. No. EEU-918)
Date Published September 1993 P
Printed in Japan

USER'S MANUAL

NEC

Phase-out/Discontinued

AS17216 DEVICE FILE

VERSION V1

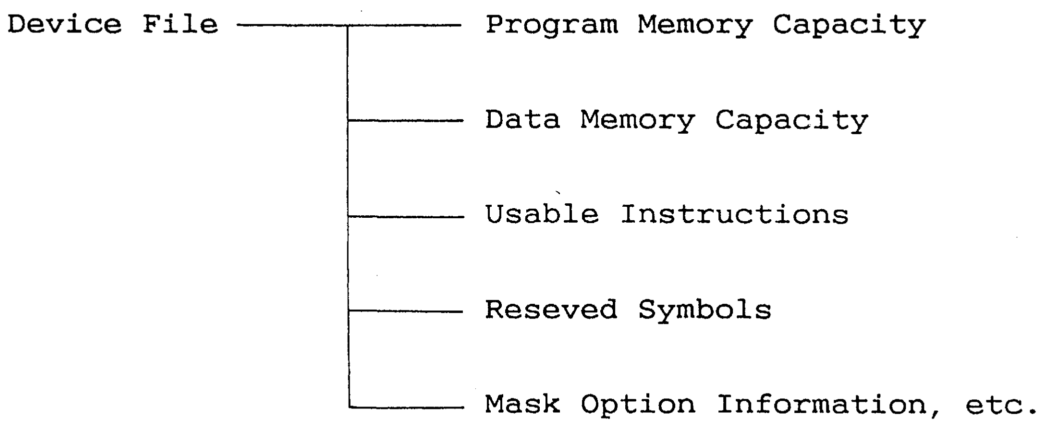
PC-9800 SERIES (MS-DOS™) BASED
IBM PC/AT™ (PC DOS) BASED

Phase-out/Discontinued

The information in this document is subject to change without notice.
No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Corporation. NEC Corporation assumes no responsibility for any errors which may appear in this document.
NEC Corporation does not assume any liability for infringement of patents, copyrights or other intellectual property rights of third parties by or arising from use of a device described herein or any other liability arising from use of such device. No license, either express, implied or otherwisé, is granted under any patents, copyrights or other intellectual property rights of NEC Corporation or of others.

PREFACE

AS17216 is a device file which is used with the AS17K assembler to assemble uPD17216 programs.



Device File

Program Memory Capacity

Data Memory Capacity

Usable Instructions

Reseved Symbols

Mask Option Information, etc.

For the AS17K assembler and AS17216 device file operating instructions, refer to the AS17K Assembler User's Manual(EEU-603).

CONTENTS

CHAPTER 1. DEVICE INFORMATION

The AS17216 device file supplies the following information related to the uPD17216 at assembly:

(1) Program memory (ROM) capacity

4096 x 16 bits (0000H to 0FFFH)

(2) Data memory (RAM) capacity

111 x 4 bits (BANK0)

(3) Usable instructions

See Chapter 2 "uPD17216 Instruction Set".

(4) Register files, port registers, and peripheral registers read and write information

See Chapter 3 "Reserved Symbols".

(5) Reserved symbols

See Chapter 3 "Reserved Symbols".

(6) Device file and device number/SE board number

In the device file, the device number peculiar to each device and the SE board number which identifies the most suitable SE board for development of each product are registered. These are also output to the ICE file and the PRO flie which are output by the assembler (AS17K). They are used when checking the development environment with an in-circuit emulator and checking when a mask is ordered.

Table 1-1 Device File and Device Number/SE Board Number Correspondence Table

Device File (Version)	Device Name	Device Number	SE Board Number	SE Board
AS17216 (V1)	uPD17216	4A	34	SE-17215

CHAPTER 2. uPD17216 INSTRUCTION SET

2.1 Instruction Set Outline

b_{14} to b_{11} \ b_{15}		0		1	
BIN	HEX				
0000	0	ADD	r, m	ADD	m, #n4
0001	1	SUB	r, m	SUB	m, #n4
0010	2	ADDC	r, m	ADDC	m, #n4
0011	3	SUBC	r, m	SUBC	m, #n4
0100	4	AND	r, m	AND	m, #n4
0101	5	XOR	r, m	XOR	m, #n4
0110	6	OR	r, m	OR	m, #n4
0111	7	INC INC MOVT BR CALL RET RETSK EI DI RETI PUSH POP GET PUT PEEK POKE RORC STOP HALT NOP	AR IX DBF, @AR @AR @AR AR AR DBF, p p, DBF WR, rf rf, WR r s h 		
1000	8	LD	r, m	ST	m, r
1001	9	SKE	m, #n4	SKGE	m, #n4
1010	A	MOV	@r, m	MOV	m, @r
1011	B	SKNE	m, #n4	SKLT	m, #n4

(to be continued)

(cont'd)

b_{14} to b_{11} \ b_{15}		0	1
BIN	HEX		
1100	C	BR addr	CALL addr
1101	D		MOV m, #4
1110	E		SKT m, #n
1111	F		SKF m, #n

2.2 LEGEND

AR	Address register
$(AR)_{ROM}$	Data of program memory represented by address register
ASR	Address stack register indicated by stack pointer
addr	Program memory address (lower 11 bits)
BANK	Bank register
CY	Carry flag
DBF	Data buffer
h	Halt release condition
INTEF	Interrupt enable flag
INTR	Automatically saved register during interrupt
INTSK	Interrupt stack register
IX	Index register
IXE	Index enable flag
MP	Data memory row address pointer
MPE	Memory pointer enable flag
m	Data memory address represented by m_R and m_C
m_R	Data memory row address (higher)
m_C	Data memory column address (lower)
n	Bit position (4 bits)
n4	Immediate data (4 bits)
PC	Program counter
p	Peripheral address
P_H	Peripheral address (upper 3 bits)
P_L	Peripheral address (lower 4 bits)
r	General register column address
rf	Register file address
rf_R	Register file address (higher 3 bits)
rf_C	Register file address (lower 4 bits)
SP	Stack pointer
s	Stop release condition
WR	Window register
(x)	Content addressed by x x: direct address such as m, r, etc. or register such as ASR, ect.

2.3 INSTRUCTION TABLE

NOTE 1	Mnemonic	Operands	Operation	Machine Code			
				Op. Code	Operand		
Addition	ADD	r, m	(r) ← (r) + (m)	00000	m_R	m_C	r
		m, #n4	(m) ← (m) + n4	10000	m_R	m_C	n4
	ADDC	r, m	(r) ← (r) + (m) + CY	00010	m_R	m_C	r
		m, #n4	(m) ← (m) + n4 + CY	10010	m_R	m_C	n4
	INC	AR	AR←AR+1	00111	000	1001	0000
		IX	IX←IX+1	00111	000	1000	0000
Subtranc-tion	SUB	r, m	(r) ← (r) − (m)	00001	m_R	m_C	r
		m, #n4	(m) ← (m) − n4	10001	m_R	m_C	n4
	SUBC	r, m	(r) ← (r) − (m) − CY	00011	m_R	m_C	r
		m, #n4	(m) ← (m) − n4 − CY	10011	m_R	m_C	n4
Logical Operation	OR	r, m	(r) ← (r) ∨ (m)	00110	m_R	m_C	r
		m, #n4	(m) ← (m) ∨ n4	10110	m_R	m_C	n4
	AND	r, m	(r) ← (r) ∧ (m)	00100	m_R	m_C	r
		m, #n4	(m) ← (m) ∧ n4	10100	m_R	m_C	n4
	XOR	r, m	(r) ← (r) ⊻ (m)	00101	m_R	m_C	r
		m, #n4	(m) ← (m) ⊻ n4	10101	m_R	m_C	n4
NOTE 2	SKT	m, #n	CMP←0, if (m) ∧ n = n, then skip	11110	m_R	m_C	n
	SKF	m, #n	CMP←0, if (m) ∧ n = 0, then skip	11111	m_R	m_C	n
Comparison	SKE	m, #n4	(m) − n4, skip if zero	01001	m_R	m_C	n4
	SKNE	m, #n4	(m) − n4, skip if not zero	01011	m_R	m_C	n4
	SKGE	m, #n4	(m) − n4, skip if not borrow	11001	m_R	m_C	n4
	SKLT	m, #n4	(m) − n4, skip if borrow	11011	m_R	m_C	n4
NOTE 3	RORC	r	$\to CY \to (r)_{b3} \to (r)_{b2} \to (r)_{b1} \to (r)_{b0}$	00111	000	0111	r
Transfer	LD	r, m	(r) ← (m)	01000	m_R	m_C	r
	ST	m, r	(m) ← (r)	11000	m_R	m_C	r

	MOV	@r, m	if MPE=1 : (MP, (r)) ← (m)
			if MPE=0 : (BANK, m_R, (r)) ← (m)
				01010	m_R	m_C	r

		m, @r	if MPE=1 : (m) ← (MP, (r))
			if MPE=0 : (m) ← (BANK, m_R, (r))
				11010	m_R	m_C	r

		m, #n4	(m) ←n4	11101	m_R	m_C	n4
	MOVT	DBF, @AR	SP←SP−1, ASR←PC, PC←AR, DBF← $(AR)_{ROM}$, PC←ASR, SP←SP+1	00111	000	0001	0000
	PUSH	AR	SP←SP−1, ASR←AR	00111	000	1101	0000
	POP	AR	AR←ASR, SP←SP+1	00111	000	1100	0000
	PEEK	WR, rf	WR← (rf)	00111	rf_R	0011	rf_C

(to be continued)

NOTE 1: Instruction Group
2: Decision
3: Rotation

(cont'd)

NOTE 1	Mnemonic	Operands	Operation	Machine Code			
				Op. Code	Operand		
Transfer	POKE	rf, WR	(rf) ←WR	00111	rf_R	0010	rf_C
	GET	DBF, p	DBF← (p)	00111	P_H	1011	P_L
	PUT	p, DBF	(p) ←DBF	00111	P_H	1010	P_L
Branch	BR	addr	PC_{10-0}←addr, PAGE←0	01100	addr		
			PC_{10-0}←addr, PAGE←1	01101			
		@AR	PC←AR	00111	000	0100	0000
Subroutine	CALL	addr	SP←SP−1, ASR←PC+1, PC_{10-0}←addr, PAGE←0	11100	addr		
		@AR	SP←SP−1, ASR←PC+1, PC←AR	00111	000	0101	0000
	RET		PC←ASR, SP←SP+1	00111	000	1110	0000
	RETSK		PC←ASR, SP←SP+1 and skip	00111	001	1110	0000
	RETI		PC←ASR, INTR←INTSK, SP←SP+1	00111	100	1110	0000
NOTE 2	EI		INTEF←1	00111	000	1111	0000
	DI		INTEF←0	00111	001	1111	0000
Others	STOP	s	STOP	00111	010	1111	s
	HALT	h	HALT	00111	011	1111	h
	NOP		No operation	00111	100	1111	0000

NOTE 1: Instruction Group
2: Interrupt

2.4 ASSEMBLER (AS17K) BUILT-IN MACRO INSTRUCTIONS

Legend

flag n	:	FLG type symbols
< >	:	Omission possible

	Mnemonic	Operands	Operation	n
Built-in macro instruction	SKTn	flag 1, ... flag n	if (flag 1) to (flag n) = all "1", then skip	1 ≦ n ≦ 4
	SKFn	flag 1, ... flag n	if (flag 1) to (flag n) = all "0", then skip	1 ≦ n ≦ 4
	SETn	flag 1, ... flag n	(flag 1) to (flag n) ← 1	1 ≦ n ≦ 4
	CLRn	flag 1, ... flag n	(flag 1) to (flag n) ← 0	1 ≦ n ≦ 4

	NOTn	flag 1, ... flag n	if (flag n) = "0", then (flag n) ← 1
			if (flag n) = "1", then (flag n) ← 0
				1 ≦ n ≦ 4

	INITFLG	<NOT> flag 1,
		...<<NOT> flag n>
			if description = NOT flag n, then (flag n) ← 0
			if description = flag n, then (flag n) ← 1
				1 ≦ n ≦ 4

CHAPTER 3. RESERVED SYMBOLS

The symbols which are defined by the uPD17216 device file are shown on the following pages.

The symbols which are defined are:

- Data buffers
- System registers
- Port registers
- Register files (control registers)
- Peripheral registers

3.1 DATA BUFFERS

Symbol	Attribute	Value	R/W	Description
DBF3	MEM	0.0CH	R/W	Data buffer bit 15 to bit 12
DBF2	MEM	0.0DH	R/W	Data buffer bit 11 to bit 8
DBF1	MEM	0.0EH	R/W	Data buffer bit 7 to bit 4
DBF0	MEM	0.0FH	R/W	Data buffer bit 3 to bit 0

3.2 SYSTEM REGISTERS

Symbol	At-trib-ute	Value	R/W	Description
AR3	MEM	0.74H	R	Address register bit 15 to bit 12 (fixed to 0)
AR2	MEM	0.75H	R/W	Address register bit 11 to bit 8
AR1	MEM	0.76H	R/W	Address register bit 7 to bit 4
AR0	MEM	0.77H	R/W	Address register bit 3 to bit 0
WR	MEM	0.78H	R/W	Window register
BANK	MEM	0.79H	R	Bank register (fixed to 0)
IXH	MEM	0.7AH	R	Index register high (bits 2, 1, 0: fixed to 0)
MPH	MEM	0.7AH	R	Memory pointer high (bits 2, 1, 0: fixed to 0)
MPE	FLG	0.7AH.3	R/W	Memory pointer enable flag
IXM	MEM	0.7BH	R/W	Index register middle (bit 3: fixed to 0)
MPL	MEM	0.7BH	R/W	Memory pointer low (bit 3: fixed to 0)
IXL	MEM	0.7CH	R/W	Index register low
RPH	MEM	0.7DH	R	General register pointer high
RPL	MEM	0.7EH	R/W	General register pointer low
PSW	MEM	0.7FH	R/W	Program status word
BCD	FLG	0.7EH.0	R/W	BCD flag
CMP	FLG	0.7FH.3	R/W	Compare flag
CY	FLG	0.7FH.2	R/W	Carry flag
Z	FLG	0.7FH.1	R/W	Zero flag
IXE	FLG	0.7FH.0	R/W	Index enable flag

3.3 PORT REGISTERS

Symbol	At-trib-ute	Value	R/W	Description
P0A0	FLG	0.70H.0	R/W	Port 0A bit 0
P0A1	FLG	0.70H.1	R/W	Port 0A bit 1
P0A2	FLG	0.70H.2	R/W	Port 0A bit 2
P0A3	FLG	0.70H.3	R/W	Port 0A bit 3
P0B0	FLG	0.71H.0	R/W	Port 0B bit 0
P0B1	FLG	0.71H.1	R/W	Port 0B bit 1
P0B2	FLG	0.71H.2	R/W	Port 0B bit 2
P0B3	FLG	0.71H.3	R/W	Port 0B bit 3
P0C0	FLG	0.72H.0	R/W	Port 0C bit 0
P0C1	FLG	0.72H.1	R/W	Port 0C bit 1
P0C2	FLG	0.72H.2	R/W	Port 0C bit 2
P0C3	FLG	0.72H.3	R/W	Port 0C bit 3
P0D0	FLG	0.73H.0	R/W	Port 0D bit 0
P0D1	FLG	0.73H.1	R/W	Port 0D bit 1
P0D2	FLG	0.73H.2	R/W	Port 0D bit 2
P0D3	FLG	0.73H.3	R/W	Port 0D bit 3
P0E0	FLG	0.6FH.0	R/W	Port 0E bit 0
P0E1	FLG	0.6FH.1	R/W	Port 0E bit 1
P0E2	FLG	0.6FH.2	R/W	Port 0E bit 2
P0E3	FLG	0.6FH.3	R/W	Port 0E bit 3

3.4 REGISTER FILES (CONTROL REGISTERS)

Symbol	At- trib- ute	Value	R/W	Description
SP	MEM	0.81H	R/W	Stack pointer
SYSCK	FLG	0.82H.0	R/W	System clock select flag
WDTRES	FLG	0.83H.3	R/W	Watchdog timer reset flag
BTMCK	FLG	0.83H.2	R/W	Basic interval timer mode selct flag
BTMRES	FLG	0.83H.1	R	Basic interval timer reset flag
INT	FLG	0.8FH.0	R	INT pin status flag
NRZBF	FLG	0.91H.0	R/W	NRZ buffer data flag
NRZ	FLG	0.92H.0	R/W	NRZ data flag
POEBPU3	FLG	0.97H.3	R/W	POE$_3$ pull-up setting flag
POEBPU2	FLG	0.97H.2	R/W	POE$_2$ pull-up setting flag
POEBPU1	FLG	0.97H.1	R/W	POE$_1$ pull-up setting flag
POEBPU0	FLG	0.97H.0	R/W	POE$_0$ pull-up setting flag
IEG	FLG	0.9FH.0	R/W	INT pin interrupt edge select flag
POEBIO3	FLG	0.0A7H.3	R/W	POE$_3$ input/output setting flag
POEBIO2	FLG	0.0A7H.2	R/W	POE$_2$ input/output setting flag
POEBIO1	FLG	0.0A7H.1	R/W	POE$_1$ input/output setting flag
POEBIO0	FLG	0.0A7H.0	R/W	POE$_0$ input/output setting flag
IPBTM	FLG	0.0AFH.2	R/W	Basic interval timer interrupt enable flag
IP	FLG	0.0AFH.1	R/W	INT pin interrupt enable flag
IPTM	FLG	0.0AFH.0	R/W	Timer interrupt enable flag

(to be continued)

(cont'd)

Symbol	Attribute	Value	R/W	Description
TMEN	FLG	0.0B3H.3	R/W	Timer enable flag
TMRES	FLG	0.0B3H.2	R/W	Timer reset flag
TMCK1	FLG	0.0B3H.1	R/W	Timer clock select flag
TMCK0	FLG	0.0B3H.0	R/W	Timer clock select flag
IRQBTM	FLG	0.0BDH.0	R/W	Basic interval timer interrupt request flag
IRQ	FLG	0.0BEH.0	R/W	INT pin interrupt request flag
IRQTM	FLG	0.0BFH.0	R/W	Timer interrupt request flag

3.5 PERIPHERAL REGISTERS

Symbol	At-tribute	Value	R/W	Description
NRZLTMM	DAT	03H	R/W	NRZ low level timer modulo register
NRZHTMM	DAT	04H	R/W	NRZ high level timer modulo register
TMC	DAT	05H	R	Timer count register
TMM	DAT	06H	W	Timer modulo register
AR	DAT	40H	R/W	Address register

3.6 RESERVED WORDS TABLE (ALPHABETICAL ORDER)

3.6.1 INSTRUCTIONS, PSEUDO-INSTRUCTIONS

ADD
ADDC
AND
BANK0
BELOW
BR
C14344
C4444
CALL
CASE
CLR1
CLR2
CLR3
CLR4
CSEG
DAT
DB
DI
DW
EI
EJECT
ELSE
END
ENDCASE
ENDIF
ENDIFC
ENDIFNC
ENDM
ENDOP
ENDP
ENDR
EOF
EXIT
EXITR
EXTRN
FLG
GET
GLOBAL
HALT
IF
IFCHAR
IFNCHAR
INC
INCLUDE
INITFLG
IRP
LAB
LBMAC
LD
LFCOND
LIST
LITERAL
LMAC
MACRO
MEM
MOV
MOVT
NIBBLE
NIBBLE1
NIBBLE2
NIBBLE2V
NIBBLE3
NIBBLE3V
NIBBLE4
NIBBLE4V
NIBBLE5
NIBBLE5V
NIBBLE6
NIBBLE6V
NIBBLE7
NIBBLE7V
NIBBLE8
NIBBLE8V
NOBMAC
NOLIST
NOMAC
NOP
NOT1
NOT2
NOT3
NOT4
OBMAC
OMAC
OPTION
OR
ORG
OTHER
PEEK
POKE
POP
PUBLIC
PURGE
PUSH
PUT
REPT
RET
RETI
RETSK
RORC
SBMAC
SET
SET1
SET2
SET3
SET4
SFCOND
SKE
SKF
SKF1
SKF2
SKF3
SKF4
SKGE
SKLT
SKNE
SKT
SKT1
SKT2
SKT3
SKT4
SMAC
ST
STOP
SUB
SUBC
SUMMARY
TAG
TITLE
XOR
ZZZERROR
ZZZMCHK
ZZZMSG
ZZZOPT

3.6.2 REGISTERS, FLAGS

AR
AR0
AR1
AR2
AR3
AR_EPA0
AR_EPA1
BANK
BCD
BTMCK
BTMRES
CMP
CY
DBF
DBF0
DBF1
DBF2
DBF3
IEG
INT
IP
IPBTM
IPTM
IRQ
IRQBTM
IRQTM
IX
IXE
IXH
IXL
IXM
MPE
MPH
MPL
NOUSEPOC
NRZ
NRZBF
NRZHTMM
NRZLTMM
OPEN
P0A0
P0A1
P0A2
P0A3
P0B0
P0B1
P0B2
P0B3
P0C0
P0C1
P0C2
P0C3
P0D0
P0D1
P0D2
P0D3
P0E0
P0E1
P0E2
P0E3
P0EBIO0
P0EBIO1
P0EBIO2
P0EBIO3
P0EBPU0
P0EBPU1
P0EBPU2
P0EBPU3
PSW
PULLUP
RPH
RPL
SP
SYSCK
TMC
TMCK0
TMCK1
TMEN
TMM
TMRES
USEPOC
WDTRES
WR
Z
ZZZ0
ZZZ1
ZZZ2
ZZZ3
ZZZ4
ZZZ5
ZZZ6
ZZZ7
ZZZ8
ZZZ9
ZZZALBMAC
ZZZALMAC
ZZZARGC
ZZZDEVID
ZZZEPA
ZZZLINE
ZZZLSARG
ZZZPRINT
ZZZSKIP
ZZZSYDOC

CHAPTER 4. MASK OPTION DEFINITION PSEUDO-INSTRUCTION

For uPD17216 programming, mask options must be specified using mask option definition pseudo-instructions in assembly source program.

Mask option should be specified for $\overline{\text{RESET}}$ pin pull-up resistor and low-voltage detection circuit.

4.1 OPTION, ENDOP PSEUDO-INSTRUCTIONS

The area from the OPTION pseudo-instruction to the ENDOP pseudo-instruction is called "mask option definition block".

The description of mask option definition block is shown below.

[Description format]

Symbol field	Mnemonic field	Operand field	Comment field
[label:]	OPTION		[; comment]
	:		
	:		
	ENDOP		

4.2 MASK OPTION DEFINITION PSEUDO-INSTRUCTION

An example of the mask option definition is shown below.

Symbol field	Mnemonic field	Operand field	Comment field
[label:]	OPTION		[; comment]
	OPTRES	PULLUP	; Pull up
	OPTPOC	NOUSEPOC	; No low-voltage detection circuit
	ENDOP		

Pseudo-instructions which can be used in a mask option definition block are shown in Table 4-1 "List of Mask Option Definition Pseudo-Instructions".

Table 4-1 List of Mask Option Definition Pseudo-Instructions

Item	Mask Option Definition Pseudo-Instruction	Number of Operands	1st Operand	2nd Operand
RESET pin pull-up resistor	OPTRES	1	RESET mask option	
			PULLUP (built-in pull-up resistor) OPEN (no pull-up resistor)	
Low-voltage detection circuit	OPTPOC	1	Low-voltage detection circuit mask option	
			USEPOC (built-in low-voltage detection circuit) NOUSEPOC (no low-voltage detection circuit)	

CHAPTER 5. LOAD MODULE FILE FORMAT

There are two types of output formats of the HEX format load module files output by the assembler (AS17K), ICE file and PRO file.

These two files should be used in accordance with their purposes, respectively. They contain the assembly environment information area, the in-circuit emulator operating environment information area, etc., in addition to the user program area.

(1) HEX format load module file formatting

Each data in the HEX format load module files output by the assembler is output as shown in the format examples below.

[HEX format load module file formatting examples]

```
: 10 0002 00 2B41000BFC80F...3A20 EC
|  |  |    |      |                |
①  ②  ③    ④      ⑤                ⑥


: 00 0000 01 FF
|  |   |   |  |
①  ②   ③   ④  ⑥
```

① Record mark

Indicates the start of a record.

② Number of codes (two digits)

Indicates the number of codes (byte data) contained in the record. The maximum of 10H (for 16 codes) is represented in hexadecimal numbers. For the last record, the value is 00H.

③ Address (4 digits)

Indicates the start address of the code represented by the record. At the final record, it becomes 0000H, which has no relation to the address.

④ Record type (2 digits)

00H indicates that the record is a data record, and 01H means that it is the final record.

⑤ Code (maximum 32 digits (16 bytes))

Up to 16 bytes are output to this field byte-wise.

⑥ Check sum (2 digits)

Byte data is output to ⑥ so that the least significant byte of the sum of data ②, ③, ④, ⑤ and ⑥ totaled in 1-byte units is 00H (even parity).

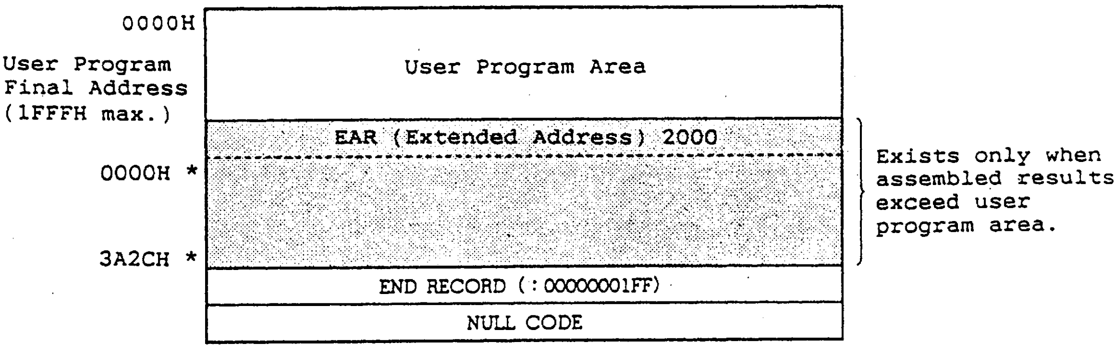
(2) ICE file

This file is output in the HEX format dedicated to the in-circuit emulator (IE-17K or IE-17K-ET) to which the AS17K outputs. The output format in the case of assembly using the uSxxxxAS17216 is shown in Figure 5-1.

The ICE file comprises 2 files. One is the program area which is divided into a user program area and patch area.
The patch area exists only when patches are used on the in-circuit emulator. The other consists of an in-circuit emulator operating environment information area, assembly environment information area and SE board environment information area. These areas contain various kinds of information which specifies in-circuit emulator operations.

Figure 5-1 ICE File Format

1st ... Program area


0000H
User Program Final Address (1FFFH max.)
User Program Area
EAR (Extended Address) 2000
0000H *
3A2CH *
END RECORD (: 00000001FF)
NULL CODE
Exists only when assembled results exceed user program area.

*: 10000H to 13A2CH in case of in-circuit emulator

2nd ... In-circuit emulator operating environment information area + assembly environment information area

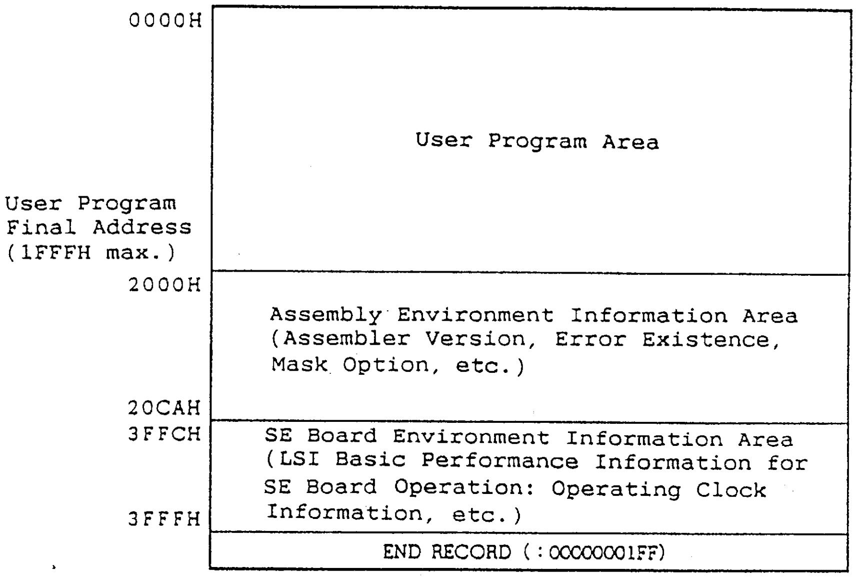
3A2DH	3E2EH	In-Circuit Emulator Operating Environment Information Area ① (Patch Information)
3E2FH	3F30H	In-Circuit Emulator Operating Environment Information Area ② (RAM Map)
3F31H	3FFBH	Assembly Environment Information Area (Assembler Version, Error Existence, Mask Option, etc.)
3FFCH	3FFFH	SE Board Environment Information Area (LSI Basic Performance Information for SE Board Operation: Operating Clock Information, etc.)
		END RECORD (: 00000001FF)

(3) PRO file

This is the HEX data dedicated PROM/one-time PROM products (uPD17P218) which are used for mask ordering or evaluation in a standalone SE board. During assembly, the PRO file is output by specifying it as /PRO by the assembly option. The output format in the case of assembly using the usxxxxAS17216 is shown in Figure 5-2.

The PRO file comprises one file which consists of a user program area, assembly environment information area, and SE board environment information area.

Figure 5-2 PRO File Format


0000H
User Program Area
User Program Final Address (1FFFH max.)
2000H
Assembly Environment Information Area (Assembler Version, Error Existence, Mask Option, etc.)
20CAH
3FFCH
SE Board Environment Information Area (LSI Basic Performance Information for SE Board Operation: Operating Clock Information, etc.)
3FFFH
END RECORD (: 00000001FF)

Remarks 1: The assembly environment information area also contains the mask option information which is used for the LSI mask creation. Thus, documents related to the mask option is not required at the mask ordering.

2: The area 20CBH to 3FFBH does not exist in the PRO file.

(4) File comparison of load module files

Even when there is no change in source files, the contents of assembler output results (assembly environment information area) may differ. This is because the assembly environment information area includes dates in which source files are created, etc.

Table 5-1 Items which may differ in assembler output results even when source file is not changed.

Item	Address	
	ICE File	PRO File
Program name (Maximum of 64 bytes of character string specified by assembly option (/'PROG='))	3F31H to 3F50H	2000H to 201FH
Mask option data	3F71H, 3F72H	2040H, 2041H
SIMPLEHOST information	3FADH	207CH
Existence of error or warning	3FB0H	207FH
Date of source file creation (year/month/date/hour/minute*)	3FBEH to 3FC7H	208DH to 2096H
Device name	3FC8H to 3FD7H	2097H to 20A6H
Device file version	3FDCH, 3FDDH	20ABH, 20ACH
Assembler version	3FDEH to 3FE1H	20ADH to 20B0H

*: When a source file is devided into two or more modules, the latest date of creation (year/month/date/hour/minute) in the module is written.

NOTE: Do not modifiy the load module file only.
The load module file should be modified by changing a source file or reassembling the file.
If only the load module file is changed, it will have a different history from that of other files, causing bugs to occur.