ABOV SEMICONDUCTOR 8-BIT SINGLE-CHIP MICROCONTROLLERS

HMS81C2232/2248 HMS81C2332/2348 MC80C2232/2248/2260 MC80C2332/2348

User's Manual (Ver. 2.03)



Version 2.03
Published by
FAE Team
©2006 ABOV semiconductor Co. Ltd. All right reserved.

Additional information of this manual may be served by ABOV semiconductor offices in Korea or Distributors and Representatives.

ABOV semiconductor reserves the right to make changes to any information here in at any time without notice.

The information, diagrams and other data in this manual are correct and reliable; however, ABOV semiconductor is in no way responsible for any violations of patents or other rights of the third party generated by the use of this manual.



REVISION HISTORY

VERSION 2.03 (OCT. 2011) This Book

Logo is changed.

The parameter name of "VFD Output Current" is changed to "FIP Output Current" in "7.4 DC Electrical Characteristics" on page 23. FIP leakage Current (I_{FL1}, I_{FL2}) is added in "7.4 DC Electrical Characteristics" on page 23.

VERSION 2.02 (NOV. 2008)

Added the MC80C2232/2248/2260 and MC80C2332/2348 built-in ESD immunity circuit and better quality. Eliminated the 80MQFP(14X14) package for HMS81C2232/48 and HMS87C2232/48. Fixed some errata.

VERSION 2.01 (NOV. 2007)

The micorocontroller division was transferred to ABOV Semiconductor Co., Ltd..

The company name, MagnaChip Semiconductor Inc. changed to ABOV Semiconductor Co., Ltd..

Figure 19-1 Block Diagram of FIP Controller/Driver was drawn in detail.

VERSION 2.00 (SEP. 2004)

The company name, Hynix Semiconductor Inc. changed to MagnaChip.

VERSION 1.01 (APR. 2004)

Fixed some errata.





Table of Contents

1. OVERVIEW	1
Description	1
HMS81C2232/48 and MC80C2232/48/60 Fea	a-
	Z
3	es.
Development Tools	3
Ordering Information	5
2. BLOCK DIAGRAM	6
3. PIN ASSIGNMENT	8
4. PACKAGE DIAGRAM	11
5. PIN FUNCTION	14
6. PORT STRUCTURES	18
7. ELECTRICAL CHARACTERISTICS	21
Absolute Maximum Ratings	21
Recommended Operating Conditions	21
A/D Converter Characteristics	22
DC Electrical Characteristics	23
AC Characteristics	25
AC Characteristics	26
Typical Characteristics	27
8. MEMORY ORGANIZATION	29
Registers	29
Program Memory	32
Data Memory	35
Control Registers	36
	41
9. I/O PORTS	45
PU and PUIO register:	45
P2 and P2IO register:	40 46
P3 Tegister:	40 46
P5 register:	40 16
P6 register:	4 0 47
10. CLOCK GENERATOR	48
11. BASIC INTERVAL TIMER	49
12. WATCHDOG TIMER	51
13. TIMER/EVENT COUNTER	54
8-bit Timer / Counter Mode	56
16-bit Timer / Counter Mode	60
8-bit Capture Mode	60
16-bit Capture Mode	64

	PWM Mode	65
	8-bit Compare Output (16-bit)	66
14.	REMOTE CONTROL TIMER Registers Controlling 8-Bit Remote Control Tim 68	. 68 ner
	Operation of 8-Bit Remote Control Timer	68
15.	ANALOG DIGITAL CONVERTER	73
16.	SERIAL PERIPHERAL INTERFACE (SPI1) . Transmission/Receiving Timing The method of Serial I/O The Method to Test Correct Transmission	76 78 79 79
17.	SERIAL PERIPHERAL INTERFACE (SPI3) . Transmission Timing The method of Serial I/O The Method to Test Correct Transmission	80 82 83 83
18.	BUZZER FUNCTION	84
19.	FIP CONTROLLER/DRIVER	86
13.	Function of FIP Controller/Driver Configuration of FIP Controller/Driver Registers Controlling FIP Controller/Driver One display period and blanking width Display Data Memory Key Scan Flag and Key Scan Data	86 87 87 89 90 91
20.	INTERRUPTS	95
	Interrupt Sequence Multi Interrupt External Interrupt	98 100 101
21.	POWER SAVING MODE	103
	Sleep Mode Stop Mode Internal RC-Oscillated Watchdog Timer Mode 107	103 104
~~		109
22.	OSCILLATOR CIRCUIT	111
23.	RESET External Reset Input Watchdog Timer Reset	112 112 112
24.	POWER FAIL PROCESSOR	113
25.	OTP PROGRAMMING	115 115 115 116



Timing Diagram of EPROM Mode120 AC/DC Requirements for Program/Read Mod 123

	Programming Flow Char	t124	
Α.	INSTRUCTION	i	ĺ
	Terminology List	i	i
	Instruction Map	ii	i

Instruction Setiii

B. MASK ORDER SHEET(HMS81C2232/2248) ... x

- C. MASK ORDER SHEET(HMS81C2332/2348) . xii
- D. MASK ORDER SHEET(MC80C2232/2248/2260) xiv
- E. MASK ORDER SHEET(MC80C2332/2348) ... xvi



HMS81C2232/2248/2332/2348 MC80C2232/2248/2260/2332/2348

CMOS Single-Chip 8-Bit Microcontroller with VFD Controller & VFD Driver

1. OVERVIEW

1.1 Description

The HMS81C2232/48 and MC80C2232/48/60 are advanced CMOS 8-bit microcontroller with 32/48/60K bytes of ROM(OTP:EPROM). This is a powerful micro-controller which provides a highly flexible and cost effective solution to many VFD applications. This provides the following standard features: 32K/48K bytes of ROM(EPROM), 896 bytes of RAM, 8-bit timer/counter, 8-bit A/D converter, 7-bit Watch dog Timer, Programmable Buzzer Driving Port, Serial Peripheral Interface, 8-bit remote control timer, on-chip oscillator and clock circuitry. It also comes with high voltage I/O pins that can directly drive a VFD(Vacuum Fluorescent Display). In addition, the HMS81C2232/48 and MC80C2232/48/60 support power saving modes to reduce power consumption.

Device Name	ROM Size	RAM Size	Display RAM	ОТР	Package	Remark
HMS81C2232	32K bytes	000 1 1	110h 400	HMS87C2232	80MQFP	
HMS81C2248	48K bytes	690 Dytes	596 Dytes 112Dytes	HMS87C2248	80TQFP	-
HMS81C2332	32K bytes			HMS87C2332	64SDIP	
HMS81C2348	48K bytes	896 bytes	896 bytes 112bytes	HMS87C2348	64MQFP 64LQFP	-
MC80C2232	32K bytes			HMS87C2232		
MC80C2248	48K bytes	896 bytes	112bytes	HMS87C2248	80MQFP 80TQFP	ESD Improvement
MC80C2260	60K bytes			-		improvement
MC80C2332	32K bytes			HMS87C2332	64SDIP	FSD
MC80C2348	48K bytes	896 bytes	112bytes	HMS87C2348	64MQFP 64LQFP	Improvement



1.2 HMS81C2232/48 and MC80C2232/48/60 Features

- 32K/48K/60K bytes ROM(OTP:EPROM)
- 896 Bytes of On-Chip Data RAM (Including STACK Area)
- 112 Bytes of On-Chip Display RAM
- Minimum Instruction Execution time:
 1uS at 4MHz (2cycle NOP Instruction)
 0.4uS at 10MHz (2cycle NOP Instruction)
- One 8-bit Basic Interval Timer
- One 8 bit Remote Timer
- One 7-bit Watch Dog Timer
- Two 8-bit Timer/Counters
- 10-bit High Speed PWM Output
- Two 8-bit Serial Peripheral Interface
 SIO1, SIO3
- Two External Interrupt Ports
- One Programmable 6-bit Buzzer Driving Port
- 53 FIP Ports - 40EA Output only pins
 - High-voltage pins Max. -40V

- Operating Temperature -40°C ~ 85°C
- 13 Interrupt Sources
 - Two External Sources (INTP0, INTP1)
 - Two Timer/Counter Sources (Timer0, Timer1)
 - Three Remote Timer Sources(FE,RE,OVF)
 - Two SPI Sources(SIO1, SIO3)
 - Three Functional Sources (ADC,WDT,BIT)
 - One Keyscan Source(KS)
- 7-Channel 8-bit On-Chip AD Converter
- Oscillator:
- Crystal
- Ceramic Resonator
- Low Power Dissipation Modes
 - STOP Mode
 - SLEEP Mode
- Operating Voltage: 2.7V ~ 5.5V (@8MHz)
 4.5V ~ 5.5V (@10MHz)
- Operating Frequency: 1MHz ~ 10MHz
- Enhanced EMS Improvement Power Fail Processor(Noise Immunity Circuit)



1.3 HMS81C2332/48 and MC80C2332/48 Features

- 32K/48K bytes ROM(EPROM)
- 896 Bytes of On-Chip Data RAM (Including STACK Area)
- 112 Bytes of On-Chip Display RAM
- Minimum Instruction Execution time:
 1uS at 4MHz (2cycle NOP Instruction)
 0.4uS at105MHz (2cycle NOP Instruction)
- One 8-bit Basic Interval Timer
- One 8 bit Remote Timer
- One 7-bit Watch Dog Timer
- Two 8-bit Timer/Counters
- 10-bit High Speed PWM Output
- One 8-bit Serial Peripheral Interface
 SIO1
- Two External Interrupt Ports
- One Programmable 6-bit Buzzer Driving Port
- 41 FIP Ports
 - 32EA Output only pins
 - High-voltage pins Max. -40V
- Operating Temperature -40°C ~ 85°C

1.4 Development Tools

The HMS81C22xx/23xx and MC80C22xx/23xx are supported by a full-featured macro assembler, an in-circuit emulator CHOICE-Dr.TM and OTP programmers. There are two different type programmers such as single type and gang type. For mode detail, Refer to "25. OTP PROGRAMMING" on page 115. Macro assembler operates under the MS-Windows 95 and upversioned Windows OS. But HMS800 C Compiler operates under the MS-Windows 2000 and upversioned Windows OS. Please contact sales part of ABOV Semiconductor.

Software	- MS-Windows Based Assembler - MS-Windows Based Debugger - HMS800 C compiler
In Circuit	- CHOICE-Dr.
Emulators	- CHOICE-Dr. EVA81C22 B/D

- 12 Interrupt Sources
 - Two External Sources (INTP0, INTP1)
 - Two Timer/Counter Sources (Timer0, Timer1)
 - Three Remote Timer Sources(FE,RE,OVF)
 - One SPI Sources(SIO1)
 - Three Functional Sources (ADC,WDT,BIT)
 - One Keyscan Source(KS)
- 5-Channel 8-bit On-Chip Analog to Digital Converter
- Oscillator:
 - Crystal
 - Ceramic Resonator
- Low Power Dissipation Modes
 - STOP Mode
 - SLEEP Mode
- Operating Voltage: 2.7V ~ 5.5V (@8MHz)
 4.5V ~ 5.5V (@10MHz)
- Operating Frequency: 1MHz ~ 10MHz
- Enhanced EMS Improvement Power Fail Processor (Noise Immunity Circuit)

	CHPOD81C23D-64SD (64SDIP)
POD	POD81C23D-64MQ-1420 (64MQFP) ¹
	POD81C23D-64LQ-1020 (64LQFP) ¹
	POD81C22D-80TQ-1414 (80TQFP) ¹
	POD81C22D-80MQ-1414 (80MQFP) ¹
	POD81C22D-80MQ-1420 (80MQFP) ¹
	OA87C23XX-64SD (64SDIP)
Cookot	OA87C23XX-64QF (64MQFP)
SOCKET Adapter for	OA87C23XX-64QT (64LQFP)
OTP	OA87C22XX-80QF (80MQFP 14X14)
0.1	OA87C22XX-80QFA (80MQFP 14X20)
	OA87C22XX-80QT (64TQFP)
	- CHOICE-SIGMA
OTP Writer	- PGM puls USB
	- Standalone GANG4

1. extra-cost options





Figure 1-1 CHOICE-Dr. (Emulator)



Figure 1-3 Standalone GANG4 II



Figure 1-2 CHOICE-SIGMA II



Figure 1-4 PGM plus USB



1.5 Ordering Information

HMS81C2232/48 ¹ HMS81C2332/48 ¹	Device name	ROM Size	RAM size	Package	Remark
Mask version	HMS81C2232 QA HMS81C2232 TQ HMS81C2248 QA HMS81C2248 TQ	32K bytes 32K bytes 48K bytes 48K bytes	896 bytes	80MQFP 80TQFP 80MQFP 80TQFP	-
OTP version	HMS87C2232 QA HMS87C2232 TQ HMS87C2248 QA HMS87C2248 TQ	32K bytes 32K bytes 48K bytes 48K bytes	896 bytes	80MQFP 80TQFP 80MQFP 80TQFP	-
Mask version	HMS81C2332 K HMS81C2332 Q HMS81C2332 LQ HMS81C2348 K HMS81C2348 Q HMS81C2348 LQ	32K bytes 32K bytes 32K bytes 48K bytes 48K bytes 48K bytes	896 bytes	64SDIP 64MQFP 64LQFP 64SDIP 64MQFP 64LQFP	-
OTP version	HMS87C2332 K HMS87C2332 Q HMS87C2332 LQ HMS87C2348 K HMS87C2348 Q HMS87C2348 LQ	32K bytes OTP 32K bytes OTP 32K bytes OTP 48K bytes OTP 48K bytes OTP 48K bytes OTP	896 bytes	64SDIP 64MQFP 64LQFP 64SDIP 64MQFP 64LQFP	-

 Pb Free Package : The "P" suffix will be added at original part number. For example: HMS81C2248 Q(Normal Package), HMS81C2248 QP(Pb Free Package) Pb/Halogen Free Package : The "B" suffix will be added at original part number. For example: HMS81C2248 Q(Normal Package), HMS81C2248 QB(Pb/Halogen Free Package)

MC80C2232/48/60 ¹ MC80C2332/48 ¹	Device name	ROM Size	RAM size	Package	Remark
	MC80C2232Q MC80C2232T MC80C2248Q MC80C2248T MC80C2260Q MC80C2260T	32K bytes 32K bytes 48K bytes 48K bytes 60K bytes 60K bytes	896 bytes	80MQFP 80TQFP 80MQFP 80TQFP 80MQFP 80TQFP	ESD Improvement
Mask version	MC80C2332K MC80C2332Q MC80C2332L MC80C2348K MC80C2348Q MC80C2348L	32K bytes 32K bytes 32K bytes 48K bytes 48K bytes 48K bytes	896 bytes	64SDIP 64MQFP 64LQFP 64SDIP 64MQFP 64LQFP	ESD Improvement

 Pb Free Package applied to MC80C2XXX basicallyt, so the "P" suffix is omitted. Halogen Free Package : The "B" suffix will be added at original part number. For example: MC80C2248Q(Pb free package), MC80C2248QB(Pb/Halogen Free Package)



2. BLOCK DIAGRAM



Figure 2-1 HMS81C2232/48 and MC80C2232/48/60 Block Diagram





Figure 2-2 HMS81C2332/48 and MC80C2332/48 Block Diagram



3. PIN ASSIGNMENT



Figure 3-1 HMS81C2232/48 and MC80C2232/48/60 80MQFP/TQFP Pin Assignment







Figure 3-2 HMS81C2332/48 and MC80C2332/48 64SDIP/MQFP Pin Assignment





Figure 3-3 HMS81C2332/48 and MC80C2332/48 64LQFP Pin Assignment



4. PACKAGE DIAGRAM



Figure 4-1 HMS81C2232/48 and MC80C2232/4/608 80MQFP(14X20) & 80TQFP(14X14) Package Diagram





Figure 4-2 HMS81C2332/48 and MC80C2332/48 64SDIP & 64MQFP Package Diagram





Figure 4-3 HMS81C2332/48 and MC80C2332/48 64LQFP Package Diagram



5. PIN FUNCTION

P00 ~ P07 (Port 0)

P00 through P07 are used as a 8-bit I/O port. These pins also have external interrupt request input and timer input functions in addition to the I/O port function. Port 0 can be set in the following operation modes in 1-bit units.

Port mode

P00 through P07 function as a 8-bit I/O port in this mode. This 8bit port can be set in the input or output mode in 1-bit units by the port 0 direction register (P0IO). When used as an input port, the internal pull-up resistor can be connected by using the pull-up resistor option register0 (PU0).

Alternate mode

P00 through P01 functions as external interrupt request input and P02 functions as timer input pin. P03 through P06 functions as ADC input pin and P07 functions as buzzer driver output pin.

INTP0, INTP1 : INTP0 and INTP1 input external interrupt requests whose valid edge can be specified (to be the rising edge, falling edge, or both the rising and falling edges).

TI : TI input timer of the 8-bit remote control timer.

ANI0, ANI1, ANDI2, ANI3 : These are input pins of the A/D converter.

BUZO : This is an output pin of buzzer driver output.

Port pin	Alternate function
P00	INTP0 (External interrupt 0)
P01	INTP1 (External interrupt 1)
P02	TI (Timer input of remote control timer)
P03	ANI0 (Analog Input 0)
P04	ANI1 (Analog Input 1)
P05	ANI2 (Analog Input 2)
P06	ANI3 (Analog Input 3)
P07	BUZO (Buzzer driver output)

P20 ~ P27 (Port 2)

P20 through P27 constitute an 8-bit I/O port. These pins also have functions to input/output data of the serial interface, clock. The following operation modes can be specified in 1-bit units.

Port mode

In this mode, P20 through P27 function as an 8-bit I/O port. This port can be set in the input or output mode in 1-bit units by using the port 2 direction register (P2IO). When the port is used as an input port, the internal pullup resistor can be used if so specified by the pull-up resistor option register 2 (PU2).

Alternate mode

In this mode, P20 through P21 are used to output serial interface data, clock. P22 through P24 functions as ADC input pin. P25 through P27 are used to input/output serial interface data, clock.

SI1, SO1, SO3 : These are I/O pins of the serial data of the serial interface.

SCK1, SCK3 : These are I/O pins of the serial clock of the serial interface.

ANI4, ANI5, ANI6 : These are input pins of the A/D converter.

Port pin	Alternate function
P20	SCK3 (Serial3 clock input/output)
P21	SO3 (Serial3 data output)
P22	ANI4 (Analog Input 4)
P23	ANI5 (Analog Input 5)
P24	ANI6 (Analog Input 6)
P25	SO1 (Serial1 data output)
P26	SI1 (Serial1data input)
P27	SCK1 (Serial1 clock input/output)

P30 ~ P37 (Port 3)

P30 through P37 constitute an 8-bit output port. These pins are also used as FIP controller/driver output pins. The following operation modes can be specified in 1-bit units.

Port mode

P30 through P37 function as an 8-bit output port in this mode. These pins are P-ch open-drain pins. Pull-down resistors can be connected to these pins of the mask ROM models by mask option. Pull-down resistor to V_{DISP} can be selected in 1-bit units.The HMS87C2232/48 does not have pull-down resistors.

· Alternate mode

In this mode, P30 through P37 function as the output pins of the FIP controller/driver (FIP24 through FIP31).

Port pin	Alternate function
P30~P37	FIP24-FIP31

P40 ~ P47 (Port 4)

P40 through P47 constitute an 8-bit output port. These pins are also used as FIP controller/driver output pins. The following operation modes can be specified in 1-bit units.

Port mode

P40 through P47 function as an 8-bit output port in this mode. These pins are P-ch open-drain pins. Pull-down resistors can be connected to these pins of the mask ROM models by mask op-



tion. Pull-down resistor to V_{DISP} can be selected in 1-bit units. The HMS87C2232/48 does not have pull-down resistors.

Alternate mode

In this mode, P40 through P47 function as the output pins of the FIP controller/driver (FIP32 through FIP39).

Port pin	Alternate function
P40~P47	FIP32-FIP39

P50 ~ P57 (Port 5)

P50 through P57 constitute an 8-bit I/O port. These pins are also used as FIP controller/driver output pins. The following operation modes can be specified in 1-bit units.

Port mode

P50 through P57 function as an 8-bit I/O port in this mode. These pins are P-ch open-drain pins. Pull-down resistors can be connected to these pins of the mask ROM models by mask option. Pull-down resistor to V_{DISP} or V_{SS0} can be selected in 1-bit units. The HMS87C2232/48 does not have pull-down resistors.

Alternate mode

In this mode, P50 through P57 function as the output pins of the FIP controller/driver (FIP40 through FIP47).

Port pin	Alternate function
P50~P57	FIP40-FIP47

P60 ~ P64 (Port 6)

P60 through P64 constitute a 5-bit I/O port. These pins are also used as FIP controller/driver output pins. The following operation modes can be specified in 1-bit units.

Port mode

P60 through P64 function as a 5-bit input/output port in this mode. These pins are P-ch open-drain pins. Pull-down resistors can be connected to these pins of the mask ROM models by mask option. Pull-down resistor to V_{DISP} or V_{SS0} can be selected in 1-bit units. The HMS87C2232/48 does not have pull-down resistors.

Alternate mode

In this mode, P60 through P64 function as the output pins of the FIP controller/driver (FIP48 through FIP52).

EC0 : This is input pin of the Event Counter0.

PWM10 : This is output pin of the PWM.

Port pin	Alternate function
P60	FIP48
P61	FIP49
P62	FIP50
P63	FIP51 / EC0
P64	FIP52 / PWM1O

FIP0 ~ FIP23

These are the output pins of the FIP controller/driver.

VDISP

This pin connects a pull-down resistor to the FIP controller/driver.

AV_{DD}

This pin supply an analog voltage to the A/D converter. Always keep this pin at the same potential as the V_{DD1} pin even when the A/D converter is not used.

AV_{SS}

This is the ground pin of the A/D converter. Always keep this pin at the same potential as the $V_{\rm SS1}$ pin even when the A/D converter is not used.

RESET

This pin inputs an active-low system reset signal.

X_{IN} and X_{OUT}

These pins connect a crystal resonator for main system clock oscillation. To supply an external clock, input it to X_{IN} , and input a signal reverse to that input to X_{IN} , to X_{OUT} .

ANI0 through ANI6

These are the input pins of the A/D converter.

$V_{DD0} \sim V_{DD2}$

 V_{DD0} supplies a positive voltage to the ports. V_{DD1} supplies a positive voltage to the internal function blocks other than the ports, analog block, and FIP controller/driver. V_{DD2} supplies a positive voltage to the FIP controller/driver.

V_{SS0} and V_{SS1}

 V_{SS0} is the ground pin for the ports. V_{SS1} is the ground pin for the internal function blocks other than the ports and analog block.



	In/Out	Functio	Function			
	m/Out	Basic	Alternate			
P00 (INTP0)	I/O (I)		External interrupt 0 input			
P01 (INTP1)	I/O (I)		External interrupt 1 input			
P02 (TI)	I/O (I)	Port0	Timer input of 8-bit remote control timer			
P03 (ANI0)	I/O (I)	8-bit I/O ports	Analog input channel 0 for A/D converter			
P04 (ANI1)	I/O (I)	Internal pull-up resistor can be used via software	Analog input channel 1 for A/D converter			
P05 (ANI2)	I/O (I)	when this port is used as input port	Analog input channel 2 for A/D converter			
P06 (ANI3)	I/O (I)		Analog input channel 3 for A/D converter			
P07 (BUZO)	I/O (O)		Buzzer driving output			
P20 (SCK3) ¹	I/O (I/O)		Serial3 clock input/output			
P21 (SO3) ¹	I/O (O)		Serial3 data output			
P22 (ANI4) ¹	I/O (I)	Port2	Analog input channel 4 for A/D converter			
P23 (ANI5) ¹	I/O (I)	8-bit I/O ports Can be set in input or output mode in 1-bit units.	Analog input channel 5 for A/D converter			
P24 (ANI6)	I/O (I)	Internal pull-up resistor can be used via software	Analog input channel 6 for A/D converter			
P25 (SO1)	I/O (O)	when this port is used as input port	Serial1 data output			
P26 (SI1)	I/O (I)		Serial1data input			
P27 (SCK1)	I/O (I/O)		Serial1clock input/output			
FIP0~FIP18 FIP19 ¹ ~FIP23 ¹	О	High voltage high-current of FIP controller/driver	-			
P30~P37 (FIP24-FIP31)	0	Port3 P-ch open-drain 8-bit high-voltage output port. Pull-down resistor for V _{DISP} can be used by mask option in 1-bit units(mask ROM models only) OTP models do not have pull-down resistor	FIP24-FIP31			
P40~P44 (FIP32-FIP36) P45 ¹ ~P47 ¹ (FIP37-FIP39) ¹	0	Port4 P-ch open-drain 8-bit high-voltage output port. Pull-down resistor for V _{DISP} can be used by mask option in 1-bit units(mask ROM models only) OTP models do not have pull-down resistor	FIP32-FIP39			
P50~P55 (FIP40-FIP45) P56 ¹ ~P57 ¹ (FIP46-FIP47) ¹	I/O (O)	Port5 P-ch open-drain 8-bit high-voltage output port. Can be set in input or output mode in 1-bit units. When it is used as input port, set the output latch to "0", and read port level read(P50 through P57). Pull-down resistor for V_{DISP} or V_{SS0} can be used by mask option in 1-bit units(mask ROM models only). OTP models do not have pull-down resistor.	FIP40-FIP47			

Table 5-1 HMS81C2232/48 and MC80C2232/48/60 Port Function Description



	lin (Quit	Functio	n
	In/Out	Basic	Alternate
P60 ¹	I/O (O)	Port6	FIP48
P61 ¹	I/O (O)	P-ch open-drain 8-bit high-voltage output port. Can be set in input or output mode in 1-bit units.	FIP49
P62	I/O (O)	When it is used as input port, set the output latch	FIP50
P63	I/O (I/O)	to "0", and read port level read(P50 through P57). Pull-down resistor for V_{DISP} or V_{SS0} can be used	FIP51/EC0
P64	I/O (O)	by mask option in 1-bit units(mask ROM models only). OTP models do not have pull-down resistor.	FIP52/Timer1 PWM 1 pulse output
AVDD	-	Analog power/reference voltage input to A/D converse Set the same potential as V_{DD}	erter
AVSS	-	Ground potential for A/D converter. Set the same potential as V_{SS}	
VDD0	-	Positive power supply to ports	
VSS0	-	Ground potential to ports.	
VDD1	-	Positive power supply to internal function block	
VSS1	-	Ground potential(except ports, analog block)	
VDD2	-	Positive power supply to FIP controller/driver.	
V _{disp}	-	Pull-down resistor connection of FIP controller/driv	er
RESET	I	System reset signal input	
XIN	I	Main system clock oscillation input	
XOUT	0	Main system clock oscillation output	

Table 5-1 HMS81C2232/48 and MC80C2232/48/60 Port Function Description

1. It is not supported to HMS81C2332/48 and MC80C2332/48.



 $V_{\underline{DD0}}$

6. PORT STRUCTURES

P00/INTP0~P01/INTP1, P02/TI, P26/SI1



P03/ANI0~P06/ANI3, P22/ANI4~P24/ANI6



Pull-up Pull-up Tr. Reg. SCK1,SO1 V_{DD0} SCK3 MUX Data Reg 本 SCK1_OUT_EN SO1_OUT_EN SCK3_OUT_EN 추 Pin Ť Data Bus √ss₀ V_{SS0} Direction Reg. MUX Rd SCK1(IN) IOSWIN Noise SCK3(IN) ¯⊾ Filter SCK1_IN_EN IOSW_EN SCK3_IN_EN P21/SO3, P07/BUZO



P20/SCK3, P25/SO1, P27/SCK1



P30/FIP24~P37/FIP31, P40/FIP32~P47/FIP39



P50/FIP40~P57/FIP47, P60/FIP48~P62/FIP50



P63/FIP51/EC0



P64/FIP52/PWM10





FIP00~FIP23



RESET



XIN, XOUT





7. ELECTRICAL CHARACTERISTICS

7.1 Absolute Maximum Ratings

Devenator	Symbol	Specif	Unit	
Parameter	Symbol	Min.	Max.	
	V _{DD}	-0.3 6.0		V
Parameter Supply Voltage Normal Voltage Pin High Voltage Pin Total Power Dissipation	AV _{DD}	V _{DD} -0.3	V _{DD} +0.3	V
	AV _{SS}	-0.3	0.3	V
	V _{DISP}	V _{DD} -45	V _{DD} +0.3	V
	V _{I1}	-0.3	V _{DD} +0.3	V
	V _{O1}	-0.3	V _{DD} +0.3	V
Normal Voltage Pin	I _{OH1}		mA	
	ΣI_{OH1}	-1	mA	
	I _{OL1}		mA	
	ΣI _{OL1}	Ę	50	mA
	V _{I2}	V _{DD} -45	V _{DD} +0.3	V
Llish) (altara Din	V _{O2}	V _{DD} -45	V _{DD} +0.3	V
High Voltage Pin	I _{OH2}	-30		mA
	ΣI_{OH2}	-120		mA
Total Power Dissipation	PT	7	00	mW
Storage Temperature	TSTG	-65	150	°C

Note: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

7.2 Recommended Operating Conditions

Parameter	Symbol	Condition	Specifi	Unit	
	Symbol	Condition	Min.	Max.	Unit
Supply Voltage	V _{DD}	f _{XI} = 1 ~ 8MHz	2.7	5.5	V
		f _{XI} = 10MHz	4.5	5.5	V
Operating Temperature	T _{OPR}	V _{DD} = 2.7V~5.5V	-40	85	°C



7.3 A/D Converter Characteristics

 $(T_A=25^{\circ}C, V_{DD}=5.12V, V_{SS}=0V, AV_{DD}=5.12V, AV_{SS}=0V @f_{XIN}=5MHz)$

				Specifications			
Parameter	Symbol	Condition	Min.	Typ. ¹	Max.	Unit	
Analog Power Supply Input Voltage Range	AV _{DD}		AV _{SS}	-	V _{DD}	V	
Analog Input Voltage Range	V _{AN}		AV _{SS}		AV _{DD}	V	
Current Following Between AV_{DD} and AV_{SS}	I _{AVDD}		-	-	200	uA	
Overall Accuracy	CA _{IN}		-	-	±2	LSB	
Non-Linearity Error	N _{NLE}		-	-	±2	LSB	
Differential Non-Linearity Error	N _{DNLE}		-	-	±2	LSB	
Zero Offset Error	N _{ZOE}		-	-	±2	LSB	
Full Scale Error	N _{FSE}		-	-	±2	LSB	
Gain Error	N _{NLE}		-	-	±2	LSB	
Conversion Time	T _{CONV}	f _{XIN} =4MHz	-	-	30	uS	

1. Data in "Typ" column is at 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.



7.4 DC Electrical Characteristics

(V_{DD} = 5.0V ± 10%, V_{SS} = 0V, T_A = -40 ~ 85°C, f_{XIN} = 5 MHz, Vdisp = V_{DD}-40V to V_{DD})

Deveryoten	Cumhal	Dim	Test Candition	S	specificatio	on	11
Parameter	Symbol	Pin	Test Condition	Min	Тур.	Мах	Unit
	V _{IH1}	RESET,SI1,INTP0,INTP1,TI, SCK1,EC0		0.8V _{DD}		V _{DD} +0.3	
Input High Voltage	V _{IH2}	P00~P07,P20~P27,P50~P57 P60~P64		0.7V _{DD}		V _{DD} +0.3	V
	V _{IH3}	XIN	External Clock	0.9V _{DD}		V _{DD} +0.3	
	V _{IL1}	RESET,SI1,INTP0,INTP1,TI, SCK1,EC0,SCK3		-0.3		0.2V _{DD}	
Input Low Voltage	V	P00~P07,P20~P27		-0.3		0.3V _{DD}	V
	VIL2	P50~P57,P60~P64		V _{DD} -40V		0.3V _{DD}	
	V _{IL3}	XIN	External Clock	-0.3		0.1V _{DD}	
	M	P00~P07,P20~P27	I _{OH} = -1.0mA	V _{DD} -1.0		V _{DD}	
Output High Voltage	VOH1	P00~P07,P20~P27	I _{OH} = -100uA	V _{DD} -0.5		V _{DD}	V
Voltage V _{OH2}		XOUT	I _{OH} = -50uA	V _{DD} -2.0		V _{DD}	
Output Low	V _{OL1}	P00~P07,P20~P27	I _{OL} = 400uA	0		0.5	
Voltage	V _{OL2}	XOUT	I _{OL} = 50uA	0		2	V
Input High Leakage Current	I _{IH1}	P00~P07,P20~P27,P50~P57 P60~P64,RESET ¹ V _{IN} = V _{DD}		3	uA		
Input Low		P00~P07,P20~P27,P50~P57 P60~P64,RESET	V _{IN} = 0			-3	uA
Leakage Current	I _{IL2}	P50~P57,P60~P64 V _{IN} = V _{DD} -40V			-10		
Input Pull-up Resistor(*Option)	R _{PU}	P00~P07,P20~P27,RESET ²	V _{DD} =5V	10	60	100	KΩ
OSC Feed Back Resistor	R _{FB}	XIN, XOUT	V _{DD} =5V	0.25	1.25	2.5	MΩ
	I _{OD1} FIP00~FIP19		V _{OD} =V _{DD} -2V			-15	A
	I _{OD2}	FIP20~FIP52	V _{OD} =V _{DD} -2V			-5	mA
FIP Leakage	I _{FL1}	FIP00~FIP19	V _{IN} =V _{DD} -40V			-10	
Current	I _{FL2} FIP20~FIP52		V _{IN} =V _{DD} -40V			-10	UA
On-Chip Mask	R _{D1}	P50~P57,P60~P64	V _{SS0} Connection	10	25	90	KΩ
Option Pull-down Resistance	R _{D2}	FIP00~FIP52	V _{DISP} Connection V _{DD} -V _{DISP} =40V	20	40	135	KΩ
Power Fail Detect Voltage	V _{PFD}	V _{DD}			2.7		V
Current Dissipation in Active Mode	I _{DD}	V _{DD}	fxin=5MHz		5	10	mA



Demonster	O	D:		S	11		
Parameter	Symbol	Pin	Test Condition	Min	Тур.	Max	Unit
Current Dissipation in SLEEP Mode	I _{SLEEP}	V _{DD}	fxin=5MHz		1	3	mA
Current Dissipation in STOPO Mode	I _{STOP}	V _{DD}	fxin=Off		1	15	uA
Internal RC WDT Frequency	T _{RCWDT}	XOUT		10		30	KHz

In case of <u>RESET</u> pull-up not connected (Mask Option).
 In case of <u>RESET</u> pull-up connected (Mask Option)



7.5 AC Characteristics

 $(T_A = -40 \sim 85^{\circ}C, V_{DD} = 5V \pm 10\%, V_{SS} = 0V)$

Deremeter	Symbol Pins		S	Unit		
Farameter	Symbol	FIIIS	Min.	Тур.	Max.	Unit
Operating Frequency	f _{CP}	XIN	1	-	10	MHz
System Clock Cycle Time	t _{SYS}	-	200		2000	nS
Oscillation Stabilizing Time	t _{ST}	XIN, XOUT	-	-	20	mS
External Clock Pulse Width	t _{CPW}	XIN	40	-	-	nS
External Clock Transition Time	t _{RCP} ,t _{FCP}	XIN	-	-	10	nS
Interrupt Input Pulse Width	t _{EPW}	INTP0, INTP1	2	-	-	t _{SYS}
Event Counter Input Pulse Width	t _{ECW}	EC0	2			t _{SYS}
Event Counter Transition Time	t _{REP} ,t _{FEP}	EC0	-	-	20	nS
RESET Input Width	t _{RST}	RESET	8	-	-	t _{SYS}



Figure 7-1 Timing Chart



7.6 AC Characteristics

(T_A=-40~+85°C, V_{DD}=5V±10%, V_{SS}=0V, f_{XIN}=4MHz)

Demonster	Symbol Bins		S	Specifications			
Parameter	Symbol	Pins	Min.	Тур.	Max.	Unit	
Serial Input Clock Pulse	t _{SCYC}	SCK1, SCK3	2t _{SYS} +200	-	-	ns	
Serial Input Clock Pulse Width	t _{SCKW}	SCK1, SCK3	t _{SYS} +70	-	-	ns	
Serial Input Clock Pulse Transition Time	t _{FSCK} t _{RSCK}	SCK1, SCK3	-	-	30	ns	
Serial Output Clock Cycle Time	t _{SCYC}	SCK1, SCK3	4t _{SYS}	-	16t _{SYS}	ns	
Serial Output Clock Pulse Width	t _{SCKW}	SCK1, SCK3	2t _{SYS} -30			ns	
Serial Output Clock Pulse Transition Time	t _{FSCK} t _{RSCK}	SCK1, SCK3			30	ns	
Serial Output Delay Time	t _{DS}	SO			100	ns	
SI Input Pulse Transition Time	t _{FSIN} t _{RSIN}	SI1	-	-	30	ns	
SI Input Setup Time (External SCK)	t _{SUS}	SI1	100	-	-	ns	
SI Input Setup Time (Internal SCK)	t _{SUS}	SI1	200	-		ns	
SI Input Hold Time	t _{HS}	SI1	t _{SYS} +100	-		ns	



Figure 7-2 Serial I/O Timing Chart



7.7 Typical Characteristics

This graphs and tables provided in this section are for design guidance only and are not tested or guaranteed.

In some graphs or tables the data presented are outside specified operating range (e.g. outside specified V_{DD} range). This is for information only and devices are guaranteed to operate properly only within the specified range.

The data presented in this section is a statistical summary of data collected on units from different lots over a period of time. "Typical" represents the mean of the distribution while "max" or "min" represents (mean + 3σ) and (mean - 3σ) respectively where σ is standard deviation









8. MEMORY ORGANIZATION

The HMS81C2232/48 and MC80C2232/48/60 have separate address spaces for Program memory and Data Memory. Program memory can only be read, not written to. It can be up to 32K/48K

8.1 Registers

This device has six registers that are the Program Counter (PC), a Accumulator (A), two index registers (X, Y), the Stack Pointer (SP), and the Program Status Word (PSW). The Program Counter consists of 16-bit register.



Figure 8-1 Configuration of Registers

Accumulator: The Accumulator is the 8-bit general purpose register, used for data operation such as transfer, temporary saving, and conditional judgement, etc.

The Accumulator can be used as a 16-bit register with Y Register as shown below.



Figure 8-2 Configuration of YA 16-bit Register

X, Y Registers: In the addressing mode which uses these index registers, the register contents are added to the specified address, which becomes the actual address. These modes are extremely effective for referencing subroutine tables and memory tables. The index registers also have increment, decrement, comparison and data transfer functions, and they can be used as simple accumulators.

Stack Pointer: The Stack Pointer is an 8-bit register used for occurrence interrupts and calling out subroutines. Stack Pointer identifies the location in the stack to be access (save or restore). bytes of Program memory. Data memory can be read and written to up to 896 bytes including the stack area.

Generally, SP is automatically updated when a subroutine call is executed or an interrupt is accepted. However, if it is used in excess of the stack area permitted by the data memory allocating configuration, the user-processed data may be lost.

The stack can be located at any position within $100_{\rm H}$ to $1{\rm FF}_{\rm H}$ of the internal data memory. The SP is not initialized by hardware, requiring to write the initial value (the location with which the use of the stack starts) by using the initialization routine. Normally, the initial value of "FF_H" is used.



Note:The Stack Pointer must be initialized by software because its value is undefined after RESET.Example:To initialize the SPLDX#0FFHTXSP; SP \leftarrow FF_H

Program Counter: The Program Counter is a 16-bit wide which consists of two 8-bit registers, PCH and PCL. This counter indicates the address of the next instruction to be executed. In reset state, the program counter has reset routine address ($PC_H:OFF_H$, $PC_L:OFE_H$).

Program Status Word: The Program Status Word (PSW) contains several bits that reflect the current state of the CPU. The PSW is described in Figure 8-3. It contains the Negative flag, the Overflow flag, the Break flag the Half Carry (for BCD operation), the Interrupt enable flag, the Zero flag, and the Carry flag.

[Carry flag C]

This flag stores any carry or borrow from the ALU of CPU after an arithmetic operation and is also changed by the Shift Instruction or Rotate Instruction.

[Zero flag Z]

This flag is set when the result of an arithmetic operation or data transfer is "0" and is cleared by any other result.





Figure 8-3 PSW (Program Status Word) Register

[Interrupt disable flag I]

This flag enables/disables all interrupts except interrupt caused by Reset or software BRK instruction. All interrupts are disabled when cleared to "0". This flag immediately becomes "0" when an interrupt is served. It is set by the EI instruction and cleared by the DI instruction.

[Half carry flag H]

After operation, this is set when there is a carry from bit 3 of ALU or there is no borrow from bit 4 of ALU. This bit can not be set or cleared except CLRV instruction with Overflow flag (V).

[Break flag B]

This flag is set by software BRK instruction to distinguish BRK from TCALL instruction with the same vector address.

[Direct page flag G]

This flag assigns RAM page for direct addressing mode. In the direct addressing mode, addressing area is from zero page 00_H to $0FF_H$ when this flag is "0". If it is set to "1", addressing area is assigned 100_H to $1FF_H$. It is set by SETG instruction and cleared by CLRG.

[Overflow flag V]

This flag is set to "1" when an overflow occurs as the result of an arithmetic operation involving signs. An overflow occurs when the result of an addition or subtraction exceeds $+127(7F_H)$ or $-128(80_H)$. The CLRV instruction clears the overflow flag. There is no set instruction. When the BIT instruction is executed, bit 6 of memory is copied to this flag.

[Negative flag N]

This flag is set to match the sign bit (bit 7) status of the result of a data or arithmetic operation. When the BIT instruction is executed, bit 7 of memory is copied to this flag.




Figure 8-4 Stack Operation



8.2 Program Memory

A 16-bit program counter is capable of addressing up to 64K bytes, but this device has 32/48K/60K bytes program memory space only physically implemented. Accessing a location above FFFF_H will cause a wrap-around to $0000_{\rm H}$.

Figure 8-5, shows a map of Program Memory. After reset, the CPU begins execution from reset vector which is stored in address $FFFE_H$ and $FFFF_H$ as shown in Figure 8-6.

As shown in Figure 8-5, each area is assigned a fixed location in Program Memory. Program Memory area contains the user program.



Figure 8-5 Program Memory Map

Page Call (PCALL) area contains subroutine program to reduce program byte length by using 2 bytes PCALL instead of 3 bytes CALL instruction. If it is frequently called, it is more useful to save program byte length.

Table Call (TCALL) causes the CPU to jump to each TCALL address, where it commences the execution of the service routine. The Table Call service area spaces 2-byte for every TCALL: $0FFC0_{\rm H}$ for TCALL15, $0FFC2_{\rm H}$ for TCALL14, etc., as shown in Figure 8-7.

Example: Usage of TCALL



The interrupt causes the CPU to jump to specific location, where it commences the execution of the service routine. The External interrupt 0, for example, is assigned to location $0FFFA_H$. The interrupt service locations spaces 2-byte interval: $0FFF8_H$ and $0FFF9_H$ for External Interrupt 1, $0FFFA_H$ and $0FFFB_H$ for External Interrupt 0, etc.

Any area from $0FF00_H$ to $0FFFF_H$, if it is not going to be used, its service location is available as general purpose Program Memory.



Figure 8-6 Interrupt Vector Area





Figure 8-7 PCALL and TCALL Memory Area

$\textbf{PCALL} \rightarrow \textbf{rel}$

4F35 PCALL 35H



$\textbf{TCALL} \rightarrow \textbf{n}$

4A TCALL 4





; * * * * * * * * * * * * * * * * * * *	******	* * * * * * * * * * * * * * * * * *	***************************************
; ******************	Interr ******	upt Vector Table *******	≥ ************************************
	ORG DW DW DW DW DW DW DW DW DW DW DW DW DW	OFFEOH Not_Used Not_Used Not_Used Not_Used Timer1_ISR Not_Used Not_Used Not_Used Not_Used Not_Used Not_Used Not_Used Not_Used Not_Used Not_Used Not_ISR Reset	<pre>; Basic Interval Timer ; Watch Dog Timer ; A/D Converter ; Not Used ; Not Used ; Timer1 ; Timer0 ; SIO3 ; SIO1 ; Key Scan Interrupt ; Remote Timer2(Overflow) ; Remote Falling Edge ; Remote Rising Edge ; Ext.INT1 ; Ext.INT0 ; Reset</pre>
;*************	******	*************	************************************;
;	Progra	m Initial Part	;
; * * * * * * * * * * * * * * * * * * *		лоь	·UMS8102218/2318 Drogram Start
Reset:	01/0040	0011	, missicz240/2340 ilogiam start
	DI		;Disable All Interrupt
	;=====		=;
	; RAM ;===== LDX LDY	Clear Routine ====================================	; =;
RAM Clear0:		11 0	
	LDA STA CMPX BNE	#0 {X}+ #0C0h RAM_Clear0	;Page0 RAM Clear(0000h ~ 00BFh)
	INC STY SETG	Y !RPR	;Page Select
	T'DX	#0	
RAM_Clear1:	LDA STA CMPX BNE	#0 {X}+ #00h RAM_Clear1	
	INC CMPY BCS	Y #5 RAM_Clear_Bye	;Page1~4 RAM Clear(0100h ~ 04FFh)
	STY SETG BRA	!RPR RAM_Clear1	;
RAM_Clear_Bye:	CLRG LDX TXSP	#0FFh	;Page0 Select ;Initial Stack Pointer
	CALL CALL EI	Initial_IO Initial_Reg	;I/O Port Initial ;Register Initial ;Enable Interrupt
;*************	******	* * * * * * * * * * * * * * * * * * *	***************************************
; ;***********************************	Main P	rogram Part ****************	; *************************************
Main:	BRA	Main	

Example: The usage software example of Vector address for HMS81C2232/48 and MC80C2232/48/60.



8.3 Data Memory

Figure 8-8 shows the internal Data Memory space available. Data Memory is divided into three groups, a user RAM (including Stack), control registers and FIP display memory.



Figure 8-8 Data Memory Map

User Memory

The HMS81C2232/48 and MC80C2232/48/60 have 896×8 bits for the user memory (RAM). RAM pages are selected by RPR.

Note: After setting RPR(RAM Page Select Register), be sure to execute SETG instruction. When executing CLRG instruction, be selected PAGE0 regardless of RPR.

Control Registers

The control registers are used by the CPU and Peripheral function blocks for controlling the desired operation of the device. Therefore these registers contain control and status bits for the interrupt system, the timer/ counters, analog to digital converters and I/O ports. The control registers are in address range of $0C0_H$ to $0FF_H$.

Note that unoccupied addresses may not be implemented on the chip. Read accesses to these addresses will in general return random data, and write accesses will have an indeterminate effect.

More detailed informations of each register are explained in each peripheral section.

Note: Write only registers can not be accessed by bit manipulation instruction. Do not use read-modify-write instruction. Use byte manipulation instruction, for example "LDM".

Example; To write at CKCTLR

LDM CLCTLR, #09H ; Divide ratio(÷16)

Stack Area

The stack provides the area where the return address is saved before a jump is performed during the processing routine at the execution of a subroutine call instruction or the acceptance of an interrupt.

When returning from the processing routine, executing the subroutine return instruction [RET] restores the contents of the program counter from the stack; executing the interrupt return instruction [RETI] restores the contents of the program counter and flags.

The save/restore locations in the stack are determined by the stack pointed (SP). The SP is automatically decreased after the saving, and increased before the restoring. This means the value of the SP indicates the stack location number for the next save. Refer to Figure 8-4 on page 31.



Figure 8-9 RPR(RAM Page Select Register)



8.4 Control Registers

Address	Symbol	Register Name	R/W	Reset Value	Addressing
Address	Cymbol				Mode
00C0	P0	Port0 data register	R/W	00000000	Byte,Bit
00C1	P0IO	Port0 I/O direction register	W	00000000	Byte
00C2	-	Reserved	-	-	-
00C3	-	Reserved	-	-	-
00C4	P2	Port2 data register	R/W	00000000	Byte,Bit
00C5	P2IO	Port2 I/O direction register	W	00000000	Byte
00C6	P3	Port3 data register	R/W	00000000	Byte,Bit
00C7	-	Reserved	-	-	-
00C8	P4	Port4 data register	R/W	00000000	Byte,Bit
00C9	-	Reserved	-	-	-
00CA	P5	Port5 data register	R/W	0000000	Byte,Bit
00CB	-	Reserved	-	-	-
00CC	P6	Port6 data register	R/W	0 0 0 0 0	Byte,Bit
00CD	-	Reserved	-	-	-
00CE	-	Reserved	-	-	-
00CF	-	Reserved	-	-	-
00D0	T0CR	Timer0 mode control register	R/W	0 0 0 0 0 0	Byte,Bit
	Т0	Timer0 register	R	00000000	Byte
00D1	T0DR	Timer0 data register	W	1 1 1 1 1 1 1 1	Byte
	CDR0	Timer0 capture data register R 0		00000000	Byte
00D2	T1CR	Timer1 mode control register	R/W	00000000	Byte,Bit
0003	T1DR	Timer1 data register	W	1 1 1 1 1 1 1 1	Byte
00D3	T1PPR	Timer1 PWM period register	W	1 1 1 1 1 1 1 1	Byte
	T1	Timer1 register	R	00000000	Byte
00D4	T1PDR	Timer1 PWM duty register	R/W	00000000	Byte,Bit
	CDR1	Timer1 capture data register R 0		00000000	Byte
00D5	PWM1HR	Timer1 PWM high register	W	0 0 0 0	Byte
00D6	-	Reserved	-	-	-
00D7	-	Reserved	-	-	-
00D8	-	Reserved	-	-	-
00D9	-	Reserved	-	-	-
00DA	-	Reserved	-	-	-
00DB	_	Reserved	-	-	-
00DC	SIO3M	SIO3 mode control register	R/W	0 0 0 0 0 1	Byte,Bit
00DD	SIO3R	SIO3 Data shift register	R/W	Undefined	Byte,Bit

Table 8-1 Control Registers



Addross	Symbol	Pogistor Namo	D/M	Reset Value	Addressing
Address	Symbol	Register Name R/W		7 6 5 4 3 2 1 0	Mode
00DE	BUR	Buzzer driver register	W	1 1 1 1 1 1 1 1	Byte
00DF	RPR	RAM page selection register	R/W	0 0 1	Byte,Bit
00E0	SIO1M	SIO1 mode control register	R/W	00000001	Byte,Bit
00E1	SIO1R	SIO1 Data shift register	R/W	Undefined	Byte,Bit
00E2	IENH	Interrupt enable register high	R/W	000000000	Byte,Bit
00E3	IENL	Interrupt enable register low	R/W	00000-	Byte,Bit
00E4	IRQH	Interrupt request flag register high	R/W	000000000	Byte,Bit
00E5	IRQL	Interrupt request flag register low	R/W	00000-	Byte,Bit
00E6	IEDS	External interrupt edge selection register	R/W	0 0 0 0	Byte,Bit
00E7	RTCR	Remote Timer control register	R/W	000000000	Byte,Bit
	RT	Remote Timer register	R	000000000	Byte
00E8	RTDR	Remote Timer data register	W	1 1 1 1 1 1 1 1	Byte
	RTCP0	Remote Timer capture register0	R	000000000	Byte
00E9	RTCP1	Remote Timer capture register1	R	000000000	Byte
00EA	ADCM	A/D converter mode register	R/W	- 0 0 0 0 0 1	Byte,Bit
00EB	ADCR	A/D converter data register	R	Undefined	Byte
	BITR	Basic interval timer register	R	Undefined	Byte
OOEC	CKCTR	Clock control register	W	0 1 0 1 1 1	Byte
0055	MOTO		R	000000000	Byte
UUED	WDIR	Watchdog Timer Register	W	0 1 1 1 1 1 1 1	Byte
00EE	-	Reserved	-	-	-
00EF	PFDR	Power fail detection register	R/W	100	Byte,Bit
00F0	DSPM0	Display mode register0	R/W	0 - 0 1 0 0 0 0	Byte,Bit
00F1	DSPM1	Display mode register1	R/W	00000001	Byte,Bit
00F2	DSPM2	Display mode register2	R/W 0000		Byte,Bit
00F3	-	Reserved	-	-	-
00F4	PSR	Port selection register	W	0 0 0 0 0	Byte
00F5	-	Reserved	-	-	-
00F6	-	Reserved	-	-	-
00F7	-	Reserved	-		
00F8	-	Reserved	-	-	-
00F9	-	Reserved		-	
00FA	SCMR	System clock mode register	R/W	0 0	Byte,Bit
00FB	SMR	Sleep mode register	W	0	Byte
00FC	PU0	Pull-up resistor option register0	R/W	000000000	Byte,Bit
00FD	-	Reserved	-	-	-

Table 8-1 Control Registers



Address	Symbol	Register Name	R/W	Reset Value 7 6 5 4 3 2 1 0	Addressing Mode
00FE	PU2	Pull-up resistor option register2	R/W	000000000	Byte,Bit
00FF	STPC	Stop control register	W	000000000	Byte

Table 8-1 Control Registers



Registers are controlled by byte manipulation instruction such as LDM etc., do not use bit manipulation instruction such as SET1, CLR1 etc. If bit manipulation instruction is used on these registers, content of other seven bits are may varied to unwanted value.

R/W

Registers are controlled by both bit and byte manipulation instruction.

- : this bit location is reserved.



Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
C0H	P0	Port0 Data	Register (Bi	t[7:0])					
C1H	P0IO	Port0 Direc	tion Registe	r (Bit[7:0])					
C2H	-	Reserved							
СЗН	-	Reserved							
C4H	P2	Port2 Data	Register (Bi	t[7:0])					
C5H	P2IO	Port2 Direc	tion Registe	r (Bit[7:0])					
C6H	P3	Port3 Data	Register (Bi	t[7:0])					
C7H	-	Reserved							
C8H	P4	Port4 Data	Register (Bi	t[7:0])					
C9H	-	Reserved							
CAH	P5	Port5 Data	Register (Bi	t[7:0])					
CBH	-	Reserved							
ССН	P6	Port6 Data	Register (Bi	t[4:0])					
CDH	-	Reserved	Reserved						
CEH	-	Reserved	Reserved						
CFH	-	Reserved	1	1	1	1	1		
D0H	T0CR	-	-	CAP0	T0CK2	T0CK1	T0CK0	T0CN	TOST
D1H	T0/T0DR/ CDR0	Timer0 Reg Capture0 D	Timer0 Register/Timer0 Data Register Capture0 Data Register						
D2H	T1CR	POL	16BIT	PWM1E	CAP1	T1CK1	T1CK0	T1CN	T1ST
D3H	T1DR T1PPR	Timer1 Dat PWM1 Per	Timer1 Data Register PWM1 Period Register						
D4H	T1/CDR1 T1PDR	Timer1 Reg PWM1 Dut	Timer1 Register/Capture1 Data Register PWM1 Duty Register						
D5H	PWM1HR	PWM1 Hig	h Register(B	sit[3:0])					
D6H	-	Reserved							
D7H	-	Reserved							
D8H	-	Reserved							
D9H	-	Reserved	Reserved						
DAH	-	Reserved							
DBH	-	Reserved							
DCH	SIO3M	POL3	-	-	SM30	SCK31	SCK30	SIO3ST	SIO3SF
DDH	SIO3R	SPI3 DATA	REGISTER	R					
DEH	BUR	BUCK1	BUCK0	BUR5	BUR4	BUR3	BUR2	BUR1	BUR0
DFH	RPR	RAM Page	Selection R	egister					
E0H	SIO1M	POL1	IOSW	SM11	SM10	SCK11	SCK10	SIO1ST	SIO1SF

Table 8-2 Control Registers of HMS81C2248

These registers of shaded area can not be access by bit manipulation instruction as " SET1, CLR1 ", but should be access by register operation instruction as " LDM dp,#imm ".



Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
E1H	SIO1R	SPI1 DATA	REGISTER	2					
E2H	IENH	INT0E	INT1E	RTRE	RTFE	RTOE	KSCE	SIO1E	SIO3E
E3H	IENL	T0E	T1E	-	-	ADCE	WDTE	BITE	-
E4H	IRQH	INT0IF	INT1IF	RTRIF	RTFIF	RTOIF	KSCIF	SIO1IF	SIO3IF
E5H	IRQL	T0IF	T1IF	-	-	ADCIF	WDTIF	BITIF	-
E6H	IEDS					IED1H	IED1L	IED0H	IED0L
E7H	RTCR	-	-	RCAP	RTCK2	RTCK1	RTCK0	RTCN	RTST
E8H	RT/RTDR RTCP0	Remote Tir Remote Tir	ner Register ner Capture	/Remote Tin Register0	ner Data Reg	gister			
E9H	RTCP0	Remote Tir	ner Capture	Register1					
EAH	ADCM	-	ADEN	ADS3	ADS2	ADS1	ADS0	ADST	ADSF
EBH	ADCR	ADC Resul	t Data Regis	ter					
ECH	BITR ¹	Basic Interval Timer Data Register							
ECH	CKCTLR ¹	-	-	RCWDT	WDTON	BTCL	BTS2	BTS1	BTS0
EDH	WDTR	WDTCL 7-bit Watchdog Counter Register							
EEH	-	Reserved							
EFH	PFDR ²	-	-	-	-	-	PFDIS	PFDM	PFDS
F0H	DSPM0	DSPEN	-	FOUT5	FOUT4	FOUT3	FOUT2	FOUT1	FOUT0
F1H	DSPM1	FBLK2	FBLK1	FBLK0	FPAT4	FPAT3	FPAT2	FPAT1	FPAT0
F2H	DSPM2	KSF	KSM	-	-	-	-	FCYC1	FCYC0
F3H	-	Reserved			-			-	
F4H	PSR	-	-	-	PWM1	EC0	BUZO	INTP1	INTP0
F5H	-	Reserved							
F6H	-	Reserved							
F7H	-	Reserved							
F8H	-	Reserved							
F9H	-	Reserved							
FAH	SCMR	-	-	-	-	-	-	CS1	CS0
FBH	SMR	-	-	-	-	-	-	-	SLP
FCH	PU0	PU07	PU06	PU05	PU04	PU03	PU02	PU01	PU00
FDH	-	Reserved			1			1	
FEH	PU2	PU27	PU26	PU25	PU24	PU23	PU22	PU21	PU20
FFH	STPC	Stop Control Register							

Table 8-2 Control Registers of HMS81C2248

These registers of shaded area can not be access by bit manipulation instruction as " SET1, CLR1 ", but should be access by register operation instruction as " LDM dp,#imm ".

1. The register BITR and CKCTLR are located at same address. Address ECH is read as BITR, written to CKCTLR.

2. The register PFDR only be implemented on devices, not on In-circuit Emulator.



8.5 Addressing Mode

The GMS800 series MCU uses six addressing modes;

- Register addressing
- Immediate addressing
- Direct page addressing
- Absolute addressing
- Indexed addressing
- Register-indirect addressing

(1) Register Addressing

Register addressing accesses the A, X, Y, C and PSW.

(2) Immediate Addressing \rightarrow #imm

In this mode, second byte (operand) is accessed as a data immediately.

Example:

0435 ADC #35H



When G-flag is 1, then RAM address is defined by 16-bit address which is composed of 8-bit RAM paging register (RPR) and 8-bit immediate data.

Example: G=1





(3) Direct Page Addressing \rightarrow dp

In this mode, a address is specified within direct page.

Example; G=0

C535 LDA 35H



(4) Absolute Addressing \rightarrow !abs

Absolute addressing sets corresponding memory data to Data, i.e. second byte (Operand I) of command becomes lower level address and third byte (Operand II) becomes upper level address.

With 3 bytes command, it is possible to access to whole memory area.

ADC, AND, CMP, CMPX, CMPY, EOR, LDA, LDX, LDY, OR, SBC, STA, STX, STY

Example;

;A ←ROM[0F035H]

;A ← RAM[35H]



!OF035H

The operation within data memory (RAM) ASL, BIT, DEC, INC, LSR, ROL, ROR

Example; Addressing accesses the address $0135_{\rm H}$ regardless of G-flag.



983501 INC !0135H ;A ←ROM[135H]





X indexed direct page (no offset) \rightarrow {X}

In this mode, a address is specified by the X register. ADC, AND, CMP, EOR, LDA, OR, SBC, STA, XMA Example; $X=15_H$, G=1 D4 LDA {X} ; ACC \leftarrow RAM[X].



X indexed direct page, auto increment \rightarrow {X}+

In this mode, a address is specified within direct page by the X register and the content of X is increased by 1.

LDA, STA

Example; G=0, X=35_H

DB LDA {X}+



X indexed direct page (8 bit offset) \rightarrow dp+X

This address value is the second byte (Operand) of command plus the data of X-register. And it assigns the memory in Direct page.

ADC, AND, CMP, EOR, LDA, LDY, OR, SBC, STA STY, XMA, ASL, DEC, INC, LSR, ROL, ROR

Example; G=0, X=0F5_H

C645 LDA 45H+X



Y indexed direct page (8 bit offset) \rightarrow dp+Y

This address value is the second byte (Operand) of command plus the data of Y-register, which assigns Memory in Direct page.

This is same with above (2). Use Y register instead of X.

Y indexed absolute \rightarrow !abs+Y

Sets the value of 16-bit absolute address plus Y-register data as Memory. This addressing mode can specify memory in whole area.

Example; $Y=55_H$



D500FA LDA !OFA00H+Y



(6) Indirect Addressing

Direct page indirect \rightarrow [dp]

Assigns data address to use for accomplishing command which sets memory data (or pair memory) by Operand. Also index can be used with Index register X,Y.

JMP, CALL

Example; G=0



X indexed indirect \rightarrow [dp+X]

Processes memory data as Data, assigned by 16-bit pair memory which is determined by pair data [dp+X+1][dp+X] Operand plus X-register data in Direct page.

ADC, AND, CMP, EOR, LDA, OR, SBC, STA

Example; G=0, X=10_H



Y indexed indirect \rightarrow [dp]+Y

Processes memory data as Data, assigned by the data [dp+1][dp] of 16-bit pair memory paired by Operand in Direct page plus Y-register data.

ADC, AND, CMP, EOR, LDA, OR, SBC, STA

Example; G=0, Y=10_H

1725 ADC [25H]+Y



Absolute indirect \rightarrow [!abs]

The program jumps to address specified by 16-bit absolute address.

JMP

Example; G=0



1F25E0 JMP [!0C025H]





9. I/O PORTS

The HMS81C2232/48 and MC80C2232/48/60 have six ports (P0, P2, P3, P4, P5, and P6). These ports pins may be multiplexed with an alternate function for the peripheral features on the device.

The HMS81C2232/48 and MC80C2232/48/60 incorporate 16 output ports and 29 input/output ports. Every port is capable of 1-bit and 8-bit manipulations and can carry out considerably varied control operations. Besides port functions, the ports can also serve as built-in hardware input/output pins.

P0 and P2 port have data direction registers which can define these ports as output or input. A "1" in the port direction register configure the corresponding port pin as output. Conversely, write "0" to the corresponding bit to specify it as input pin. For example, to use the even numbered bit of P0 as output ports and the odd numbered bits as input ports, write " $55_{\rm H}$ " to address $0C1_{\rm H}$ (P0 port direction register) during initial setting as shown in Figure 9-1.

All the port direction registers in the HMS81C2232/48 and MC80C2232/48/60 have 0 written to them by reset function. On the other hand, its initial status is input.



Figure 9-1 Example of Port I/O Assignment

9.1 P0 and P0IO register:

P0 is an 8-bit CMOS bidirectional I/O port (address $0C0_H$). Each port can be set individually as input and output through the P0IO register (address $0C1_H$).

Port pin	Alternate function
P00	INTP0 (External interrupt 0)
P01	INTP1 (External interrupt 1)
P02	TI (Timer input of remote control timer)
P03	AN0 (Analog Input 0)
P04	AN1 (Analog Input 1)
P05	AN2 (Analog Input 2)
P06	AN3 (Analog Input 3)
P07	BUZO (Buzzer driver output)

When P00 through P07 pins are used as input ports, an on-chip pull-up resistor can be connected to them in 1-bit units with a pull-up resistor option register 0 (PU0). Alternate functions include external interrupt request input and timer input. RESET input sets port 0 to input mode.



P00~P01 ports are multiplexed with External Interrupt Input Port(INTP1, INTP0), and P02 port is multiplexed with Timer input port of remote control timer(TI). P03~P06 ports are multiplexed with Analog Input Port and P07 port is multiplexed with Buzzer Output Port(BUZO).

The control register PSR (address $F4_H$) controls to select alternate function. After reset, this value is "0", port may be used as general I/O ports. To select alternate function such as Buzzer Output and External Interrupt Input, write "1" to the corresponding bit of PSR. Regardless of the direction register P0IO, PSR is selected to use as alternate functions, port pin can be used as a corresponding alternate features (BUZO, INTP1, INTP0)



9.2 P2 and P2IO register:

P2 is an 8-bit CMOS bidirectional I/O port (address $0C4_{H}$). Each port can be set individually as input and output through the P2IO register (address $0C5_{H}$).

Port pin	Alternate function
P20	SCK3 (Serial3 clock input/output)
P21	SO3 (Serial3 data output)
P22	AN4 (Analog Input 4)
P23	AN5 (Analog Input 5)
P24	AN6 (Analog Input 6)
P25	SO1 (Serial1 data output)
P26	SI1 (Serial1data input)
P27	SCK1 (Serial1 clock input/output)

When P20 to P27 pins are used as input ports, an on-chip pull-up resistor can be connected to them in 1-bit units with a pull-up resistor option register 2 (PU2). Alternate functions include serial interface data input/output and clock input/output. RESET input sets port 2 to input mode.



P20 to P21 port is multiplexed with serial interface3 data input/ output(SO3), clock input/output(SCK3). P22~P24 ports are multiplexed with Analog Input Port (ANI4~ANI6). P25~P27 port is multiplexed with serial interface1 data input(SI1)/output(SO1), clock input/output(SCK1).

9.3 P3 register:

P3 is an 8-bit high-voltage output only port(address $0C6_{\rm H}$). Onchip pull-down resistors can be connected in 1-bit units with the mask option in case of mask ROM model. The HMS87C2232/48 has no pull-down resistor. In addition, FIP controller/driver segment/digit output is provided as an alternate function.

Port pin	Alternate function
P30~P37	FIP24-FIP31

On-chip pull-down resistors can be connected in 1-bit units with the mask option. Pull-down resistor to V_{DISP} can be selected in 1-bit units. The HMS87C2232/48 has no pull-down resistor. In addition, FIP controller/driver output is provided as an alternate function.



9.4 P4 register:

P4 is an 8-bit high-voltage output only port(address $0C8_{\rm H}$). Onchip pull-down resistors can be connected in 1-bit units with the mask option in case of mask ROM model. The HMS87C2232/48 has no pull-down resistor. In addition, FIP controller/driver segment/digit output is provided as an alternate function.

Port Pin	Alternate Function
P40~P47	FIP32-FIP39

On-chip pull-down resistors can be connected in 1-bit units with the mask option. Pull-down resistor to V_{DISP} can be selected in 1-bit units. The HMS87C2232/48 has no pull-down resistor. In addition, FIP controller/driver output is provided as an alternate function.

P4 Da	ta Register	ADDRESS: 0C8 _H RESET VALUE: 00 _H
P4	P47 P46 P45 P	44 P43 P42 P41 P40
		Output data

9.5 P5 register:

P5 is an 8-bit high-voltage bidirectional I/O port (address $0CA_H$). Port 5 is an 8-bit input/output port with output latch. When using this port as an output port, the value assigned to the output latch (P50 through P57) is output. When it is used as an input port, set the output latch (P50 through P57) to "0", and read the port level



read (P50 through P57).

Port Pin	Alternate Function
P50~P57	FIP40-FIP47

On-chip pull-down resistors can be connected in 1-bit units with the mask option. Pull-down resistor to V_{DISP} or V_{SS0} can be selected in 1-bit units. The HMS87C2232/48 has no pull-down resistor. In addition, FIP controller/driver output is provided as an alternate function. RESET input sets port 5 to input mode.



9.6 P6 register:

P6 is an 5-bit high-voltage bidirectional I/O port (address $0CC_H$). Port 64 is multiplexed with Pulse Width Modulator (PWM). Port63 is multiplexed with Event Counter Input Port(EC0).

Port 6 is an 5-bit input/output port with output latch. When using this port as an output port, the value assigned to the output latch (P60 through P64) is output. When it is used as an input port, set the output latch (P60 through P64) to "0", and read the port level read (P60 through P64).

Port pin	Alternate function
P60	FIP48
P61	FIP49
P62	FIP50
P63	FIP51 / EC0
P64	FIP52 / PWM1O

On-chip pull-down resistors can be connected in 1-bit units with the mask option. Pull-down resistor to V_{DISP} or V_{SS0} can be selected in 1-bit units. The HMS87C2232/48 has no pull-down resistor. In addition, FIP controller/driver output is provided as an alternate function. RESET input sets port 6 to input mode.



The control register PSR (address $F4_H$) controls to select alternate function. After reset, this value is "0", port may be used as general I/O ports. To select alternate function such as PWM(Pulse Width Modulation) output and External Event Counter Input, write "1" to the corresponding bit of PSR. PSR is selected to use as alternate functions, port pin can be used as a corresponding alternate features (PWM1, EC0). When P63 pin are used as EC0 input, FIP51 output must be set "0", not used to FIP controller/driver output.



10. CLOCK GENERATOR

As shown in Figure 10-1, the clock generator produces the basic clock pulses which provide the system clock to be supplied to the CPU and the peripheral hardware. The clock generator produces the basic clock pulses which provide the system clock to be supplied to the CPU and peripheral hardware. The main system clock oscillator oscillates with a crystal resona-

tor or a ceramic resonator connected to the Xin and Xout pins. External clocks can be input to the main system clock oscillator. In this case, input a clock signal to the Xin pin and open the Xout pin The system clock can also be obtained from the external oscillator.



Figure 10-1 Block Diagram of Clock Generator

The clock generator produces the system clocks forming clock pulse, which are supplied to the CPU and the peripheral hardware. The internal system clock can be selected by bit1, and bit0 of the System Clock Mode Register(SCMR). The register is shown in Figure 10-2.

To the peripheral block, the clock among the not-divided original clocks, divided by 2, 4,..., up to 4096 can be provided. Peripheral clock is enabled or disabled by STOP instruction. On the initial reset, internal system clock is PS1 which is the fastest and other clock can be provided by bit1 and bit0 of SCMR.



Figure 10-2 System Clock Control Register



11. BASIC INTERVAL TIMER

The HMS81C2232/48 and MC80C2232/48/60 have one 8-bit Basic Interval Timer that is free-run, can not stop. Block diagram is shown in Figure 11-1. In addition, the Basic Interval Timer generates the time base for watchdog timer counting. It also provides a Basic interval timer interrupt (BITIF).

The 8-bit Basic interval timer register (BITR) is increased every internal count pulse which is divided by prescaler. Since prescaler has divided ratio by 8 to 1024, the count rate is 1/8 to 1/1024 of the oscillator frequency. As the count overflows from FF_H to $00_{\rm H}$, this overflow causes to generate the Basic interval timer interrupt. The BITIF is interrupt request flag of Basic interval timer. The Basic Interval Timer is controlled by the clock control register (CKCTLR) shown in Figure 11-2.

When write "1" to bit BTCL of CKCTLR, BITR register is cleared to "0" and restart to count-up. The bit BTCL becomes "0" after one machine cycle by hardware.

If the STOP instruction executed after writing "1" to bit RCWDT of CKCTLR, it goes into the internal RC oscillated watchdog timer mode. In this mode, all of the block is halted except the internal RC oscillator, Basic Interval Timer and Watchdog Timer. More detail informations are explained in Power Saving Function. The bit WDTON decides Watchdog Timer or the normal 7-bit timer.

Source clock can be selected by lower 3 bits of CKCTLR. BITR and CKCTLR are located at same address, and address $0EC_H$ is read as a BITR, and written to CKCTLR.



Figure 11-1 Block Diagram of Basic Interval Timer

CKCTLR [2:0]	Source clock	Interrupt (overflow) Period (ms) @ f _{XIN} = 4MHz
000 001 010 011 100 101 110 111	$\begin{array}{c} PS3(f_{XIN}\div8) \\ PS4(f_{XIN}\div16) \\ PS5(f_{XIN}\div32) \\ PS6(f_{XIN}\div64) \\ PS7(f_{XIN}\div128) \\ PS8(f_{XIN}\div256) \\ PS9(f_{XIN}\div512) \\ PS10(f_{XIN}\div1024) \end{array}$	0.512 1.024 2.048 4.096 8.192 16.384 32.768 65.536





Figure 11-2 BITR: Basic Interval Timer Mode Register

Example 1:

Example 2:

Basic Interval Timer Interrupt request flag is generated every 4.096ms at 4MHz.

```
LDM CKCTLR,#03H
SET1 BITE
EI
:
```

Basic Interval Timer Interrupt request flag is generated every 1.024ms at 4MHz.

: LDM CKCTLR,#01H SET1 BITE EI :



12. WATCHDOG TIMER

The watchdog timer rapidly detects the CPU malfunction such as endless looping caused by noise or the like, and resumes the CPU to the normal state.

The watchdog timer signal for detecting malfunction can be selected either a reset CPU or a interrupt request.

When the watchdog timer is not being used for malfunction detection, it can be used as a timer to generate an interrupt at fixed intervals. The purpose of the watchdog timer is to detect the malfunction (runaway) of program due to external noise or other causes and return the operation to the normal condition.

The watchdog timer has two types of clock source.

The first type is an on-chip RC oscillator which does not require any external components. This RC oscillator is separate from the external oscillator of the Xin pin. It means that the watchdog timer will run, even if the clock on the Xin pin of the device has been stopped, for example, by entering the STOP mode.

The other type is a prescaled system clock.

The watchdog timer consists of 7-bit binary counter and the watchdog timer data register. When the value of 7-bit binary counter is equal to the lower 7 bits of WDTR, the interrupt request flag is generated. This can be used as WDT interrupt or reset the CPU in accordance with the bit WDTON.



ter clearing Basic Interval Timer, after the bit WDTON set to "1", maximum error of timer is depend on prescaler ratio of Basic Interval Timer. The 7-bit binary counter is cleared by setting WDTCL(bit7 of WDTR) and the WDTCL is cleared automatically after 1 machine cycle.

The RC oscillated watchdog timer is activated by setting the bit RCWDT as shown below.

LDM CKCTLR,#3FH; enable the RC-osc WDT LDM WDTR,#0FFH; set the WDT period STOP ; enter the STOP mode NOP NOP ; RC-osc WDT running

The RCWDT oscillation period is vary with temperature, V_{DD} and process variations from part to part (approximately, 33~100uS). The following equation shows the RCWDT oscillated watchdog timer time-out.

```
T_{RCWDT} = CLK_{RCWDT} \times 2^8 \times [WDTR.6 - 0] + (CLK_{RCWDT} \times 2^8)/2
where, CLK_{RCWDT} = 33 - 100 uS
```

In addition, this watchdog timer can be used as a simple 7-bit timer by interrupt WDTIF. The interval of watchdog timer interrupt is decided by Basic Interval Timer. Interval equation is as below.

 $T_{WDT} = [WDTR.6 \sim 0] \times Interval of BIT$



Figure 12-1 Block Diagram of Watchdog Timer



Watchdog Timer Control

Figure 12-2 shows the watchdog timer data register. The watchdog timer is automatically enabled after reset.

The CPU malfunction is detected during setting of the detection time, selecting of output, and clearing of the binary counter. Clearing the binary counter is repeated within the detection time.

If the malfunction occurs for any cause, the watchdog timer output will become active at the rising overflow from the binary counters unless the binary counter is cleared. At this time, when WDTON=1, a reset is generated, which reset the internal hard-ware. When WDTON=0, a watchdog timer interrupt (WDTIF) is generated.

The watchdog timer temporarily stops counting in the STOP mode, and when the STOP mode is released, it automatically restarts (continues counting).



Figure 12-2 WDTR: Watchdog Timer Data Register

Example: Sets the watchdog timer detection time to 1.0 sec at 4.19MHz

	LDM LDM	CKCTLR,#1FH WDTR,#08FH	; Select 1/1024 clock source,	WDTON \leftarrow 1, Clear Counter
Within WDT detection time	LDM : :	WDTR,#08FH	; Clear counter	
Within WDT		WDTR,#08FH	; Clear counter	
detection time	LDM	WDTR,#08FH	; Clear counter	



Enable and Disable Watchdog

Watchdog timer is enabled by setting WDTON (bit 4 in CKCTLR) to "1". WDTON is initialized to "1".

Example: Enables watchdog timer for Reset

```
LDM CKCTLR, #xxx1_xxxxB; WDTON ← 1
:
```

The watchdog timer is disabled by clearing bit 4 (WDTON) of CKCTLR. The watchdog timer is halted in STOP mode and restarts automatically after STOP mode is released.

Watchdog Timer Interrupt

The watchdog timer can be also used as a simple 7-bit timer by clearing bit4 of CKCTLR to "0". The interval of watchdog timer interrupt is decided by Basic Interval Timer. Interval equation is shown as below.

 $T = WDTR \times Interval of BIT$

The stack pointer (SP) should be initialized before using the watchdog timer output as an interrupt source.

Example: 7-bit timer interrupt set up.

```
LDM CKCTLR, #xxx0_xxxxB; WDTON ←0
LDM WDTR, #8FH ; WDTCL ←1
```



Figure 12-3 Watchdog timer Timing

If the watchdog timer output becomes active, a reset is generated,

which reset the internal hardware.



13. TIMER/EVENT COUNTER

The HMS81C2232/48 and MC80C2232/48/60 have two Timer/ Counter registers. Each module can generate an interrupt to indicate that an event has occurred (i.e. timer match).

Timer 0 and Timer 1 can be used either two 8-bit Timer/Counter or one 16-bit Timer/Counter with combine them.

In the "timer" function, the register is increased every internal clock input. Thus, you can think of it as counting internal clock input. Since a least clock consists of 2 and most clock consists of 2048 oscillator periods, the count rate is 1/2 to 1/2048 of the oscillator frequency in Timer0. And Timer1 can use the same clock source too. In addition, Timer1 has more fast clock source (1/1 to 1/8).

In the "counter" function, the register is increased in response to

a 0-to-1(rising edge) transition at its corresponding external input pin, EC0.

In addition the "capture" function, the register is increased in response external or internal clock sources same with timer or counter function. When external clock edge input, the count register is captured into capture data register CDRx.

Timer1 is shared with "PWM" function and "Compare output" function

It has seven operating modes: "8-bit timer/counter", "16-bit timer/counter", "8-bit capture", "16-bit capture", "8-bit compare output", "16-bit compare output" and "10-bit PWM" which are selected by bit in Timer mode register TM0 and TM1 as shown in Figure 13-1 and Table 13-1.

16BIT	CAP0	CAP1	PWM1E	Т0СК [2:0]	T1CK [1:0]	PWM1O	TIMER 0	TIMER 1
0	0	0	0	XXX	XX	0	8-bit Timer	8-bit Timer
0	0	1	0	111	XX	0	8-bit Event counter	8-bit Capture
0	1	0	0	XXX	XX	1	8-bit Capture (internal clock)	8-bit Compare Output
0	Х	0	1	XXX	XX	1	8-bit Timer/Counter	10-bit PWM
1	0	0	0	XXX	11	0	16-bit Timer	
1	0	0	0	111	11	0	16-bit Event counter	
1	1	Х	0	XXX	11	0	16-bit Capture (internal clock)	
1	0	0	0	XXX	11	1	16-bit Compare Output	

Table 13-1 Operating Modes of Timer0 and Timer1



HMS81C2232/2248/2332/2348 MC80C2232/2248/2260/2332/2348



Figure 13-1 TM0, TM1 Registers



13.1 8-bit Timer / Counter Mode

The HMS81C2232/48 and MC80C2232/48/60 have two 8-bit Timer/Counters, Timer 0, Timer 1 as shown in Figure 13-2.

TMx as shown in Figure 13-1 and Table 13-1. To use as an 8-bit timer/counter mode, bit CAP0 of T0CR is cleared to "0" and bits 16BIT of T1CR should be cleared to "0"(Table 13-1).

The "timer" or "counter" function is selected by mode registers



Figure 13-2 8-bit Timer/Counter 0, 1



Example 1:

Timer0 = 2ms 8-bit timer mode at 4MHz Timer1 = 0.5ms 8-bit timer mode at 4MHz

LDM	TDR0,#249
LDM	TDR1,#249
LDM	TOCR,#0000 1111B
LDM	T1CR,#0000 ^{-1011B}
SET1	TOE —
SET1	T1E
ΕI	

Example 2:

Timer0 = 8-bit event counter mode Timer1 = 0.5ms 8-bit timer mode at 4MHz

LDM	TDR0,#249
LDM	TDR1 , #249
LDM	TOCR, #0001 1111E
LDM	T1CR,#0000_1011B
SET1	тое —
SET1	T1E
ΕI	

Note: The contents of Timer data register TDRx should be initialized $1_H \sim FF_H$, not 0_H , because it is undefined after reset.

These timers have each 8-bit count register and data register. The count register is increased by every internal or external clock input. The internal clock has a prescaler divide ratio option of 2, 4, 8, 32,128, 512, 2048 selected by control bits T0CK[2:0] of register (T0CR) and 1, 2, 8 selected by control bits T1CK[1:0] of register (T1CR). In the Timer 0, timer register T0 increases from 00_H until it matches T0DR and then reset to 00_H . The match output of Timer 0 generates Timer 0 interrupt (latched in T0IF bit). As TDRx and Tx register are in same address, when reading it as a Tx, written to TDRx.

In counter function, the counter is increased every 0-to-1 (rising edge) transition of EC0 pin. In order to use counter function, the bit EC0 of the Port Selection Register(PSR.3) is set to "1". The Timer 0 can be used as a counter by pin EC0 input, but Timer 1 can not.



8-bit Timer Mode

In the timer mode, the internal clock is used for counting up. Thus, you can think of it as counting internal clock input. The contents of TDR*n* are compared with the contents of up-counter, Tn. If match is found, a timer 1 interrupt (T1IF) is generated and the up-counter is cleared to 0. Counting up is resumed after the

up-counter is cleared.

As the value of TDRn is changeable by software, time interval is set as you want



Figure 13-3 Timer Mode Timing Chart



Figure 13-4 Timer Count Example



1

8-bit Event Counter Mode

In this mode, counting up is started by an external trigger. This trigger means rising edge of the EC0 pin input. Source clock is used as an internal clock selected with timer mode register T0CR. The contents of timer data register T0DR is compared with the contents of the up-counter T0. If a match is found, an timer interrupt request flag T0IF is generated, and the counter is cleared to "0". The counter is restart and count up continuously by every rising edge of the EC0 pin input.

The maximum frequency applied to the EC0 pin is $f_{XIN}/2$ [Hz].

In order to use event counter function, the bit 3 of the Port Selection Register(PSR.3) is required to be set to "1".

After reset, the value of timer data register T0DR is undefined, it should be initialized to between $0_H \sim FE_H$, not to "0". The interval period of Timer is calculated as below equation.

Period (sec) =
$$\frac{1}{f_{XIN}} \times 2 \times Divide \times (TDR0+1)$$



Figure 13-5 Event Counter Mode Timing Chart



Figure 13-6 Count Operation of Timer / Event counter



13.2 16-bit Timer / Counter Mode

The Timer register is being run with 16 bits. A 16-bit timer/counter register T0, T1 are increased from $0000_{\rm H}$ until it matches T0DR, T1DR and then resets to $0000_{\rm H}$. The match output generates Timer 0 interrupt not Timer 1 interrupt.

The clock source of the Timer 0 is selected either internal or external clock by bit T0CK[2:0].

In 16-bit mode, the bits T1CK[1:0] and 16BIT of T1CR should be set to "1" respectively.



Figure 13-7 16-bit Timer/Counter

13.3 8-bit Capture Mode

The Timer 0 capture mode is set by bit CAP0 of timer mode register T0CR (bit CAP1 of timer mode register T1CR for Timer 1) as shown in Figure 13-8.

As mentioned above, not only Timer 0 but Timer 1 can also be used as a capture mode.

The Timer/Counter register is increased in response internal or external input. This counting function is same with normal timer mode, and Timer interrupt is generated when timer register T0 (T1) increases and matches T0DR (T1DR).

This timer interrupt in capture mode is very useful when the pulse width of captured signal is more wider than the maximum period of Timer.

For example, in Figure 13-10, the pulse width of captured signal is wider than the timer data value (FF_H) over 2 times. When external interrupt is occurred, the captured value (13_H) is more little than wanted value. It can be obtained correct value by counting



the number of timer overflow occurrence.

Timer/Counter still does the above, but with the added feature that a edge transition at external input INTx pin causes the current value in the Timer x register (T0,T1), to be captured into registers CDRx (CDR0, CDR1), respectively. After captured, Timer x register is cleared and restarts by hardware.

It has three transition modes: "falling edge", "rising edge", "both

edge" which are selected by interrupt edge selection register IEDS (Refer to External interrupt section). In addition, the transition at INTx pin generate an interrupt.

Note: The CDRx, TDRx and Tx are in same address. In the capture mode, reading operation is read the CDRx, not Tx because path is opened to the CDRx, and TDRx is only for writing operation.





Figure 13-8 8-bit Capture Mode





Figure 13-9 Input Capture Operation



Figure 13-10 Excess Timer Overflow in Capture Mode



13.4 16-bit Capture Mode

16-bit capture mode is the same as 8-bit capture, except that the Timer register is being run will 16 bits.

The clock source of the Timer 0 is selected either internal or ex-

ternal clock by bit T0CK2, T0CK1 and T0CK0.

In 16-bit mode, the bits T1CK1,T1CK0 and 16BIT of T1CR should be set to "1" respectively.



Figure 13-11 16-bit Capture Mode



Example 1:

Timer0 = 16-bit timer mode, 0.5s at 4MHz

```
LDM TOCR,#0000_1111B;8uS
LDM T1CR,#0100_1100B;16bit Mode
LDM TDR0,#<62499 ;8uS X 62500
LDM TDR1,#>62499 ;=0.5s
SET1 TOE
EI
:
```

Example 2:

Timer0 = 16-bit event counter mode

```
LDM PSR,#0000 1000B;EC0 Set

LDM T0CR,#0001_1111B;CounterMode

LDM T1CR,#0100_1100B;16bit Mode

LDM TDR0,#<0FFH ;

LDM TDR1,#>0FFH ;

SET1 T0E

EI

:
```

13.5 PWM Mode

The HMS81C2232/48 and MC80C2232/48/60 have a high speed PWM (Pulse Width Modulation) functions which shared with Timer1.

In PWM mode, pin R64/PWM1O outputs up to a 10-bit resolution PWM output. This pin should be configured as a PWM output by setting "1" bit PWM1O in PSR.4 register.

The period of the PWM output is determined by the T1PPR (PWM1 Period Register) and PWM1HR[3:2] (bit3,2 of PWM1 High Register) and the duty of the PWM output is determined by the T1PDR (PWM1 Duty Register) and PWM1HR[1:0] (bit1,0 of PWM1 High Register).

The user writes the lower 8-bit period value to the T1PPR and the higher 2-bit period value to the PWM1HR[3:2]. And writes duty value to the T1PDR and the PWM1HR[1:0] same way.

The T1PDR is configured as a double buffering for glitchless PWM output. In Figure 13-12, the duty data is transferred from the master to the slave when the period data matched to the counted value. (i.e. at the beginning of next duty cycle)

PWM Period = [PWM1HR[3:2]T1PPR] X Source Clock

PWM Duty = [PWM1HR[1:0]T1PDR] X Source Clock

The relation of frequency and resolution is in inverse proportion. Table 13-2 shows the relation of PWM frequency vs. resolution.

If it needed more higher frequency of PWM, it should be reduced

Example 3:

Timer0 = 16-bit capture mode

:

```
PSR,#0000_0001B;INT0 set
TM0,#0010_1111B;Capture Mode
TM1,#0100_1100B;16bit Mode
LDM
LDM
LDM
          TDR0,#<0F\overline{F}H
LDM
                                    ;
LDM
          TDR1, #>0FFH
LDM
          IEDS, #01H; Falling Edge
SET1
          TOE
ΕI
:
:
```

resolution.

	Frequency				
Resolution	T1CK[1:0] = 00(250nS)	T1CK[1:0] = 01(500nS)	T1CK[1:0] = 10(2uS)		
10-bit	3.9KHz	1.95KHz	0.49KHz		
9-bit	7.8KHz	3.90KHz	0.97KHz		
8-bit	15.6KHz	7.81KHz	1.95KHz		
7-bit	31.2KHz	15.62KHz	3.90KHz		

Table 13-2 PWM Frequency vs. Resolution at 4MHz

The bit POL of T1CR decides the polarity of duty cycle.

If the duty value is set same to the period value, the PWM output is determined by the bit POL (1: High, 0: Low). And if the duty value is set to " 00_{H} ", the PWM output is determined by the bit POL (1: Low, 0: High).

It can be changed duty value when the PWM output. However the changed duty value is output after the current period is over. And it can be maintained the duty value at present output when changed only period value shown as Figure 13-14. As it were, the absolute duty time is not changed in varying frequency. But the changed period value must greater than the duty value.

Note: If changing the Timer1 to PWM function, it should be stop the timer clock firstly, and then set period and duty register value. If user writes register values while timer is in operation, these register could be set with certain values.

Ex) Sample Program @4MHz 2uS



LDM	T1CR,#1010_	1010b; Set Clock & PWM1E
LDM	T1PPR,#199	;
LDM	T1PDR,#99	;

13.6 8-bit Compare Output (16-bit)

The HMS81C2232/48 and MC80C2232/48/60 have a function of Timer Compare Output. To pulse out, the timer match can goes to port pin(P64/PWM1O) as shown in Figure 13-2 and Figure 13-7. Thus, pulse out is generated by the timer match. These operation is implemented to pin, P64/PWM1O.

In this mode, the bit P64/PWM1O of Port Selection register (PSR.4) should be set to "1", and the bit PWM1E of timer1 mode register (T1CR) should be set to "0". In addition, 16-bit Compare output mode is available, also.

LDM PWM1HR,00H LDM T1CR,#1010_1011b ; Start timer1

This pin output the signal having a 50 : 50 duty square wave, and output frequency is same as below equation.

 $f_{COMP} = \frac{Oscillation \ Frequency}{2 \times Prescaler \ Value \times (TDR + 1)}$



Figure 13-12 PWM Mode




Figure 13-13 Example of PWM at 4MHz



Figure 13-14 Example of Changing the Period in Absolute Duty Cycle (@4MHz)

14. REMOTE CONTROL TIMER

The 8-bit remote control timer has a pulse width measurement function with a resolution of 8 bits. Pulse width is measured from

14.1 Registers Controlling 8-Bit Remote Control Timer

The following three types of registers control the 8-bit remote control timer.

• Remote control timer control register (RTCR)

• Remote control timer capture registers (RTCP0 and RTCP1)

• 8-bit timer register (RT)

(1) Remote control timer control register(RTCR)

This register enables or disables the operation of the 8-bit timer (RT), and sets the count clock. RTCR is set by using a 1-bit or 8-bit memory manipulation instruction. This register is initialized to $00_{\rm H}$ by RESET input.

(2) Remote control timer capture registers (RTCP0 and RTCP1)

14.2 Operation of 8-Bit Remote Control Timer

The 8-bit remote control timer operates as a pulse width measuring circuit. The width of a high-level or low-level external pulse input to the TI pin is measured by operating the 8-bit timer (RT) in the free-running mode. Detection of the valid edge is sampled every 2 cycles of the count clock selected by RTCK0~ RTCK2, and the capture operation is not performed until the valid level has been detected two times. Therefore, the pulse width input to the TI pin must be 5 or more of the count clock set by RTCK0~ RTCK2, regardless of whether the level is high or low. If the a difference in count value when the valid edge has been detected while the timer operates in the free-running mode.

These 8-bit registers capture the contents of the 8-bit timer (RT). The capture operation is performed in synchronization with the valid edge input to the TI pin (capture trigger). The contents of RTCP0 are retained until the next rising edge of the TI pin is detected. The contents of RTCP1 are retained until the next falling edge of the TI pin is detected.

RTCP0 and RTCP1 can be read by using an 8-bit memory manipulation instruction. The values of these registers are initialized to $00_{\rm H}$ by RESET input.

(3) 8-bit timer register (RT)

This 8-bit register counts the count pulse. It can be read by using an 8-bit memory manipulation instruction. The value of this register is initialized to 00H by RESET input or by clearing the RTST bit.

pulse width is less than 5 clocks, it cannot be detected, and the capture operation is not performed. The value of remote timer register(RT) is loaded to and retained in the capture registers (RTCP0 and RTCP1) in synchronization with the valid edge of the pulse input to the TI pin, as shown in Figure 14-2. Figure 14-3 and Figure 14-4 shows the timing of pulse width measurement.

The Remote Timer control register is shown in Figure 14-





1.

Bit Name	Bit Position	Description
RCAP	RTCR.5	0: Timer/Counter mode
PTCK2	PTCP /	1. Capture mode selection hay
RTCK1	RTCR.3	000. 5-bit Timer, Clock source is t _{XIN} + 5
RTCK0	RTCR.2	010: 8-bit Timer, Clock source is $f_{XIN} \neq 32$
		011: 8-bit Timer, Clock source is $f_{XIN} + 256$
		100: 8-bit Timer, Clock source is $f_{XIN} \div 512$
		101: 8-bit Timer, Clock source is f _{XIN} ÷ 1024
		110: 8-bit Timer, Clock source is f _{XIN} ÷ 2048
		111: 8-bit Timer, Clock source is f _{XIN} ÷ 4096
RTCN	RTCR.1	0: Timer count pause 1: Timer count start
RTST	RTCR.0	0: When cleared, stop the counting.
		1: When set, Timer 0 Count Register is cleared and start again.
R/V 7 RTDR	V R/W R/W R/ 6 5 4	W R/W R/W R/W 3 2 1 0 ADDRESS: 0E8 _H

Figure 14-1 RTCR Register

Note: The RTCP0, RTDR and RT are in same address. In the capture mode, reading operation is read the RTCP0, not RT because path is opened to the RTCP0, and RTDR is only for writing operation.





Figure 14-2 Block Diagram of Remote Control Timer





Figure 14-3 To measure pulse width in synchronization with rising edge





Figure 14-4 To measure pulse width in synchronization with both rising and falling edge



15. ANALOG DIGITAL CONVERTER

The analog-to-digital converter (A/D) allows conversion of an analog input signal to a corresponding 8-bit digital value. The A/D module has seven analog inputs, which are multiplexed into one sample and hold. The output of the sample and hold is the input into the converter, which generates the result via successive approximation. The analog supply voltage is connected to AV_{DD} of ladder resistance of A/D module.

The A/D module has two registers which are the control register ADCM and A/D result register ADCR. The register ADCM, shown in Figure 15-1, controls the operation of the A/D converter module. The port pins can be configured as analog inputs or digital I/O.

It is selected the corresponding channel to be converted by setting ADS[3:0].

How to Use A/D Converter

The processing of conversion is start when the start bit ADST is set to "1". After one cycle, it is cleared by hardware. The register ADCR contains the results of the A/D conversion. When the conversion is completed, the result is loaded into the ADCR, the A/D conversion status bit ADSF is set to "1", and the A/D interrupt flag ADIF is set. The block diagram of the A/D module is shown in Figure 15-2. The A/D status bit ADSF is set automatically when A/D conversion is completed, cleared when A/D conversion is in process. The conversion time takes maximum 30 uS (at fX_{IN}=4 MHz).



Figure 15-1 A/D Converter Control Register





Figure 15-2 A/D Block Diagram





Figure 15-3 A/D Converter Operation Flow

A/D Converter Cautions

(1) Input range of AN0 to AN6

The input voltage of AN6 to AN0 should be within the specification range. In particular, if a voltage above AV_{DD} or below AV_{SS} is input (even if within the absolute maximum rating range), the conversion value for that channel can not be indeterminate. The conversion values of the other channels may also be affected. (2) Noise countermeasures

In order to maintain 8-bit resolution, attention must be paid to noise on pins AV_{DD} and AN6 to AN0. Since the effect increases in proportion to the output impedance of the analog input source, it is recommended that a capacitor be connected externally as shown in Figure 15-4 in order to reduce noise.



Figure 15-4 Analog Input Pin Connecting Capacitor

(3) Pins AN0/P03 to AN3/P06 and AN4/P22 to AN6/P24

The analog input pins AN6 to AN0 also function as input/output port (PORT P0 and P2) pins. When A/D conversion is performed with any of pins AN6 to AN0 selected, be sure not to execute a PORT input instruction while conversion is in progress, as this may reduce the conversion resolution.

Also, if digital pulses are applied to a pin adjacent to the pin in the process of A/D conversion, the expected A/D conversion value may not be obtainable due to coupling noise. Therefore, avoid applying pulses to pins adjacent to the pin undergoing A/D conversion.

(4) AV_{DD} pin input impedance

A series resistor string of approximately 10K Ω is connected between the AV_{DD} pin and the AV_{SS} pin.

Therefore, if the output impedance of the reference voltage source is high, this will result in parallel connection to the series resistor string between the AV_{DD} pin and the AV_{SS} pin, and there will be a large reference voltage error.



16. SERIAL PERIPHERAL INTERFACE (SPI1)

The Serial Peripheral Interface (SPI) module is a serial interface useful for communicating with other peripheral of microcontroller devices. These peripheral devices may be serial EEPROMs, shift registers, display drivers, A/D converters, etc. The Serial Peripheral Interface(SPI) is 8-bit clock synchronous type and consists of serial I/O register, serial I/O mode register, clock selection circuit octal counter and control circuit. The SO1 pin is designed to input and output. So Serial Peripheral Interface(SPI) can be operated with minimum two pin



Figure 16-1 SPI1 Block Diagram



Serial I/O 1 Mode Register(SIO1M) controls serial I/O function. According to SCK1 and SCK0, the internal clock or external clock can be selected. The serial transmission operation mode is decided by setting the SM1 and SM0, and the polarity of transfer clock is selected by setting the POL.

Serial I/O Data Register(SIO1R) is a 8-bit shift register. First LSB is send or is received. When receiving mode, serial input pin is selected by IOSW. The SPI allows 8-bits of data to be synchronously transmitted and received.

To accomplish communication, typically three pins are used:

Serial Data In	P26/SI1
Serial Data Out	P25/SO1
Serial Clock	P27/SCK1



.

Figure 16-2 SPI1 Control Register



16.1 Transmission/Receiving Timing

The serial transmission is started by setting SIO1ST(bit1 of SIO1M) to "1". After one cycle of SCK1, SIO1ST is cleared automatically to "0". The serial output data from 8-bit shift register is output at falling edge of SCK1. And input data is latched at ris-

ing edge of SCK1 pin. When transmission clock is counted 8 times, serial I/O counter is cleared as '0". Transmission clock is halted in "H" state and serial I/O interrupt(IFSIO1) occurred.



Figure 16-3 SPI1 Timing Diagram at POL=0



Figure 16-4 SPI Timing Diagram at POL=1



16.2 The method of Serial I/O

1 Select transmission/receiving mode

Note: When external clock is used, the frequency should be less than 1MHz and recommended duty is 50%.

② In case of sending mode, write data to be send to SIO1R.

③ Set SIO1ST to "1" to start serial transmission.

Note: If both transmission mode is selected and transmission is performed simultaneously it would be made error.

④ The SIO1 interrupt is generated at the completion of SIO1 and SIO1SF is set to "1". In SIO1 interrupt service routine, correct transmission should be tested.

16.3 The Method to Test Correct Transmission

⑤ In case of receiving mode, the received data is acquired by reading the SIO1R.

LDM	SIO1R,#OAAh ;SIO1R Initia	1
	;Value	
LDM	SIO1M,#0011_1100b;SIO1M Select	
	NOP	
	NOP	
	SI01M,#0011 1110b;SI01 Start	



Figure 16-5 Serial1 Method to Test Transmission



17. SERIAL PERIPHERAL INTERFACE (SPI3)

The Serial Peripheral Interface (SPI3) module is a serial interface useful for communicating with other peripheral of microcontroller devices. These peripheral devices may be serial EEPROMs, shift registers, display drivers, A/D converters, etc. The Serial Peripheral Interface(SPI3) is 8-bit clock synchronous type and consists of serial I/O register, serial I/O mode register, clock selection circuit octal counter and control circuit.



Figure 17-1 SPI3 Block Diagram



Serial I/O Mode Register(SIO3M) controls serial I/O function. According to SCK31 and SCK30, the internal clock or external clock can be selected. The serial transmission operation mode is decided by setting the SM30, and the polarity of transfer clock is selected by setting the POL3.

Serial I/O Data Register(SIO3R) is a 8-bit shift register. First LSB is send or is received. The SPI3 allows 8-bits of data to be synchronously transmitted and received.

To accomplish communication, typically two pins are used:

- Serial3 Data Out	P21/SO3
- Serial3 Clock	P20/SCK3



Figure 17-2 SPI3 Control Register



17.1 Transmission Timing

The serial transmission is started by setting SIOST(bit1 of SIOM) to "1". After one cycle of SCK, SIOST is cleared automatically to "0". The serial output data from 8-bit shift register is output at falling edge of SCLK. And input data is latched at rising edge of

SCLK pin. When transmission clock is counted 8 times, serial I/ O counter is cleared as '0". Transmission clock is halted in "H" state and serial I/O interrupt(IFSIO) occurred.



Figure 17-3 SPI3 Timing Diagram at POL=0







17.2 The method of Serial I/O

1 Select transmission mode

Note: When external clock is used, the frequency should be less than 1MHz and recommended duty is 50%.

⁽²⁾ Write data to be send to SIO3R.

③ Set SIO3ST to "1" to start serial transmission.

Note: If both transmission mode is selected and transmission is performed simultaneously it would be made error.

The SIO3 interrupt is generated at the completion of SIO3 and SIO3SF is set to "1". In SIO3 interrupt service routine, correct transmission should be tested.

LDM	SIO3R,#0AAh	;SIO1R Initial
		;Value
LDM	SIO3M,#0001	1100b;SIO3M Select
	NOP	_
	NOP	
	SIO3M,#0001	1110b;SIO3 Start

17.3 The Method to Test Correct Transmission



Figure 17-5 Serial3 Method to Test Transmission



18. BUZZER FUNCTION

The buzzer driver block consists of 6-bit binary counter, buzzer register BUR, and clock source selector. It generates square-wave which has very wide range frequency (480Hz ~ 250kHz at f_{XIN} = 4MHz) by user software.

A 50% duty pulse can be output to P07/BUZO pin to use for piezo-electric buzzer drive. Pin P07 is assigned for output port of Buzzer driver by setting the bit 2 of PSR(address $0F4_{\rm H}$) to "1". At this time, the pin P07 must be defined as output mode (the bit 7 of P0IO=1).

Example: 5kHz output at 4MHz.

LDM	P0IO,#1XXX_XX1XXB
LDM	BUR,#0011_0010B
LDM	PSR,#XXXX_X1XXB

X means don't care

The bit 0 to 5 of BUR determines output frequency for buzzer driving.

Equation of frequency calculation is shown below.

$$f_{BUZ} = \frac{f_{XIN}}{2 \times DivideRatio \times (BUR + 1)}$$

 $\label{eq:started} \begin{array}{l} f_{BUZ} : \mbox{Buzzer frequency} \\ f_{XIN} : \mbox{Oscillator frequency} \\ \mbox{Divide Ratio: Prescaler divide ratio by BUCK[1:0]} \\ \mbox{BUR: Lower 6-bit value of BUR. Buzzer period value.} \end{array}$

The frequency of output signal is controlled by the buzzer control register BUR. The bit 0 to bit 5 of BUR determine output frequency for buzzer driving.



Figure 18-1 Block Diagram of Buzzer Driver



Figure 18-2 PSR and Buzzer Register



Note: BUR is a write-only register.

The 6-bit counter is cleared and starts the counting by writing signal at BUR register. It is incremental from 00_H until it matches 6-

BUR	BUR[7:6]				
[5:0]	00	01	10	11	
00	250.000	125.000	62.500	31.250	
01	125.000	62.500	31.250	15.625	
02	83.333	41.667	20.833	10.417	
03	62.500	31.250	15.625	7.813	
04	50.000	25.000	12.500	6.250	
05	41.667	20.833	10.417	5.208	
06	35.714	17.857	8.929	4.464	
07	31.250	15.625	7.813	3.906	
08	27.778	13.889	6.944	3.472	
09	25.000	12.500	6.250	3.125	
0A	22.727	11.364	5.682	2.841	
0B	20.833	10.417	5.208	2.604	
0C	19.231	9.615	4.808	2.404	
0D	17.857	8.929	4.464	2.232	
0E	16.667	8.333	4.167	2.083	
0F	15.625	7.813	3.906	1.953	
10	14.706	7.353	3.676	1.838	
11	13.889	6.944	3.472	1.736	
12	13.158	6.579	3.289	1.645	
13	12.500	6.250	3.125	1.563	
14	11.905	5.952	2.976	1.488	
15	11.364	5.682	2.841	1.420	
16	10.870	5.435	2.717	1.359	
17	10.417	5.208	2.604	1.302	
18	10.000	5.000	2.500	1.250	
19	9.615	4.808	2.404	1.202	
1A	9.259	4.630	2.315	1.157	
1B	8.929	4.464	2.232	1.116	
1C	8.621	4.310	2.155	1.078	
1D	8.333	4.167	2.083	1.042	
1E	8.065	4.032	2.016	1.008	
1F	7.813	3.906	1.953	0.977	

bit BUR value.

When main-frequency is 4MHz, buzzer frequency is shown as below table.

BUR	BUR[7:6]			
[5:0]	00	01	10	11
20	7.576	3.788	1.894	0.947
21	7.353	3.676	1.838	0.919
22	7.143	3.571	1.786	0.893
23	6.944	3.472	1.736	0.868
24	6.757	3.378	1.689	0.845
25	6.579	3.289	1.645	0.822
26	6.410	3.205	1.603	0.801
27	6.250	3.125	1.563	0.781
28	6.098	3.049	1.524	0.762
29	5.952	2.976	1.488	0.744
2A	5.814	2.907	1.453	0.727
2B	5.682	2.841	1.420	0.710
2C	5.556	2.778	1.389	0.694
2D	5.435	2.717	1.359	0.679
2E	5.319	2.660	1.330	0.665
2F	5.208	2.604	1.302	0.651
30	5.102	2.551	1.276	0.638
31	5.000	2.500	1.250	0.625
32	4.902	2.451	1.225	0.613
33	4.808	2.404	1.202	0.601
34	4.717	2.358	1.179	0.590
35	4.630	2.315	1.157	0.579
36	4.545	2.273	1.136	0.568
37	4.464	2.232	1.116	0.558
38	4.386	2.193	1.096	0.548
39	4.310	2.155	1.078	0.539
3A	4.237	2.119	1.059	0.530
3B	4.167	2.083	1.042	0.521
3C	4.098	2.049	1.025	0.512
3D	4.032	2.016	1.008	0.504
3E	3.968	1.984	0.992	0.496
3F	3.907	1.953	0.977	0.488



19. FIP CONTROLLER/DRIVER

19.1 Function of FIP Controller/Driver

The FIP controller/driver of the HMS81C2232/48 and MC80C2232/48/60 have the following functions.

(1) Can output display signals (DMA operation) by automatically reading display data.

(2) The pins not used for FIP display can be used as I/O port or output port pins (FIP24 through FIP52 pins only).

(3) Luminance can be adjusted in 8 steps by display mode register 1 (DSPM1).

(4) Hardware for key scan application

• Generates an interrupt signal (INTKS) indicating key scan timing

• Timing in which key scan data is output can be detected by key scan flag (KSF).

• Whether key scan timing is inserted or not can be selected.

(5) High-voltage output buffer that can directly drive FIP.

(6) FIP0 through FIP52 pins can be connected to pull-down resis-

tors by mask option (mask ROM model only). The HMS87C2232/48 does not have pull-down resistors)

Of the 53 FIP output pins of the HMS81C2232/48 and MC80C2232/48/60, FIP24 through FIP52 are multiplexed with port pins. FIP0 through FIP23 are dedicated output pins.

FIP24 through FIP52 can be used as port pins when FIP display is disabled by bit 7 (DSPEN) of the display mode

register 0 (DSPM0). Even when FIP display is enabled, the FIP output pins not used for display signal output can be used as port pins.

FIP Pin Name	Multiplexed Port Name	I/O	
FIP24-FIP31	P30-P37	Output only port	
FIP32-FIP39	P40-P47	Output only port	
FIP40-FIP47	P50-P57	I/O port	
FIP48-FIP52	P60-P64	I/O port	

Table 19-1 FIP Output Pins and Multiplexed Port Pins



19.2 Configuration of FIP Controller/Driver

The FIP controller/driver consists of the following hardware.



Figure 19-1 Block Diagram of FIP Controller/Driver

19.3 Registers Controlling FIP Controller/ Driver

The following three types of registers control the FIP controller/ driver.

- Display Mode Register 0 (DSPM0)
- Display Mode Register 1 (DSPM1)
- Display Mode Register 2 (DSPM2)

ltem	Configuration
Display	53
Control Register	Display mode register 0 (DSPM0) Display mode register 1 (DSPM1) Display mode register 2 (DSPM2)

Table 19-2 Configuration of FIP Controller/Driver

(1) Display mode register 0 (DSPM0)

DSPM0 performs the following setting.

- Enables or disables display
- Number of FIP output pins

DSPM0 is set by using a 1-bit or 8-bit memory manipulation instruction. The value of this register is set to $10_{\rm H}$ by RESET input.

(2) Display mode register 1 (DSPM1)

DSPM1 performs the following setting:

- Blanking width of FIP output signal
- Number of display patterns

DSPM1 is set by using a 1-bit or 8-bit memory manipulation instruction. The value of this register is set to 01_{H} by RESET input.

(3) Display mode register 2 (DSPM2)

DSPM2 performs the following setting. It also indicates the status of the display timing/key scan.

- · Insertion of key scan timing
- Display cycle (TDSP)



DSPM2 is set by using a 1-bit or 8-bit memory manipulation instruction. However, only bit 7 (KSF) can be read by a 1-bit memory manipulation instruction. The value of this register is initialized to 00H by RESET input.



Figure 19-2 Format of Display Mode Register 0



Figure 19-3 Format of Display Mode Register 1





Figure 19-4 Format of Display Mode Register 2

19.4 One display period and blanking width

The FIP output signals are blanked equally at the beginning and end of the display period by the blanking width set by bits 5 through 7 (FBLK0 through FBLK2) of the display mode register 1 (DSPM1).



Figure 19-5 Blanking Width of FIP Output Signal





Figure 19-6 VFD Output Operation Timings in Display Mode

19.5 Display Data Memory

The display data memory is a 112-byte RAM area that stores data to be displayed, and is mapped to addresses $0400_{\rm H}$ through $046F_{\rm H}$. The FIP controller reads the data stored in the display data memory independently of the CPU operation for FIP display (DMA operation).

The area of the display data memory not used for display can be

used as a normal RAM area.

At key scan timing (T_{KS}), all the FIP output pins are cleared to "0", and the data of the output latches of ports 3 through 6 are output to FIP24/P30 through FIP52/P64.

The address location of the display data memory is as follows:



• With 53 FIP output pins and 16 patterns

The addresses of the display data memory corresponding to the data output at each display timing (T0 through T15) are as shown in Figure 19-7 (for example, $T0 = 0400_{\text{H}}$ through 0406_{H} , and T1 = 0407_{H} through $040D_{\text{H}}$).

When 53 FIP output pins (FIP0 through FIP52) are used, one block of display data consists of 7 bytes. FIP output pins 0 (FIP0) through 52 (FIP52) correspond to one block of display data sequentially, starting from the least significant bit toward the most significant bit.



Figure 19-7 Relation between Address Location of Display Data Memory and FIP Output HMS81C2232/48 and MC80C2232/48/60 (with 53 FIP output pins and 16 patterns)

19.6 Key Scan Flag and Key Scan Data

Key scan flag

The key scan flag (KSF) is set to 1 during key scan timing, and is automatically reset to 0 at display timing.

KSF is mapped to bit 7 of the display mode register 2 (DSPM2) and can be tested in 1-bit units. It cannot be written, however. By testing KSF, it can be determined whether key scan timing is in progress, and whether key input data is correct can be checked.

Whether key scan timing is inserted or not can be selected by using the key scan timing insertion specification flag(KSM) (bit 6 of the display mode register 2 (DSPM2)).

Key scan data

Data stored to ports 3 through 6 are output from the FIP24 through FIP52 pins during key scan timing.

Note: If scanning is performed in such a manner that both a segment and a digit turn ON during key scan timing, the display may flicker.















Figure 19-10 Relationship between Display Data Memory and FIP Output with 13 Segments-11 Digits Displayed



20. INTERRUPTS

The HMS81C2232/48 and MC80C2232/48/60 interrupt circuits consist of Interrupt enable register (IENH, IENL), Interrupt request flags of IRQH, IRQL, Priority circuit, and Master enable flag ("I" flag of PSW). Thirteen interrupt sources are provided. The configuration of interrupt circuit is shown in Figure 20-2.

The External Interrupts INT0 and INT1 each can be transition-activated (1-to-0 or 0-to-1 transition) by selection IEDS.

The flags that actually generate these interrupts are bit INT0F and INT1F in register IRQH. When an external interrupt is generated, the flag that generated it is cleared by the hardware when the service routine is vectored to only if the interrupt was transition-activated.

The Timer $0 \sim \text{Timer 1}$ Interrupts are generated by TxIF which is set by a match in their respective timer/counter register. The Basic Interval Timer Interrupt is generated by BITIF which is set by an overflow in the timer register.

The AD converter Interrupt is generated by ADCIF which is set by finishing the analog to digital conversion. The Watchdog timer Interrupt is generated by WDTIF which set by a match in Watchdog timer register. The Basic Interval Timer Interrupt is generated by BITIF which are set by a overflow in the timer counter register. The interrupts are controlled by the interrupt master enable flag I-flag (bit 2 of PSW on page 30), the interrupt enable register (IENH, IENL), and the interrupt request flags (in IRQH and IRQL) except Power-on reset and software BRK interrupt. Below table shows the Interrupt priority.

Reset/Interrupt	Symbol	Priority
Hardware Reset	RESET	-
External Interrupt 0	INTP0	1
External Interrupt 1	INTP1	2
Remote Timer Rising	RTR	3
Remote Timer Falling	RTF	4
Remote Timer Overflow	RTO	5
Key Scan Interrupt	KS	6
SIO1 Interrupt	SIO1	7
SIO3 Interrupt	SIO3	8
Timer/Counter 0	TIMER0	9
Timer/Counter 1	TIMER1	10
ADC Interrupt	ADC	11
Watchdog Timer	WDT	12
Basic Interval Timer	BIT	13





Figure 20-1 Interrupt Request Flag

Vector addresses are shown in Figure 8-6 on page 32. Interrupt enable registers are shown in Figure 20-3. These registers are composed of interrupt enable flags of each interrupt source and these flags determines whether an interrupt will be accepted or not. When enable flag is "0", a corresponding interrupt source is prohibited. Note that PSW contains also a master enable bit, I-flag, which disables all interrupts at once.



Figure 20-2 Block Diagram of Interrupt



HMS81C2232/2248/2332/2348 MC80C2232/2248/2260/2332/2348



Figure 20-3 Interrupt Enable Flag



20.1 Interrupt Sequence

An interrupt request is held until the interrupt is accepted or the interrupt latch is cleared to "0" by a reset or an instruction. Interrupt acceptance sequence requires 8 f_{XIN} (2 µs at f_{MAIN} =4.19MHz) after the completion of the current instruction execution. The interrupt service task is terminated upon execution of an interrupt return instruction [RETI].

Interrupt acceptance

- 1. The interrupt master enable flag (I-flag) is cleared to "0" to temporarily disable the acceptance of any following maskable interrupts. When a non-maskable interrupt is accepted, the acceptance of any following interrupts is temporarily disabled.
- 2. Interrupt request flag for the interrupt source accepted is cleared to "0".
- 3. The contents of the program counter (return address) and the program status word are saved (pushed) onto the stack area. The stack pointer decreases 3 times.
- 4. The entry address of the interrupt service program is read from the vector table address and the entry address is loaded to the program counter.
- 5. The instruction stored at the entry address of the interrupt service program is executed.



Figure 20-4 Timing chart of Interrupt Acceptance and Interrupt Return Instruction



Correspondence between vector table address for BIT interrupt and the entry address of the interrupt service program.

A interrupt request is not accepted until the I-flag is set to "1" even if a requested interrupt has higher priority than that of the current interrupt being serviced.

When nested interrupt service is required, the I-flag should be set to "1" by "EI" instruction in the interrupt service program. In this case, acceptable interrupt sources are selectively enabled by the individual interrupt enable flags.

Saving/Restoring General-purpose Register

During interrupt acceptance processing, the program counter and the program status word are automatically saved on the stack, but accumulator and other registers are not saved itself. These registers are saved by the software if necessary. Also, when multiple interrupt services are nested, it is necessary to avoid using the same data memory area for saving registers.

The following method is used to save/restore the general-purpose



registers.

Example: Register save using push and pop instructions

INTxx:	PUSH PUSH PUSH	A X Y	;SAVE ACC. ;SAVE X REG. ;SAVE Y REG.
	interrupt proc	essing	
	POP POP POP RETI	Y X A	;RESTORE Y REG. ;RESTORE X REG. ;RESTORE ACC. ;RETURN

General-purpose register save/restore using push and pop instructions;



20.2 BRK Interrupt

Software interrupt can be invoked by BRK instruction, which has the lowest priority order.

Interrupt vector address of BRK is shared with the vector of TCALL 0 (Refer to Program Memory Section). When BRK interrupt is generated, B-flag of PSW is set to distinguish BRK from TCALL 0.

Each processing step is determined by B-flag as shown in Figure 20-5.



Figure 20-5 Execution of BRK/TCALL0



20.3 Multi Interrupt

If two requests of different priority levels are received simultaneously, the request of higher priority level is serviced. If requests of the interrupt are received at the same time simultaneously, an internal polling sequence determines by hardware which request is serviced.



Figure 20-6 Execution of Multi Interrupt

However, multiple processing through software for special features is possible. Generally when an interrupt is accepted, the Iflag is cleared to disable any further interrupt. But as user sets Iflag in interrupt routine, some further interrupt can be serviced even if certain interrupt is in progress.

Example: During Timer1 interrupt is in progress, INT0 interrupt serviced without any suspend.

TIMER1:	PUSH PUSH PUSH LDM EI : :	A X Y IENH,#80H IENL,#0	; Enable INT0 only ; Disable other ; Enable Interrupt
	: : LDM LDM POP POP POP RETI	ienh,#0f0h ienl,#0f0h Y X A	; Enable all interrupts



20.4 External Interrupt

The external interrupt on INT0 and INT1 pins are edge triggered depending on the edge selection register IEDS (address $0E6_H$) as shown in Figure 20-7.

The edge detection of external interrupt has three transition activated mode: rising edge, falling edge, and both edge.



Figure 20-7 External Interrupt Block Diagram

INT0 and INT1 are multiplexed with general I/O ports (P00 and P01). To use as an external interrupt pin, the bit of PSR should be set to "1" correspondingly.

Example: To use as an INT0 and INT1				
:				
:				
; **** Set port as an input port R00,R01				
LDM ROIO,#1111_1100B				
· · · · · · · · · · · · · · · · · · ·				
; **** Set port as an interrupt port				
LDM PSR,#0000_0011B				
;				
; **** Set Falling-edge Detection				
LDM IEDS,#0000_0101B				
:				
:				
:				

Response Time

The INT0 and INT1 edge are latched into INT0IF and INT1IF at every machine cycle. The values are not actually polled by the circuitry until the next machine cycle. If a request is active and conditions are right for it to be acknowledged, a hardware subroutine call to the requested service routine will be the next instruction to be executed. The DIV itself takes twelve cycles. Thus, a minimum of twelve complete machine cycles elapse between activation of an external interrupt request and the beginning of execution of the first instruction of the service routine.

Figure 20-8 shows interrupt response timings.



Figure 20-8 Interrupt Response Timing Diagram








21. POWER SAVING MODE

The GMS81C2232/48 has two power-down modes. In powerdown mode, power consumption is reduced considerably that in Battery operation Battery life can be extended a lot. For applications where power consumption is a critical factor, device provides two kinds of power saving functions, STOP mode and

21.1 Sleep Mode

In this mode, the internal oscillation circuits remain active. Oscillation continues and peripherals are operate normally but CPU stops. Movement of all peripherals is shown in Table 21-1. SLEEP mode is entered by setting the bit SLP of SMR to "1". (This register should be written by byte operation. If this register is set by bit manipulation instruction, for example "set1" or "clr1" instruction, it may be undesired operation). It is released by RE-SET or interrupt. To be release by interrupt, interrupt should be enabled before SLEEP mode. SLEEP mode. Table 21-1 shows the status of each Power Saving Mode. SLEEP mode is entered by setting bit 0 of SMR(Sleep Mode Register), and STOP mode is entered by STOP instruction.

Note: After SLEEP instruction, at least two or more NOP instruction should be written Ex) LDM SMR,#0000_0001B NOP NOP NOP





Release the SLEEP mode

The exit from SLEEP mode is hardware reset or all interrupts. Reset re-defines all the Control registers but does not change the on-chip RAM. Interrupts allow both on-chip RAM and Control registers to retain their values.

If I-flag = 1, the normal interrupt response takes place. If I-flag = 0, the chip will resume execution starting with the instruction following the SLEEP instruction. It will not vector to interrupt service routine. (refer to Figure 21-5)

When exit from SLEEP mode by reset, enough oscillation stabi-

lization time is required to normal operation. Figure 21-3 shows the timing diagram. When release the SLEEP mode, the Basic interval timer is activated on wake-up. It is increased from 00_H until FF_H. The count overflow is set to start normal operation. Therefore, before SLEEP instruction, user must be set its relevant prescaler divide ratio to have long enough time (more than 20msec). This guarantees that oscillator has started and stabilized. By interrupts, exit from SLEEP mode is shown in Figure 21-2. By reset, exit from SLEEP mode is shown in Figure 21-3.





Figure 21-2 SLEEP Mode Release Timing by External Interrupt



Figure 21-3 Timing of SLEEP Mode Release by RESET

21.2 Stop Mode

In the Stop mode, the on-chip oscillator is stopped. With the clock frozen, all functions are stopped, but the on-chip RAM and Control registers are held. The port pins out the values held by their respective port data register, port direction registers. Oscillator stops and the systems internal operations are all held up.

- The states of the RAM, registers, and latches valid immediately before the system is put in the STOP state are all held.
- The program counter stop the address of the instruction to be executed after the instruction "STOP" which starts the STOP operating mode.

Note: The Stop mode is activated by execution of STOP instruction after setting the STPC to " 0101_1010_B ". (This register should be written by byte operation. If this register is set by bit manipulation instruction, for example "set1" or "clr1" instruction, it may be undesired operation)

In the Stop mode of operation, V_{DD} can be reduced to minimize power consumption. Care must be taken, however, to ensure that V_{DD} is not reduced before the Stop mode is invoked, and that V_{DD} is restored to its normal operating level, before the Stop mode is terminated.

The reset should not be activated before $V_{\mbox{\scriptsize DD}}$ is restored to its



normal operating level, and must be held active long enough to allow the oscillator to restart and stabilize.

Note:	After STO	P instruction, at least two or more NOP in-									
	struction should be written										
	Ex) LDM CKCTLR,#0000_1110B										
		NOP									
		LDM STPC,#0101_1010B									
		NOP									
		STOP									
		NOP									
		NOP									

In the STOP operation, the dissipation of the power associated with the oscillator and the internal hardware is lowered; however, the power dissipation associated with the pin interface (depending on the external circuitry and program) is not directly determined by the hardware operation of the STOP feature. This point should be little current flows when the input level is stable at the power voltage level (V_{DD}/V_{SS}); however, when the input level gets higher than the power voltage level (by approximately 0.3 to 0.5V), a current begins to flow. Therefore, if cutting off the output transistor at an I/O port puts the pin signal into the high-impedance state, a current flow across the ports input transistor, requiring to fix the level by pull-up or other means.

	W 7	W 6	W 5	W 4	W 3	W 2	W 1	W 0	
STPC	0	1	0	1	1	0	1	0	INITIAL VALUE: 0000 0000 _B

Cautions: 1. To get into STOP mode, STOP Control Register must be enabled just before STOP instruction.
2. When STOP mode is released, STOP Control Register(STPC) value is cleared automatically.
3. It is prohibited to write another value into STPC.

Figure 21-4 STOP Control Register

Peripheral	STOP Mode	SLEEP Mode
CPU	Stop	Stop
RAM	Retain	Retain
Basic Interval Timer	Halted(Only operate in RCWDT mode)	Operates Continuously
Watchdog Timer	Stop(Only operate in RCWDT mode)	Stop
Timer/Event0,1	Halted(Only when the event counter mode is enable, timer operates normaly)	Operates Continuously
VFD Controller, ADC	Stop	Stop
BUZ, Remote Timer	Stop	Operates Continuously
SIO1,SIO3	Only operate with external clock	Operate
Oscillation	Stop(Xin=L, Xout=H)	Oscillation
I/O Ports	Retain	Retain
Control Registers	Retain	Retain
Internal Circuit	Stop mode	Sleep mode
Prescaler	Retain	Active
Address Data Bus	Retain	Retain
Release Source	Reset, Timer Interrupt(EC0), SIO1(External Clock), SIO3(External Clock), External Interrupt	Reset, All Interrupts

 Table 21-1Peripheral Operation During Power Saving Mode



Release the STOP mode

The exit from STOP mode is hardware reset or external interrupt. Reset re-defines all the Control registers but does not change the on-chip RAM. External interrupts allow both on-chip RAM and Control registers to retain their values.

If I-flag = 1, the normal interrupt response takes place. If I-flag = 0, the chip will resume execution starting with the instruction following the STOP instruction. It will not vector to interrupt service routine. (refer to Figure 21-5)

When exit from Stop mode by external interrupt, enough oscillation stabilization time is required to normal operation. Figure 21-6 shows the timing diagram. When release the Stop mode, the Basic interval timer is activated on wake-up. It is increased from $00_{\rm H}$ until FF_H. The count overflow is set to start normal operation. Therefore, before STOP instruction, user must be set its relevant prescaler divide ratio to have long enough time (more than 20msec). This guarantees that oscillator has started and stabilized.

By reset, exit from Stop mode is shown in Figure .



Figure 21-5 STOP Releasing Flow by Interrupts





Figure 21-6 STOP Mode Release Timing by External Interrupt



Figure 21-7 Timing of STOP Mode Release by RESET

21.3 Internal RC-Oscillated Watchdog Timer Mode

In the Internal RC-Oscillated Watchdog Timer mode, the on-chip oscillator is stopped. But internal RC oscillation circuit is oscillated in this mode. The on-chip RAM and Control registers are held. The port pins out the values held by their respective port data register, port direction registers.

The Internal RC-Oscillated Watchdog Timer mode is activated by execution of STOP instruction after setting the bit RCWDT of CKCTLR to "1". (This register should be written by byte operation. If this register is set by bit manipulation instruction, for example "set1" or "clr1" instruction, it may be undesired operation) **Note:** Caution: After STOP instruction, at least two or more NOP instruction should be written

Ex)	LDM WDTR,#1111_1111B
	LDM CKCTLR,#0010_1110B
	NOP
	LDM STPC,#0101_1010B
	NOP
	STOP
	NOP
	NOP

The exit from Internal RC-Oscillated Watchdog Timer mode is hardware reset or external interrupt. Reset re-defines all the Control registers but does not change the onchip RAM. External interrupts allow both on-chip RAM



and Control registers to retain their values.

If I-flag = 1, the normal interrupt response takes place. In this case, if the bit WDTON of CKCTLR is set to "0" and the bit WDTE of IENH is set to "1", the device will execute the watchdog timer interrupt service routine.(Figure 21-8) However, if the bit WDTON of CKCTLR is set to "1", the device will generate the internal RESET signal and execute the reset processing. (Figure 21-9) -flag = 0, the chip will resume execution starting with the instruction following the STOP instruction. It will not vector to interrupt service routine.(refer to Figure 21-5) When exit from Internal RC-Oscillated Watchdog Timer mode by external interrupt, the oscillation stabilization time is required to normal operation. Figure 21-8 shows the timing diagram. When release the Internal RC-Oscillated Watchdog Timer mode, the basic interval timer is activated on wake-up. It is increased from $00_{\rm H}$ until FF_H. The count overflow is set to start normal operation. Therefore, before STOP instruction, user must be set its relevant prescaler divide ratio to have long enough time (more than 20msec). This guarantees that oscillator has started and stabilized. By reset, exit from internal RC-Oscillated Watchdog Timer mode is shown in Figure 21-9.



Figure 21-8 Internal RCWDT Mode Releasing by External Interrupt or WDT Interrupt



Figure 21-9 Internal RCWDT Mode Releasing by RESET



21.4 Minimizing Current Consumption

The Stop mode is designed to reduce power consumption. To minimize current drawn during Stop mode, the user should turn-



off output drivers that are sourcing or sinking current, if it is practical. .



Figure 21-10 Application Example of Unused Input Port



Figure 21-11 Application Example of Unused Output Port

Note: In the STOP operation, the power dissipation associated with the oscillator and the internal hardware is lowered; however, the power dissipation associated with the pin interface (depending on the external circuitry and program) is not directly determined by the hardware operation of the STOP feature. This point should be little current flows when the input level is stable at the power voltage level (V_{DD}/V_{SS}) ; however, when the input level becomes higher than the power voltage level (by approximately 0.3V), a current begins to flow. Therefore, if cutting off the output transistor at an I/O port puts the pin signal into the highimpedance state, a current flow across the ports input transistor, requiring it to fix the level by pull-up or other means.

It should be set properly in order that current flow through port doesn't exist.



First conseider the setting to input mode. Be sure that there is no current flow after considering its relationship with external circuit. In input mode, the pin impedance viewing from external MCU is very high that the current doesn't flow.

But input voltage level should be V_{SS} or V_{DD} . Be careful that if unspecified voltage, i.e. if unfirmed voltage level (not V_{SS} or V_{DD}) is applied to input pin, there can be little current (max. 1mA at around 2V) flow.

If it is not appropriate to set as an input mode, then set to output mode considering there is no current flow. Setting to High or Low is decided considering its relationship with external circuit. For example, if there is external pull-up resistor then it is set to output mode, i.e. to High, and if there is external pull-down register, it is set to low.



22. OSCILLATOR CIRCUIT

The HMS81C2232/48 and MC80C2232/48/60 have two oscillation circuits internally. X_{IN} and X_{OUT} are input and output for main frequency. Respectively, inverting amplifier which can be

configured for being used as an on-chip oscillator, as shown in Figure 22-1.



Figure 22-1 Oscillation Circuit

Oscillation circuit is designed to be used either with a ceramic resonator or crystal oscillator. Since each crystal and ceramic resonator have their own characteristics, the user should consult the crystal manufacturer for appropriate values of external components.

In addition, see Figure 22-2 for the layout of the crystal.

Note: Minimize the wiring length. Do not allow the wiring to intersect with other signal conductors. Do not allow the wiring to come near changing high current. Set the potential of the grounding position of the oscillator capacitor to that of Vss. Do not ground it to any ground pattern where high current is present. Do not fetch signals from the oscillator.



Figure 22-2 Layout of Oscillator PCB circuit



23. RESET

The HMS81C20xxA have two types of reset generation procedures; one is an external reset input, the other is a watch-dog tim-

er reset. Table 23-1	shows on-chip	hardware	initialization by
reset action.			

On-chip Hardw	Initial Value	
Program counter	(PC)	(FFFF _H) - (FFFE _H)
RAM page register	(RPR)	0
G-flag	(G)	0
Operation mode		Main-frequency clock

On-chip Hardware	Initial Value
Peripheral clock	Off
Watchdog timer	Disable
Control registers	Refer to Table 8-1 on page 36
Power fail detector	Disable

Table 23-1 Initializing Internal Status by Reset Action

23.1 External Reset Input

The reset input is the RESET pin, which is the input to a Schmitt Trigger. A reset in accomplished by holding the RESET pin low for at least 8 oscillator periods, within the operating voltage range and oscillation stable, it is applied, and the internal state is initialized. After reset, 64ms (at 4 MHz) add with 7 oscillator periods are required to start execution as shown in Figure 23-2.

Internal RAM is not affected by reset. When V_{DD} is turned on, the RAM content is indeterminate. Therefore, this RAM should be initialized before read or tested it.

When the RESET pin input goes to high, the reset operation is released and the program execution starts at the vector address stored at addresses $FFFE_H$ - $FFFF_H$.



Figure 23-1 Simple Power-on-Reset Circuit



Figure 23-2 Timing Diagram after RESET

23.2 Watchdog Timer Reset

Refer to "12. WATCHDOG TIMER" on page 51.



24. POWER FAIL PROCESSOR

The HMS81C2232/48 and MC80C2232/48/60 have an on-chip power fail detection circuitry to immunize against power noise. A configuration register, PFDR, can enable or disable the power fail detect circuitry. Whenever V_{DD} falls close to or below power fail voltage for 100ns, the power fail situation may reset or freeze MCU according to PFDM bit of PFDR. Refer to "7.4 DC Electrical Characteristics" on page 23.

In the in-circuit emulator, power fail function is not implemented and user can not experiment with it. Therefore, after final development of user program, this function may be experimented or evaluated.

Note: User can select power fail voltage level according to PFD0, PFD1 bit of CONFIG register($307F_H$) at the OTP (HMS87C2232/48) but **must select** the power fail voltage level to define PFD option of "Mask Order & Verification Sheet" at the mask chip(HMS81C2232/48 and MC80C2232/48/60).

Because the power fail voltage level of mask chip (HMS81C2232/48 and MC80C2232/48/60) is determined according to mask option.

Note: If power fail voltage is selected to 3.0V on 3V operation, MCU is freezed at all the times.

Power FailFunction	ОТР	MASK		
Enable/Disable	PFDIS flag	PFDIS flag		
Level Selection	PFS0 bit PFS1 bit	Mask option		

Table 24-1 Power fail processor







Figure 24-2 Example S/W of RESET flow by Power fail



Figure 24-3 Power Fail Processor Situations



25. OTP PROGRAMMING

25.1 DEVICE CONFIGURATION AREA

The Device Configuration Area can be programmed or left unprogrammed to select device configuration such as security bit. This area is not accessible during normal execution but is readable and writable during program / verify.



Figure 25-1 Device Configuration Area

Pin No.			User Mode	EPROM MODE							
80TQFP	64SDIP	64MQFP	64LQFP	Pin Name	Pin Name	Description					
6	14	7	6	RESET	VPP	Programming Power (0V, 11.5V)					
3	11	4	3	XIN	PXEN	OTP Clock					
7	15	8	7	P27	CTL3						
8	16	9	8	P26	CTL2	Program/verity/Read Control					
9	17	10	9	P25	CTL1	Dramman Mada Cantral					
10	18	11	10	P24	CTL0	 Program Mode Control 					
15	19	12	11	P00	A_D0		A8	A0	D0		
16	20	13	12	P01	A_D1		A9	A1	D1		
17	21	14	13	P02	A_D2		A10	A2	D2		
19	23	16	15	P03	A_D3	Address Input	A11	A3	D3		
20	24	17	16	P04	A_D4	Data Input/Output	A12	A4	D4		
21	25	18	17	P05	A_D5		A13	A5	D5		
22	26	19	18	P06	A_D6		A14	A6	D6		
5	13	6	5	P07	A_D7		A15	A7	D7		
23	27	20	19	VSS0	VSS0	Connect to 0V					
2	10	3	2	VSS1	VSS1	Connect to 0V					
25	29	22	21	VDD0	VDD0	Connect to 5.0V (Port Power)					
1	9	2	1	VDD1	VDD1	Connect to 5.0V (Logic Power)					

25.2 Pin Description in EPROM Mode

Table 25-1 Pin Description in EPROM Mode



25.3 OTP Programming Pin Assignment



Figure 25-2 HMS87C2232/48 80MQFP Programming Pin Assignment





Figure 25-3 HMS87C2232/48 TQFP Programming Pin Assignment





Figure 25-4 HMS87C2332/48 64SDIP/MQFP Programming Pin Assignment





Figure 25-5 HMS87C2332/48 64LQFP Programming Pin Assignment





25.4 Timing Diagram of EPROM Mode

Figure 25-6 Timing Diagram in Program (Write & Verify) Mode





Figure 25-7 Timing Diagram in READ Mode



Figure 25-8 Timing Diagram in Configuration bits Program

\BOV

SEMICONDUCTOR



25.5 AC/DC Requirements for Program/Read Mod

Parameter	Symbol	MIN	TYP	MAX	Unit
Programming Supply Current	I _{VPP}	-	-	50	mA
Supply Current in EPROM Mode	I _{VDDP}	-	-	20	mA
VPP Level during Programming	V _{IHP}	11.25	11.5	11.75	V
VDD Level in Program Mode	V _{DD1H}	4.8	5.0	5.2	V
CTL3~0(P27~P24) High Level in EPROM Mode	V _{IHC}	$0.9V_{DD}$	-	-	V
CTL3~0(P27~P24) Low Level in EPROM Mode	V _{ILC}	-	-	$0.1V_{DD}$	V
A_D7~A_D0(P07~P00) High Level in EPROM Mode	V _{IHAD}	$0.9V_{DD}$	-	-	V
A_D7~A_D0(P07~P00) Low Level in EPROM Mode	V _{ILAD}	-	-	$0.1V_{DD}$	V
VDD Saturation Time	T _{VDDS}	1	-	-	mS
VPP Setup Time	T _{VPPR}	-	-	1	mS
VPP Saturation Time	T _{VPPS}	1	-	-	mS
EPROM Enable Setup Time after Data Input	T _{SET1}	-	500±20	-	nS
EPROM Enable Hold Time after T _{SET1}	T _{HLD1}	-	500±50	-	nS
EPROM Enable Delay Time after T _{HLD1}	T _{DLY1}	-	500±20	-	nS
EPROM Enable Hold Time in Write Mode	T _{HLD2}	-	50±10	-	uS
EPROM Enable Delay Time after T _{HLD2}	T _{DLY2}	-	500±20	-	nS
Reset Pin Pulse Input Duty	T _{DUTY}	-	1.0±0.1	-	uS
Reset Pin Pulse Input Period	T _{PERIOD}	-	2.0±0.1	-	uS



25.6 Programming Flow Chart



Figure 25-9 Programming Flow Chart







A. INSTRUCTION

A.1 Terminology List

Terminology	Description
A	Accumulator
X	X - register
Y	Y - register
PSW	Program Status Word
#imm	8-bit Immediate data
dp	Direct Page Offset Address
!abs	Absolute Address
[]	Indirect expression
{ }	Register Indirect expression
{ }+	Register Indirect expression, after that, Register auto-increment
.bit	Bit Position
A.bit	Bit Position of Accumulator
dp.bit	Bit Position of Direct Page Memory
M.bit	Bit Position of Memory Data (000 _H ~0FFF _H)
rel	Relative Addressing Data
upage	U-page (0FF00 _H ~0FFFF _H) Offset Address
n	Table CALL Number (0~15)
+	Addition
x	Upper Nibble Expression in Opcode
У	Upper Nibble Expression in Opcode
_	Subtraction
×	Multiplication
/	Division
()	Contents Expression
^	AND
~	OR
\oplus	Exclusive OR
~	NOT
\leftarrow	Assignment / Transfer / Shift Left
\rightarrow	Shift Right
\leftrightarrow	Exchange
=	Equal
≠	Not Equal



A.2 Instruction Map

LOW HIGH	00000 00	00001 01	00010 02	00011 03	00100 04	00101 05	00110 06	00111 07	01000 08	01001 09	01010 0A	01011 0B	01100 0C	01101 0D	01110 0E	01111 0F
000	-	SET1 dp.bit	BBS A.bit,rel	BBS dp.bit,rel	ADC #imm	ADC dp	ADC dp+X	ADC !abs	ASL A	ASL dp	TCALL 0	SETA1 .bit	BIT dp	POP A	PUSH A	BRK
001	CLRC	ű	"	cc	SBC #imm	SBC dp	SBC dp+X	SBC !abs	ROL A	ROL dp	TCALL 2	CLRA1 .bit	COM dp	POP X	PUSH X	BRA rel
010	CLRG	"	"	66	CMP #imm	CMP dp	CMP dp+X	CMP !abs	LSR A	LSR dp	TCALL 4	NOT1 M.bit	TST dp	POP Y	PUSH Y	PCALL Upage
011	DI	"	**	cc	OR #imm	OR dp	OR dp+X	OR !abs	ROR A	ROR dp	TCALL 6	OR1 OR1B	CMPX dp	POP PSW	PUSH PSW	RET
100	CLRV	"	"	cc	AND #imm	AND dp	AND dp+X	AND !abs	INC A	INC dp	TCALL 8	AND1 AND1B	CMPY dp	CBNE dp+X	TXSP	INC X
101	SETC	"	"	cc	EOR #imm	EOR dp	EOR dp+X	EOR !abs	DEC A	DEC dp	TCALL 10	EOR1 EOR1B	DBNE dp	XMA dp+X	TSPX	DEC X
110	SETG	"	"	cc	LDA #imm	LDA dp	LDA dp+X	LDA !abs	TXA	LDY dp	TCALL 12	LDC LDCB	LDX dp	LDX dp+Y	XCN	DAS (N/A)
111	EI	"	"	£6	LDM dp,#imm	STA dp	STA dp+X	STA !abs	TAX	STY dp	TCALL 14	STC M.bit	STX dp	STX dp+Y	XAX	STOP

LOW HIGH	10000 10	10001 11	10010 12	10011 13	10100 14	10101 15	10110 16	10111 17	11000 18	11001 19	11010 1A	11011 1B	11100 1C	11101 1D	11110 1E	11111 1F
000	BPL rel	CLR1 dp.bit	BBC A.bit,rel	BBC dp.bit,rel	ADC {X}	ADC !abs+Y	ADC [dp+X]	ADC [dp]+Y	ASL !abs	ASL dp+X	TCALL 1	JMP !abs	BIT !abs	ADDW dp	LDX #imm	JMP [!abs]
001	BVC rel	"	££	"	SBC {X}	SBC !abs+Y	SBC [dp+X]	SBC [dp]+Y	ROL !abs	ROL dp+X	TCALL 3	CALL !abs	TEST !abs	SUBW dp	LDY #imm	JMP [dp]
010	BCC rel	"	££	"	CMP {X}	CMP !abs+Y	CMP [dp+X]	CMP [dp]+Y	LSR !abs	LSR dp+X	TCALL 5	MUL	TCLR1 !abs	CMPW dp	CMPX #imm	CALL [dp]
011	BNE rel	**	**	"	OR {X}	OR !abs+Y	OR [dp+X]	OR [dp]+Y	ROR !abs	ROR dp+X	TCALL 7	DBNE Y	CMPX !abs	LDYA dp	CMPY #imm	RETI
100	BMI rel	**	**	"	AND {X}	AND !abs+Y	AND [dp+X]	AND [dp]+Y	INC !abs	INC dp+X	TCALL 9	DIV	CMPY !abs	INCW dp	INC Y	TAY
101	BVS rel	"	66	"	EOR {X}	EOR !abs+Y	EOR [dp+X]	EOR [dp]+Y	DEC !abs	DEC dp+X	TCALL 11	XMA {X}	XMA dp	DECW dp	DEC Y	TYA
110	BCS rel	"	66	"	LDA {X}	LDA !abs+Y	LDA [dp+X]	LDA [dp]+Y	LDY !abs	LDY dp+X	TCALL 13	LDA {X}+	LDX !abs	STYA dp	XAY	DAA (N/A)
111	BEQ rel	"	"	"	STA {X}	STA !abs+Y	STA [dp+X]	STA [dp]+Y	STY !abs	STY dp+X	TCALL 15	STA {X}+	STX !abs	CBNE dp	XYX	NOP



A.3 Instruction Set

Arithmetic / Logic Operation

NO.	MNEMONIC	OP CODE	BYTE NO	CYCLE NO	OPERATION	FLAG NVGBHIZC
1	ADC #imm	04	2	2		
2	ADC dp	05	2	3		
3	ADC dp + X	06	2	4		
4	ADC !abs	07	3	4	Add with carry.	NVH-ZC
5	ADC !abs + Y	15	3	5	$A \leftarrow (A) + (M) + C$	
6	ADC [dp + X]	16	2	6		
7	ADC [dp]+Y	17	2	6		
8	ADC {X}	14	1	3		
9	AND #imm	84	2	2		
10	AND dp	85	2	3		
11	AND dp + X	86	2	4		
12	AND labs	87	3	4	Logical AND	
13	AND !abs + Y	95	3	5	$A \leftarrow (A) \land (M)$	NZ-
14	AND [dp + X]	96	2	6		
15	AND [dp]+Y	97	2	6		
16	AND {X}	94	1	3		
17	ASL A	08	1	2	Arithmetic shift left	
18	ASL dp	09	2	4	C 7 6 5 4 3 2 1 0	
19	ASL dp + X	19	2	5		NZC
20	ASL !abs	18	3	5		
21	CMP #imm	44	2	2		
22	CMP dp	45	2	3		
23	CMP dp + X	46	2	4		NZC
24	CMP !abs	47	3	4	Compare accumulator contents with memory contents	
25	CMP !abs + Y	55	3	5	(A)-(M)	
26	CMP [dp + X]	56	2	6		
27	CMP [dp]+Y	57	2	6		
28	CMP {X}	54	1	3		
29	CMPX #imm	5E	2	2		
30	CMPX dp	6C	2	3	Compare X contents with memory contents	NZC
31	CMPX !abs	7C	3	4		
32	CMPY #imm	7E	2	2		
33	CMPY dp	8C	2	3	Compare Y contents with memory contents $(X) = (M)$	NZC
34	CMPY !abs	9C	3	4		
35	COM dp	2C	2	4	1'S Complement : (dp) $\leftarrow \sim$ (dp)	NZ-
36	DAA	DF	1	3	Decimal adjust for addition	NZC
37	DAS	CF	1	3	Decimal adjust for subtraction	NZC
38	DEC A	A8	1	2		
39	DEC dp	A9	2	4		
40	DEC dp + X	B9	2	5	Decrement	
41	DEC labs	B8	3	5	M ← (M) - 1	NZ-
42	DEC X	AF	1	2		
43	DEC Y	BE	1	2		
44	DIV	9B	1	12	Divide : YA / X Q: A, R: Y	NVH-Z-



NO.	MNEMONIC	OP CODE	BYTE NO	CYCLE NO	OPERATION	FLAG NVGBHIZC
45	EOR #imm	A4	2	2		
46	EOR dp	A5	2	3		
47	EOR dp + X	A6	2	4		
48	EOR !abs	A7	3	4	Exclusive OR	
49	EOR !abs + Y	B5	3	5	$A \leftarrow (A) \oplus (M)$	NZ-
50	EOR [dp + X]	B6	2	6		
51	EOR [dp]+Y	B7	2	6		
52	EOR {X}	B4	1	3		
53	INC A	88	1	2		NZC
54	INC dp	89	2	4		
55	INC dp + X	99	2	5	Increment	
56	INC labs	98	3	5	M ← (M) + 1	NZ-
57	INC X	8F	1	2		
58	INC Y	9E	1	2		
59	LSR A	48	1	2	Logical shift right	
60	LSR dp	49	2	4	76543210 C	
61	LSR dp + X	59	2	5	"0" —> => => => => =>	NZC
62	LSR !abs	58	3	5		
63	MUL	5B	1	9	Multiply : $YA \leftarrow Y \times A$	NZ-
64	OR #imm	64	2	2		
65	OR dp	65	2	3		
66	OR dp + X	66	2	4		
67	OR !abs	67	3	4	Logical OR	
68	OR !abs + Y	75	3	5	$A \leftarrow (A) \lor (M)$	NZ-
69	OR [dp + X]	76	2	6		
70	OR [dp]+Y	77	2	6		
71	OR { X }	74	1	3		
72	ROL A	28	1	2	Rotate left through carry	
73	ROL dp	29	2	4	C 7 6 5 4 3 2 1 0	
74	ROL dp + X	39	2	5		NZC
75	ROL labs	38	3	5		
76	ROR A	68	1	2	Rotate right through carry	
77	ROR dp	69	2	4	76543210 C	
78	ROR dp + X	79	2	5	~- 	NZC
79	ROR !abs	78	3	5		
80	SBC #imm	24	2	2		
81	SBC dp	25	2	3		
82	SBC dp + X	26	2	4		
83	SBC labs	27	3	4	Subtract with carry	
84	SBC !abs + Y	35	3	5	A ← (A) - (M) - ~(C)	NVHZC
85	SBC [dp + X]	36	2	6		
86	SBC [dp]+Y	37	2	6		
87	SBC {X}	34	1	3		
88	TST dp	4C	2	3	Test memory contents for negative or zero (dp) - 00 _H	NZ-
89	XCN	CE	1	5	Exchange nibbles within the accumulator $A_7 \sim A_4 \leftrightarrow A_3 \sim A_0$	NZ-



Register / Memory Operation

NO.	MNEMONIC	OP	BYTE	CYCLE	OPERATION	FLAG
1	LDA #imm	CODE C4	<u>NO</u> 2	NO 2		NVGBHIZC
2	LDA dp	C5	2	- 3		
3	LDA dp + X	C6	2	4		
4	LDA labs	C7	3	4	Load accumulator	
5	LDA !abs + Y	D5	3	5	$A \leftarrow (M)$	NZ-
6	LDA $[dp + X]$	D6	2	6	$\Lambda \subset (W)$	
7	LDA [dp]+Y	D7	2	6		
8	LDA {X}	D4	1	3		
9	LDA { X }+	DB	1	4	X ₋ register auto-increment : $A \leftarrow (M)$ X \leftarrow X + 1	1
10	LDM dp.#imm	E4	3	5	Load memory with immediate data : $(M) \leftarrow imm$	
11	LDX #imm	1E	2	2		
12	LDX dp	CC	2	3	Load X-register	
13	LDX dp + Y	CD	2	4	$X \leftarrow (M)$	NZ-
14	LDX labs	DC	3	4	// · · ()	
15	LDY #imm	3E	2	2		
16	LDY dp	C9	2	3	Load Y-register	
17	LDY dp + X	D9	2	4	$Y \leftarrow (M)$	NZ-
18	LDY labs	D8	3	4		
19	STA dp	E5	2	4		
20	STA dp + X	E6	2	5		
21	STA labs	E7	3	5		
22	STA !abs + Y	F5	3	6	Store accumulator contents in memory	
23	STA [dp + X]	F6	2	7	$(M) \leftarrow A$	
24	STA [dp]+Y	F7	2	7		
25	STA {X}	F4	1	4		
26	STA {X}+	FB	1	4	X- register auto-increment : (M) \leftarrow A. X \leftarrow X + 1	
27	STX dp	EC	2	4		
28	STX dp + Y	ED	2	5	Store X-register contents in memory	
29	STX !abs	FC	3	5	$(M) \leftarrow X$	
30	STY dp	E9	2	4	.	
31	STY dp + X	F9	2	5	Store Y-register contents in memory	
32	STY labs	F8	3	5	(M) ← Y	
33	TAX	E8	1	2	Transfer accumulator contents to X-register : $X \leftarrow A$	NZ-
34	TAY	9F	1	2	Transfer accumulator contents to Y-register : $Y \leftarrow A$	NZ-
35	TSPX	AE	1	2	Transfer stack-pointer contents to X-register : $X \leftarrow sp$	NZ-
36	ТХА	C8	1	2	Transfer X-register contents to accumulator: $A \leftarrow X$	NZ-
37	TXSP	8E	1	2	Transfer X-register contents to stack-pointer: sp \leftarrow X	NZ-
38	TYA	BF	1	2	Transfer Y-register contents to accumulator: A \leftarrow Y	NZ-
39	XAX	EE	1	4	Exchange X-register contents with accumulator : $X \leftrightarrow A$	
40	XAY	DE	1	4	Exchange Y-register contents with accumulator : $Y \leftrightarrow A$	
41	XMA dp	BC	2	5	Exchange memory contents with accumulator	
42	XMA dp+X	AD	2	6	$(M) \leftrightarrow A$	NZ-
43	XMA {X}	BB	1	5		
44	XYX	FE	1	4	Exchange X-register contents with Y-register : $X \leftrightarrow Y$	



16-BIT Operation

NO.	MNEMONIC	OP CODE	BYTE NO	CYCLE NO	OPERATION	FLAG NVGBHIZC
1	ADDW dp	1D	2	5	16-Bits add without carry YA \leftarrow (YA) + (dp +1) (dp)	NVH-ZC
2	CMPW dp	5D	2	4	Compare YA contents with memory pair contents : $(YA) - (dp+1)(dp)$	NZC
3	DECW dp	BD	2	6	Decrement memory pair (dp+1)(dp) \leftarrow (dp+1) (dp) - 1	NZ-
4	INCW dp	9D	2	6	Increment memory pair (dp+1) (dp) \leftarrow (dp+1) (dp) + 1	NZ-
5	LDYA dp	7D	2	5	Load YA YA ← (dp +1) (dp)	NZ-
6	STYA dp	DD	2	5	Store YA (dp +1) (dp) ← YA	
7	SUBW dp	3D	2	5	16-Bits substact without carry YA \leftarrow (YA) - (dp +1) (dp)	NVH-ZC

Bit Manipulation

NO.	MNEMONIC	OP CODE	BYTE NO	CYCLE NO	OPERATION	FLAG NVGBHIZC
1	AND1 M.bit	8B	3	4	Bit AND C-flag : C \leftarrow (C) \land (M .bit)	C
2	AND1B M.bit	8B	3	4	Bit AND C-flag and NOT $: C \leftarrow (C) \land \sim (M .bit)$	C
3	BIT dp	0C	2	4	Bit test A with memory :	204 5
4	BIT !abs	1C	3	5	$Z \leftarrow (A) \land (M), N \leftarrow (M_7), V \leftarrow (M_6)$	MMZ-
5	CLR1 dp.bit	y1	2	4	Clear bit : (M.bit) \leftarrow "0"	
6	CLRA1 A.bit	2B	2	2	Clear A bit ∶ (A.bit)← "0"	
7	CLRC	20	1	2	Clear C-flag : C \leftarrow "0"	0
8	CLRG	40	1	2	Clear G-flag : $G \leftarrow "0"$	0
9	CLRV	80	1	2	Clear V-flag : V \leftarrow "0"	-00
10	EOR1 M.bit	AB	3	5	Bit exclusive-OR C-flag $: C \leftarrow (C) \oplus (M.bit)$	C
11	EOR1B M.bit	AB	3	5	Bit exclusive-OR C-flag and NOT : C \leftarrow (C) \oplus ~(M .bit)	C
12	LDC M.bit	CB	3	4	Load C-flag : $C \leftarrow (M.bit)$	C
13	LDCB M.bit	CB	3	4	Load C-flag with NOT $: C \leftarrow \sim (M .bit)$	C
14	NOT1 M.bit	4B	3	5	Bit complement : (M .bit) $\leftarrow \sim$ (M .bit)	
15	OR1 M.bit	6B	3	5	Bit OR C-flag : C \leftarrow (C) \lor (M .bit)	C
16	OR1B M.bit	6B	3	5	Bit OR C-flag and NOT $: C \leftarrow (C) \lor \sim (M .bit)$	C
17	SET1 dp.bit	x1	2	4	Set bit : (M.bit) \leftarrow "1"	
18	SETA1 A.bit	0B	2	2	Set A bit : (A.bit) \leftarrow "1"	
19	SETC	A0	1	2	Set C-flag : $C \leftarrow "1"$	1
20	SETG	C0	1	2	Set G-flag ∶ G ← "1"	1
21	STC M.bit	EB	3	6	Store C-flag ∶ (M .bit) ← C	
22	TCLR1 !abs	5C	3	6	Test and clear bits with A : A - (M), (M) \leftarrow (M) \wedge ~(A)	NZ-
23	TSET1 !abs	3C	3	6	Test and set bits with A : A - (M) , (M) ← (M) ∨ (A)	NZ-



Branch / Jump Operation

NO.	MNEMONIC	OP CODF	BYTE NO	CYCLE NO	OPERATION	FLAG NVGBHIZC	
1	BBC A.bit,rel	y2	2	4/6	Branch if bit clear :		
2	BBC dp.bit,rel	у3	3	5/7	if (bit) = 0, then $pc \leftarrow (pc) + rel$		
3	BBS A.bit,rel	x2	2	4/6	Branch if bit set :		
4	BBS dp.bit,rel	x3	3	5/7	if (bit) = 1, then $pc \leftarrow (pc) + rel$		
E	D 00 I	50	2	2/4	Branch if carry bit clear		
5	BCC rel	50	2	2/4	if (C) = 0, then $pc \leftarrow (pc) + rel$		
6	BCS rel	D0	2	2/4	Branch if carry bit set if $(C_{i}) = 1$, then $p_{i} \leftarrow (p_{i}) + rel$		
					Branch if equal		
7	BEQ rel	F0	2	2/4	if $(7) = 1$ then $pc \leftarrow (pc) + rel$		
•		00	•	0/4	Branch if minus		
8	BMI rel	90	2	2/4	if (N) = 1, then $pc \leftarrow (pc) + rel$		
0		70	2	2/4	Branch if not equal		
9	BINE IEI	70	2	2/4	if (Z) = 0 , then $pc \leftarrow (pc) + rel$		
10	BDI rol	10	2	2/4	Branch if plus		
	DILIG		_		if (N) = 0, then $pc \leftarrow (pc) + rel$		
11	BRA rel	2F	2	4	Branch always		
					$pc \leftarrow (pc) + rel$		
12	BVC rel	30	2	2/4	Branch if overflow bit clear		
					If $(V) = 0$, then $pc \leftarrow (pc) + rel$		
13	BVS rel	B0	2	2/4	$\frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{1000} \frac{1}{10000} \frac{1}{10000000000000000000000000000000000$		
14		3B	3	8	$II(V) = I$, then $pc \leftarrow (pc) + Iei$		
	CALL :abs				$M(sp) \leftarrow (p_{CL}), sp \leftarrow sp - 1, M(sp) \leftarrow (p_{CL}), sp \leftarrow sp - 1$		
15	CALL [dp]	5F	2	8	if labs, $pc \leftarrow abs$: if [dp], $pc_{i} \leftarrow (dp)$, $pc_{i} \leftarrow (dp+1)$.		
16	CBNE dp,rel	FD	3	5/7	Compare and branch if not equal :		
17	CBNE dp+X,rel	8D	3	6/8	if $(A) \neq (M)$, then $pc \leftarrow (pc) + rel$.		
18	DBNE dp,rel	AC	3	5/7	Decrement and branch if not equal :		
19	DBNE Y,rel	7B	2	4/6	if $(M) \neq 0$, then $pc \leftarrow (pc) + rel$.		
20	JMP !abs	1B	3	3			
21	JMP [labs]	1F	3	5	Unconditional jump		
22	JMP [dp]	3F	2	4	pc ← jump address		
23	PCALL upage	4F	2	6	U-page call M(sp) ←(pc _H), sp ←sp - 1, M(sp) ← (pc _L), sp ← sp - 1, pc _L ← (upage), pc _H ← "0FF _H ".		
24	TCALL n	nA	1	8	Table call : (sp) \leftarrow (pc _H), sp \leftarrow sp - 1,M(sp) \leftarrow (pc _L),sp \leftarrow sp - 1,pc _L \leftarrow (Table vector L), pc _H \leftarrow (Table vector H)		



Control Operation & Etc.

NO.	MNEMONIC	OP CODE	BYTE NO	CYCLE NO	OPERATION	FLAG NVGBHIZC
1	BRK	0F	1	8	Software interrupt : $B \leftarrow "1"$, $M(sp) \leftarrow (pc_H)$, $sp \leftarrow sp-1$, $M(s) \leftarrow (pc_L)$, $sp \leftarrow sp - 1$, $M(sp) \leftarrow (PSW)$, $sp \leftarrow sp - 1$, $pc_L \leftarrow (0FFDE_H)$, $pc_H \leftarrow (0FFDF_H)$.	1-0
2	DI	60	1	3	Disable interrupts : $I \leftarrow "0"$	0
3	EI	E0	1	3	Enable interrupts ∶ I ← "1"	1
4	NOP	FF	1	2	No operation	
5	POP A	0D	1	4	$sp \leftarrow sp + 1, A \leftarrow M(sp)$	
6	POP X	2D	1	4	$sp \leftarrow sp + 1, X \leftarrow M(sp)$	
7	POP Y	4D	1	4	$sp \leftarrow sp + 1, Y \leftarrow M(sp)$	
8	POP PSW	6D	1	4	$sp \leftarrow sp + 1$, $PSW \leftarrow M(sp)$	restored
9	PUSH A	0E	1	4	$M(sp) \leftarrow A, sp \leftarrow sp - 1$	
10	PUSH X	2E	1	4	$M(sp) \leftarrow X, sp \leftarrow sp - 1$	
11	PUSH Y	4E	1	4	$M(sp) \leftarrow Y, sp \leftarrow sp - 1$	
12	PUSH PSW	6E	1	4	$M(sp) \leftarrow PSW$, $sp \leftarrow sp - 1$	
13	RET	6F	1	5	$\begin{array}{l} \mbox{Return from subroutine} \\ \mbox{sp} \leftarrow \mbox{sp +1, pc}_L \leftarrow \mbox{M(sp), sp} \leftarrow \mbox{sp +1, pc}_H \leftarrow \mbox{M(sp)} \end{array}$	
14	RETI	7F	1	6	Return from interrupt $sp \leftarrow sp +1$, PSW $\leftarrow M(sp)$, $sp \leftarrow sp + 1$, $pc_L \leftarrow M(sp)$, $sp \leftarrow sp + 1$, $pc_H \leftarrow M(sp)$	restored
15	STOP	EF	1	3	Stop mode (halt CPU, stop oscillator)	



B. MASK ORDER SHEET(HMS81C2232/2248)

MASK ORDER & VERIFICATION SHEET								
HMS8 ²	1C22							
	→ 32 : 32K → 48 : 48K		QA : 80 TQ : 80	MQFP TQFP	Blank : Normal PKG ➤ P : Pb Free PKG B : Pb/Halogen Free PKG			
Customer should write ins	ide thick line box.	2. D	2. Device Information					
Company Name		Pac	kage 80	MQFP	80TQFP			
Application			File Name		().OTP			
Order Date	YYYY MM DD		ROM Size (bytes)	32K48K			
Tel:	Fax:	sk Data	Check Sum	() (48K) 4000 _H (32K) 8000 _H			
E-mail address: Name & Signature:		Ma	Set "00 _H " in blanked area					
	32 or 48		Customer's logo					
YYWW KOREA		H Y C	HMS81C22XX-HL					
If the customer logo n	nust be used in the special mark, pl	ease subm	t a clean origina	l of the lo	ogo.			
Customer's part numb	ber							
4. Delivery Schedu	le							
	(Quantity		ABOV Confirmation				
Customer sample	YYYY MM DD		pcs					
Risk order	YYYY MM DD • •		pcs					
5. ROM Code Verific Please confirm out verific	cation cation data.							

Verification date:	YYYY	MM •	DD	Tel:	Fax:
Check sum:				Name &	
E-mail address:				Signature:	





Customer should write inside thick line box. (Please check mark $\sqrt{into \Box}$)

1. PFD Level Selection



PFD Level Selection 00: PFD = 2.7V 01: PFD = 2.7V 10: PFD = 3.0V 11: PFD = 2.4V

2. H/V Port Option Check (Pull-down Option Check)

Port	Option				
FUIL	Off	Vdisp			
FIP0					
FIP1					
FIP2					
FIP3					
FIP4					
FIP5					
FIP6					
FIP7					

 Option

 Off
 Vdisp

 FIP8

 FIP9

 FIP10

 FIP11

 FIP12

 FIP13

 FIP14

Dort	Option				
FUIL	Off	Vdisp			
FIP16					
FIP17					
FIP18					
FIP19					
FIP20					
FIP21					
FIP22					
FIP23					

Port	Option		
FUIL	Off	Vdisp	
P30/FIP24			
P31/FIP25			
P32/FIP26			
P33/FIP27			
P34/FIP28			
P35/FIP29			
P36/FIP30			
P37/FIP31			

Port	Option		
FUIL	Off	Vdisp	
P40/FIP32			
P41/FIP33			
P42/FIP34			
P43/FIP35			
P44/FIP36			
P45/FIP37			
P46/FIP38			
P47/FIP39			

Port	Option			
FUIL	Off	Vdisp	Vss	
P50/FIP40				
P51/FIP41				
P52/FIP42				
P53/FIP43				
P54/FIP44				
P55/FIP45				
P56/FIP46				
P57/FIP47				

Dort	Option			
FUIL	Off	Vdisp	Vss	
P60/FIP48				
P61/FIP49				
P62/FIP50				
P63/FIP51				
P64/FIP52				

Off : without pull-down resistorr Vdisp : with pull-down resistor connected to Vdisp Vss : with pull-down resistor connected to Vss

3. Reset Port Option Check (Pull-up Option Check)

DEOFT	Off	On
RESET		

Off : without pull-up resistorr On : with pull-up resistor



C. MASK ORDER SHEET(HMS81C2332/2348)

MASK ORDER & VERIFICATION SHEET				
HMS	81C23 -HL		K : 64SDIP Q : 64MQFP LQ : 64LQFP Blank : Normal PKG P : Pb Free PKG B : Pb/Halogen Free PKG	
Customer should write in 1. Customer Inform	iside thick line box.	2. Dev	vice Information	
Company Name		Pack	kage 64SDIP 64MQFP 64LQFP	
Application		_	File Name ().OTP	
Order Date	YYYY MM DD		ROM Size (bytes) 32K 48K	
Tel: E-mail address:	Fax:	Mask Data	Check Sum ()	
Name & Signature:			Set "00 _H " in blanked area	
Joint of the customer logo must be used in the special mark, please submit a clean original of the logo. Customer's part number 4. Delivery Schedule				
	Date	Qua	antity ABOV Confirmation	
Customer sample	YYYY MM DD • •		pcs	
Risk order	YYYY MM DD • •		pcs	
5. ROM Code Verifi Please confirm out verifi	ication cation data.			
Verification date:	YYYY MM DD • •	Tel:	Fax:	
Check sum:		Name &		
E-mail address:				



٦



Customer should write inside thick line box. (Please check mark $\sqrt{into \Box}$)

1. PFD Level Selection



2. H/V Port Option Check (Pull-down Option Check)

Port	Option		
	Off	Vdisp	
FIP0			
FIP1			
FIP2			
FIP3			
FIP4			
FIP5			
FIP6			
FIP7			

Port	Option		
FUIL	Off	Vdisp	
FIP8			
FIP9			
FIP10			
FIP11			
FIP12			
FIP13			
FIP14			
FIP15			

Option		
Off	Vdisp	
	Op ⁱ Off	

Port	Option		
	Off	Vdisp	
P30/FIP24			
P31/FIP25			
P32/FIP26			
P33/FIP27			
P34/FIP28			
P35/FIP29			
P36/FIP30			
P37/FIP31			

Port	Option		
	Off	Vdisp	
P40/FIP32			
P41/FIP33			
P42/FIP34			
P43/FIP35			
P44/FIP36			

Dort	Option			
FUIL	Off	Vdisp	Vss	
P50/FIP40				
P51/FIP41				
P52/FIP42				
P53/FIP43				
P54/FIP44				
P55/FIP45				

Port	Option		
	Off	Vdisp	Vss
P62/FIP50			
P63/FIP51			
P64/FIP52			

Off : without pull-down resistorr Vdisp : with pull-down resistor connected to Vdisp Vss : with pull-down resistor connected to Vss

3. Reset Port Option Check (Pull-up Option Check)

RESET	Off	On

Off : without pull-up resistorr On : with pull-up resistor


D. MASK ORDER SHEET(MC80C2232/2248/2260)

MASK ORDER & VERIFICATION SHEET					
МС	80C22 -VA				
	→ 32 : 32K 48 : 48K 60 : 60K		Q : 80MQF T : 80TQFI	P	Blank : Pb Free PKG B : Pb/Halogen Free PKG
Customer should write ins 1. Customer Informa	ide thick line box. ation	2. D	evice Inform	ation	
Company Name		Pack	kage 80	MQFP	80TQFP
Application			File Name		().OTP
Order Date	YYYY MM DD • •		ROM Size (bytes)	☐ 32K <u></u> 48K <u>60K</u>
Tel:	Fax:	k Data	Check Sum	() (48K) 4000 H
E-mail address:		Mas		0-+ "00	.OTP file
Name & Signature:				Set "00 _H	FFFF
ABO MC80C22XX-VA YYWW KOREA If the customer logo n Customer's part numb 4. Delivery Schedu	nust be used in the special mark, ple	ase submi	Customer's	logo	Customer logo is not required.
	Date	(Quantity		ABOV Confirmation
Customer sample	• •		pcs		
Risk order	MM DD		pcs		
5. ROM Code Verifi Please confirm out verific	cation cation data.				
Verification date:	YYYY MM DD • •	Tel:		Fax:	
Check sum:		Name 8) X		
E-mail address:		Signatu	re:		





Customer should write inside thick line box. (Please check mark $\sqrt{into \Box}$)

1. PFD Level Selection



PFD Level Selection 00: PFD = 2.7V 01: PFD = 2.7V 10: PFD = 3.0V 11: PFD = 2.4V

2. H/V Port Option Check (Pull-down Option Check)

Port	Option		
FUIL	Off	Vdisp	
FIP0			
FIP1			
FIP2			
FIP3			
FIP4			
FIP5			
FIP6			
FIP7			

 Option

 Off
 Vdisp

 FIP8

 FIP9

 FIP10

 FIP11

 FIP12

 FIP13

 FIP14

 FIP15

Dort	Option		
FUIL	Off	Vdisp	
FIP16			
FIP17			
FIP18			
FIP19			
FIP20			
FIP21			
FIP22			
FIP23			

Port	Option		
i olt	Off	Vdisp	
P30/FIP24			
P31/FIP25			
P32/FIP26			
P33/FIP27			
P34/FIP28			
P35/FIP29			
P36/FIP30			
P37/FIP31			

Port	Op	otion
FUIL	Off	Vdisp
P40/FIP32		
P41/FIP33		
P42/FIP34		
P43/FIP35		
P44/FIP36		
P45/FIP37		
P46/FIP38		
P47/FIP39		

Dort	Option				
FUIL	Off	Vdisp	Vss		
P50/FIP40					
P51/FIP41					
P52/FIP42					
P53/FIP43					
P54/FIP44					
P55/FIP45					
P56/FIP46					
P57/FIP47					

Dort	Option			
FUIL	Off	Vdisp	Vss	
P60/FIP48				
P61/FIP49				
P62/FIP50				
P63/FIP51				
P64/FIP52				

Off : without pull-down resistorr Vdisp : with pull-down resistor connected to Vdisp Vss : with pull-down resistor connected to Vss

3. Reset Port Option Check (Pull-up Option Check)

DEOFT	Off	On
RESET		

Off : without pull-up resistorr On : with pull-up resistor



E. MASK ORDER SHEET(MC80C2332/2348)



Please confirm out verification data.

Verification date:	YYYY •	MM •	DD	Tel: Fax:
Check sum:				Name &
E-mail address:				Signature:





Customer should write inside thick line box. (Please check mark $\sqrt{into \Box}$)

1. PFD Level Selection



PFD Level Selection 00: PFD = 2.7V 01: PFD = 2.7V 10: PFD = 3.0V 11: PFD = 2.4V

2. H/V Port Option Check (Pull-down Option Check)

Dort	Option		
FUIL	Off	Vdisp	
FIP0			
FIP1			
FIP2			
FIP3			
FIP4			
FIP5			
FIP6			
FIP7			

Port	Op	tion
FUIL	Off	Vdisp
FIP8		
FIP9		
FIP10		
FIP11		
FIP12		
FIP13		
FIP14		
FIP15		

Port	Option		
FUIL	Off	Vdisp	
FIP16			
FIP17			
FIP18			

Port	Option		
	Off	Vdisp	
P30/FIP24			
P31/FIP25			
P32/FIP26			
P33/FIP27			
P34/FIP28			
P35/FIP29			
P36/FIP30			
P37/FIP31			

Port	Option	
	Off	Vdisp
P40/FIP32		
P41/FIP33		
P42/FIP34		
P43/FIP35		
P44/FIP36		

Port	Option		
	Off	Vdisp	Vss
P50/FIP40			
P51/FIP41			
P52/FIP42			
P53/FIP43			
P54/FIP44			
P55/FIP45			

Port	Option		
	Off	Vdisp	Vss
P62/FIP50			
P63/FIP51			
P64/FIP52			

Off : without pull-down resistorr Vdisp : with pull-down resistor connected to Vdisp Vss : with pull-down resistor connected to Vss

3. Reset Port Option Check (Pull-up Option Check)

RESET	Off	On

Off : without pull-up resistorr On : with pull-up resistor

