LG Semicon
8-bit Microcontrollers

GMS81604/08

Revision History

Rev 1.2 (Dec. 1998)

Redraw package dimension on page 5~6.

Rev 1.1 (Nov. 1998)

Operating Voltage, 2.7~5.5V is extended with 2.4~5.5V.
Operating Temperature, -20~80°C is extended with -20~85°C.
Add the "Typical Characteristics" on page 16, 17.
Add the unused port guidance on page 48.
Revision the information for the OTP programming guidance, recommand using "Intelligent Mode" on page 49.
Add the chapter for OTP programming specification as an appendix.

Rev 1.0 (Nov. 1997)

First Edition

Second Edition

Published by
MCU Application Team

Additional information of this manual may be served by LG Semicon offices in Korea or Distributors and Representatives listed at address directory.

LG Semicon 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, LG Semicon Co,. Ltd. is in no way responsible for any violations of patents or other rights of the third party generated by the use of this manual.

Table of Contents

APPENDIX

GMS81604 / GMS81608
CMOS SINGLE-CHIP 8-BIT MICROCONTROLLER

OVERVIEW

Description

The GMS81604/08 is a high-performance CMOS 8-bit microcontroller with 4K or 8K bytes of ROM. The device is one of GMS800 family. The LG Semicon GMS81604/08 is a powerful microcontroller which provides a highly flexible and cost effective solution to many embedded control applications. The GMS81604/08 provides the following standard features: 8K bytes of ROM, 256 bytes of RAM, 35 I/O lines(33 lines for 40PDIP), 16-bit or 8-bit timer/counter, a precision analog to digital converter, on-chip oscillator and clock circuitry. In addition, the GMS81604/08 supports power saving modes to reduce power consumption. The Stop Mode saves the RAM contents but freezes the oscillator disabling all other chip functions until the next hardware reset or external interrupt.

Features

- **4K/ 8K On-chip Program Memory**
- **256 Bytes of On-Chip Data RAM**
- **Instruction execution time: 0.5us at 8MHz**
- **2.4V to 5.5V Operating Range**
- **1~8 MHz Operating frequency**
- **Basic Interval Timer**
- **Four 8-Bit Timer/ Counters (can be used as two 16-bit)**
- **Four external interrupt ports**
- **Two Programmable Clock Out**
- **One Buzzer Driving port**
- **31 Programmable I/O, 4 Input pins,**
- **Twelve Interrupt Sources**
- **All LED Direct Drive Output Ports**
- **8-Channel 8-Bit On-Chip Analog to Digital Converter**
- **Power Fail Processor (Noise immunity circuit)**
- **Power Down Mode (Stop Mode)**

Memory Proliferation

Device	ROM Bytes	RAM Bytes
GMS81604	4K	256
GMS81608	8K	256
GMS81608T	8K EPROM	256

Development Tools

The GMS800 family is supported by a full-featured macro assembler, an in-circuit emulators CHOICE-Jr.TM, socket adapters for OTP device.

The availability of OTP devices are especially useful for customers expecting frequent code changes and updates. The OTP devices, packaged in plastic packages permit the user to program them once. In addition to the program memory, the configuration fuses must be programmed.

	GMS81604, GMS81608
In-Circuit Emulators	CHOICE-Jr.TM
OTP devices	GMS81608T (40 DIP) GMS81608T K (42 SDIP) GMS81608T PL (44 pin PLCC)
Socket Adapters for OTP Devices	OA816A-40PD (40 DIP) OA816A-42SD (42 SDIP) OA816A-44PL (44 PLCC)
Assembler	LGS Macro Assembler

Device Selection Guide

ROM size	Package	Ordering code
4K bytes	40DIP	GMS81604
	42SDIP	GMS81604 K
	44PLCC	GMS81604 PL
8K bytes	40DIP	GMS81608
	42SDIP	GMS81608 K
	44PLCC	GMS81608 PL
8K bytes (OTP)	40DIP	GMS81608T
	42SDIP	GMS81608T K
	44PLCC	GMS81608T PL

BLOCK DIAGRAM

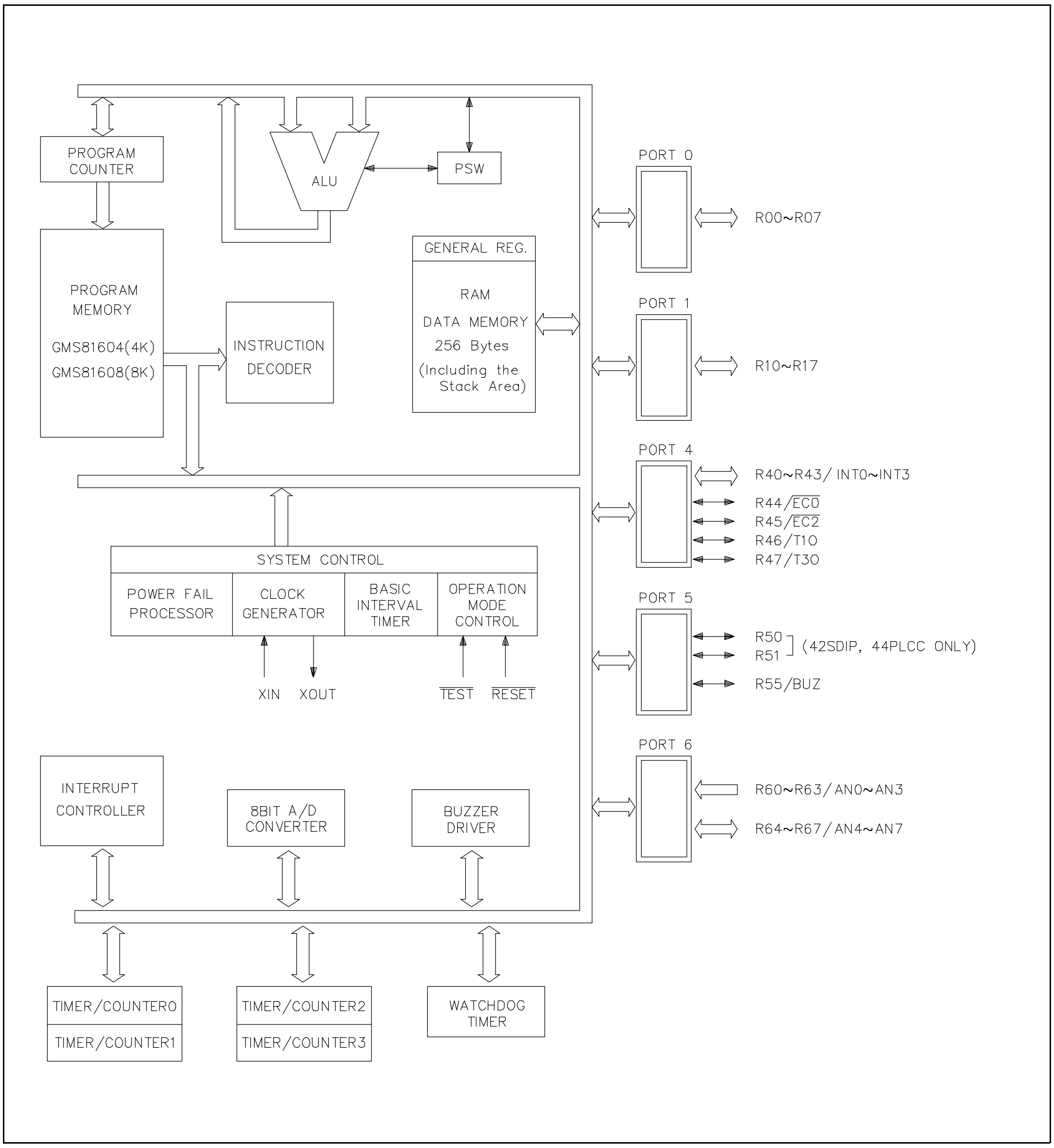
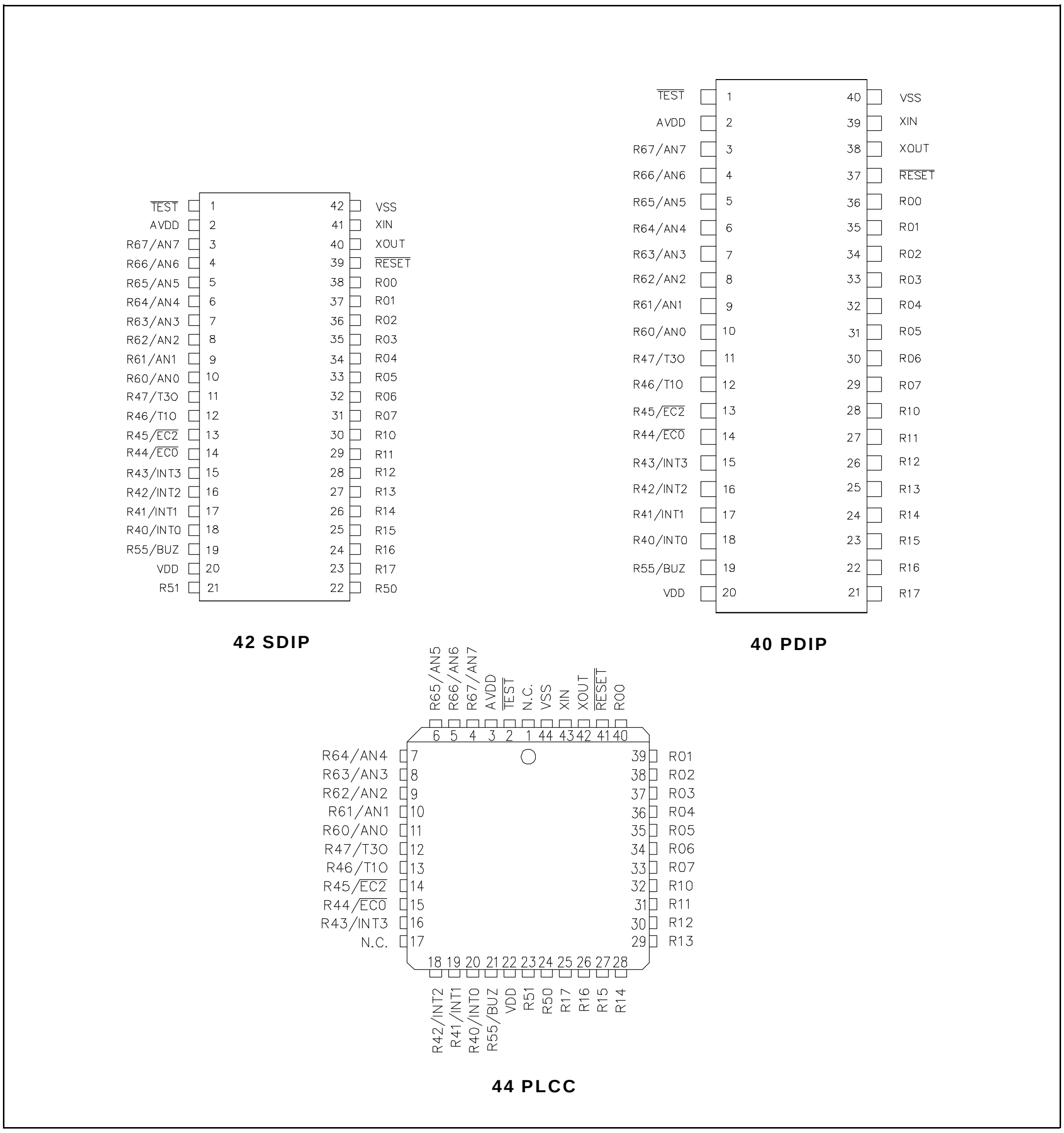


Figure 1. Block Diagram

PIN ASSIGNMENT



42 SDIP

40 PDIP

44 PLCC

Figure 2. Pin Connections

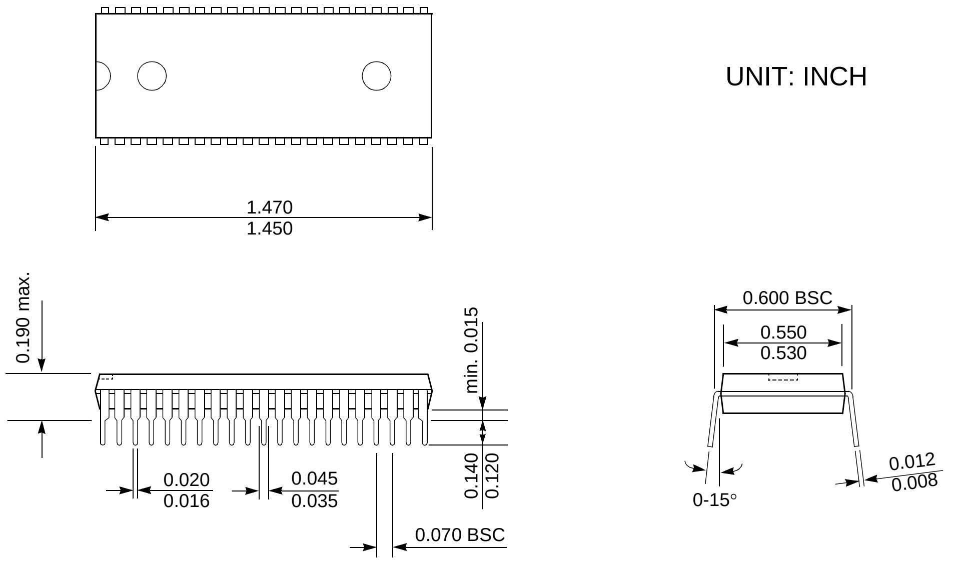
PACKAGES

Part	Package Type
GMS8160X GMS8160X K GMS8160X PL	40DIP 42SDIP 44PLCC

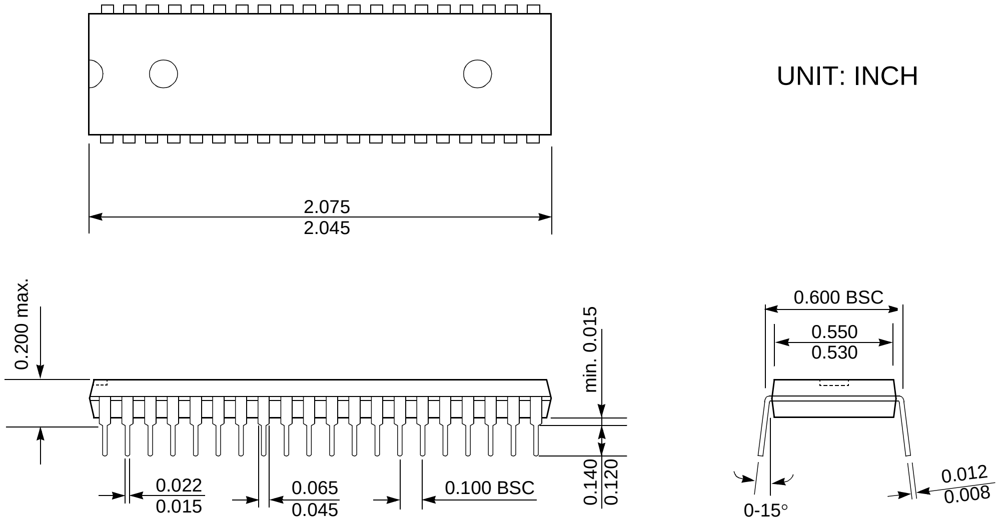
← "X" means 4(4K bytes) or 8(8K bytes).

PACKAGE

42SDIP

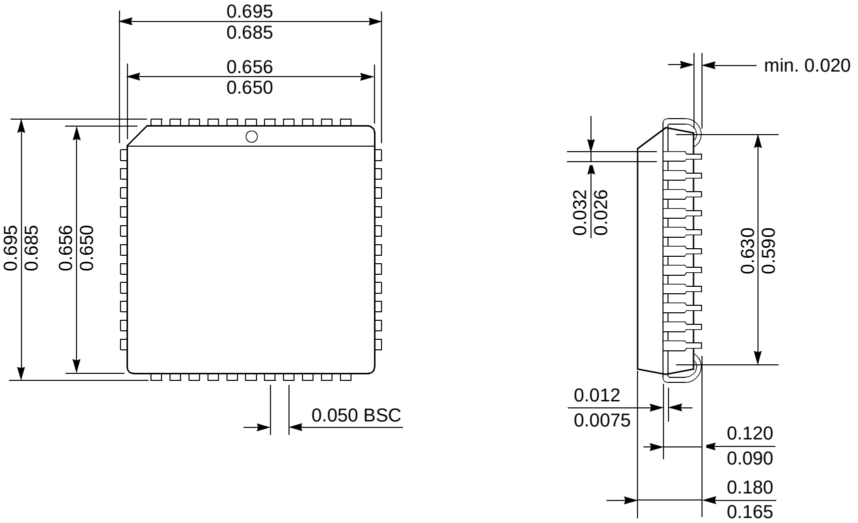


40DIP



44PLCC

UNIT: INCH



PIN DESCRIPTIONS

V_{DD}: Supply voltage.

V_{SS}: Circuit Ground.

$\overline{\text{TEST}}$: For test purposes only. Connect it to V_{DD}.

$\overline{\text{RESET}}$: Reset the MCU.

X_{IN}: Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

X_{OUT}: Output from the inverting oscillator amplifier.

R00~R07: R0 is an 8-bit, CMOS, bidirectional I/O port. As an output port each pin can sink several LS TTL inputs. R0 pins that have 1 or 0 written to their Port Direction Mode Register, can be used as outputs or inputs.

R10~R17: R1 is an 8-bit, CMOS, bidirectional I/O port. As an output port each pin can sink several LS TTL inputs. R1 pins that have 1 or 0 written to their Port Direction Mode Register, can be used as outputs or inputs.

R40~R47: R4 is an 8-bit, CMOS, bidirectional I/O port. As an output port each pin can sink several LS TTL inputs. R4 pins that have 1 or 0 written to their Port Direction Mode Register, can be used as outputs or inputs.

In addition, Port 4 serves the functions of the various following special features.

Port Pin	Alternate Function
R40	INT0 (External Interrupt 0)
R41	INT1 (External Interrupt 1)
R42	INT2 (External Interrupt 2)
R43	INT3 (External Interrupt 3)
R44	$\overline{\text{EC0}}$ (External Count Input to Timer/ Counter 0)
R45	$\overline{\text{EC2}}$ (External Count Input to Timer/ Counter 2)
R46	T1O (Timer 1 Clock-Out)
R47	T3O (Timer 3 Clock-Out)

R50, R51, R55: R5 is a 3-bit, CMOS, bidirectional I/O port. As an output port each pin can sink several LS TTL inputs. R5 pins that have 1 or 0 written to their Port Direction Mode Register, can be used as outputs or inputs. R50 and R51 differs in having internal pull-ups.

Port R55 serves the functions of special features.

Port Pin	Alternate Function
R55	BUZ (Square wave output for Buzzer driving)

R60~R67: R6 is an 8-bit, CMOS, I/O port. R60~R63 can be used as only input, can not be output, R64~R67 are bidirectional I/O port. As an output port each pin can sink several LS TTL inputs. R64~R67 pins that have 1 or 0 written to their Port Direction Mode Register, can be used as outputs or inputs.

R6 serves the functions of following special features.

Port Pin	Alternate Function
R60	AN0 (ADC input 0)
R61	AN1 (ADC input 1)
R62	AN2 (ADC input 2)
R63	AN3 (ADC input 3)
R64	AN4 (ADC input 4)
R65	AN5 (ADC input 5)
R66	AN6 (ADC input 6)
R67	AN7 (ADC input 7)

AV_{DD}: Supply voltage to the ladder resistor of ADC circuit. To enhance the resolution of analog to digital converter, use independent power source as well as possible, other than digital power source.

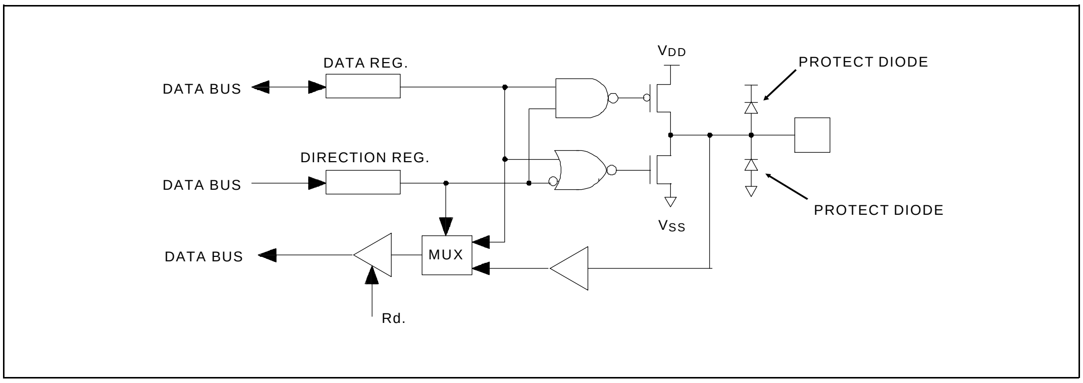
Port Pin	I/O	Descriptions		Pull-up/ Pull-down	RESET	STOP Mode
		Primary Functions	Secondary Functions			
V_{DD}	-	Power supply to MCU	-	-	-	-
V_{SS}	-	Ground	-	-	-	-
AV_{DD}	-	Power supply for ADC	-	-	-	-
$\overline{\text{TEST}}$	I	Test mode	-	-	-	-
$\overline{\text{RESET}}$	I	Reset the MCU	-	Pull-up	Low	Last state
X_{IN}	I	Oscillation input	-	-	Oscillation	Low
X_{OUT}	O	Oscillation output	-	-	Oscillation	High
R00~R07	I/O	General I/O	-	-	Input [3]	Last state
R10~R17	I/O	General I/O	-	-	Input [3]	Last state
R40/INT0	I/O	General I/O	External interrupt 0	-	Input [3]	Last state
R41/INT1	I/O	"	External interrupt 1			
R42/INT2	I/O	"	External interrupt 2			
R43/$\underline{\text{INT3}}$	I/O	"	External interrupt 3			
R44/$\underline{\text{EC0}}$	I/O	"	External count input 0			
R45/EC2	I/O	"	External count input 2			
R46/T1O	I/O	"	Timer 1 output			
R47/T3O	I/O	"	Timer 3 output			
R50[1]	I/O	General I/O	-	Pull-up [2]	Input [3]	Last state
R51[1]	I/O	"	-	Pull-up [2]		
R55/BUZ	I/O	"	Buzzer driving output	-		
R60/AN0	I	General Input	Analog input 0	-	Input [3]	Last state
R61/AN1	I	"	Analog input 1			
R62/AN2	I	"	Analog input 2			
R63/AN3	I	"	Analog input 3			
R64/AN4	I/O	General I/O	Analog input 4			
R65/AN5	I/O	"	Analog input 5			
R66/AN6	I/O	"	Analog input 6			
R67/AN7	I/O	"	Analog input 7			

NOTES:

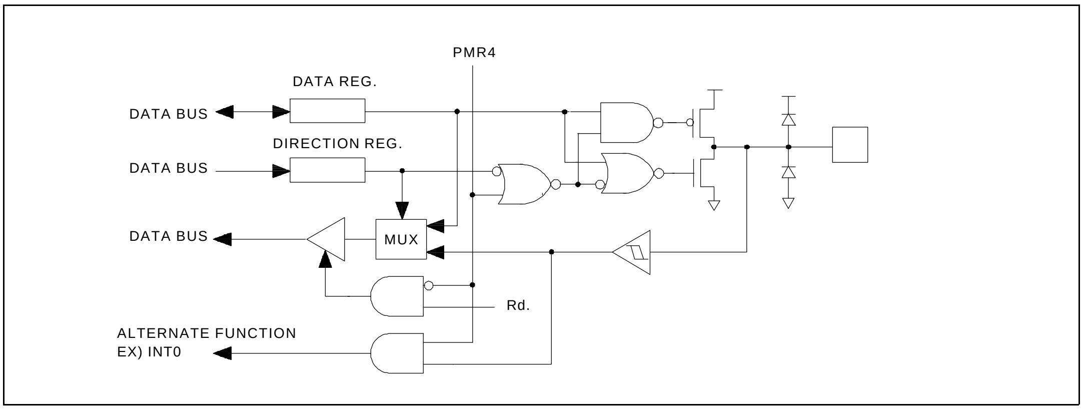
1. R50 and R51 are not physically served on 40 pin package.
2. When input mode is selected, pull-up is activated. In output mode, pull-up is de-activated.
3. In reset status, status of R50,R51 are weak high (Typ. impedance 50~100kΩ). Other pin impedance is very high(High-Z).

PORT STRUCTURES

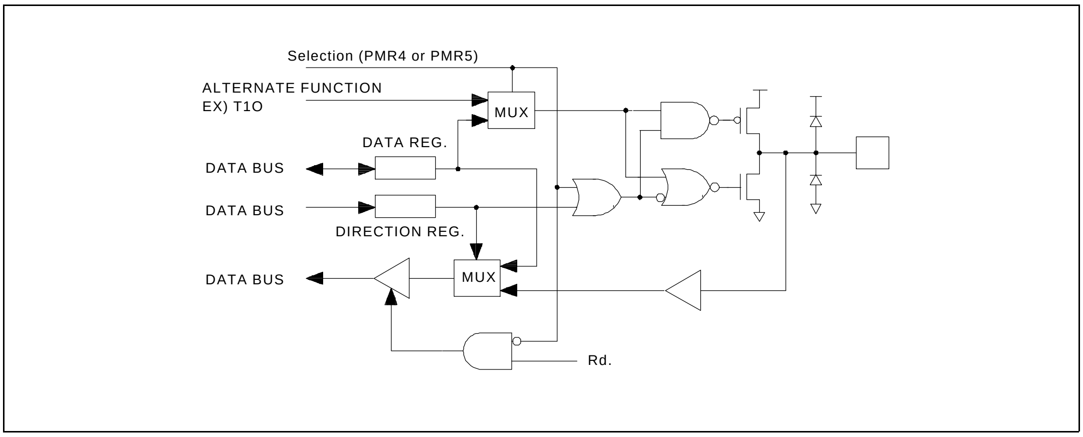
R00~R07, R10~R17



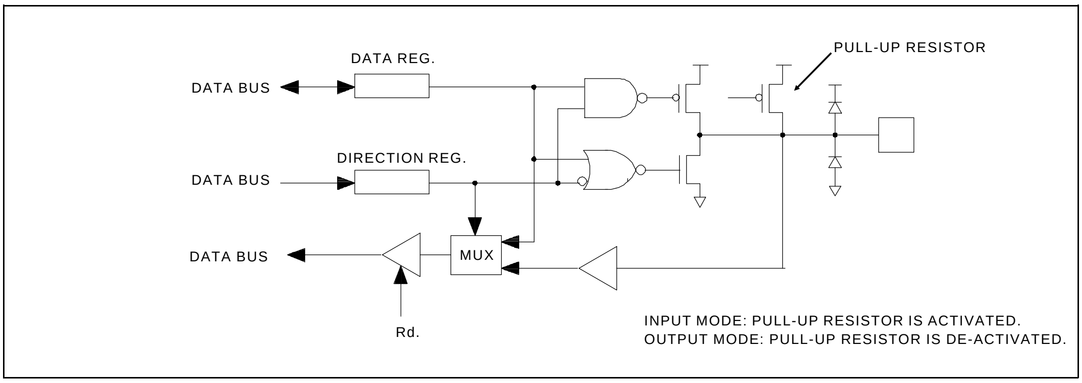
R40/INT0, R41/INT1, R42/INT2, R43/INT3, R44/$\overline{\text{EC0}}$, R45/$\overline{\text{EC2}}$



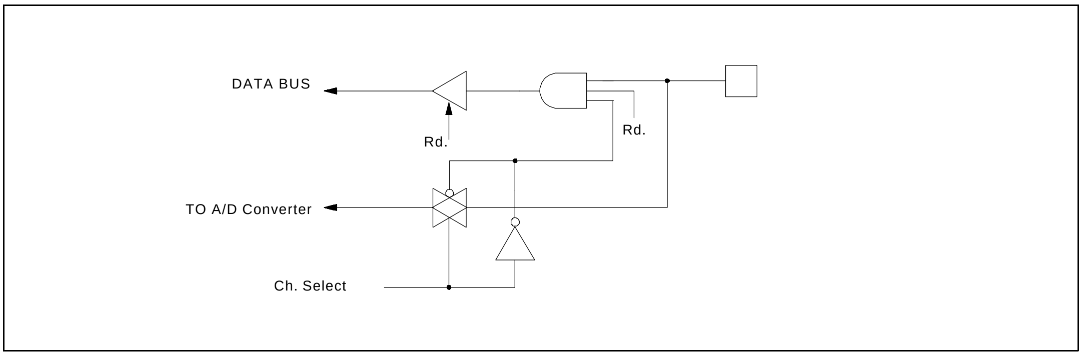
R46/T1O, R47/T3O, R55/BUZ



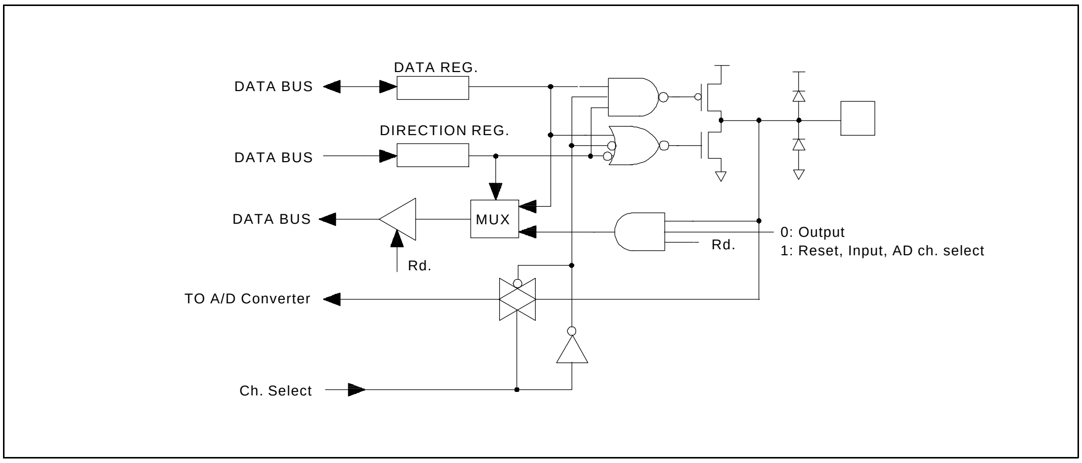
R50, R51


PULL-UP RESISTOR
DATA REG.
DATA BUS
DIRECTION REG.
DATA BUS
DATA BUS
MUX
Rd.
INPUT MODE: PULL-UP RESISTOR IS ACTIVATED.
OUTPUT MODE: PULL-UP RESISTOR IS DE-ACTIVATED.

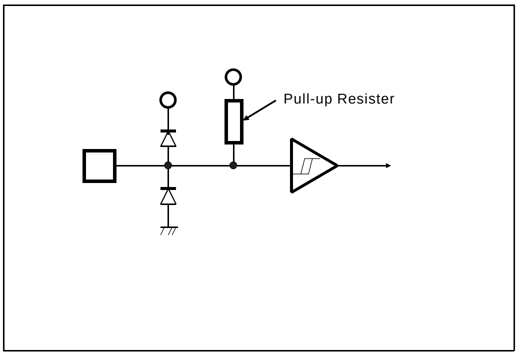
R60/AN0, R61/AN1, R62/AN2, R63/AN3



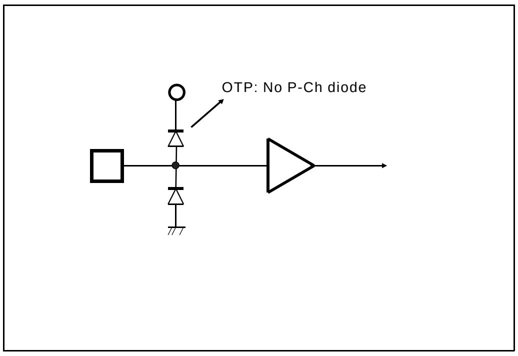
R64/AN4, R65/AN5, R66/AN6, R67/AN7



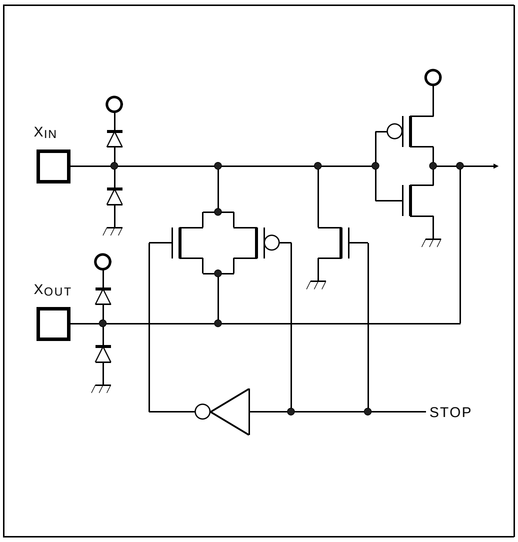
$\overline{\text{RESET}}$



$\overline{\text{TEST}}$



X_{IN}, X_{OUT}



ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings

Supply Voltage	-0.3 to +6.0 V
Storage Temperature	-40 to +125 °C
Voltage on any pin with respect to Ground (V_{SS})	-0.3 to V_{DD}+0.3 V
Maximum current out of V_{SS} pin	150 mA
Maximum current into V_{DD} pin	100 mA
Maximum current sunk by (I_{OL} per I/O Pin)	20 mA
Maximum output current sourced by (I_{OH} per I/O Pin)	8 mA
Maximum current (Σ I_{OL})	120 mA
Maximum current (Σ I_{OH})	50 mA

Notice:
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 these of 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.

Recommended Operating Conditions

Parameter	Symbol	Condition	Specifications		Unit
			Min.	Max.	
Supply Voltage	V_{DD}	f_{XIN} = 8 MHz f_{XIN} = 4 MHz	4.5 2.4	5.5 5.5	V
Operating Frequency	f_{XIN}	V_{DD} = 4.5~5.5V V_{DD} = 2.4~5.5V	1 1	8 4.2	MHz
Operating Temperature	T_{OPR}		-20	85	°C

DC Characteristics (5V)

(V_{DD} = 5.0V± 10%, V_{SS} = 0V, T_A = -20 ~ 85 °C, f_{XIN} = 8 MHz)

Parameter	Pin	Symbol	Test Condition	Specifications			Unit
				Min.	Typ.*	Max.	
Input High Voltage	X_{IN}, $\overline{\text{RESET}}$, R40~R45	V_{IH1}	-	0.8V_{DD}	-	V_{DD}	V
	R0,R1,R46,R47 R5,R6	V_{IH2}	-	0.7V_{DD}	-	V_{DD}	V
Input Low Voltage	X_{IN}, $\overline{\text{RESET}}$, R40~R45	V_{IL1}	-	0	-	0.2V_{DD}	V
	R0,R1,R46,R47 R5,R6	V_{IL2}	-	0	-	0.3V_{DD}	V
Output High Voltage	R0,R1,R4,R5,R6	V_{OH}	V_{DD} = 5V I_{OH} = -2mA	V_{DD} -1.0	V_{DD} -0.4	-	V
Output Low Voltage	R0,R1,R4,R5,R6	V_{OL}	V_{DD} = 5V I_{OL} = 10mA	-	0.6	1.0	V
Power Fail Detect Voltage	V_{DD}	V_{PFD}	V_{DD}=3~4V	3.0	-	4.0	V
Input Leakage Current	$\overline{\text{RESET}}$, R0, R1, R4, R5, R6	I_{IH}	V_I = V_{DD}	-5.0	-	5.0	uA
		I_{IL}	V_I = 0V	-5.0	-	5.0	uA
Input Pull-up Current	$\overline{\text{RESET}}$	I_{P1}	V_{DD} = 5V	-180	-120	-30	uA
	R50, R51	I_{P2}	V_{DD} = 5V	-90	-60	-15	uA
Power Current	Operating mode	I_{DD}	f_{XIN}=4MHz f_{XIN}=8MHz	-	4.5 8	8 15	mA
	STOP mode	I_{STOP}	V_{DD} = 5V	-	2	20	uA
Hysteresis	$\overline{\text{RESET}}$, R40~R45	V_{T+} ~V_{T-}	V_{DD} = 5V	0.5	0.8	-	V

* : Data in "Typ" column is at 5 V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

A/D Converter Characteristics (5V)

(V_{DD} = 5.0V± 10%, V_{AIN} = 5.0V, V_{SS} = 0V, T_A = 25 °C)

Parameter	Symbol	Specifications			Unit
		Min.	Typ.*	Max.	
Analog Input Range	V_{AIN}	V_{SS}	-	V_{AVDD}	V
Non-linearity Error	N_{LE}	-	0.7	±1.5	LSB
Differential Non-linearity Error	N_{DIF}	-	0.1	±0.5	LSB
Zero Offset Error	N_{OFF}	-	1.5	±2.5	LSB
Full Scale Error	N_{FS}	-	1.0	±1.5	LSB
Accuracy	A_{CC}	-	2.0	±3.0	LSB
AV_{DD} Input Current	I_{AVDD}	-	0.5	1.0	mA
Conversion Time	T_{CONV}	-	-	40	uS
Analog power supply Input Range	V_{AVDD}	4.5	5.0	5.5	V

* : Data in "Typ" column is at 5 V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

DC Characteristics (3V)

(V_{DD} = 3.0V± 10%, V_{SS} = 0V, T_A = -20 ~ 85 °C, f_{XIN} = 4 MHz)

Parameter	Pin	Symbol	Test Condition	Specifications			Unit
				Min.	Typ.*	Max.	
Input High Voltage	X_{IN}, $\overline{\text{RESET}}$, R40~R45	V_{IH1}	-	0.8V_{DD}	-	V_{DD}	V
	R0,R1,R46,R47 R5,R6	V_{IH2}	-	0.7V_{DD}	-	V_{DD}	V
Input Low Voltage	X_{IN}, $\overline{\text{RESET}}$, R40~R45	V_{IL1}	-	0	-	0.2V_{DD}	V
	R0,R1,R46,R47 R5,R6	V_{IL2}	-	0	-	0.3V_{DD}	V
Output High Voltage	R0,R1,R4,R5,R6	V_{OH}	V_{DD} = 3V I_{OH} = -1mA	V_{DD} -0.5	V_{DD} -0.3	-	V
Output Low Voltage	R0,R1,R4,R5,R6	V_{OL}	V_{DD} = 3V I_{OL} = 5mA	-	0.5	0.7	V
Power Fail Detect Voltage**	-	-	-	-	-	-	V
Input Leakage Current	$\overline{\text{RESET}}$, R0, R1, R4, R5, R6	I_{IH}	V_I = V_{DD}	-3.0	-	3.0	uA
		I_{IL}	V_I = 0V	-3.0	-	3.0	uA
Input Pull-up Current	$\overline{\text{RESET}}$	I_{P1}	V_{DD} = 3V	-60	-40	-15	uA
	R50, R51	I_{P2}	V_{DD} = 3V	-30	-20	-7.5	uA
Power Current	Operating mode	I_{DD}	f_{XIN}=4MHz	-	2	5	mA
	STOP mode	I_{STOP}	V_{DD} = 3V	-	1	10	uA
Hysteresis	$\overline{\text{RESET}}$, R40~R45	V_{T+} ~V_{T-}	V_{DD} = 3V	0.3	0.6	-	V

* : Data in "Typ" column is at 3 V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**: Power Fail Detection function is not available on 3V operation.

A/D Converter Characteristics (3V)

(V_{DD} = 3.0V± 10%, V_{AIN} = 3.0V, V_{SS} = 0V, T_A = 25 °C)

Parameter	Symbol	Specifications			Unit
		Min.	Typ.*	Max.	
Analog Input Range	V_{AIN}	V_{SS}	-	V_{AVDD}	V
Non-linearity Error	N_{LE}	-	0.2	±1.0	LSB
Differential Non-linearity Error	N_{DIF}	-	0.1	±0.5	LSB
Zero Offset Error	N_{OFF}	-	2.0	±2.5	LSB
Full Scale Error	N_{FS}	-	1.0	±1.5	LSB
Accuracy	A_{CC}	-	2.0	±3.0	LSB
AV_{DD} Input Current	I_{AVDD}	-	0.3	0.5	mA
Conversion Time	T_{CONV}	-	-	40	uS
Analog power supply Input Range	V_{AVDD}	2.7	3.0	3.3	V

* : Data in "Typ" column is at 3 V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

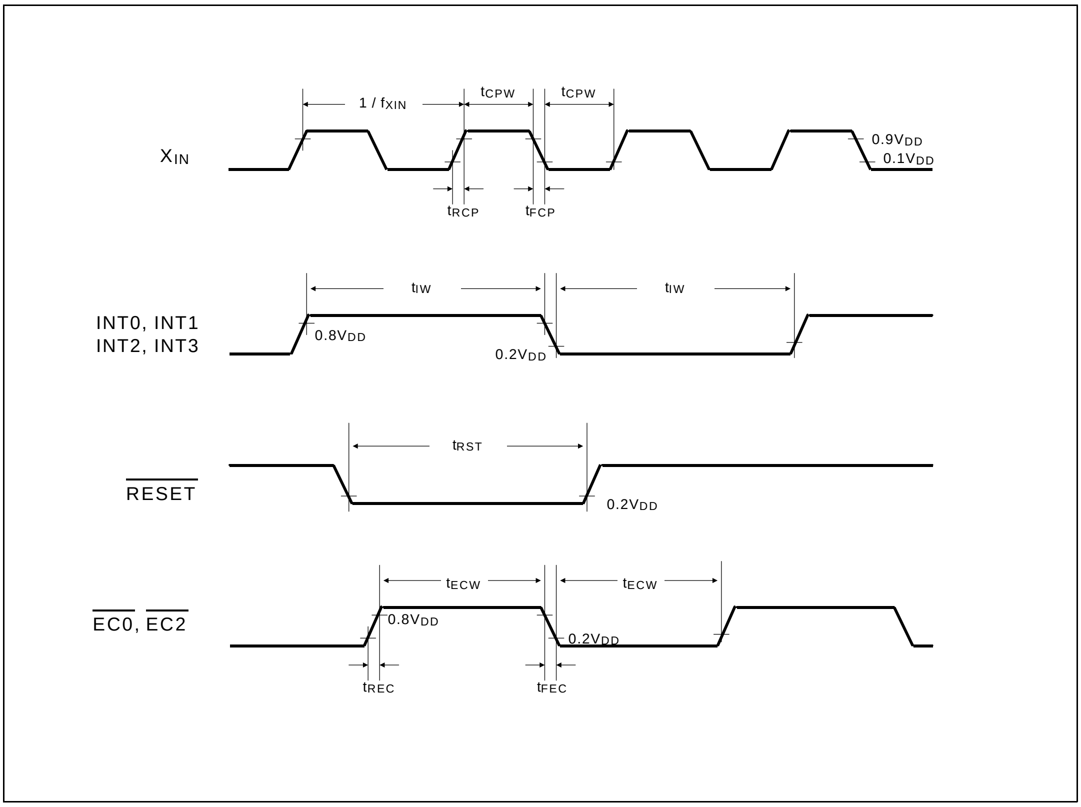
AC Characteristics

(V_{DD} = 2.7~5.5V, V_{SS} = 0V, T_A = -20 ~ 85 °C)

Parameter	Pin	Symbol	Specifications			Unit
			Min.	Typ.	Max.	
Main clock frequency	X_{IN}	f_{XIN}	1	-	8	MHz
Oscillation stabilization Time	X_{IN}, X_{OUT}	t_{ST}	20	-	-	ms
External Clock Pulse Width	X_{IN}	t_{CPW}	80	-	-	ns
External Clock Transition Time	X_{IN}	t_{RCP}, t_{FCP}	-	-	20	ns
Interrupt Pulse Width	INT0, INT1, INT2, INT3	t_{IW}	2	-	-	t_{SYS}*
RESET Input Low Width	$\overline{\text{RESET}}$	t_{RST}	8	-	-	t_{SYS}*
Event Counter Input Pulse Width	$\overline{\text{EC0}}$, $\overline{\text{EC2}}$	t_{ECW}	2	-	-	t_{SYS}*
Event Counter Transition Time	$\overline{\text{EC0}}$, $\overline{\text{EC2}}$	t_{REC}, t_{FEC}	-	-	20	ns

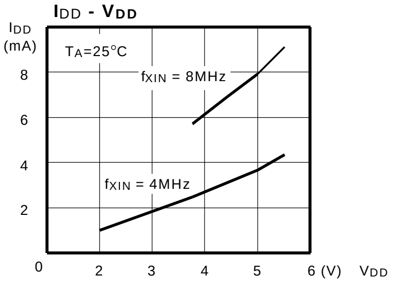
*: t_{SYS} is 2/f_{XIN}.

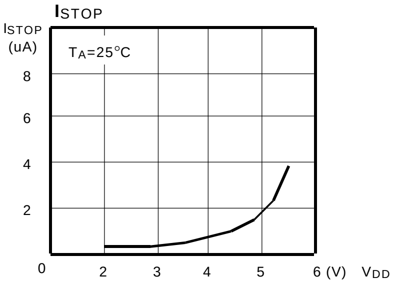
Timing Chart



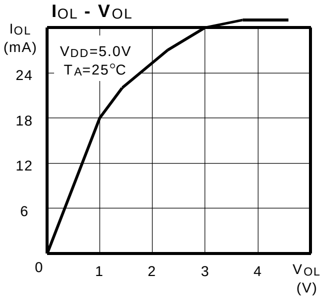
TYPICAL CHARACTERISTICS

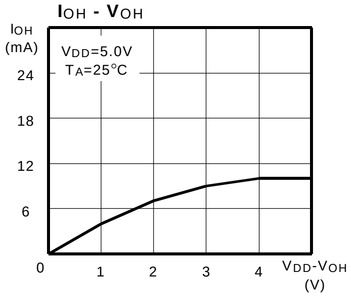
These parameters are for design guidance only and are not tested.

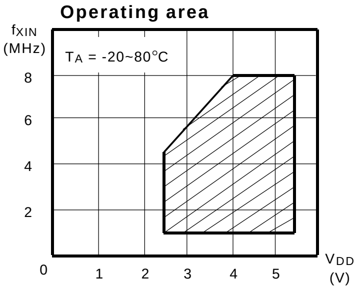




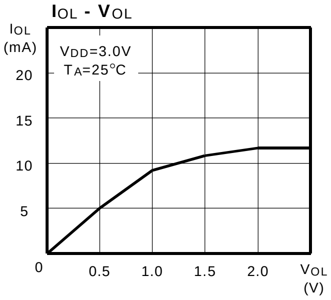
V_{DD}=5V

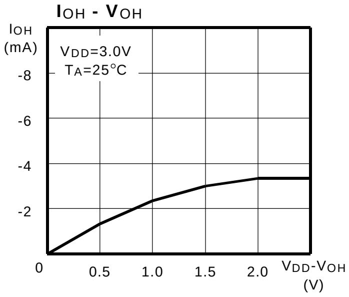






V_{DD}=3.0V


IOL - VOL
IOL
(mA)
VDD=3.0V
TA=25°C
20
15
10
5
0
0.5
1.0
1.5
2.0
VOL
(V)


IOH - VOH
IOH
(mA)
VDD=3.0V
TA=25°C
-8
-6
-4
-2
0
0.5
1.0
1.5
2.0
VDD-VOH
(V)

MEMORY ORGANIZATION

The GMS81604 has separate address spaces for Program and Data Memory. Program memory can only be read, not written to. It can be up to 4K (8K for GMS81608) bytes of Program Memory. Data memory can be read and written to up to 256 bytes including the stack area.

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.

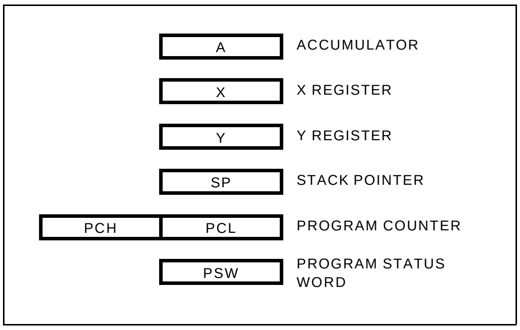


Figure 3. Configuration of Registers

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

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

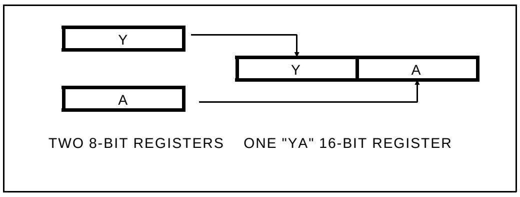


Figure 4. Configuration of YA 16-bit register

X register, Y register: In the addressing modes which use these index registers, the register contents are added to the specified address and this becomes the actual address. These modes are extremely effective for referencing subroutine tables and memory tables. The index registers also have increment, decrement, compare 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. The stack can be located at any position within 100_H to $13F_H$ of the internal data memory. Data store and restore sequence to(from) stack area is shown in Figure 0.

Caution:
The stack pointer must be initialized by software because its value is undefined after reset.

```
Ex)  LDX     #03FH
     TXSP              ; SP ← 3FH
```

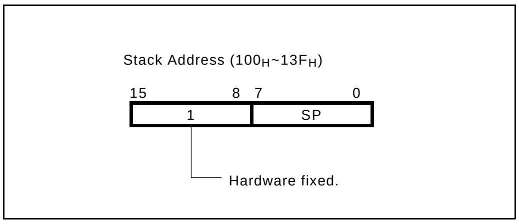


Figure 5. Stack Pointer

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

Program Status Word : The Program Status Word (PSW) contains several status bits that reflect the current state of the CPU. The PSW shown in Figure 6. It contains the Negative flag, the Overflow flag, the Direct page flag, the Break flag, the Half Carry (for BCD operations), the Interrupt enable flag, the Zero flag and the Carry bit.

[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.

[Interrupt disable flag I] This flag enables/disables all interrupts except interrupt caused by Reset or software

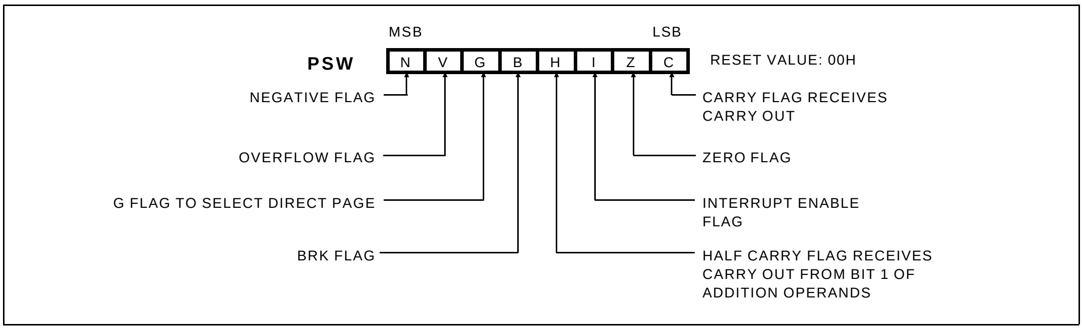

MSB
LSB
PSW
N
V
G
B
H
I
Z
C
RESET VALUE: 00H
NEGATIVE FLAG
CARRY FLAG RECEIVES CARRY OUT
OVERFLOW FLAG
ZERO FLAG
G FLAG TO SELECT DIRECT PAGE
INTERRUPT ENABLE FLAG
BRK FLAG
HALF CARRY FLAG RECEIVES CARRY OUT FROM BIT 1 OF ADDITION OPERANDS

Figure 6. PSW (Program Status Word) Register

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, cleared by the DI instruction.

[Half carry flag H]
After operation, set when there is a carry from bit 3 of ALU or there is not a borrow from bit 4 of ALU. This bit can not be set or cleared except CLRV instruction, clearing with Overflow flag (V).

[Break flag B]
This flag set by software BRK instruction to distinguish BRK from TCALL instruction which as the same vector address.

[Direct page flag G]
This flag assign direct page for direct addressing mode. In the direct addressing mode, addressing area is within zero page 00_H to FF_H when this flag is "0". If it is set to "1", addressing area is 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 in 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, for other than the above, bit 6 of memory is copy 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 copy to this flag.

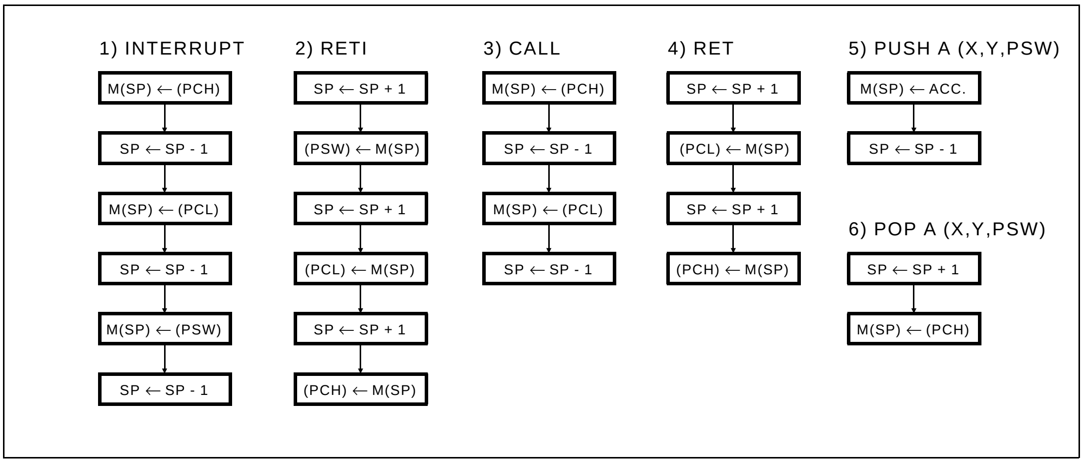


Figure 7. Stack Operation

Program Memory

A 16-bit program counter is capable of addressing up to 64K bytes, but this devices have 4K bytes (8K for GMS81608) program memory space only the physically implemented. Accessing a location above FFFF$_H$ will cause a wrap-around to 0000$_H$.

Figure 8, shows a map of the upper part of the Program Memory. After reset, the CPU begins execution from reset vector which is stored in address FFFE$_H$, FFFF$_H$.

As shown in Figure 8, each area is assigned a fixed location in Program Memory. Program Memory area contains the user program, Page Call (PCALL) area contains subroutine program, to reduce program byte length because of using by 2 bytes PCALL instead of 3 bytes CALL instruction. If it is frequently called, more useful to save program byte length.

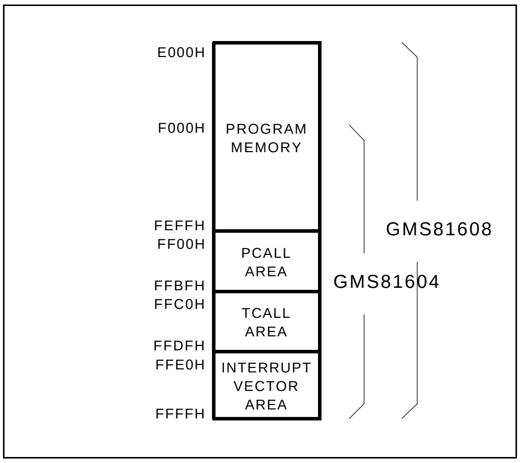


Figure 8. Program Memory

Table Call (TCALL) causes the CPU to jump to each TCALL address, where it commences execution of the service routine. The Table Call service locations are spaced at 2-byte interval : FFC0$_H$ for TCALL15, FFC2$_H$ for TCALL14, etc.

Address	TCALL Name
FFC0H	TCALL15
FFC2H	TCALL14
FFC4H	TCALL13
FFC6H	TCALL12
FFC8H	TCALL11
FFCAH	TCALL10
FFCCH	TCALL9
FFCEH	TCALL8
FFD0H	TCALL7
FFD2H	TCALL6
FFD4H	TCALL5
FFD6H	TCALL4
FFD8H	TCALL3
FFDAH	TCALL2
FFDCH	TCALL1
FFDEH	TCALL0/ BRK1)

1) The BRK software interrupt is using same address with TCALL0.

The interrupt causes the CPU to jump to specific location, where it commences execution of the service routine. The External interrupt 0, for example, is assigned to location FFFA$_H$. The interrupt service locations are spaced at 2-byte interval : FFF8$_H$ for External Interrupt 1, FFFA$_H$ for External Interrupt 0, etc.

Any area from FF00$_H$ to FFFF$_H$, if it not going to be used, its service location is available as general purpose Program Memory.

Address	Vector Name
FFE0H	-
FFE2H	-
FFE4H	-
FFE6H	Basic Interval Timer
FFE8H	Watch Dog Timer
FFEAH	Analog to Digital Converter
FFECH	Timer/ Counter 3
FFEEH	Timer/ Counter 2
FFF0H	Timer/ Counter 1
FFF2H	Timer/ Counter 0
FFF4H	External Interrupt 3
FFF6H	External Interrupt 2
FFF8H	External Interrupt 1
FFFAH	External Interrupt 0
FFFCH	-
FFFEH	RESET

Data Memory

Figure 9 shows the internal Data Memory space available. Data Memory are divided into three groups, a user RAM, control registers and Stack.

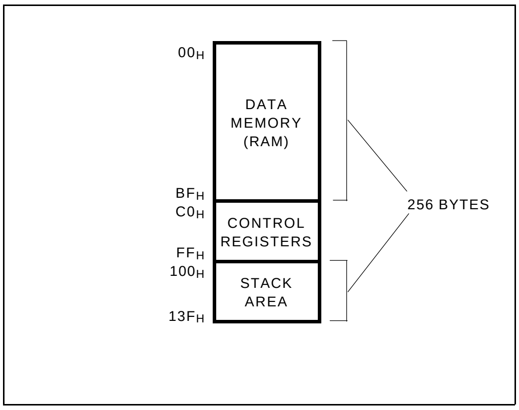


Figure 9. Data Memory

Internal Data Memory addresses are always one byte wide, which implies an address space of 256 bytes including the stack area. To access above $\mathrm{FF_H}$, G-flag should be set to "1" before, because after MCU reset, G-flag is "0".

The stack pointer should be initialized within $\mathrm{00_H}$ to $\mathrm{3F_H}$ by software because of implemented area of internal data memory.

The control registers are used by the CPU and Peripheral functions 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, I/O ports. The control registers are in address $\mathrm{C0_H}$ to $\mathrm{FF_H}$.

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 detail informations of each register are explained in each peripheral sections.

Caution:
Write only registers can not be accessed by bit manipulation instruction.

Address	Symbol	R/W	Power-on Reset Value
$\mathrm{C0_H}$	R0	R/W	X
$\mathrm{C1_H}$	R0DD	W[1]	00000000
$\mathrm{C2_H}$	R1	R/W	X
$\mathrm{C3_H}$	R1DD	W [1]	00000000
$\mathrm{C8_H}$	R4	R/W	X
$\mathrm{C9_H}$	R4DD	W [1]	00000000
$\mathrm{CA_H}$	R5	R/W	X
$\mathrm{CB_H}$	R5DD	W [1]	--0---00
$\mathrm{CC_H}$	R6	R/W	X
$\mathrm{CD_H}$	R6DD	W [1]	00000000
$\mathrm{D0_H}$	PMR4	W [1]	00000000
$\mathrm{D1_H}$	PMR5	W [1]	--0-----
$\mathrm{D3_H}$[2]	BITR	R	00000000
$\mathrm{D3_H}$[2]	CKCTLR	W [1]	--010111
$\mathrm{E0_H}$	WDTR	W [1]	-0111111
$\mathrm{E2_H}$	TM0	R/W	00000000
$\mathrm{E3_H}$	TM2	R/W	00000000
$\mathrm{E4_H}$	☞ Note 3	R/W	X
$\mathrm{E5_H}$	☞ Note 3	R/W	X
$\mathrm{E6_H}$	☞ Note 3	R/W	X
$\mathrm{E7_H}$	☞ Note 3	R/W	X
$\mathrm{E8_H}$	ADCM	R/W [4]	--000001
$\mathrm{E9_H}$	ADR	R	X
$\mathrm{EC_H}$	BUR	W [1]	X
$\mathrm{ED_H}$	PFDR	R/W	-----100
$\mathrm{F4_H}$	IENL	R/W	000-----
$\mathrm{F5_H}$	IRQL	R/W	000-----
$\mathrm{F6_H}$	IENH	R/W	00000000
$\mathrm{F7_H}$	IRQH	R/W	00000000
$\mathrm{F8_H}$	IEDS	W [1]	00000000

Legend - = Unimplemented locations.
X= Undefined value.

NOTES:

1) The all write only registers can not be accessed by bit manipulation instruction.
2) The register BITR and CKCTLR are located at same address. Address D3H is read as BITR, as written to CKCTLR.
3) Several names are given at same address. Refer to below table.

Address	When read		When write
	Timer mode	Capture Mode	
E4H E5H E6H E7H	T0 T1 T2 T3	CDR0 CDR1 CDR2 CDR3	TDR0 TDR1 TDR2 TDR3

4) Only bit 0 of ADCM can be read.

Control Registers for the GMS81604/08

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
C0$_H$	R0	R0 port data register							
C1$_H$	R0DD	R0 port direction register							
C2$_H$	R1	R1 port data register							
C3$_H$	R1DD	R1 port direction register							
C8$_H$	R4	R4 port data register							
C9$_H$	R4DD	R4 port direction register							
CA$_H$	R5	R5 port data register							
CB$_H$	R5DD	R5 port direction register							
CC$_H$	R6	R6 port data register							
CD$_H$	R6DD	R6 port direction register							
D0$_H$	PMR4	T3S	T1S	EC2S	EC0S	INT3S	INT2S	INT1S	INT0S
D1$_H$	PMR5	-	-	BUZS	-	-	-	-	-
D3$_H$[1)]	BITR	Basic Interval Timer data register							
D3$_H$[1)]	CKCTLR	-	-	WDTON	ENPCK	BTCL	BTS2	BTS1	BTS0
E0$_H$	WDTR	-	WDTCL	6-bit Watch Dog Counter register					
E2$_H$	TM0	CAP0	T1ST	T1SL1	T1SL0	T0ST	T0CN	T0SL1	T0SL0
E3$_H$	TM2	CAP2	T3ST	T3SL1	T3SL0	T2ST	T2CN	T2SL1	T2SL0
E4$_H$	T0/ TDR0/ CDR0	Timer 0 register/ Timer data register 0/ Capture data register 0							
E5$_H$	T1/ TDR1/ CDR1	Timer 1 register/ Timer data register 1/ Capture data register 1							
E6$_H$	T2/ TDR2/ CDR2	Timer 2 register/ Timer data register 2/ Capture data register 2							
E7$_H$	T3/ TDR3/ CDR3	Timer 3 register/ Timer data register 3/ Capture data register 3							
E8$_H$	ADCM	-	-	ADEN	ADS2	ADS1	ADS0	ADST	ADSF
E9$_H$	ADR	ADC result data register							
EC$_H$	BUR	BUCK1	BUCK0	BU5	BU4	BU3	BU2	BU1	BU0
ED$_H$[2)]	PFDR	-	-	-	-	-	PFD	PFR	PFS
F4$_H$	IENL	AE	WDTE	BITE	-	-	-	-	-
F5$_H$	IRQL	AIF	WDTIF	BITIF	-	-	-	-	-
F6$_H$	IENH	INT0E	INT1E	INT2E	INT3E	T0E	T1E	T2E	T3E
F7$_H$	IRQH	INT0IF	INT1IF	INT2IF	INT3IF	T0IF	T1IF	T2IF	T3IF
F8$_H$	IEDS	IED3H	IED3L	IED2H	IED2L	IED1H	IED1L	IED0H	IED0L

Legend - = Unimplemented locations.

NOTES:

1) The register BITR and CKCTLR are located at same address. Address D3$_H$ is read as BITR, written to CKCTLR.

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

I/O PORTS

The GMS81604/08 have five ports, R0, R1, R4, R5, R6. These ports pins may be multiplexed with an alternate function for the peripheral features on the device. In general, when a initial reset state, all ports are used as a general purpose input port.

All pins have data direction registers which can configure these pins as output or input.
A "1" in the port direction register configures the corresponding port pin as output. Conversely, write "0" to the corresponding bit to specify as an input pin. For example, to use the even numbered bit of R1 as output ports and the odd numbered bits as input ports, write "55$_H$" to address C1$_H$ (R0 direction register) during initial setting as shown in Figure 10.

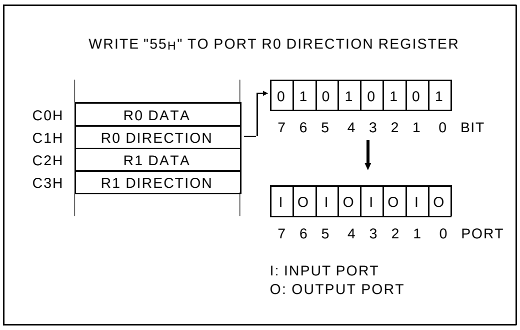
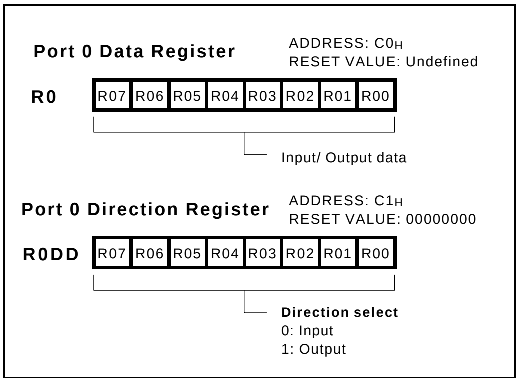


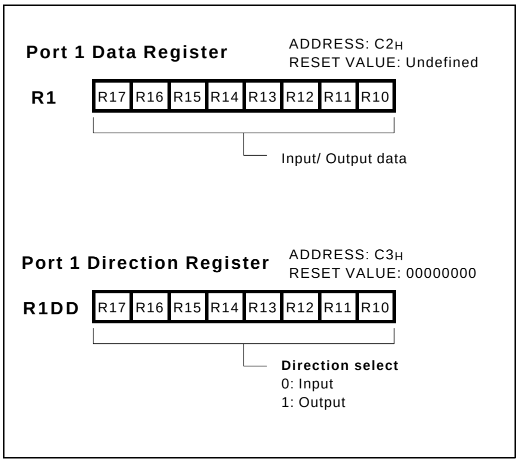
Figure 10. Example port I/O assignment

Reading data register reads the status of the pins whereas writing to it will write to the port latch.

R0 and R0DD registers: R0 is a 8-bit bidirectional I/O port (address C0$_H$). Each pin is individually configurable as input and output through the R0DD register (address C1$_H$).



R1 and R1DD registers: R1 is an 8-bit bidirectional I/O port (address C2$_H$). Each pin is individually configurable as input and output through the R1DD register (address C3$_H$).

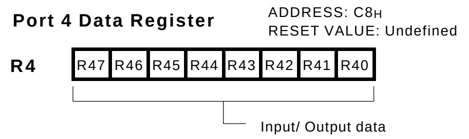


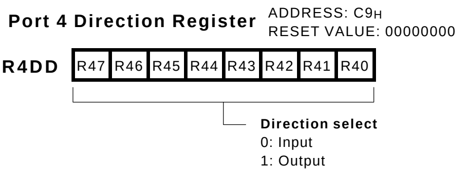
R4 and R4DD registers: R4 is an 8-bit bidirectional I/O port (address C8$_H$). Each pin is individually configurable as input and output through the R4DD register (address C9$_H$).

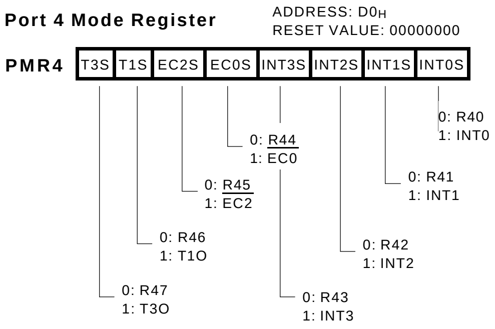
In addition, Port R4 is multiplexed with various special features. The control register PMR4 (address D0$_H$) controls to select alternate function. After reset, this value is "0", port may be used as general I/O ports. To select alternate function such as External interrupt or External counter or Timer clock out, write "1" to the corresponding bit of PMR4.

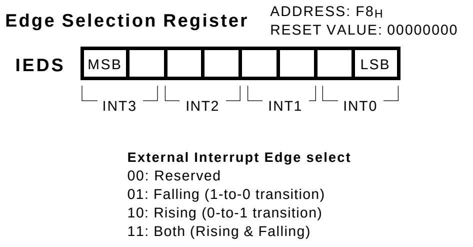
Port Pin	Alternate Function
R40	INT0 (External Interrupt 0)
R41	INT1 (External Interrupt 1)
R42	INT2 (External Interrupt 2)
R43	INT3 (External Interrupt 3)
R44	EC0 (External Count Input to Timer/ Counter 0)
R45	EC2 (External Count Input to Timer/ Counter 2)
R46	T1O (Timer 1 Clock-Out)
R47	T3O (Timer 3 Clock-Out)

Regardless of the direction register R4DD, PMR4 is selected to use as alternate functions, port pin can be used as a corresponding alternate features.








Edge Selection Register
ADDRESS: F8H
RESET VALUE: 00000000
IEDS
MSB
LSB
INT3
INT2
INT1
INT0
External Interrupt Edge select
00: Reserved
01: Falling (1-to-0 transition)
10: Rising (0-to-1 transition)
11: Both (Rising & Falling)

R5 and R5DD registers: R5 is a 3-bit bidirectional I/O port (address CA_H). R50, R51 and R55 only are physically implemented on this device.

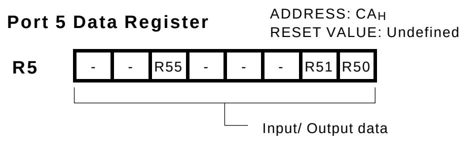
R50, R51 have internal pullups which is activated on input but deactivated on output. As input, these pins that are externally pull low will source current (I_{P2} on the DC characteristics) because of the internal pullups.

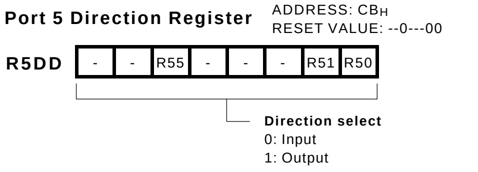
> *Caution:*
> *Pins R50, R51 are present on 42SDIP, 44PLCC package only, but not on 40DIP . Refer to Pin assignment.*

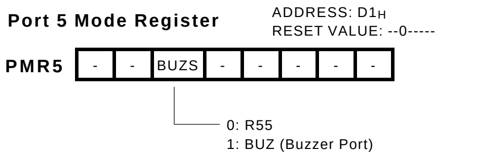
Each pin is individually configurable as input and output through the R5DD register (address CB_H).

Port Pin	Alternate Function
R55	BUZ (Square-wave output for Buzzer driving)

The control register PMR5 (address $D1_H$) controls the selection alternate function. After reset, this value is "0", port may be used as general I/O ports. To use buzzer function, write "1" to the PMR5.





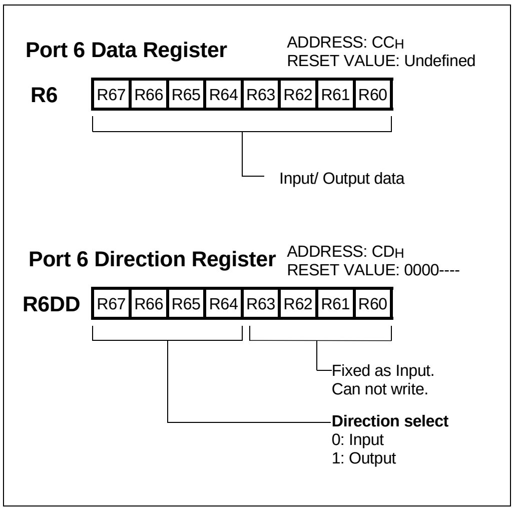


R6 and R6DD registers: R6 is an 8-bit port (address CC$_H$). Pins R64~R67 are individually configurable as input and output through the R6DD register (address CD$_H$), but pins R60~R63 are input only.

Port Pin	Alternate Function
R60	AN0 (ADC input 0)
R61	AN1 (ADC input 1)
R62	AN2 (ADC input 2)
R63	AN3 (ADC input 3)
R64	AN4 (ADC input 4)
R65	AN5 (ADC input 5)
R66	AN6 (ADC input 6)
R67	AN7 (ADC input 7)

R6DD (address CD$_H$) controls the direction of the R6 pins, even when they are being used as analog inputs. The user must make sure to keep the pins configured as inputs when using them as analog inputs.

On the initial RESET, R60 can not be used digital input port, because this port is selected as an analog input port by ADCM register. To use this port as a digital I/O port, change the value of lower 4 bits of ADCM (address 0E8$_H$).
On the other hand, R6 port, all eight pins can not be used as digital I/O port simultaneousely. At least one pin is used as an analog input.



BASIC INTERVAL TIMER

The GMS81604 has one 8-bit Basic Interval Timer that is free-run, can not stop. Block diagram is shown in Figure 11.

The 8-bit Basic interval timer register (BITR) is incremented every internal count pulse which is divided by prescaler. Since prescaler has divided ratio by 16 to 2048, the count rate is 1/16 to 1/2048 of the oscillator frequency. As the count overflows from FF_H to 00_H, this overflow causes to generate the Basic interval timer interrupt. The BITR is interrupt request flag of Basic interval timer.

> ***Caution:***
> *All control bits of Basic interval timer are in CKCTLR register which is located at same address of BITR (address $D3_H$). Address $D3_H$ is read as BITR, written to CKCTLR.*

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

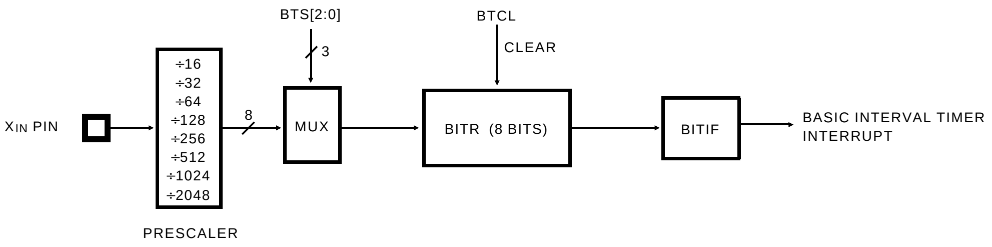


Figure 11. Block Diagram of The Basic Interval Timer

CKCTLR

-	-	WDTON	ENPCK	BTCL	BTS2	BTS1	BTS0

ADDRESS: $D3_H$
RESET VALUE: --010111

Symbol	Position	Name and Significance

WDTON	CKCTLR.5	WDTON=1, enables Watch Dog Timer operation,
		WDTON=0, operates as a 6-bit timer

ENPCK	CKCTLR.4	Enable Peripheral clock.
BTCL	CKCTLR.3	BTCL is set to "1", BITR is cleared. BTCL becomes "0" automatically after one machine cycle, and starts counting.

BASIC INTERVAL TIMER CLOCK SELECTION

BTS2	BTS1	BTS0	Prescale value
0	0	0	16
0	0	1	32
0	1	0	64
0	1	1	128
1	0	0	256
1	0	1	512
1	1	0	1024
1	1	1	2048

Figure 12. CKCTLR: Control Clock Register

TIMER/COUNTER

The GMS81604 has four 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 are can be used either the two 8-bit Timer/Counter or one 16-bit Timer/Counter to combine them. Also Timer 2 and Timer 3 are same.

In the "timer" function, the register is incremented every internal clock input. Thus, one can think of it as counting internal clock input. Since a least clock consists of 4 and most clock consists of 64 oscillator periods, the count rate is 1/4 to 1/64 of the oscillator frequency.

In the "counter" function, the register is incremented in response to a 1-to-0 (falling edge) transition at its corresponding external input pin, EC0 or EC2.

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

It has four operating modes: "8-bit timer/counter", "16-bit timer/counter", "8-bit capture", "16-bit capture" which are selected by bit in Timer mode register TM0 and TM2 as shown in right Table.

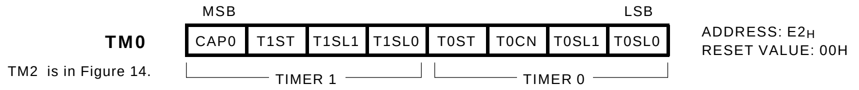
In operation of Timer 2, Timer 3, their operations are same with Timer 0, Timer 1, respectively.

TM0 FOR TIMER 0, TIMER 1

CAP0	T1SL1	T1SL0	Timer 0	Timer 1
0	0	0	16-bit Timer/Counter	
1	0	0	16-bit Capture	
0	X	X	8-bit Timer	8-bit Timer
1	X	X	8-bit Capture	8-bit Timer

TM2 FOR TIMER 2, TIMER 3

CAP2	T3SL1	T3SL0	Timer 2	Timer 3
0	0	0	16-bit Timer/Counter	
1	0	0	16-bit Capture	
0	X	X	8-bit Timer	8-bit Timer
1	X	X	8-bit Capture	8-bit Timer



CAP0 Capture mode selection flag, When set, timer operate as one 16-bit capture timer combine two 8-bit timers.

T1ST When set, Timer 1 count register is cleared and start again.
When cleared, stop the counting.

T0ST When set, The Timer 0 Count Register is cleared and start again.
When cleared, stop the counting.

T0CN Start/Stop control for Timer 0. A logic 1 starts the timer.

TIMER 1

T1SL1	T1SL0	INPUT CLOCK
0	0	16-BIT TIMER MODE (NOTE 1)
0	1	8-BIT TIMER, ÷ 4 ← PRESCALER
1	0	8-BIT TIMER, ÷ 16
1	1	8-BIT TIMER, ÷ 64

TIMER 0

T0SL1	T0SL0	INPUT CLOCK
0	0	Timer or Counter select
0	1	÷ 4 ← PRESCALER
1	0	÷ 16
1	1	÷ 64

NOTE:
If this mode selected, the Timer 0 are used as a 16-bit timer mode. The Timer 1 is engaged to the Timer 0.
The source clock is selected by bits T0SL1 and T0SL0.

Figure 13. TM0: Timer 0, Timer 1 Mode Register

TM2

MSB							LSB
CAP2	T3ST	T3SL1	T3SL0	T2ST	T2CN	T2SL1	T2SL0
TIMER 3				TIMER 2			

ADDRESS: E3$_H$
RESET VALUE: 00$_H$

CAP2 Capture mode selection flag, When set, timer operate as one 16-bit timer combine two 8-bit timers. See Figure 21 and Figure 22.

T3ST When set, Timer 3 count register is cleared and start again.
When cleared, stop the counting.

T2ST When set, Timer 2 count register is cleared and start again.
When cleared, stop the counting.

T2CN Start/Stop control for Timer 2. A logic 1 starts the timer.

TIMER 3

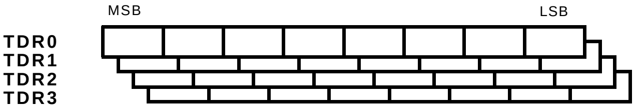
T3SL1	T3SL0	INPUT CLOCK
0	0	16-BIT TIMER MODE (NOTE 1)
0	1	8-BIT TIMER, ÷4 ←PRESCALER
1	0	8-BIT TIMER, ÷16
1	1	8-BIT TIMER, ÷64

TIMER 2

T2SL1	T2SL0	INPUT CLOCK
0	0	Timer or Counter select
0	1	÷4 ← PRESCALER
1	0	÷16
1	1	÷64

NOTE:
If this mode selected, the Timer 2 and Timer 3 are used as a 16-bit timer mode. The Timer 3 is engaged to the Timer 2. The source clock is selected by bits T2SL1 and T2SL0.

Figure 14. TM2: Timer 2, Timer 3 Mode Register



ADDRESS: E4$_H$
RESET VALUE: 00$_H$

ADDRESS: E5$_H$
RESET VALUE: 00$_H$

ADDRESS: E6$_H$
RESET VALUE: 00$_H$

ADDRESS: E7$_H$
RESET VALUE: 00$_H$

Figure 15. TDRx : Timer x Data Register

8-bit Timer/Counter Mode

The GMS81604 has four 8-bit Timer/Counters, Timer 0, Timer 1, Timer 2, Timer 3. The Timer 0, Timer 1 only as shown in Figure 16. because other timer/counters are same with Timer 0 and Timer 1.

The "timer" or "counter" function is selected by control registers TM0, TM2 as shown in Figure 13 and Figure 14. To use as an 8-bit timer/counter mode, bit CAP0 of TM0 is cleared to "0" and bits T1SL1, T1SL0 of TM0 or bits T3SL1, T3SL0 of TM2 should not set to zero (Figure 16).

These timers have each 8-bit count register and data register. The count register is incremented by every internal or external clock input. The internal clock has a prescaler divide ratio option of 4, 16, 64 (selected by control bits TxSL1, TxSL0 of register TMx).

In the Timer 0, timer register T0 increments from 00_H until it matches TDR0 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.

Caution:
The contents of Timer data register TDRx should be initialized 1_H~FF_H except 0_H, because it is undefined after reset.

In counter function, the counter is incremented every 1-to 0 (falling edge) transition of $\overline{\text{EC0}}$ or $\overline{\text{EC2}}$ pin. In order to use counter function, the bit EC0S, EC2S of the Port mode register PMR4 are set to "1". The Timer 0 can be used as a counter by pin $\overline{\text{EC0}}$ input, but Timer 1 can not. Similarly, Timer 2 can be used by pin $\overline{\text{EC2}}$ input but Timer 3 can not.

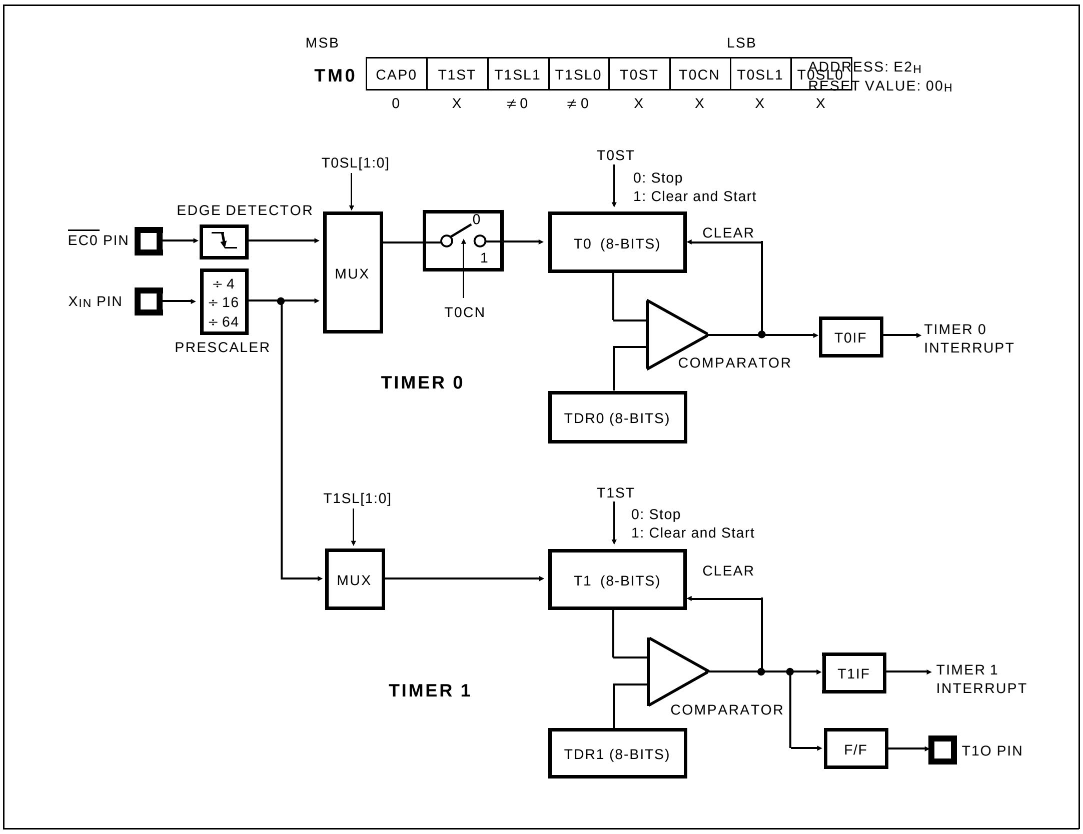


Figure 16. 8-bit Timer/Counter Mode

To pulse out, the timer match can goes to port pin as shown in Figure 16. Thus, pulse out is generated by the timer match. These operation is implemented to pin, T1O and T3O. The pin T1O is output from Timer 1, the T3O is from Timer 3. Operation of T3O is omitted in this document, but still presents and same architecture with T1O.

$$f_{TxO}\,(Hz) = \frac{OscillatorFrequency}{2 \cdot Prescaler \cdot TDR}$$

	MSB							LSB	
PMR4	T3S	T1S	EC2S	EC0S	INT3S	INT2S	INT1S	INT0S	ADDRESS: D0$_H$ RESET VALUE: 00$_H$

T3S	0: R47 1: T3O (TIMER 3 OUTPUT)	INT3S	0: R43 1: INT3 (EXTERNAL INTERRUPT 3)
T1S	0: R46 1: T1O (TIMER 1 OUTPUT)	INT2S	0: R42 1: INT2 (EXTERNAL INTERRUPT 2)
EC2S	0: R45 1: $\overline{\text{EC2}}$ **(EXTERNAL INPUT PIN FOR TIMER 2**	INT1S	0: R41 1: INT1 (EXTERNAL INTERRUPT 1)
EC0S	0: R44 1: $\overline{\text{EC0}}$ **(EXTERNAL INPUT PIN FOR TIMER 0**	INT0S	0: R40 1: INT0 (EXTERNAL INTERRUPT 0)

Figure 17. PMR4: R4 Port Mode Register

EX) When TM0: 00110111 (PRESCALER= 16)
TDR0: F9$_H$ = 249$_D$
OSCILLATOR FREQ.= 4MHz

$$\text{INTERRUPT PERIOD} = \frac{1}{4 \times 10^6\,Hz} \times 16 \times (249 + 1) = 1\text{ms}$$

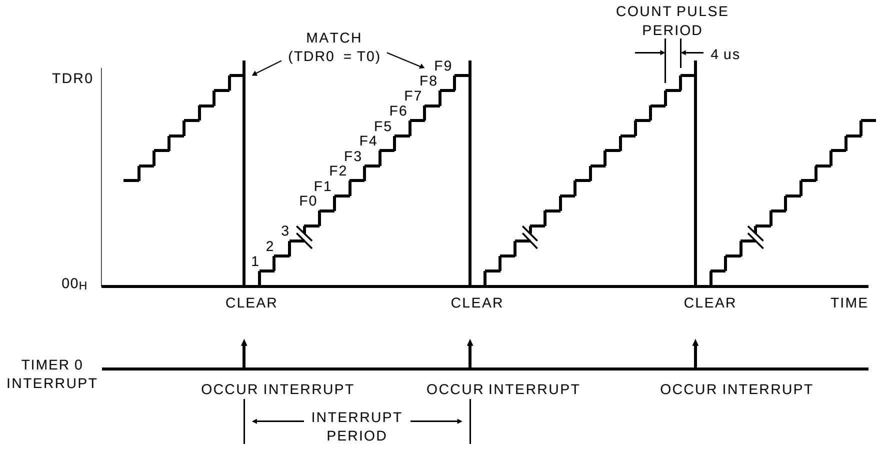


Figure 18. Timer Count Example

16-bit Timer/Counter Mode

The Timer register is being run with all 16 bits. A 16-bit timer/counter register T0, T1 are incremented from 0000_H until it matches TDR0, TDR1 and then resets to 0000_H. The match output generates Timer 0 interrupt.

The clock source of the Timer 0 is selected either internal or external clock by bit T0SL1, T0SL0.

Even if the Timer 0 (including the Timer 1) is used as a 16-bit timer, the Timer 2 and Timer 3 can still be used as either two 8-bit timer or one 16-bit timer by setting the TM2. Reversely, even if the Timer 2 (including the Timer 3) is used as a 16-bit timer, the Timer 0 and Timer 1 can still be used as 8-bit timer independently.

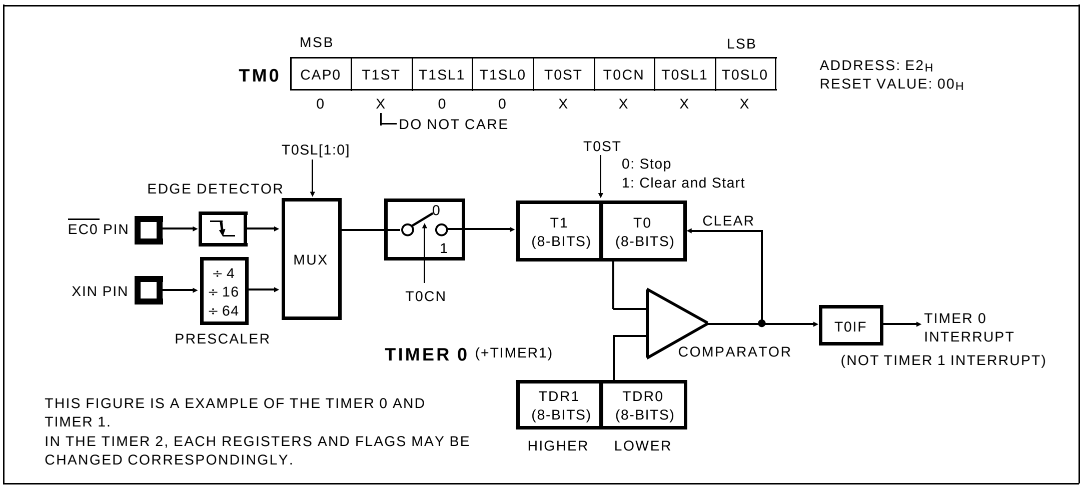


Figure 19. 16-bit Timer/Counter Mode

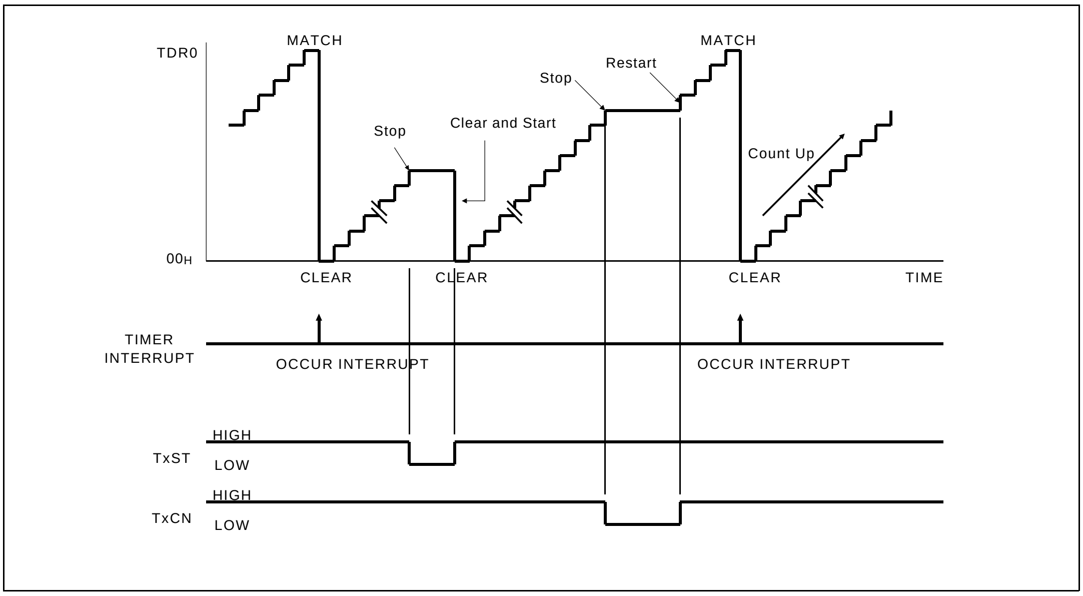


Figure 20. Timer Count Operation

8-bit Capture Mode

The Timer 0 capture mode is set by bit CAP0 of timer mode register TM0 (bit CAP2 of timer mode register TM2 for Timer 2) as shown in Figure 21. In this mode, Timer 1 still operates as an 8-bit timer/counter.

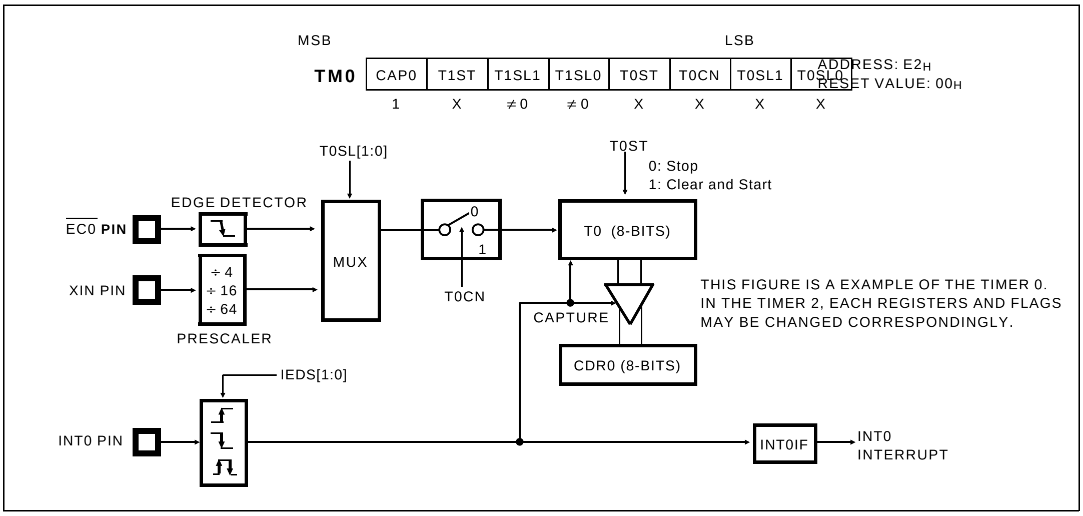
As mentioned above, not only Timer 0 but Timer 2 can also be used as a capture mode.

In 8-bit capture mode, Timer 1 and Timer 3 are can not be used as a capture mode.

The Timer/Counter register is incremented in response internal or external input. This counting function is same with normal timer mode, but Timer interrupt is not generated. 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,T2), to be captured into registers CDRx (CDR0, CDR2), respectively. After captured, Timer x register is cleared and restarts by hardware.

> ***Caution:***
> *The CDRx and TDRx are in same address.*
> *In the capture mode, reading operation is read the CDRx, not TDRx because path is opened to the CDRx.*

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.



TM0	CAP0	T1ST	T1SL1	T1SL0	T0ST	T0CN	T0SL1	T0SL0
	1	X	≠ 0	≠ 0	X	X	X	X

Figure 21. 8-bit Capture Mode

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.

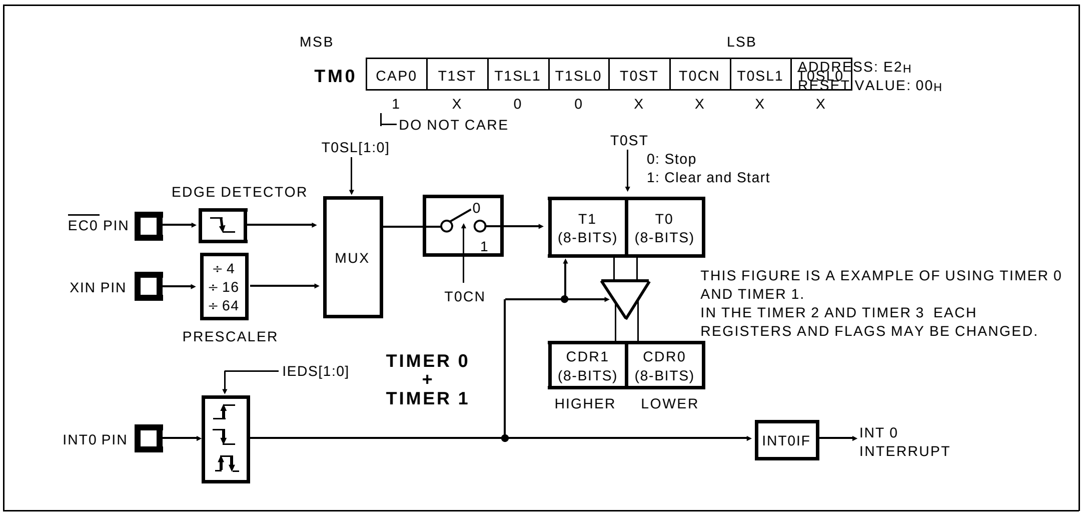


Figure 22. 16-bit Capture Mode

ANALOG TO 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 eight 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 ADR. The register ADCM, shown in Figure 24, controls the operation of the A/D converter module. The port pins can be configured as analog inputs or digital I/O. To use analog inputs, I/O is selected input mode by R6DD direction register.

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 ADR contains the results of the A/D conversion. When the conversion is completed, the result is loaded into the ADR, the A/D conversion status bit ADSF is set to "1", and the A/D interrupt flag AIF is set. The block diagram of the A/D module is shown in Figure 23. 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 40 uS (at f_{XIN}=4 MHz).

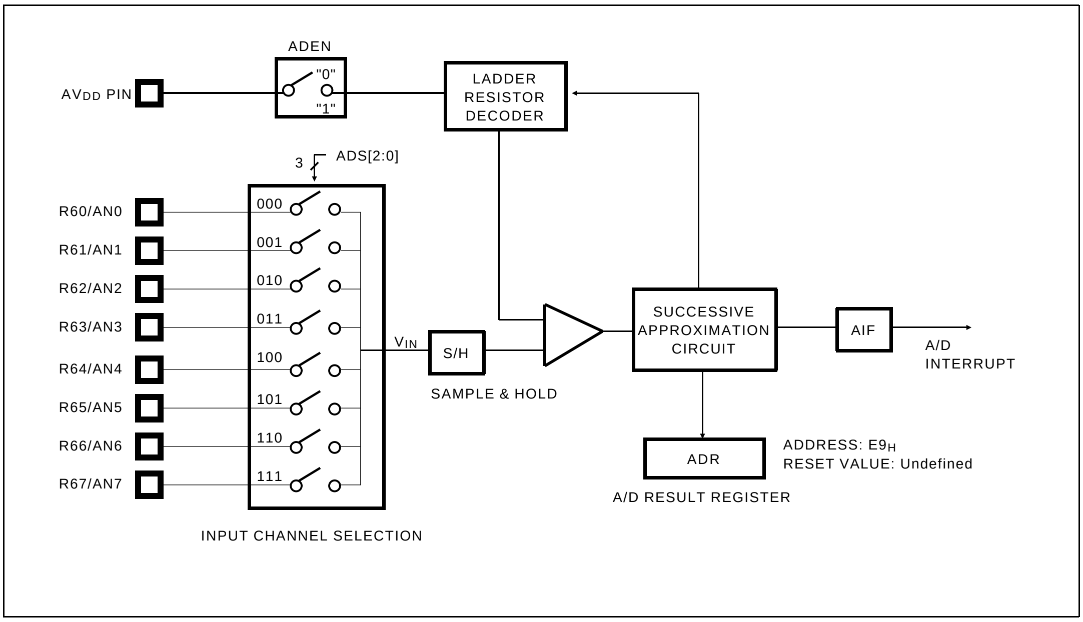
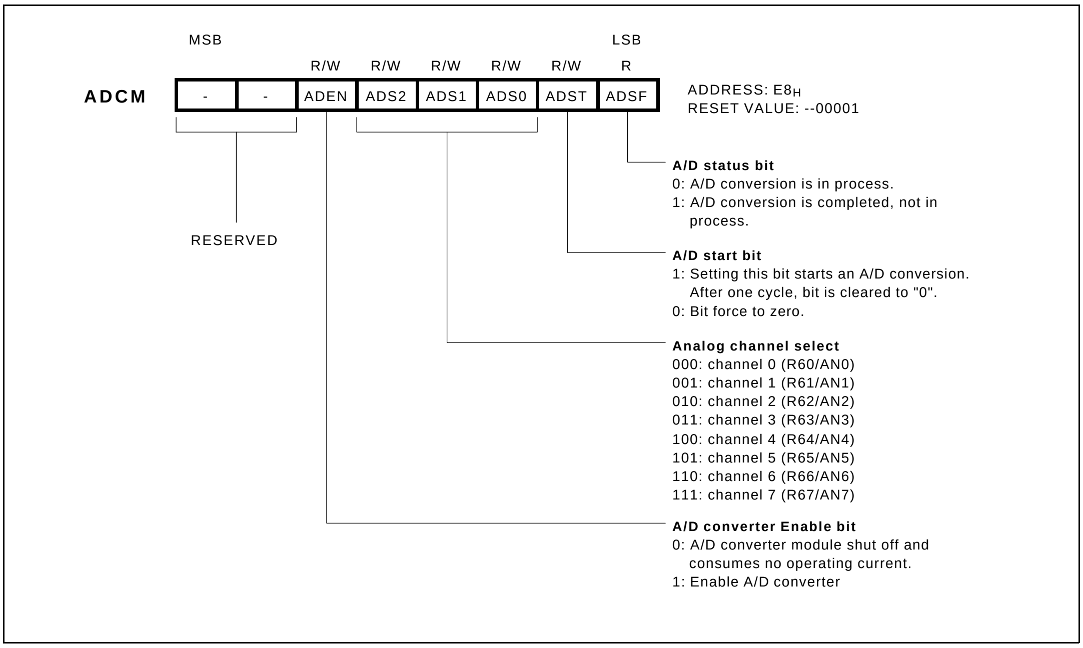


Figure 23. A/D Block Diagram



ADDRESS: $E8_H$
RESET VALUE: --00001

A/D status bit
0: A/D conversion is in process.
1: A/D conversion is completed, not in process.

A/D start bit
1: Setting this bit starts an A/D conversion. After one cycle, bit is cleared to "0".
0: Bit force to zero.

Analog channel select
000: channel 0 (R60/AN0)
001: channel 1 (R61/AN1)
010: channel 2 (R62/AN2)
011: channel 3 (R63/AN3)
100: channel 4 (R64/AN4)
101: channel 5 (R65/AN5)
110: channel 6 (R66/AN6)
111: channel 7 (R67/AN7)

A/D converter Enable bit
0: A/D converter module shut off and consumes no operating current.
1: Enable A/D converter

Figure 24. ADCM: A/D Converter Control Register

BUZZER FUNCTION

The buzzer driver consists of 6-bit binary counter, the buzzer register BUR and the clock selector. It generates square-wave which is very wide range frequency (250 Hz~125 kHz at f_{XIN}=4 MHz) by user programmable counter.

Pin R55 is assigned for output port of Buzzer driver by setting the bit 5 of PMR5 (address $D1_H$) to "1". At this time, the pin R55 must be defined as output mode (the bit 5 of R5DD=1)

The bit 0 to 5 of BUR determines output frequency for buzzer driving.
Frequency calculation is following below.

$$f_{BUZ}\ (Hz) = \frac{f_{XIN}}{2 \cdot Prescaler\ ratio \cdot BUR value}$$

f_{BUZ}: Buzzer frequency
f_{XIN}: Min oscillator frequency
Prescaler: Prescaler divide ratio by BUCK1, BUCK0
BUR:Lower 6-bit of BUR. Buzzer period data value

The bits BUCK1, BUCK0 of BUR selects the source clock from prescaler output.

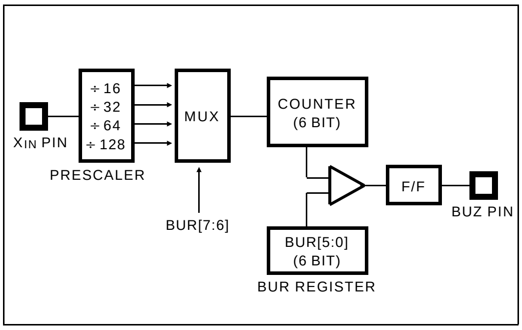
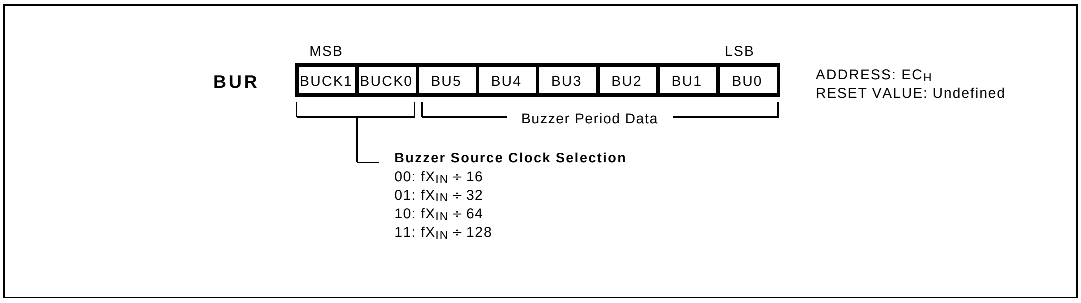


Figure 25. Buzzer Driver

The 6-bit buzzer counter is cleared and start the counting by writing signal to the register BUR. It is increment from 00_H until it matches 6-bit register BUR.

> ***Caution:***
> *The register BUR contains undefined value after reset. It must be initialized none 0_H(1_H~$3F_H$).*



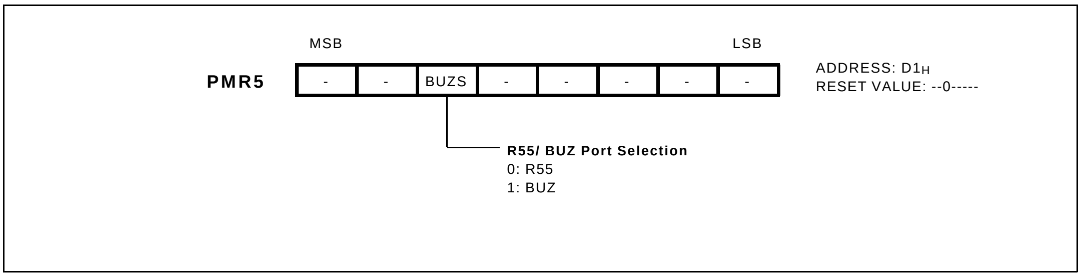
BUR	MSB						LSB
BUCK1	BUCK0	BU5	BU4	BU3	BU2	BU1	BU0

ADDRESS: EC_H
RESET VALUE: Undefined

BUCK1–BU5…BU0: Buzzer Period Data

Buzzer Source Clock Selection
00: fX_{IN} ÷ 16
01: fX_{IN} ÷ 32
10: fX_{IN} ÷ 64
11: fX_{IN} ÷ 128

Figure 26. BUR: Buzzer Period Data Register



PMR5	MSB						LSB
-	-	BUZS	-	-	-	-	-

ADDRESS: $D1_H$
RESET VALUE: --0-----

R55/ BUZ Port Selection
0: R55
1: BUZ

Figure 27. PMR5: Port 5 Mode Register

INTERRUPTS

The GMS81604/08 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). The configuration of interrupt circuit is shown in Figure 28.

12 interrupt sources are provided including the Reset.

Interrupt source	Symbol	Priority
Hardware RESET	RST	1
External Interrupt 0	INT0IF	2
External Interrupt 1	INT1IF	3
External Interrupt 2	INT2IF	4
External Interrupt 3	INT3IF	5
Timer/Counter 0	T0IF	6
Timer/Counter 1	T1IF	7
Timer/Counter 2	T2IF	8
Timer/Counter 3	T3IF	9
AD Converter	AIF	10
Watch dog timer	WDTIF	11
Basic interval timer	BITIF	12

*Vector addresses are shown in Program Memory section.

The External Interrupts INT0~INT3 can each be transition-activated, depending on interrupt edge selection register.

The Timer 0~Timer 3 Interrupts are generated by T0IF ~T3IF, which are set by a match in their respective timer/counter register.
The AD converter Interrupt is generated by AIF which is set by finishing the analog to digital conversion.
The Watch dog timer Interrupt is generated by WDTIF which set by a match in Watch dog 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), the interrupt enable register (IENH, IENL) and the interrupt request flags (in IRQH, IRQL) except Power-on reset and software BRK interrupt.

Interrupt enable registers are shown in Figure 29. These registers are composed of interrupt enable flags of each interrupt source, these flags determines

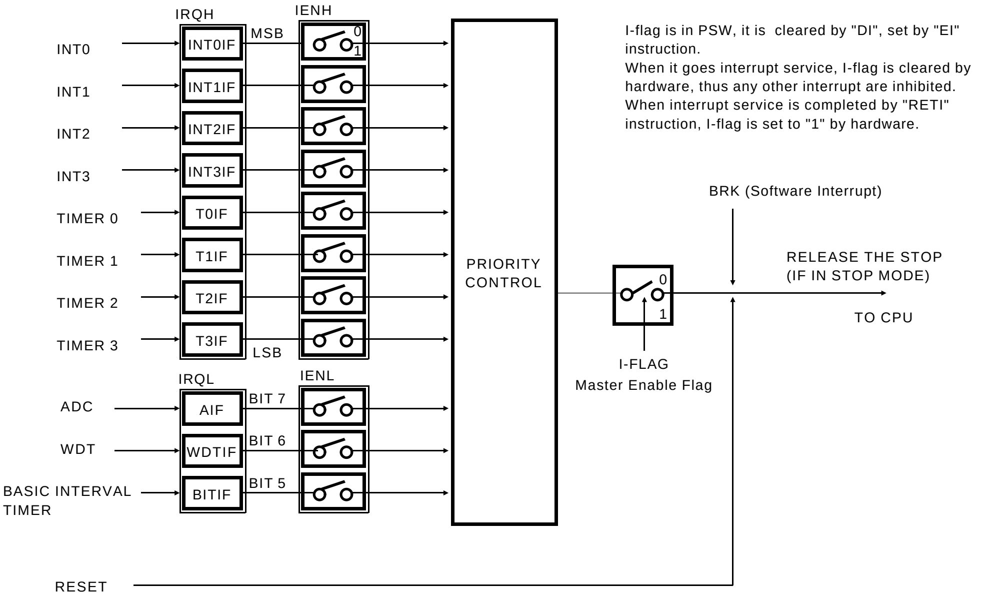

IRQH
IENH
INT0
INT1
INT2
INT3
TIMER 0
TIMER 1
TIMER 2
TIMER 3
INT0IF
INT1IF
INT2IF
INT3IF
T0IF
T1IF
T2IF
T3IF
MSB
LSB
0
1
IRQL
IENL
ADC
WDT
BASIC INTERVAL TIMER
AIF
WDTIF
BITIF
BIT 7
BIT 6
BIT 5
PRIORITY CONTROL
RESET
I-flag is in PSW, it is cleared by "DI", set by "EI" instruction.
When it goes interrupt service, I-flag is cleared by hardware, thus any other interrupt are inhibited.
When interrupt service is completed by "RETI" instruction, I-flag is set to "1" by hardware.
BRK (Software Interrupt)
RELEASE THE STOP (IF IN STOP MODE)
TO CPU
0
1
I-FLAG
Master Enable Flag

Figure 28. Block Diagram of Interrupt Function

	MSB							LSB	
IENH	INT0E	INT1E	INT2E	INT3E	T0E	T1E	T2E	T3E	ADDRESS: F6$_H$ RESET VALUE: 00$_H$

	MSB							LSB	
IENL	AE	WDTE	BITE	-	-	-	-	-	ADDRESS: F4$_H$ RESET VALUE: 000-----

Enables or disables the interrupt individually.
If flag is cleared, the interrupt is disabled.
0: Disable
1: Enable

Figure 29. IENH, IENL: Interrupt Enable Registers

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.

When an interrupt is responded to, the I-flag is cleared to disable any further interrupt, the return address is pushed into the stack and the PC is vectored to. Once in the interrupt service routine the source(s) of the interrupt can be determined by polling the interrupt flag bits.

The interrupt flag bit(s) must be cleared in software before reenabling interrupts to avoid recursive interrupts. The Interrupt Request flags are able to be read and write.

External Interrupt

External interrupt on INT0~INT3 pins are edge triggered depending the edge selection register IEDS.

The edge detection of external interrupt has three transition activated mode: rising edge, falling edge, both edge. INT0~INT3 are multiplexed with general I/O ports (R40~R43). To use external interrupt pin, set bit 0 to bit 3 of the port mode register PMR4.

The PMR4 and IEDS registers are shown in Figure 32.

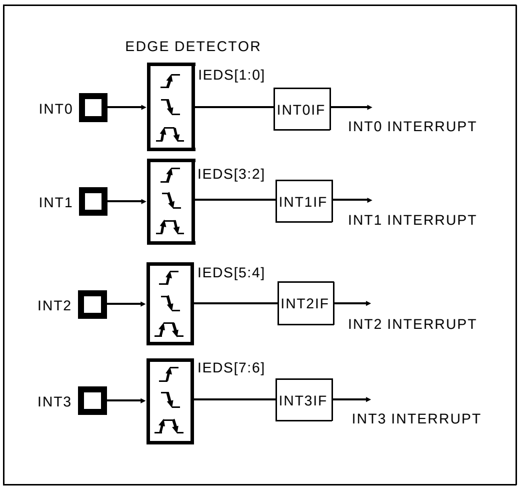


Figure 30. External Interrupt

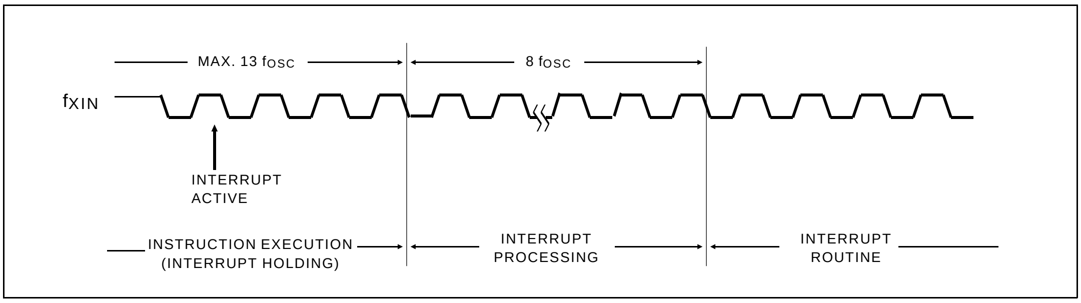


Figure 31. INT Pin Interrupt Timing

PMR4

MSB							LSB
T3S	T1S	EC2S	EC0S	INT3S	INT2S	INT1S	INT0S

ADDRESS: $D0_H$
RESET VALUE: 00_H

Relation with Timer/Counter Function

T3S	0: R47 1: T3O (TIMER/COUNTER 3 OUTPUT)
T1S	0: R46 1: T1O (TIMER/COUNTER 1 OUTPUT)
EC2S	0: R45 1: EC2 (EXTERNAL INPUT PIN FOR TIMER/COUNTER 2
EC0S	0: R44 1: EC0 (EXTERNAL INPUT PIN FOR TIMER/COUNTER 0

Relation with External Interrupt function

INT3S	0: R43 1: INT3 (EXTERNAL INTERRUPT 3)
INT2S	0: R42 1: INT2 (EXTERNAL INTERRUPT 2)
INT1S	0: R41 1: INT1 (EXTERNAL INTERRUPT 1)
INT0S	0: R40 1: INT0 (EXTERNAL INTERRUPT 0)

IEDS

MSB							LSB
IED3H	IED3L	IED2H	IED2L	IED1H	IED1L	IED0H	IED0L
INT3		INT2		INT1		INT0	

ADDRESS: $F8_H$
RESET VALUE: 00_H

Edge selection register
00: Reserved
01: Falling (1-to-0 transition)
10: Rising (0-to-1 transition)
11: Both (Rising & Falling)

Figure 32. PMR4 and IEDS Registers

BRK Interrupt

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

Interrupt vector address of BRK is shared with the vector of TCALL0 (Refer to Program Memory Section). When BRK interrupt is generated, B-flag of PSW is set to distinguish BRK from TCALL0.

Each processing step is determined by B-flag as shown below.

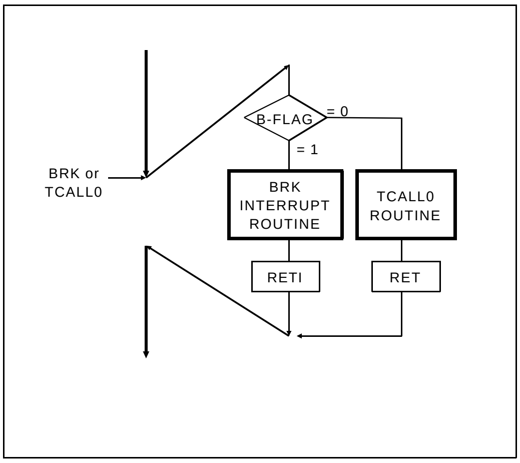


Figure 33. Execution of BRK/ TCALL0

Multiple Interrupt

If two requests of different priority levels are received simultaneously, the request of higher priority level is serviced. If requests of the same priority level are received simultaneously, an internal polling sequence determines by hardware which request is serviced. Hardware interrupt priority is shown in Page37.

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

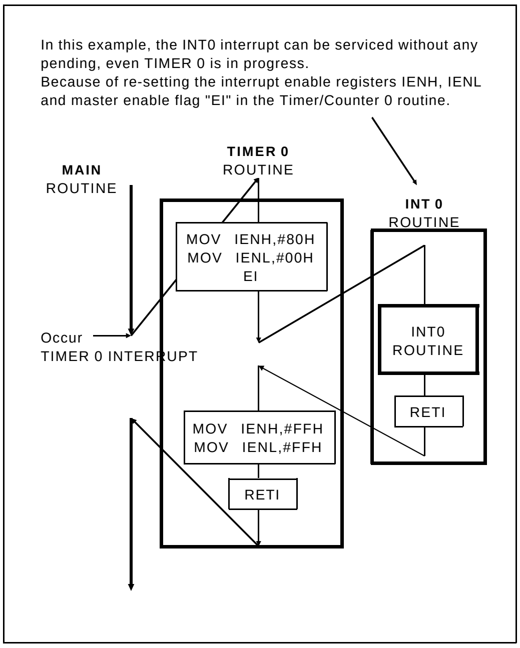

In this example, the INT0 interrupt can be serviced without any pending, even TIMER 0 is in progress.
Because of re-setting the interrupt enable registers IENH, IENL and master enable flag "EI" in the Timer/Counter 0 routine.
MAIN ROUTINE
TIMER 0 ROUTINE
INT 0 ROUTINE
MOV IENH,#80H
MOV IENL,#00H
EI
Occur TIMER 0 INTERRUPT
INT0 ROUTINE
RETI
MOV IENH,#FFH
MOV IENL,#FFH
RETI

Figure 34. Execution of Multi-Interrupt

WATCHDOG TIMER

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 consists of 6-bit binary counter, 6-bit comparator and the watchdog timer data register. When the value of 6-bit binary counter is equal to the lower 6 bits of WDTR, the match is generated to go to reset the CPU.

The 6-bit binary counter is cleared by WDTCL=1.

Caution:
Because the watchdog timer counter is enabled after clearing Basic Interval Timer .
After the bit WDTON set to "1", maximum error of timer is depend on prescaler ratio of Basic Interval Timer.

This watchdog timer can also be used as a simple 6-bit timer by interrupt WDTIF. The interval of watchdog timer interrupt is decided by Basic Interval Timer. Interval equation is as below.

$$T_{WDT} = WDTR \cdot Interval\ of\ BIT$$

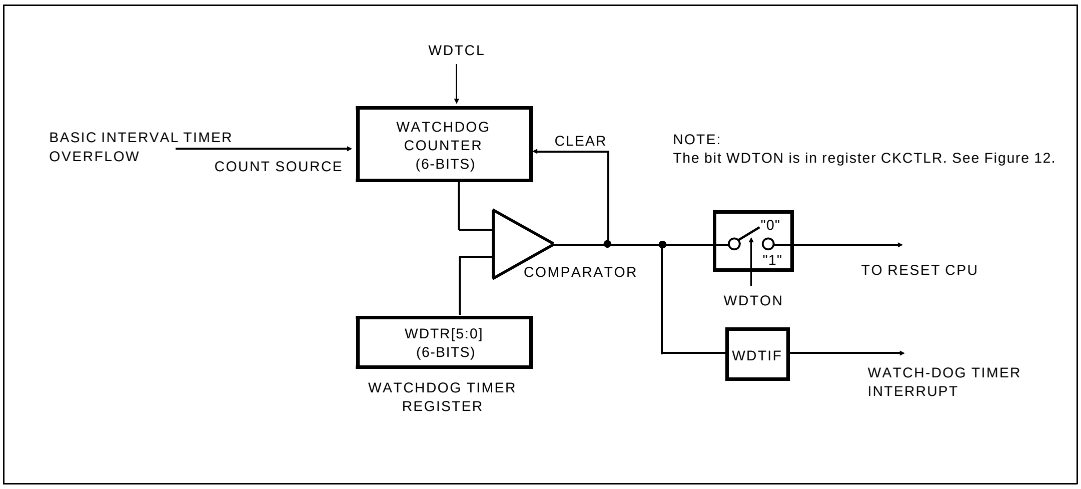


Figure 35. Block Diagram of Watch-dog Timer

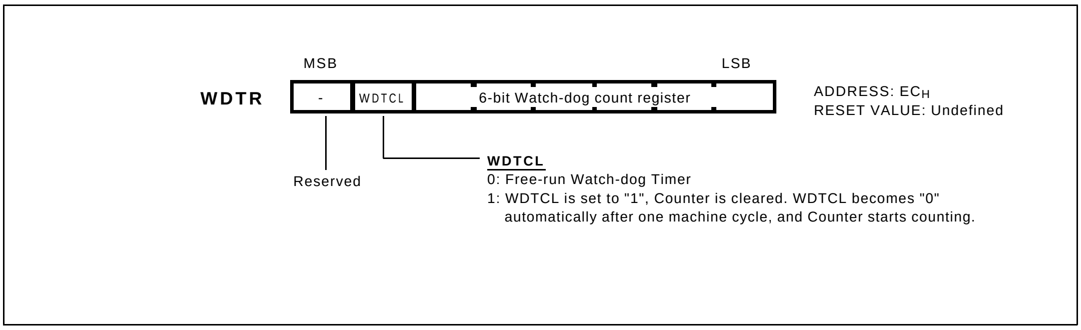

MSB
LSB
WDTR
-
WDTCL
6-bit Watch-dog count register
ADDRESS: EC$_H$
RESET VALUE: Undefined
Reserved
WDTCL
0: Free-run Watch-dog Timer
1: WDTCL is set to "1", Counter is cleared. WDTCL becomes "0" automatically after one machine cycle, and Counter starts counting.

Figure 36. WDTR: Watch-dog Timer Data Register

STOP MODE

For applications where power consumption is a critical factor, device provides reduced power of STOP.

An instruction that STOP causes that to be the last instruction executed before going into the 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 Rx, port direction register RxDD. The status of peripherals during Stop mode is shown below.

Peripheral	Status
RAM	Retain
Control registers	Retain
I/O	Retain
Oscillation	Stop
X_{IN}	Low
X_{OUT}	High

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_{DD} is restored to its normal operating level, and must be held active long enough to allow the oscillator to restart and stabilize (minimum 20 msec).

> ***Caution:***
> *The NOP instruction have to be written more than two to next line of the STOP instruction.*
> *Ex)*
> *STOP*
> *NOP*
> *NOP*

Release Stop Mode

The exit from Stop mode is hardware reset or external interrupt. Reset redefines 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.

When exit from Stop mode by external interrupt from Stop mode, enough oscillation stabilization time is required to normal operation. Figure 37 shows the timing diagram. When release the Stop mode, the

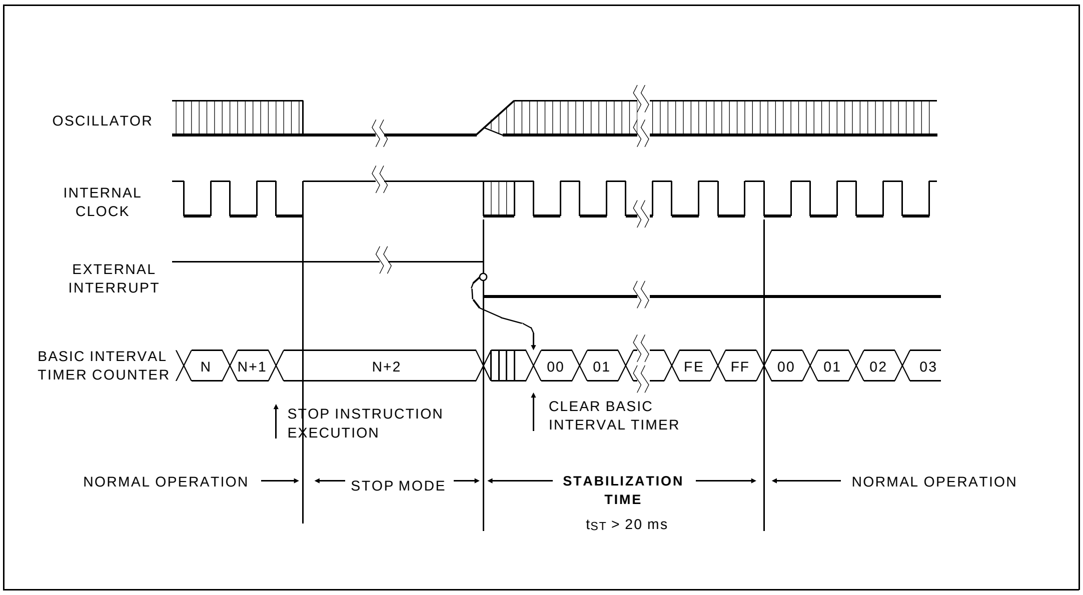


Figure 37. Timing of Stop Release by External Interrupt

Wake-up and Reset Function Table

Event	Chip Status before event	Chip function after event	
		PC	Oscillator Circuit
RESET	Do not care	Vector	on
STOP instruction	Normal operation	N+1	off
External Interrupt	Normal operation	Vector	on
External Interrupt Wake-up	Stop, I-flag = 1 Stop, I-flag = 0	Vector N+1	on on

PC: Program Counter contents after the event.
N: Address of STOP instruction.

Basic interval timer is activated on wake-up. It is incremented from 00_H until FF_H then 00_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 crystal oscillator has started and stabilized.

By reset, exit from Stop mode is shown in Figure 38.

Minimizing Current Consumption in Stop Mode

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. Weak pull-ups on port pins should be turned off, if possible. All inputs should be either as V_{SS} or at V_{DD} (or as close to rail as possible). An intermediate voltage on an input pin causes the input buffer to draw a significant amount of current.

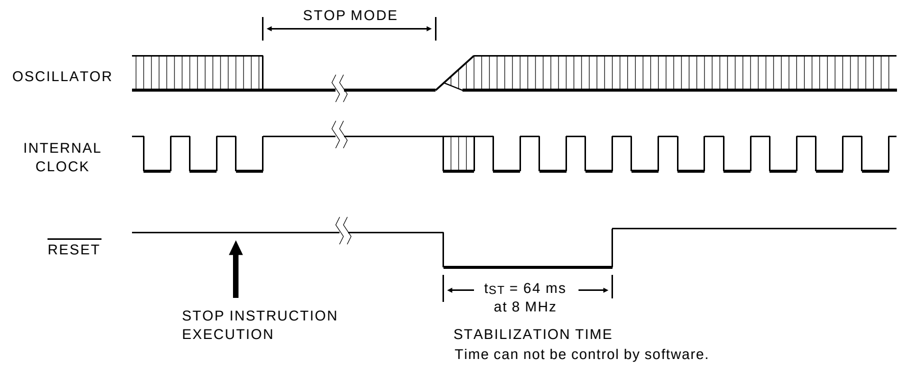

STOP MODE
OSCILLATOR
INTERNAL CLOCK
$\overline{\text{RESET}}$
STOP INSTRUCTION EXECUTION
t_{ST} = 64 ms at 8 MHz
STABILIZATION TIME
Time can not be control by software.

Figure 38. Timing of Stop Mode Release by Reset

RESET

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, while the oscillator running. After reset, 64ms (at 8 MHz) plus 7 oscillator periods are required to start execution as shown in Figure 40.

Internal RAM is not affected by reset. When V_{DD} is turned on, the RAM content is indeterminate. Initial state of each register is as follow. Therefore, this RAM should be initialized before reading or testing it.

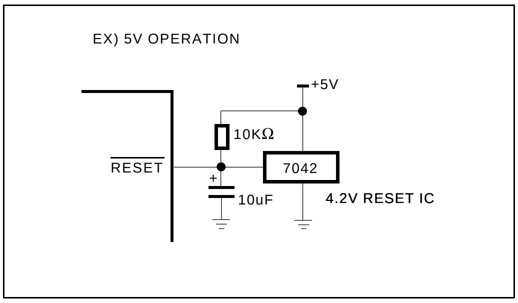


Figure 39. Example of Reset circuit

Register	Content
A	X
X	X
Y	X
PSW	00H
PC	X
SP	X
R0	X
R0DD	00000000
R1	X
R1DD	00000000
R4	X
R4DD	00000000
R5	X
R5DD	--0---00
R6	X
R6DD	00000000
PMR4	00000000
PMR5	--0-----
BITR	00H
CKCTLR	--010111
WDTR	-0111111
TM0	00H
TM2	00H
TDR0/ T0/ CDR0	X
TDR1/ T1/ CDR1	X
TDR2/ T2/ CDR2	X
TDR3/ T3/ CDR3	X
ADCM	--000001
ADR	X
BUR	X
PFDR	-----100
IENH	00H
IENL	000-----
IRQH	00H
IRQL	000-----
IEDS	00H

- = unimplemented bit
X= unknown

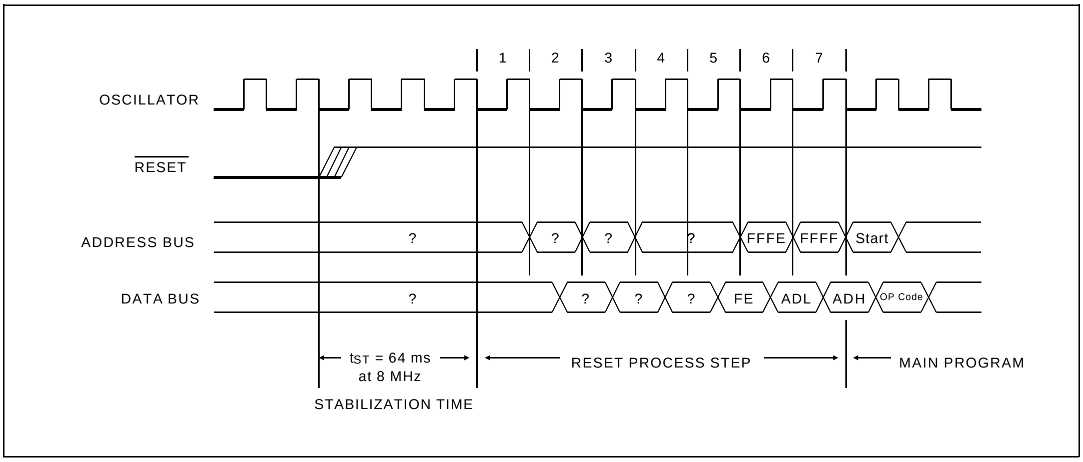


Figure 40. Timing Diagram after Reset

POWER FAIL PROCESSOR

The GMS81604/08 have on-chip power fail detection circuitry to immunize against power noise. A configuration register, PFDR, can enable (if clear/programmed) or disable (if set) the Power-fail Detect circuitry. If V_{DD} falls below 3.0~4.0V range for longer than 100 ns, the Power fail situation may reset MCU according to PFR bit of PFDR.

> ***Caution:***
> *Power fail processor function is not available on 3V operation, because this function will detect power fail all the time.*

As below PFDR register is not implemented on the in-circuit emulator, user can not experiment with it. Therefore, after final development of user program, this function may be experimented.

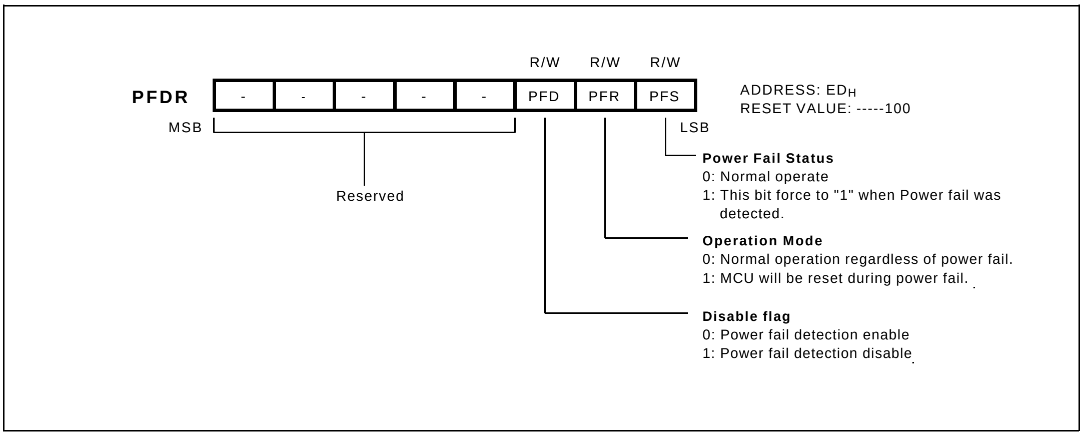

R/W
R/W
R/W
PFDR
-
-
-
-
-
PFD
PFR
PFS
ADDRESS: ED$_H$
RESET VALUE: -----100
MSB
LSB
Reserved
Power Fail Status
0: Normal operate
1: This bit force to "1" when Power fail was detected.
Operation Mode
0: Normal operation regardless of power fail.
1: MCU will be reset during power fail.
Disable flag
0: Power fail detection enable
1: Power fail detection disable

Figure 41. PFDR: Power Fail Detector Register

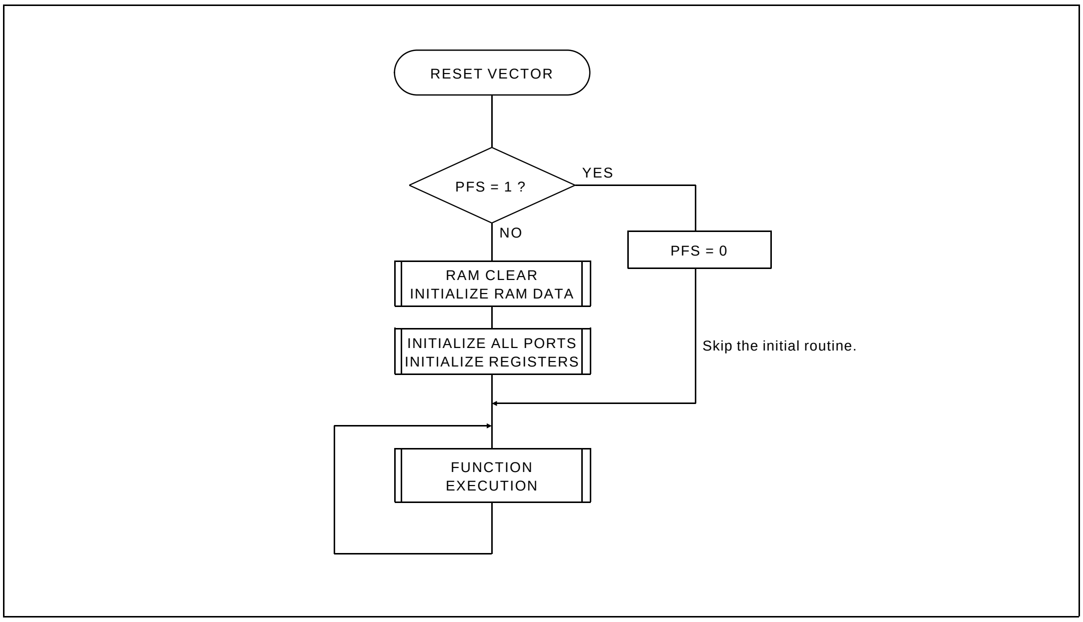


Figure 42. Example S/W of Reset flow by Power Fail

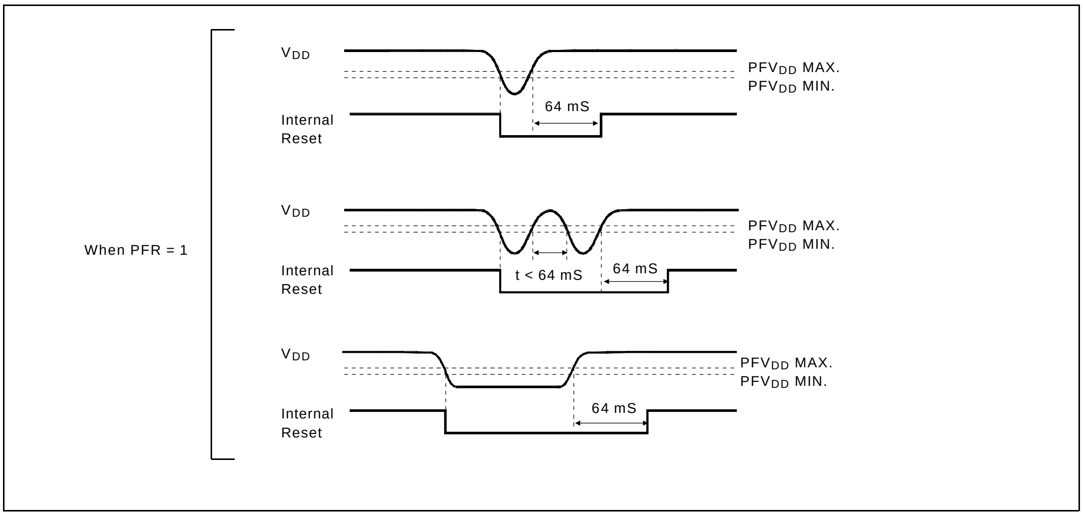
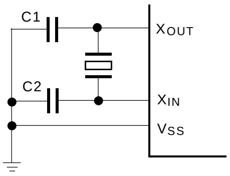


Figure 43. Power Fail Processor Situations

OSCILLATOR CIRCUIT

X_{IN} and X_{OUT} are the input and output, respectively, of a inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 44.



Recommend:
C1,C2 = 30 pF ± 10 pF for Crystals.

Figure 44. Oscillator Connections

To drive the device from an external clock source, X_{OUT} should be left unconnected while X_{IN} is driven as shown in Figure 45. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum high and low times specified on the data sheet must be observed.

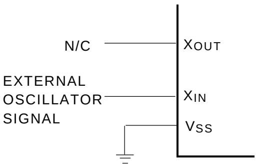


Figure 45. External Clock Drive Configuration

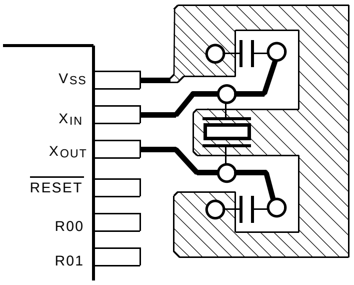


Figure 46. Layout of Crystal

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 46. for the layout of the crystal. In all cases, an external clock operation is available.

UNUSED PORTS

All unused ports should be set properly that current flow through the port does not 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 does not flow.

But input voltage level should be V_{SS} or V_{DD}. Be careful that if unspecified voltage, i.e. if unfirmed voltage level is applied to input pin, there can be little current (max. 1mA at around 2V) flow.

If it is not appropriate to set to 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.

GMS81608T (OTP) PROGRAMMING

The GMS81608T is one-time PROM (OTP) microcontroller with 8K bytes electrically programmable read only memory for the GMS81604/08 system evaluation, first production and fast mass production.

The programming to the OTP device, user can have two way. One is using the universal programmer which is support LGS microcontrollers, other is using the general EPROM programmer.

1. Using the Universal programmer

Third party universal programmer support to program the GMS81608T microcontrollers and lists are shown as below.

Manufacturer: **Advantech**
Web site: http://www.aec.com.tw
Programmer: LabTool-48

Manufacturer: **Hi-Lo systems**
Web site: http://www.hilosystems.com.tw
Programmer: ALL-11, GANG-08

Socket adapters are supported by third party programmer manufacturer.

2. Using the general EPROM(27C256) programmer

The programming algorithm is simmilar with the standart EPROM 27C256. It give some convience that user can use standard EPROM programmer. ***Make sure that 1ms programming pulse must be used, it generally called "Intelligent Mode".*** Do not use 100us programming pulse mode, "Quick Pulse Mode".

When user use general EPROM programmer, socket adaper is essencially required. It convert pin to fit the pin of general 27C256 EPROM.

Three type socket adapters are provided according to package variation as below table.

Socket Adapter	Package Type
OA816A-40SD	40 pin DIP
OA816A-42SD	42 pin SDIP
OA816A-42PL	44 pin PLCC

With these socket adapters, the GMS81608T can easy be programming and verifying using 27C256 EPROM mode on general-purpose PROM programmer.

In assembler and file type, two files are generated after compiling. One is "*.HEX", another is "*.OTP". The "*.HEX" file is used for emulation in circuit emulator (CHOICE-Dr^{TM} or CHOICE-Jr^{TM}) and "*.OTP" file is used for programming to the OTP device.

Programming Procedure

1. Select the EPROM device and manufacturer on EPROM programmer (Intel 27C256).

2. Select the programming algorithm as an Intelligent mode (apply 1ms writing pulse), not a Quick pulse mode.

3. Load the file (*.OTP) to the programmer.

4. Set the programming address range as below table.

Address	Set Value
Buffer start address	6000_H
Buffer end address	$7FFF_H$
Device start address	6000_H

5. Mount the socket adapter with the GMS81608T on the PROM programmer.

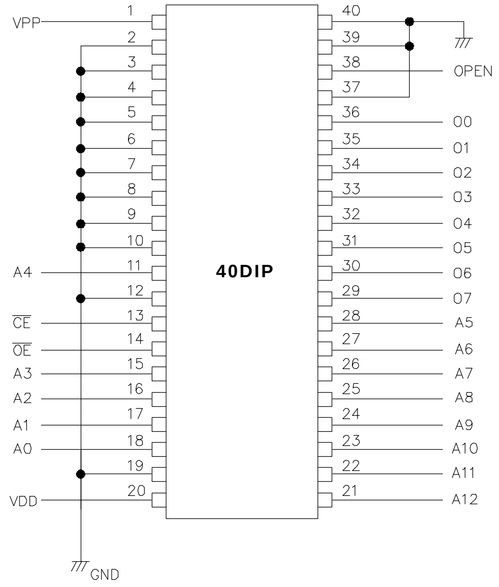
6. Start the PROM programmer to programming/ verifying.

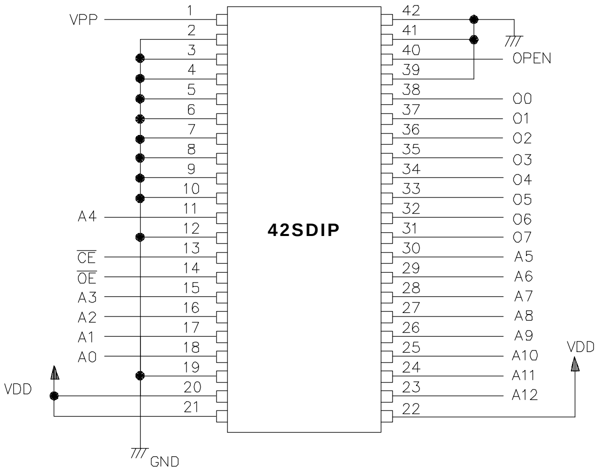
GMS81608T PROGRAMMING MANUAL

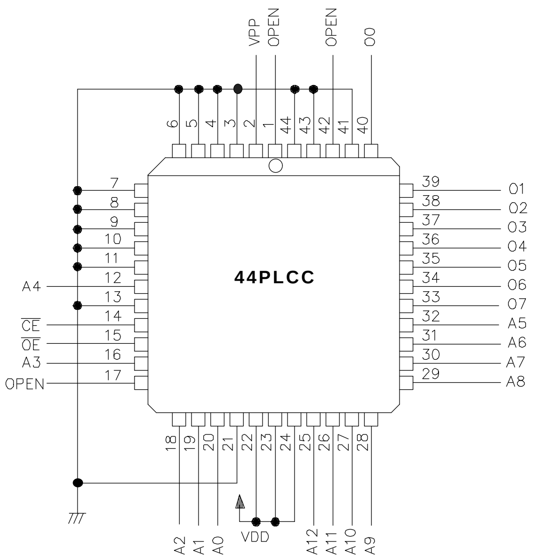
GMS815045T PACKAGE

DEVICE NAME	PACKAGE
GMS81608T	40DIP
GMS81608T K	42SDIP
GMS81608T PL	44PLCC

PIN CONFIGURATION




42SDIP
VPP
1
2
3
4
5
6
7
8
9
10
A4
11
12
13
CE
14
OE
15
A3
16
A2
17
A1
18
A0
19
VDD
20
21
GND
42
41
40
OPEN
39
38
O0
37
O1
36
O2
35
O3
34
O4
33
O5
32
O6
31
O7
30
A5
29
A6
28
A7
27
A8
26
A9
25
A10
24
A11
23
A12
22
VDD


44PLCC
VPP
OPEN
OPEN
O0
6
5
4
3
2
1
44
43
42
41
40
7
8
9
10
11
A4
12
13
14
CE
15
OE
16
A3
17
OPEN
39
O1
38
O2
37
O3
36
O4
35
O5
34
O6
33
O7
32
A5
31
A6
30
A7
29
A8
18
19
20
21
22
23
24
25
26
27
28
A2
A1
A0
VDD
A12
A11
A10
A9

40DIP Package for GMS81608T

Pin No.	MCU Mode		OTP Mode	
1	$\overline{TEST}$	I	V_{PP}	-
2	AV_{DD}	-	(1)	-
3	R67/AN7	I/O	(1)	-
4	R66/AN6	I/O	(1)	-
5	R65/AN5	I/O	(1)	-
6	R64/AN4	I/O	(1)	-
7	R63/AN3	I	(1)	-
8	R62/AN2	I	(1)	-
9	R61/AN1	I	(1)	-
10	R60/AN0	I	(1)	-
11	R47/T3O	I/O	A4	I
12	R46/T1O	I/O	(1)	-
13	R45/$\overline{EC2}$	I/O	$\overline{CE}$	I
14	R44/$\overline{EC0}$	I/O	$\overline{OE}$	I
15	R43/INT3	I/O	A3	I
16	R42/INT2	I/O	A2	I
17	R41/INT1	I/O	A1	I
18	R40/INT0	I/O	A0	I
19	R55/BUZ	I/O	(1)	-
20	V_{DD}	-	V_{DD}	-

Pin No.	MCU Mode		OTP Mode	
21	R17	I/O	A12	I
22	R16	I/O	A11	I
23	R15	I/O	A10	I
24	R14	I/O	A9	I
25	R13	I/O	A8	I
26	R12	I/O	A7	I
27	R11	I/O	A6	I
28	R10	I/O	A5	I
29	R07	I/O	O7	O
30	R06	I/O	O6	O
31	R05	I/O	O5	O
32	R04	I/O	O4	O
33	R03	I/O	O3	O
34	R02	I/O	O2	O
35	R01	I/O	O1	O
36	R00	I/O	O0	O
37	$\overline{RESET}$	I	(1)	-
38	X_{OUT}	O	(3)	-
39	X_{IN}	I	(1)	-
40	V_{SS}	-	(1)	-

NOTES:

(1) Pins must be connected to V_{SS}, because these pins are input ports during programming, program verify and reading
(2) Pins must be connected to V_{DD}.
(3) X_{OUT} pin must be opened during programming.

I/O: Input/Output Pin
I: Input Pin
O: Output Pin

42SDIP Package for GMS81608T

Pin No.	MCU Mode		OTP Mode	
1	$\overline{\mathrm{TEST}}$	I	V_{PP}	-
2	AV_{DD}	-	(1)	-
3	R67/AN7	I/O	(1)	-
4	R66/AN6	I/O	(1)	-
5	R65/AN5	I/O	(1)	-
6	R64/AN4	I/O	(1)	-
7	R63/AN3	I	(1)	-
8	R62/AN2	I	(1)	-
9	R61/AN1	I	(1)	-
10	R60/AN0	I	(1)	-
11	R47/T3O	I/O	A4	I
12	R46/T1O	I/O	(1)	-
13	R45/$\overline{\mathrm{EC2}}$	I/O	$\overline{\mathrm{CE}}$	I
14	R44/$\overline{\mathrm{EC0}}$	I/O	$\overline{\mathrm{OE}}$	I
15	R43/INT3	I/O	A3	I
16	R42/INT2	I/O	A2	I
17	R41/INT1	I/O	A1	I
18	R40/INT0	I/O	A0	I
19	R55/BUZ	I/O	(1)	-
20	V_{DD}	-	V_{DD}	-
21	R51	I/O	(2)	-

Pin No.	MCU Mode		OTP Mode	
22	R50	I/O	(2)	-
23	R17	I/O	A12	I
24	R16	I/O	A11	I
25	R15	I/O	A10	I
26	R14	I/O	A9	I
27	R13	I/O	A8	I
28	R12	I/O	A7	I
29	R11	I/O	A6	I
30	R10	I/O	A5	I
31	R07	I/O	O7	O
32	R06	I/O	O6	O
33	R05	I/O	O5	O
34	R04	I/O	O4	O
35	R03	I/O	O3	O
36	R02	I/O	O2	O
37	R01	I/O	O1	O
38	R00	I/O	O0	O
39	$\overline{\mathrm{RESET}}$	I	(1)	-
40	X_{OUT}	O	(3)	-
41	X_{IN}	I	(1)	-
42	V_{SS}	-	(1)	-

NOTES:

(1) Pins must be connected to V_{SS}, because these pins are input ports during programming, program verify and reading

(2) Pins must be connected to V_{DD}.

(3) X_{OUT} pin must be opened during programming.

I/O: Input/Output Pin
I: Input Pin
O: Output Pin

44PLCC Package for GMS81608T

Pin No.	MCU Mode		OTP Mode	
1	N.C.	-	N.C.	-
2	$\overline{\text{TEST}}$	I	V_{PP}	-
3	AV_{DD}	-	(1)	-
4	R67/AN7	I/O	(1)	-
5	R66/AN6	I/O	(1)	-
6	R65/AN5	I/O	(1)	-
7	R64/AN4	I/O	(1)	-
8	R63/AN3	I	(1)	-
9	R62/AN2	I	(1)	-
10	R61/AN1	I	(1)	-
11	R60/AN0	I	(1)	-
12	R47/T3O	I/O	A4	I
13	R46/T1O	I/O	(1)	-
14	R45/$\overline{\text{EC2}}$	I/O	$\overline{\text{CE}}$	I
15	R44/$\overline{\text{EC0}}$	I/O	$\overline{\text{OE}}$	I
16	R43/INT3	I/O	A3	I
17	N.C.	-	N.C.	-
18	R42/INT2	I/O	A2	I
19	R41/INT1	I/O	A1	I
20	R40/INT0	I/O	A0	I
21	R55/BUZ	I/O	(1)	-
22	V_{DD}	-	V_{DD}	-
23	R51	I/O	(2)	-
24	R50	I/O	(2)	-
25	R17	I/O	A12	I
26	R16	I/O	A11	I
27	R15	I/O	A10	I
28	R14	I/O	A9	I
29	R13	I/O	A8	I
30	R12	I/O	A7	I
31	R11	I/O	A6	I
32	R10	I/O	A5	I
33	R07	I/O	O7	O
34	R06	I/O	O6	O
35	R05	I/O	O5	O
36	R04	I/O	O4	O
37	R03	I/O	O3	O
38	R02	I/O	O2	O
39	R01	I/O	O1	O
40	R00	I/O	O0	O
41	$\overline{\text{RESET}}$	I	(1)	-
42	X_{OUT}	O	(3)	-
43	X_{IN}	I	(1)	-
44	V_{SS}	-	(1)	-

NOTES:

(1) Pins must be connected to V_{SS}, because these pins are input ports during programming, program verify and reading

(2) Pins must be connected to V_{DD}.

(3) X_{OUT} pin must be opened during programming.

I/O: Input/Output Pin
I: Input Pin
O: Output Pin

PIN FUNCTION (OTP Mode)

V_{PP} (Program Voltage)

V_{PP} is the input for the program voltage for programming the EPROM.

$\overline{CE}$ (Chip Enable)

CE is the input for programming and verifying internal EPROM.

$\overline{OE}$ (Output Enable)

OE is the input of data output control signal for verify.

A_0~A_{12} (Address Bus)

A_0~A_{12} are address input pins for internal EPROM.

O_0~O_7 (EPROM Data Bus)

These are data bus for internal EPROM.

PROGRAMMING

The GMS81608T has address A_0~A_{12} pins. Therefore, the programmer just program 8K bytes data of addresses 6000_H to $7FFF_H$ into the GMS81608T OTP device. During the programming addresses A_{13}, A_{14}, A_{15} of programmer must be pulled to a logic high.
When the programmer write the data from 6000_H to $7FFF_H$, consequently, the data actually will be written into addresses $E000_H$ to $FFFF_H$ of the OTP device.

Programming Flow

1. The data format to be programmed is made up of Motorola S1 format.

Ex) "Motorola S1" format;

```
S00B00005741544348363038DF
S1246000E1FF3BFF04A13F8F06E1C1711BFF3F1B003E1B00371B00361BFF3D1B003C1BFF3385
S12460211BFF321BFF351B92131B7FCC1BF3D61B17FD1BFCFC1B821B1BE01D1B8E191BFD18B1
:
:
S1057FF2941FD6
S1057FFEFF1F5F
S9030000FC
```

2. Down load above data into programmer from PC.

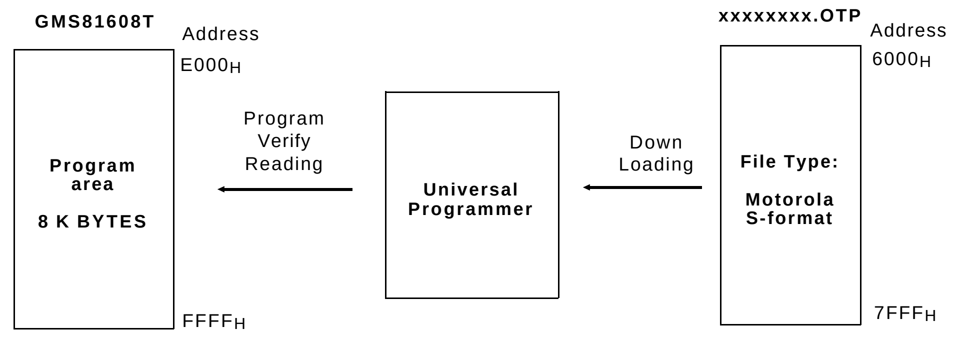
3. Programming the data from address 6000_H to $7FFF_H$ into the OTP MCU, the data must be turned over respectively, and then record the data into the OTP device. When read the data, it also must be turned over.
Ex) 00(00000000)→FF(11111111), 76(01110110)→89(10001001), FF(11111111)→00(00000000) etc.

4. Of course, the check sum is result of the sum of whole data from address 6000_H to $7FFF_H$ in the file (not reverse data of the OTP MCU).

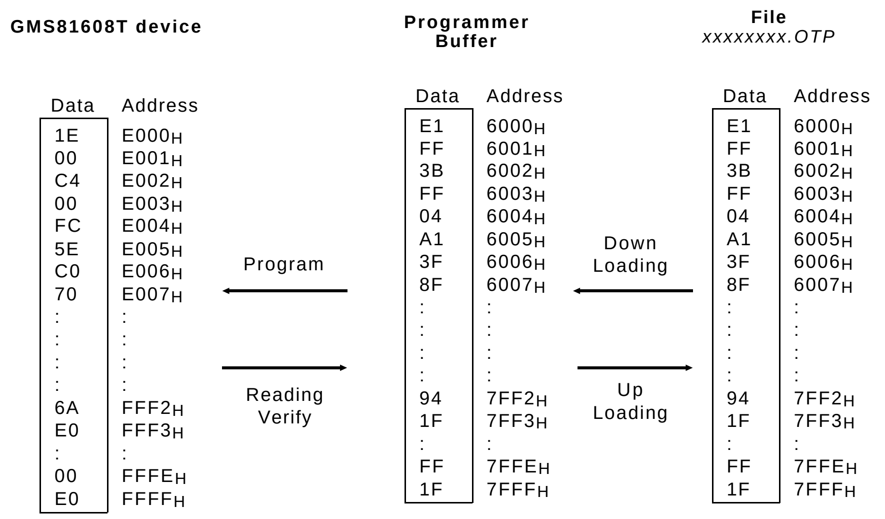
* When GMS81608T shipped, the blank data of GMS81608T is initially 00_H (not FF_H).

Programming Flow

Buffer Start Address: 6000$_H$
Buffer End Address: 7FFF$_H$
Device Start Address: E000$_H$



Programming Example



GMS81608T device

Data	Address
1E	E000$_H$
00	E001$_H$
C4	E002$_H$
00	E003$_H$
FC	E004$_H$
5E	E005$_H$
C0	E006$_H$
70	E007$_H$
:	:
:	:
:	:
:	:
6A	FFF2$_H$
E0	FFF3$_H$
:	:
00	FFFE$_H$
E0	FFFF$_H$

Programmer Buffer

Data	Address
E1	6000$_H$
FF	6001$_H$
3B	6002$_H$
FF	6003$_H$
04	6004$_H$
A1	6005$_H$
3F	6006$_H$
8F	6007$_H$
:	:
:	:
:	:
:	:
94	7FF2$_H$
1F	7FF3$_H$
:	:
FF	7FFE$_H$
1F	7FFF$_H$

File
xxxxxxxx.OTP

Data	Address
E1	6000$_H$
FF	6001$_H$
3B	6002$_H$
FF	6003$_H$
04	6004$_H$
A1	6005$_H$
3F	6006$_H$
8F	6007$_H$
:	:
:	:
:	:
:	:
94	7FF2$_H$
1F	7FF3$_H$
:	:
FF	7FFE$_H$
1F	7FFF$_H$

Checksum = E1+FF+3B+FF+04+A1+3F+8F+ · · · · · · · · + 94+1F+ · · · · +FF+1F

DEVICE OPERATION MODE

(T_A = 25°C ± 5°C)

Mode	$\overline{CE}$	$\overline{OE}$	A0~A15	VPP	VDD	O0~O7
Read	X	⎺╲⎽	X	V_{DD}	5.0V	D_{OUT}
Output Disable	V_{IH}	V_{IH}	X	V_{DD}	5.0V	Hi-Z
Programming	V_{IL}	V_{IH}	X	V_{PP}	V_{DD}	D_{IN}
Program Verify	X	⎺╲⎽	X	V_{PP}	V_{DD}	D_{OUT}

NOTES:
1. X = Either V_{IL} or V_{IH}
3. See DC Characteristics Table for V_{DD} and V_{PP} voltages during programming.

DC CHARACTERISTICS

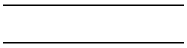
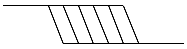
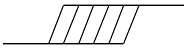
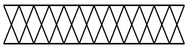
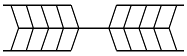
(V_{SS}=0 V, T_A = 25°C ± 5°C)

Symbol	Item	Min	Typ	Max	Unit	Test condition
V_{PP}	V_{PP} supply voltage	12.0	-	13.0	V	
$V_{DD}^{(1)}$	V_{DD} supply voltage	5.75	-	6.25	V	
I_{PP} (2)	V_{PP} supply current			50	mA	$\overline{CE}$=V_{IL}
I_{DD} (2)	V_{DD} supply current			30	mA	
V_{IH}	Input high voltage	0.8 V_{DD}			V	
V_{IL}	Input low voltage			0.2 V_{DD}	V	
V_{OH}	Output high voltage	V_{DD}-1.0			V	I_{OH} = -2.5 mA
V_{OL}	Output low voltage			0.4	V	I_{OL} = 2.1 mA
I_{IL}	Input leakage current			5	uA	

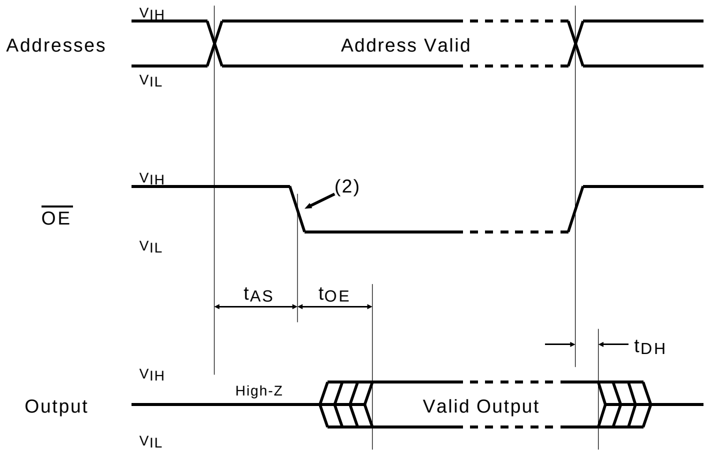
NOTES:

1. V_{DD} must be applied simultaneously or before V_{PP} and removed simultaneously or after V_{PP}.
2. The maximum current value is with outputs O_0 to O_7 unloaded.

SWITCHING WAVEFORMS

WAVEFORM	INPUTS	OUTPUTS
	Must be steady	Will be steady
	May change from H to L	Will be changing from H to L
	May change from L to H	Will be changing from L to H
	Do not care any change permitted	Changing state unknown
	Does not apply	Center line is high impedance "Off" state

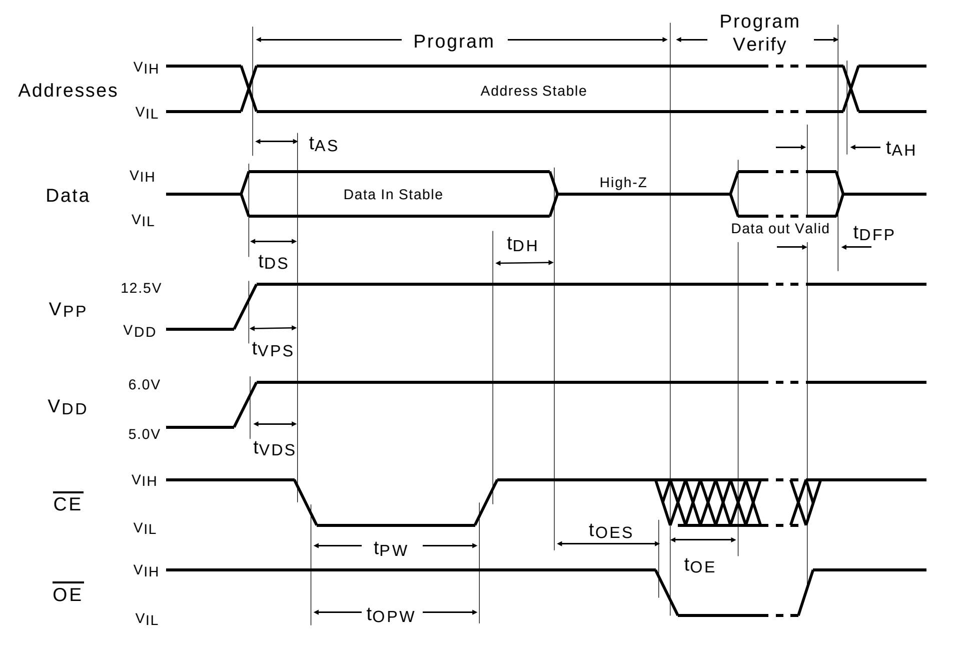
READING WAVEFORMS



NOTES:

1. The input timing reference level is 1.0 V for a V_{IL} and 4.0V for a V_{IH} at V_{DD}=5.0V
2. To read the output data, transition requires on the $\overline{OE}$ from the high to the low after address setup time t_{AS}.

PROGRAMMING ALGORITHM WAVEFORMS



NOTES:

1. The input timing reference level is 1.0 V for a V_{IL} and 4.0V for a V_{IH} at V_{DD}=5.0V

AC READING CHARACTERISTICS

(V_{SS}=0 V, T_A = 25°C ± 5°C)

Symbol	Item	Min	Typ	Max	Unit	Test condition
t_{AS}	Address setup time	2			us	
t_{OE}	Data output delay time			200	ns	
t_{DH}	Data hold time	0			ns	

NOTES:

1. V_{DD} must be applied simultaneously or before V_{PP} and removed simultaneously or after V_{PP}.

AC PROGRAMMING CHARACTERISTICS

(V_{SS}=0 V, T_A = 25°C ± 5°C; See DC Characteristics Table for V_{DD} and V_{PP} voltages.)

Symbol	Item	Min	Typ	Max	Unit	Condition* (Note 1)
t_{AS}	Address set-up time	2			us	
t_{OES}	$\overline{\text{OE}}$ set-up time	2			us	
t_{DS}	Data setup time	2			us	
t_{AH}	Address hold time	0			us	
t_{DH}	Data hold time	1			us	
t_{DFP}	Output disable delay time	0			us	
t_{VPS}	V_{PP} setup time	2			us	
t_{VDS}	V_{DD} setup time	2			us	
t_{PW}	Program pulse width	0.95	1.0	1.05	ms	
t_{OPW}	$\overline{\text{CE}}$ pulse width when over programming	2.85		78.75	ms	(Note 2)
t_{OE}	Data output delay time			200	ns	

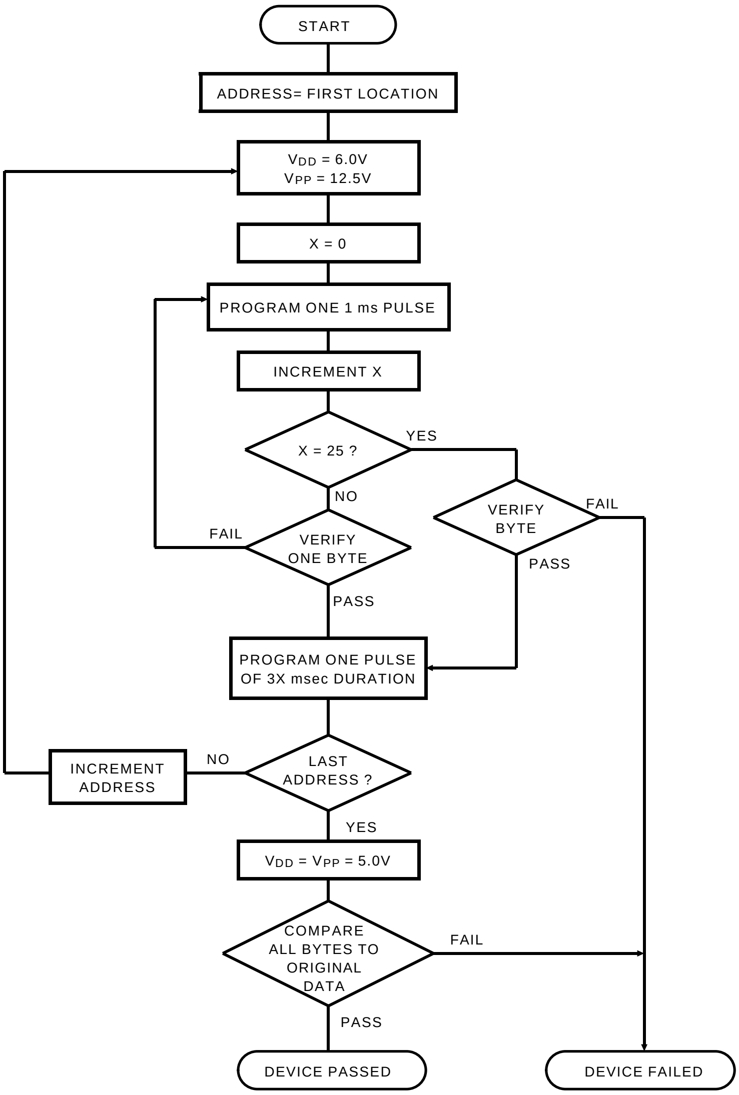
*AC CONDITIONS OF TEST

Input Rise and Fall Times (10% to 90%)	20 ns
Input Pulse Levels	0.45V to 4.55V
Input Timing Reference Level	1.0V to 4.0V
Output Timing Reference Level	1.0V to 4.0V

NOTES:

1. V_{DD} must be applied simultaneously or before V_{PP} and removed simultaneously or after V_{PP}.
2. The length of the overprogram pulse may vary from 2.85 msec to 78.75 msec as a function of the iteration counter value X (Intelligent Programming Algorithm).Refer to flow chart of page 13.

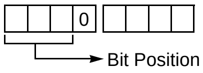
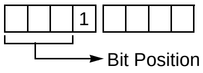
Intelligent Programming Algorithm



APPENDIX

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
y	 Upper Nibble Expression in Opcode
–	Subtraction
×	Multiplication
/	Division
( )	Contents Expression
∧	AND
∨	OR
⊕	Exclusive OR
~	NOT
←	Assignment / Transfer / Shift Left
→	Shift Right
↔	Exchange
=	Equal
≠	Not Equal

A.2 Instruction Map

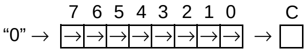
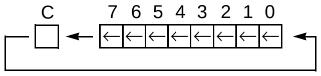
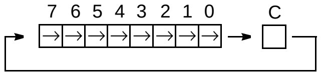
LOW HIGH	00000 00	00001 01	00010 02	00011 03	00100 04	00101 05	00110 06	00111 07	01000 08	01001 09	01010 0A	01011 0B	01100 0C	01101 0D	01110 0E	01111 0F
000	-	SET1 dp.bit	BBS A.bit,rel	BBS dp.bit,rel	ADC #imm	ADC dp	ADC dp+X	ADC !abs	ASL A	ASL dp	TCALL 0	SETA1 .bit	BIT dp	POP A	PUSH A	BRK
001	CLRC				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				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				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				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				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				LDA #imm	LDA dp	LDA dp+X	LDA !abs	TXA	LDY dp	TCALL 12	LDC LDCB	LDX dp	LDX dp+Y	XCN	DAS
111	EI				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				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				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
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	Add with carry. A ← (A) + (M) + C	NV--H-ZC
2	ADC dp	05	2	3		
3	ADC dp + X	06	2	4		
4	ADC !abs	07	3	4		
5	ADC !abs + Y	15	3	5		
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	Logical AND A ← (A) ∧ (M)	N-----Z-
10	AND dp	85	2	3		
11	AND dp + X	86	2	4		
12	AND !abs	87	3	4		
13	AND !abs + Y	95	3	5		
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 C 7 6 5 4 3 2 1 0 □ ← ←←←←←←←← ← "0"	N-----ZC
18	ASL dp	09	2	4		
19	ASL dp + X	19	2	5		
20	ASL !abs	18	3	5		
21	CMP #imm	44	2	2	Compare accumulator contents with memory contents (A) - (M)	N-----ZC
22	CMP dp	45	2	3		
23	CMP dp + X	46	2	4		
24	CMP !abs	47	3	4		
25	CMP !abs + Y	55	3	5		
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	Compare X contents with memory contents (X) - (M)	N-----ZC
30	CMPX dp	6C	2	3		
31	CMPX !abs	7C	3	4		
32	CMPY #imm	7E	2	2	Compare Y contents with memory contents (Y) - (M)	N-----ZC
33	CMPY dp	8C	2	3		
34	CMPY !abs	9C	3	4		
35	COM dp	2C	2	4	1'S Complement : (dp) ← ~(dp)	N-----Z-
36	DAA	DF	1	3	Decimal adjust for addition	N-----ZC
37	DAS	CF	1	3	Decimal adjust for subtraction	N-----ZC
38	DEC A	A8	1	2	Decrement	N-----ZC

No.	Mnemonic	Op Code	Byte No	Cycle No	Operation	Flag NVGBHIZC
39	DEC dp	A9	2	4	M ← (M) - 1	N-----Z-
40	DEC dp + X	B9	2	5		N-----Z-
41	DEC !abs	B8	3	5		N-----Z-
42	DEC X	AF	1	2		N-----Z-
43	DEC Y	BE	1	2		N-----Z-
44	DIV	9B	1	12	Divide : YA / X Q: A, R: Y	NV--H-Z-
45	EOR #imm	A4	2	2	Exclusive OR	N-----Z-
46	EOR dp	A5	2	3	A ← (A) ⊕ (M)	
47	EOR dp + X	A6	2	4		
48	EOR !abs	A7	3	4		
49	EOR !abs + Y	B5	3	5		
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	Increment	N-----ZC
54	INC dp	89	2	4	M ← (M) + 1	N-----Z-
55	INC dp + X	99	2	5		N-----Z-
56	INC !abs	98	3	5		N-----Z-
57	INC X	8F	1	2		N-----Z-
58	INC Y	9E	1	2		N-----Z-
59	LSR A	48	1	2	Logical shift right	N-----ZC
60	LSR dp	49	2	4		
61	LSR dp + X	59	2	5		
62	LSR !abs	58	3	5		
63	MUL	5B	1	9	Multiply : YA ← Y × A	N-----Z-
64	OR #imm	64	2	2	Logical OR	N-----Z-
65	OR dp	65	2	3	A ← (A) ∨ (M)	
66	OR dp + X	66	2	4		
67	OR !abs	67	3	4		
68	OR !abs + Y	75	3	5		
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	N-----ZC
73	ROL dp	29	2	4		
74	ROL dp + X	39	2	5		
75	ROL !abs	38	3	5		
76	ROR A	68	1	2	Rotate right through Carry	N-----ZC
77	ROR dp	69	2	4		
78	ROR dp + X	79	2	5		
79	ROR !abs	78	3	5		
80	SBC #imm	24	2	2	Subtract with Carry	

No.	Mnemonic	Op Code	Byte No	Cycle No	Operation	Flag NVGBHIZC
81	SBC dp	25	2	3	A ← (A) - (M) - ~(C)	NV--HZC
82	SBC dp + X	26	2	4		
83	SBC !abs	27	3	4		
84	SBC !abs + Y	35	3	5		
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	N-----Z-
89	XCN	CE	1	5	Exchange nibbles within the accumulator A_7~A_4 ↔ A_3~A_0	N-----Z-

Register / Memory Operation

No.	Mnemonic	Op Code	Byte No	Cycle No	Operation	Flag NVGBHIZC
1	LDA #imm	C4	2	2	Load accumulator	N-----Z-
2	LDA dp	C5	2	3	A ← (M)	
3	LDA dp + X	C6	2	4		
4	LDA !abs	C7	3	4		
5	LDA !abs + Y	D5	3	5		
6	LDA [dp + X]	D6	2	6		
7	LDA [dp] + Y	D7	2	6		
8	LDA { X }	D4	1	3		
9	LDA { X }+	DB	1	4	X- register auto-increment : A ← (M) , X ← X + 1	
10	LDM dp,#imm	E4	3	5	Load memory with immediate data : (M) ← imm	--------
11	LDX #imm	1E	2	2	Load X-register	N-----Z-
12	LDX dp	CC	2	3	X ← (M)	
13	LDX dp + Y	CD	2	4		
14	LDX !abs	DC	3	4		
15	LDY #imm	3E	2	2	Load Y-register	N-----Z-
16	LDY dp	C9	2	3	Y ← (M)	
17	LDY dp + X	D9	2	4		
18	LDY !abs	D8	3	4		
19	STA dp	E5	2	4	Store accumulator contents in memory	--------
20	STA dp + X	E6	2	5	(M) ← A	
21	STA !abs	E7	3	5		
22	STA !abs + Y	F5	3	6		
23	STA [dp + X]	F6	2	7		
24	STA [dp] + Y	F7	2	7		
25	STA { X }	F4	1	4		
26	STA { X }+	FB	1	4	X- register auto-increment : (M) ← A, X ← X + 1	
27	STX dp	EC	2	4	Store X-register contents in memory	--------
28	STX dp + Y	ED	2	5	(M) ← X	
29	STX !abs	FC	3	5		
30	STY dp	E9	2	4	Store Y-register contents in memory	--------
31	STY dp + X	F9	2	5	(M) ← Y	
32	STY !abs	F8	3	5		
33	TAX	E8	1	2	Transfer accumulator contents to X-register : X ← A	N-----Z-
34	TAY	9F	1	2	Transfer accumulator contents to Y-register : Y ← A	N-----Z-
35	TSPX	AE	1	2	Transfer stack-pointer contents to X-register : X ← sp	N-----Z-
36	TXA	C8	1	2	Transfer X-register contents to accumulator: A ← X	N-----Z-
37	TXSP	8E	1	2	Transfer X-register contents to stack-pointer: sp ← X	N-----Z-
38	TYA	BF	1	2	Transfer Y-register contents to accumulator: A ← Y	N-----Z-
39	XAX	EE	1	4	Exchange X-register contents with accumulator :X ↔ A	--------

40	XAY	DE	1	4	Exchange Y-register contents with accumulator :Y ↔ A	--------
41	XMA dp	BC	2	5	Exchange memory contents with accumulator (M) ↔ A	N-----Z-
42	XMA dp+X	AD	2	6		
43	XMA {X}	BB	1	5		
44	XYX	FE	1	4	Exchange X-register contents with Y-register : X ↔ 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 ← (YA) (dp +1) (dp)	NV--H-ZC
2	CMPW dp	5D	2	4	Compare YA contents with memory pair contents : (YA) – (dp+1)(dp)	N-----ZC
3	DECW dp	BD	2	6	Decrement memory pair (dp+1)(dp) ← (dp+1) (dp) - 1	N-----Z-
4	INCW dp	9D	2	6	Increment memory pair (dp+1) (dp) ← (dp+1) (dp) + 1	N-----Z-
5	LDYA dp	7D	2	5	Load YA YA ← (dp +1) (dp)	N-----Z-
6	STYA dp	DD	2	5	Store YA (dp +1) (dp) ← YA	--------
7	SUBW dp	3D	2	5	16-Bits subtract without carry YA ← (YA) - (dp +1) (dp)	NV--H-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 ← (C) ∧ (M .bit)	-------C
2	AND1B M.bit	8B	3	4	Bit AND C-flag and NOT : C ← (C) ∧ ~(M .bit)	-------C
3	BIT dp	0C	2	4	Bit test A with memory : Z ← (A) ∧ (M) , N ← (M_7) , V ← (M_6)	MM----Z-
4	BIT !abs	1C	3	5		
5	CLR1 dp.bit	y1	2	4	Clear bit : (M.bit) ← “0”	--------
6	CLRA1 A.bit	2B	2	2	Clear A bit : (A.bit) ← “0”	--------
7	CLRC	20	1	2	Clear C-flag : C ← “0”	-------0
8	CLRG	40	1	2	Clear G-flag : G ← “0”	--0-----
9	CLRV	80	1	2	Clear V-flag : V ← “0”	-0--0---
10	EOR1 M.bit	AB	3	5	Bit exclusive-OR C-flag : C ← (C) ⊕ (M .bit)	-------C
11	EOR1B M.bit	AB	3	5	Bit exclusive-OR C-flag and NOT : C ← (C) ⊕ ~(M .bit)	-------C
12	LDC M.bit	CB	3	4	Load C-flag : C ← (M .bit)	-------C
13	LDCB M.bit	CB	3	4	Load C-flag with NOT : C ← ~(M .bit)	-------C
14	NOT1 M.bit	4B	3	5	Bit complement : (M .bit) ← ~(M .bit)	--------
15	OR1 M.bit	6B	3	5	Bit OR C-flag : C ← (C) ∨ (M .bit)	-------C
16	OR1B M.bit	6B	3	5	Bit OR C-flag and NOT : C ← (C) ∨ ~(M .bit)	-------C

17	SET1 dp.bit	x1	2	4	Set bit : (M.bit) ← “1”	--------
18	SETA1 A.bit	0B	2	2	Set A bit : (A.bit) ← “1”	--------
19	SETC	A0	1	2	Set C-flag : C ← “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) ← (M) ∧ ~(A)	N-----Z-
23	TSET1 !abs	3C	3	6	Test and set bits with A : A - (M) , (M) ← (M) ∨ (A)	N-----Z-

Branch / Jump Operation

No.	Mnemonic	Op Code	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	y3	3	5/7	if (bit) = 0 , then pc ← (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 ← (pc) + rel	
5	BCC rel	50	2	2/4	Branch if carry bit clear if (C) = 0 , then pc ← (pc) + rel	- - - - - - - -
6	BCS rel	D0	2	2/4	Branch if carry bit set if (C) = 1 , then pc ← (pc) + rel	- - - - - - - -
7	BEQ rel	D0	2	2/4	Branch if equal if (Z) = 1 , then pc ← (pc) + rel	- - - - - - - -
8	BMI rel	90	2	2/4	Branch if minus if (N) = 1 , then pc ← (pc) + rel	- - - - - - - -
9	BNE rel	70	2	2/4	Branch if not equal if (Z) = 0 , then pc ← (pc) + rel	- - - - - - - -
10	BPL rel	10	2	2/4	Branch if minus if (N) = 0 , then pc ← (pc) + rel	- - - - - - - -
11	BRA rel	2F	2	4	Branch always pc ← (pc) + rel	- - - - - - - -
12	BVC rel	30	2	2/4	Branch if overflow bit clear if (V) = 0 , then pc ← (pc) + rel	- - - - - - - -
13	BVS rel	B0	2	2/4	Branch if overflow bit set if (V) = 1 , then pc ← (pc) + rel	- - - - - - - -
14	CALL !abs	3B	3	8	Subroutine call	- - - - - - - -
15	CALL [dp]	5F	2	8	M(sp)←(pc_H), sp←sp - 1, M(sp)← (pc_L), sp ←sp - 1, if !abs, pc← abs ; if [dp], pc_L← (dp), pc_H← (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) ≠ (M) , then pc ← (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) ≠ 0 , then pc ← (pc) + rel.	
20	JMP !abs	1B	3	3	Unconditional jump	- - - - - - - -
21	JMP [!abs]	1F	3	5	pc ← jump address	
22	JMP [dp]	3F	2	4		
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) ←(pc_H), sp ← sp - 1, M(sp) ← (pc_L),sp ← sp - 1, pc_L ← (Table vector L), pc_H ← (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 ← ”1”, M(sp) ← (pc_H), sp ←sp-1, M(s) ← (pc_L), sp ← sp - 1, M(sp) ← (PSW), sp ← sp -1, pc_L ← (0FFDE$_H$) , pc_H ← (0FFDF$_H$) .	---1-0--
2	DI	60	1	3	Disable all interrupts : I ← “0”	-----0--
3	EI	E0	1	3	Enable all interrupt : I ← “1”	-----1--
4	NOP	FF	1	2	No operation	--------
5	POP A	0D	1	4	sp ← sp + 1, A ← M(sp)	--------
6	POP X	2D	1	4	sp ← sp + 1, X ← M(sp)	
7	POP Y	4D	1	4	sp ← sp + 1, Y ← M(sp)	
8	POP PSW	6D	1	4	sp ← sp + 1, PSW ← M(sp)	restored
9	PUSH A	0E	1	4	M(sp) ← A , sp ← sp - 1	--------
10	PUSH X	2E	1	4	M(sp) ← X , sp ← sp - 1	
11	PUSH Y	4E	1	4	M(sp) ← Y , sp ← sp - 1	
12	PUSH PSW	6E	1	4	M(sp) ← PSW , sp ← sp - 1	
13	RET	6F	1	5	Return from subroutine sp ← sp +1, pc_L ← M(sp), sp ← sp +1, pc_H ← M(sp)	--------
14	RETI	7F	1	6	Return from interrupt sp ← sp +1, PSW ← M(sp), sp ← sp + 1, pc_L ← M(sp), sp ← sp + 1, pc_H ← M(sp)	restored
15	STOP	EF	1	3	Stop mode (halt CPU, stop oscillator)	--------

MASK ORDER & VERIFICATION SHEET

GMS81604-HC□□□

Customer should write inside thick line box.

1. Customer Information

Company Name	
Application	
Order Date	YYYY MM DD . .

Tel: Fax:

Name & Signature:

2. Device Information

Package	□ 40DIP □ 42SDIP □ 44PLCC
Mask Data	File Name: (.OTP)

Check Sum: ( )

□ Hitel

□ Chollian

□ Internet

0000H 6FFFH	Set “FF” in this area
7000H 7FFFH	ROM (4K)

(Please check mark into □)

3. Marking Specification

LGS
GMS81604-HC□□□
□□□□□□□□□□□□□□
YYWW KOREA

Customer’s part number

□□□□□□□□□□□□□□

4. Delivery Schedule

	Date	Quantity	LG Confirmation
Customer Sample	YYYY MM DD . .	pcs	
Risk Order	YYYY MM DD . .	pcs	

5. ROM Code Verification

Verification Date: YYYY MM DD . .

Please confirm our verification data.

Check Sum:

Tel: Fax:

Name & Signature:

This box is written after “5.Verification”.

Approval Date: YYYY MM DD . .

I agree with your verification data and confirm you to make mask set.

Tel: Fax:

Name & Signature:

LG Semicon

MASK ORDER & VERIFICATION SHEET

GMS81608-HC□□□

Customer should write inside thick line box.

1. Customer Information

Company Name	
Application	
Order Date	YYYY MM DD . .
Tel: Fax: Name & Signature:	

2. Device Information

Package	□ 40DIP □ 42SDIP □ 44PLCC
Mask Data	File Name: (.OTP) Check Sum: ( )

□ Hitel

□ Chollian

□ Internet

0000H 5FFFH	Set "FF" in this area
6000H 7FFFH	ROM (8K)

(Please check mark into □)

3. Marking Specification

LGS
GMS81608-HC□□□
□□□□□□□□□□□□□□
YYWW KOREA

Customer's part number

□□□□□□□□□□□□□□

4. Delivery Schedule

	Date	Quantity	LG Confirmation
Customer Sample	YYYY MM DD . .	pcs	
Risk Order	YYYY MM DD . .	pcs	

5. ROM Code Verification

Verification Date: YYYY MM DD . .
Please confirm our verification data.
Check Sum:
Tel: Fax: Name & Signature:

This box is written after "5.Verification".

Approval Date: YYYY MM DD . .
I agree with your verification data and confirm you to make mask set.
Tel: Fax: Name & Signature:

LG Semicon