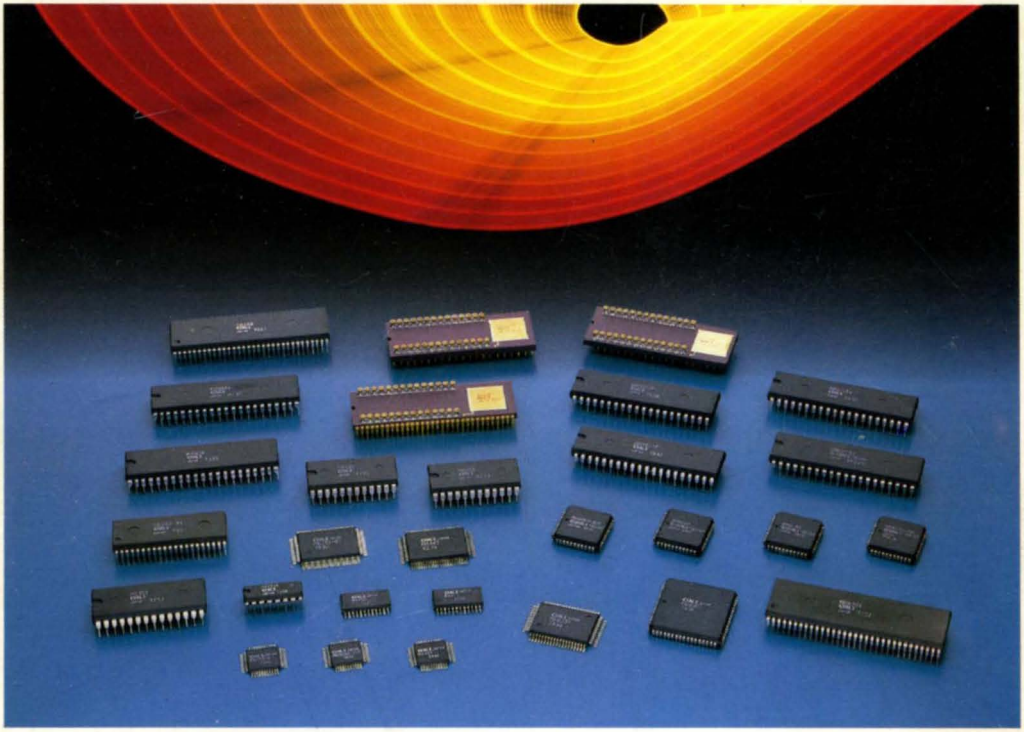
DATA BOOK

OKI

MICROCONTROLLER



THIRD EDITION
ISSUE DATE: MAR 1988

PREFACE

A high technology company with an aggresive approach to innovation, OKI has been supplying single-chip microcontrollers since 1975. OKI's single-chip microcontrollers find wide application in various types of electronic equipment in the consumer and the industrial fields. Our products have been enjoying a good reputation for their high quality and high performance. The most outstanding feature employed in all of OKI's microcontrollers is CMOS technology which ensures low power operation.

OKI will continue to enhance the its microcontroller series and program development systems to cater to cutomers' requirements.

CONTENTS

1

PROGRAM DEVELOPMENT SYSTEMS

PROGRAM DEVELOPMENT SYSTEMS FOR OKI MICROCONTROLLERS

OLMS-40 SERIES

target chip	development tool package name	standard software (included in package)	adaptor module (if necessary)	field debugging tool
MSM5840	EASE 40 (SERIAL INTERFACE) (BUS INTERFACE)	DB400 debugger ASM40 cross assembler	—	MPB202
MSM5842				
MSM58421			MPB421	
MSM58422			MPB422	
MSM5847	dedicated hardware simulator		—	—

[Note]
1. The standard software DB400 and ASM40 are avalable on CP/M-80 for most personal computers, or ISIS-II for INTEL MDS.

* CP/M is a registered trade mark of Digital Research, and ISIS-II, MDS of Intel.

LOW POWER SERIES

target chip	development tool package name	standard software	
MSM5052	EASE5052/56	EASE host monitor	ASM50 cross assember
MSM5056			
MSM5054	EASE5054/55		
MSM5055			
MSM6051	(EASE6051)		
MSM6351	(EASE6351/6353)		
MSM6353			
MSM6052	EASE6052, (EASE6352/6052)		
MSM6352	(EASE6352/6052)		

[Note]
1. The standard software is avilable under following operating system.
   CP/M-80 for most of personal computers
   MSDOS for OKI if800, NEC PC9801 etc.
   PCDOS for IBM PC-XT, AT, IBM 5550

• PROGRAM DEVELOPMENT SYSTEMS •

OLMS-64 SERIES

target chip	development tool package name	I/O adaptor module	standard software (included in package)		field debugging tool	Piggy back
MSM6404	EASE6400	–	EASE host monitor	(ASM6408)*	–	MSM6404VS
MSM6408		–			–	
MSM6402		–		ASM6400	–	
MSM6422		PAM6422			PEM6422	
MSM6411		PAM6411			PEM6411	
MSM6442		PAM6442			PEM6442	
MSC6458	EASE6458	–		ASM6458	–	MSC6458VS

[Note] The standard software is available under following operating system.
EASE, ASM6400, ASM6458 ..... CP/M-80 for most personal computers,
MSDOS for OKI if800, NEC PC9801 etc.
PCDOS for IBM PC-XT, AT, IBM 5550
* ASM6408 ..... CP/M-80 version.
ASM6400 Covers MSDOS and PCDOS for MSM6408.

OLMS-65 SERIES

target chip	development tool package name	standard software (included in package)	field debugging tool
MSM6502/6512	EASE6502	EASE65 ASM6502	MPB6502EVA

[Note] The standard software is available under CP/M-80, MSDOS, or PCDOS.

8 BIT SERIES (INTEL compatible)

target chip	development tool	standard software	optional software	piggy back
MSM80C48	EASE80C49	EASE49 ASM49	–	–
MSM80C49				
MSM80C50				
MSM80C51F	EASE80C51mKII	EASE	See Note 1.	MSM85C154VS
MSM83C154		ASM51		

[Note]
1. Optional Software for MSM80C51/83C154
PASM preprocessor, MAC51 relocatable assembler, RL51 object linker, LIB51 librarian SID51 symbolic debugger
2. The softwares are available under following operating system.
EASE49 ..... CP/M-80, ISIS-II
ASM49 ..... CP/M-80, ISIS-II
EASE & ASM51 ..... CP/M-80 for OKI if 800, NEC PC8801 etc.
MSDOS for OKI if800, NEC PC9801 etc.
PCDOS for IBM PC-XT, AT, IBM 5550
PASM, MAC51, RL51, LIB51 ..... MSDOS for OKI if80, NEC PC9801 etc.
SID51 PCDOS for IBM PC-XT, AT, IBM 5550

8 BIT SERIES (OKI original)

target chip	development tool package name	standard software (included in package)	optional software
MSM66301	EASE66301	ASM66301 EASE	See Note 1.
MSM62580	EASE62580	AS62580 EASE	

[Note]

1. Optional Software for MSM66301

c-compiler (VMS for uVAX-II)	* under development
relocatable assembler (VMS)	"
object linker and librarian (VMS)	"
cc66 debuger (VMS)	"
symbolic debugger (VMS)	"

2

LINE-UP AND TYPICAL CHARACTERISTICS

• OLMS-40 SERIES

TYPE NO.	PROCESS	POWER SUPPLY VOLTAGE	CLOCK FREQUENCY	ROM (BIT)	RAM (BIT)	INPUT PORT	OUTPUT PORT	I/O PORT	TIMER COUNTER	INTER-RUPT	STACK	INSTRUC-TION	MACHINE CYCLE	POWER CONSUMPTION		PACKAGE	REMARKS
														ACTIVE	STAND-BY		
MSM5840	CMOS	5V	4.2MHz	2048 × 8	128 × 4	6	16	8	8 Bit R/W	2	4	98	7.6μS	1.6mA	–	42DIP/44FLAT	
MSM5842	CMOS	5V	4.2MHz	768 × 8	32 × 4	5	8	8	8 Bit	–	1	52	7.6μS	1.5mA	–	28DIP/32FLAT	
MSM58421	CMOS	5V	4.2MHz	1536 × 8	40 × 4	5	35LCD Seg. 5LCD Seg. or LOGIC	8	12 Bit	–	1	52	7.6μS	2.0mA	–	60FLAT	
MSM58422	CMOS	5V	4.2MHz	1536 × 8	40 × 4	5	7 × 5 FLT Seg. 5 Discreate	8	12 Bit	–	1	52	7.6μS	2.0mA	–	60FLAT	
MSM5847	CMOS	3V	32kHz	1536 × 8	96 × 4	–	24 × 3LCD Seg.	7	13 Bit	–	2	43	610μS	50μA	–	44FLAT/CHIP	

• OLMS-50/60 SERIES

TYPE NO.	PROCESS	POWER SUPPLY VOLTAGE	CLOCK FREQUENCY	ROM (BIT)	RAM (BIT)	INPUT PORT	OUTPUT PORT	I/O PORT	TIMER COUNTER	INTER-RUPT	STACK	INSTRUC-TION	MACHINE CYCLE	POWER CONSUMPTION		PACKAGE	REMARKS
														ACTIVE	STAND-BY		
MSM5052	CMOS	1.5V	32kHz	1280 × 14	62 × 4	8	26 × 2LCD Seg. 5 LOGIC	–	–	–	–	42	122μS	3μA	–	CHIP	Built-in temper-ature detector
MSM5054	CMOS	1.5V/3V	32kHz	1024 × 14	62 × 4	6	44 × 2LCD Seg. 4 LOGIC	–	–	–	–	40	122μS	3μA	–	CHIP	
MSM5055	CMOS	1.5V/3V	32kHz	1792 × 14	96 × 4	8	60 × 2LCD Seg. 4 LOGIC	–	–	–	–	42	122μS	3μA	–	CHIP	VOICE CONTROLLER
MSM5056	CMOS	1.5V	32kHz	1792 × 14	90 × 4	4	38 × 2LCD Seg. 4 LOGIC	–	–	–	–	42	122μS	3μA	–	CHIP	Connection with solar cell available
MSM6051	CMOS	1.5V/3V	32kHz	2560 × 14	120 × 4	9	63 × 3LCD Seg. 4 LOGIC	–	–	1	2	59	91.5μS	3μA	–	CHIP	
MSM6351	CMOS	1.5V/3V	32kHz	4096 × 15	1024 × 4	–	59 × 3LCD Seg. or 58 × 4LCD Seg.	20	–	3	7	65	61.0μS	3μA	–	CHIP/100FLAT	
MSM6353	CMOS	1.5V/3V	32kHz	4096 × 15	1024 × 4	–	–	20	–	3	7		60.0μA	3μA	–	CHIP/ 42PINS-DIP	
MSM6052	CMOS	3V	3.58MHz	2048 × 14	640 × 4	12	12	4	4 Bit	1	5	52	17.9μS	1.2mA	0.2μA	28DIP/40DIP/ 44FLAT	Built-in DTMF
MSM6352	CMOS	3V	3.58MHz	2048 × 14	640 × 4	12	12	4	4 Bit	2	5	52	17.9μS	1.8mA	0.2μA	28DIP/40DIP/ 44FLAT	Built-in DTMF on-hook dialing

• OLMS-64 SERIES

TYPE NO.	PROCESS	POWER SUPPLY VOLTAGE	CLOCK FREQUENCY	ROM (BIT)	RAM (BIT)	INPUT PORT	OUTPUT PORT	I/O PORT	TIMER COUNTER	INTER-RUPT	STACK	INSTRUC-TION	MACHINE CYCLE	POWER CONSUMPTION		PACKAGE	REMARKS
														ACTIVE	STAND-BY		
MSM6404	CMOS	5V	4.2MHz	4000 × 8	256 × 4	4	–	32	12 Bit 12 Bit R/W 8 bit R/W	5	32	121	952nS	6mA	1μA	42DIP/44FLAT	
MSM6408	CMOS	5V	4.0MHz	8096 × 8	256 × 4	4	–	32	12 Bit 12 Bit R/W 8 Bit R/W	5	32	121	1μS	6mA	1μA	42DIP/44FLAT	
MSM6411	CMOS	5V	4.2MHz	1024 × 8	32 × 4	4	–	8	–	2	8	63	952nS	6mA	1μA	16DIP/24FLAT	
MSM6422	CMOS	5V	4.2MHz	2048 × 8	64 × 4	1	–	18	12 Bit	2	16	63	952nS	6mA	1μA	16DIP/24FLAT	
MSM6442	CMOS	5V	4.2MHz	2048 × 8	128 × 4	1	46 × 2LCD Seg.	16	8 Bit R/W	4	16	76	952nS	6mA	1μA	80FLAT	Built-in LCD Controller/Driver
MSC6458	Bi-CMOS	5V	4.3MHz	8192 × 8	512 × 4	9	12 × 12FLT Seg.	24	12 Bit 16 Bit R/W 8 Bit R/W	8	32	147	930nS	9mA	1μA	64 Shrink DIP/ 64 FLAT	Built-in FLT Controller/Driver

• OLMS-65 SERIES

TYPE NO.	PROCESS	POWER SUPPLY VOLTAGE	CLOCK FREQUENCY	ROM (BIT)	RAM (BIT)	INPUT PORT	OUTPUT PORT	I/O PORT	TIMER COUNTER	INTER-RUPT	STACK	INSTRUC-TION	MACHINE CYCLE	POWER CONSUMPTION		PACKAGE	REMARKS
														ACTIVE	STAND-BY		
MSM6502	CMOS	3V	32kHz	2000 × 8	128 × 4	4	108LCD Seg.	8	12 Bit	3	32	68	91.5μS	45μA	30μA	44FLAT/CHIP	
MSM6512	CMOS	3V	32kHz	2000 × 8	128 × 4	4	108LCD Seg.	8	12 Bit	3	32	68	91.5μS	30μA	12μA	44FLAT/CHIP	

• 8 Bit SERIES (INTEL compatible)

TYPE NO.	PROCESS	POWER SUPPLY VOLTAGE	CLOCK FREQUENCY	ROM (BIT)	RAM (BIT)	INPUT PORT	OUTPUT PORT	I/O PORT	TIMER COUNTER	INTER-RUPT	STACK	INSTRUC-TION	MACHINE CYCLE	POWER CONSUMPTION		PACKAGE	REMARKS
														ACTIVE	STAND-BY		
MSM80C35	CMOS	5V	11MHz	—	64 × 8	—		24	8 Bit R/W	2	8	111	1.36μS	10mA	1μA	40DIP/44FLAT	
MSM80C39	CMOS	5V	11MHz	—	128 × 8	—		24	8 Bit R/W	2	8	111	1.36μS	10mA	1μA	40DIP/44FLAT	
MSM80C40	CMOS	5V	6MHz	—	256 × 8	—		24	8 Bit R/W	2	8	111	2.5μS	10mA	1μA	40DIP/44FLAT	
MSM80C31F	CMOS	5V	16MHz	—	128 × 8	—		32	16 Bit R/W × 2	5	64	111	0.75μS	20mA	1μA	40DIP/44FLAT/ 44PLCC	
MSM80C154	CMOS	5V	16MHz	—	256 × 8	—		32	16 Bit R/W × 3	6	128	111	0.75μS	20mA	1μA	40DIP/44FALT/ 44PLCC	
MSM80C48	CMOS	5V	11MHz	1024 × 8	64 × 8	—		24	8 Bit R/W	2	8	111	1.36μS	10mA	1μA	40DIP/44FLAT	
MSM80C49	CMOS	5V	11MHz	2048 × 8	128 × 8	—		24	8 Bit R/W	2	8	111	1.36μS	10mA	1μA	40DIP/44FLAT	
MSM80C50	CMOS	5V	6MHz	4096 × 8	256 × 8	—		24	8 Bit R/W	2	8	111	2.5μS	10mA	1μA	40DIP/44FLAT	
MSM80C51F	CMOS	5V	16MHz	4096 × 8	128 × 8	—		32	16 Bit R/W × 2	5	64	111	0.75μS	20mA	1μA	40DIP/44FLAT/ 44PLCC	
MSM83C154	CMOS	5V	16MHz	16384 × 8	256 × 8	—		32	16 Bit R/W × 3	6	128	111	0.75μS	20mA	1μA	40DIP/44FLAT/ 44PLCC	

• 8 BIT SERIES (OKI original)

TYPE NO.	PROCESS	POWER SUPPLY VOLTAGE	CLOCK FREQUENCY	ROM (BIT)	RAM (BIT)	INPUT PORT	OUTPUT PORT	I/O PORT	TIMER COUNTER	INTER-RUPT	STACK	INSTRUC-TION	MACHINE CYCLE	POWER CONSUMPTION		PACKAGE	REMARKS
														ACTIVE	STAND-BY		
MSM62580	CMOS	5V	5MHz	3072 × 8	128 × 8	—		1	—	—	32	95	800nS	4mA	10μA	C.O.B.	For IC cards under develop-ment
MSM66301	CMOS	5V	10MHz	16384 × 8	512 × 8	8		40	16 Bit × 4	17	256	99	400nS	—	—	64 Shrink DIP/ 68 PLCC/ 64 FLAT	Under develop-ment (PGB type) • A/DC 8ch, 10bit • UART • PWM • Chapter register

2

LOW POWER

HIGH SPEED

OLM-50/60 SERIES

TYPE NO.	ROM	RAM	POWER CONSUMP-TION	FEATURES
MSM5052	1280 x 14	62 x 4	3μA	52 Seg. LCD Driver
MSM5054	1024 x 14	62 x 4	3μA	88 Seg. LCD Driver
MSM5055	1792 x 14	96 x 4	3μA	120 Seg. LCD Driver
MSM5056	1792 x 14	90 x 4	3μA	76 Seg. LCD Driver
MSM6051	2560 x 14	120 x 4	3μA	189 Seg. LCD Driver
MSM6052 6352	2048 x 14	640 x 4	1.2mA 1.8mA	DTMF Generator
MSM6351	4096 x 15	1024 x 4	3μA	232 Seg. LCD Driver
MSM6353	4096 x 15	1024 x 4	3μA	Serial board (Synchronised/ Non synchronised)

OLM-64 SERIES

TYPE NO.	ROM	RAM	MACHINE CYCLE	FEATURES
MSM6404	4000 x 8	256 x 4	952nS	I/O; 36
MSM6408	8096 x 8	256 x 4	952nS	I/O; 36
MSM6411	1024 x 8	32 x 4	952nS	I/O; 11
MSM6422	2048 x 8	64 x 4	952nS	I/O; 19
MSM6442	2048 x 8	128 x 4	952nS	92 Seg. LCD Driver
MSM6458	8192 x 8	512 x 4	930nS	144 Seg. FLT Driver

OLM-40/65 SERIES

TYPE NO	ROM	RAM	MACHINE CYCLE	FEATURES
MSM5840	2048 x 8	128 x 4	7.6μS	I/O; 30
MSM5842	768 x 8	32 x 4	7.6μS	I/O; 21
MSM58421	1536 x 8	40 x 4	7.6μS	35 Seg. LCD Driver
MSM58422	1536 x 8	40 x 4	7.6μS	35 Seg. FLT Driver
MSM5847	1536 x 8	96 x 4	600μS	72 Seg. LCD Driver
MSM6502	2000 x 8	128 x 4	91.5μS	108 Seg. LCD Driver
MSM6512	2000 x 8	128 x 4	91.5μS	104 Seg. LCD Driver

IC CARD

ONE CHIP MICROCOMPUTER FOR IC CARD

TYPE NO.	ROM	RAM	MACHINE CYCLE	FEATURES
MSM62580	3072 x 8	128 x 8	800nS	Built-in EEPROM 2048 x 8

4BIT

8BIT

OKI ONE CHI

HIGH PERFORMANCE

HIGH PERFORMANCE nX SERIES

MSM80C154 SERIES

TYPE NO.	ROM	RAM	MACHINE CYCLE
MSM83C154	16384 x 8	256 x 8	750nS
MSM80C154	—	256 x 8	750nS

OLMS-66K SERIES

TYPE NO.	ROM	RAM	MACHINE CYCLE
MSM66301*	16384 x 8	512 x 8	400nS

16BIT

MSM80C51 SERIES

TYPE NO.	ROM	RAM	MACHINE CYCLE
MSM80C51/51F	4096 x 8	128 x 8	0.75μS
MSM80C31/31F	—	128 x 8	0.75μS

MSM80C48 SERIES

TYPE NO.	ROM	RAM	MACHINE CYCLE
MSM80C48	1024 x 8	64 x 8	1.36μS
MSM80C49	2048 x 8	128 x 8	1.36μS
MSM80C50	4095 x 8	256 x 8	2.5μS
MSM80C35	—	64 x 8	1.36μS
MSM80C39	—	128 x 8	1.36μS
MSM80C40	—	256 x 8	2.5μS

8BIT

IICROCOMPUTER

3

CODE ENTRY

CODE ENTRY

The program code ENTERING method is outlined below.

1. USABLE MEDIA

(1) 2 pieces of same type EPROMs containing identical DATA

EPROM specification

- 2716
- 2732
- 27C32
- 27C32A
- 2764
- 27C64
- 27128
- 27256

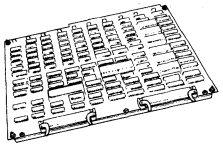
(2) 1 copy of object machine code list

2. SINGLE CHIP MICROCONTROLLER DEVELOPMENT STAGES

USER 1

● Program with OKI development tool.

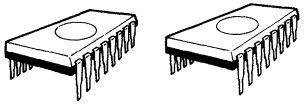
if 800



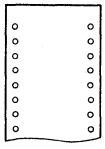
TOOL

USER 2

● Prepare 2 pcs of EPROM and programming list



EPROM



Printout

OKI 3

● Print out identical EPROM data.

OKI 4

● Engineering Sample
Actual production sample of your microcontroller chip prepared for final approval.

OKI 5

● Volume production of single chip microcontrollers.

USER 6

START
PRODUCT/SYSTEM
MANUFACTURE

4

PACKAGING

PACKAGING

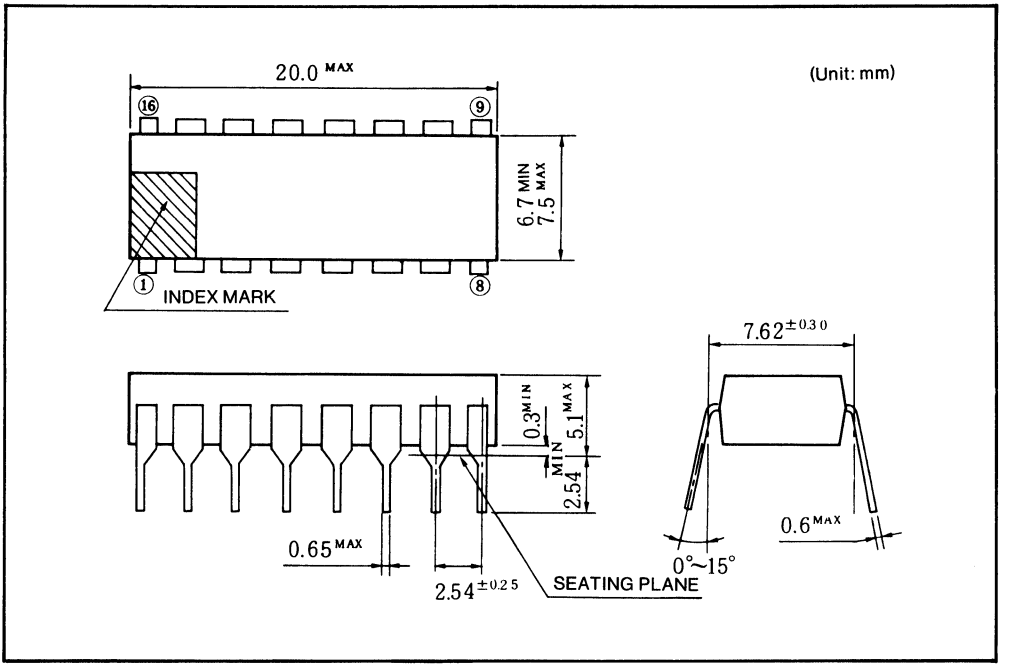
	PACKAGE/PIN COUNT		
	DIP	FLAT	PLCC
MSM5840	42 PIN	44 PIN	–
MSM5842	28 PIN	32 PIN	–
MSM58421	–	60 PIN	–
MSM58422	–	60 PIN	–
MSM5847 *	–	44 PIN	–
MSM5052 *	–	56 PIN	–
MSM5054 *	–	56 PIN	–
MSM5055 *	–	80 PIN	–
MSM5056 *	–	–	–
MSM6051 *	–	–	–
MSM6351 *	–	100 PIN	–
MSM6052	28/40 PIN	44 PIN	–
MSM6352	28/40 PIN	44 PIN	–
MSM6404	42 PIN	44 PIN	44 PIN
MSM6404VS	42 PIN PIGGY BACK	–	–
MSM6408	42 PIN	44 PIN	44 PIN
MSM6411	16 PIN	24 PIN	–
MSM6422	24 PIN	24 PIN	–
MSM6442	–	80 PIN	–
MSC6458	64 PIN SHRINK	64 PIN	68 PIN
MSC6458VS	64 PIN SHRINK PIGGY BACK	–	–

NOTE: * CHIP TYPE is available.

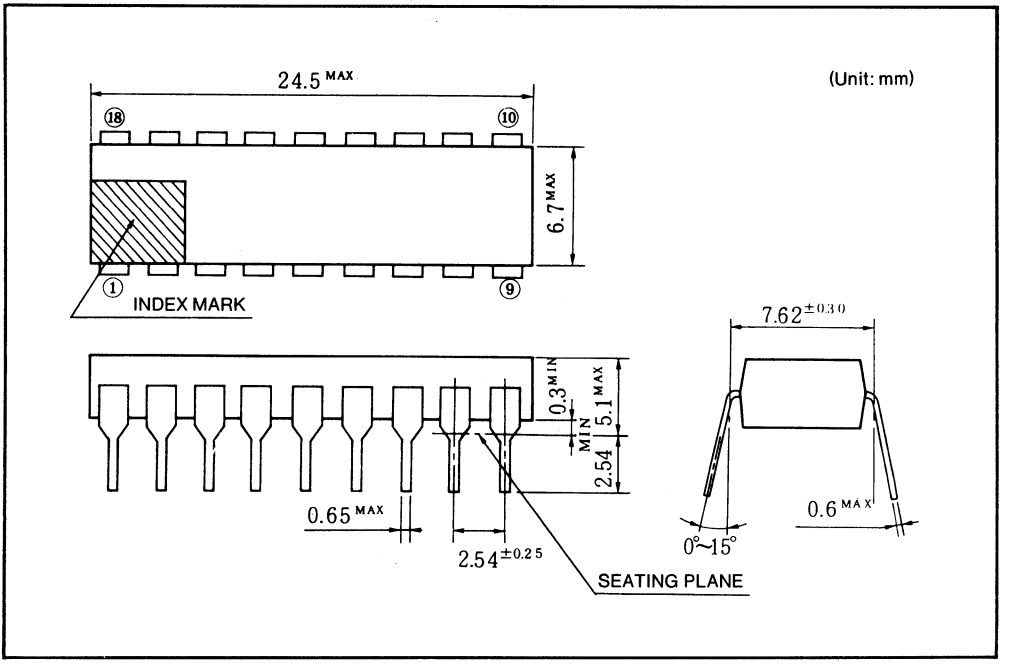
	PACKAGE/PIN COUNT		
	DIP	FLAT	PLCC
MSM6502*	–	56 PIN(S)	–
MSM6512*	–	56 PIN(S)	–
MSM62580	–	COB (5PIN)	–
MSM66301	64 PIN SHRINK	64 PIN	68 PIN
MSM80C35	40 PIN	44 PIN	44 PIN
MSM80C39	40 PIN	44 PIN	44 PIN
MSM80C40	40 PIN	44 PIN	44 PIN
MSM80C31	40 PIN	44 PIN	44 PIN
MSM80C48	40 PIN	44 PIN	44 PIN
MSM80C49	40 PIN	44 PIN	44 PIN
MSM80C50	40 PIN	44 PIN	44 PIN
MSM80C31F	40 PIN	44 PIN	44 PIN
MSM80C51FVS	40 PIN PIGGY BACK	–	–
MSM80C51F	40 PIN	44 PIN	44 PIN
MSM80C154	40 PIN	44 PIN	44 PIN
MSM83C154	40 PIN	44 PIN	44 PIN
MSM85C154VS	40 PIN PIGGY BACK	–	–

NOTE: (S) means Small pachage

● 16 PIN PLASTIC DIP

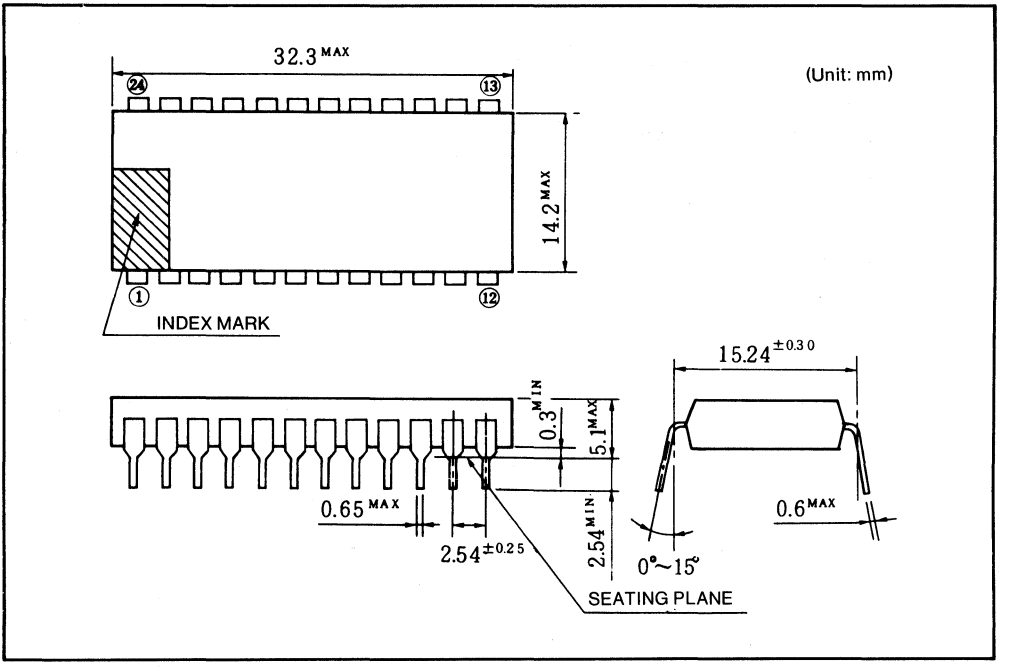


● 18 PIN PLASTIC DIP

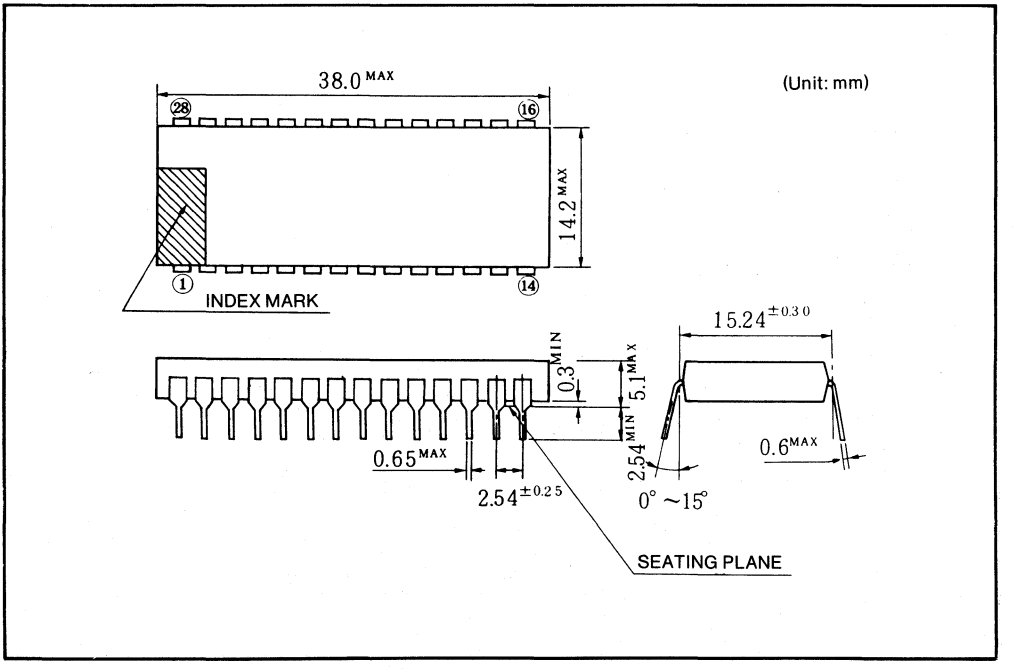


• PACKAGING •

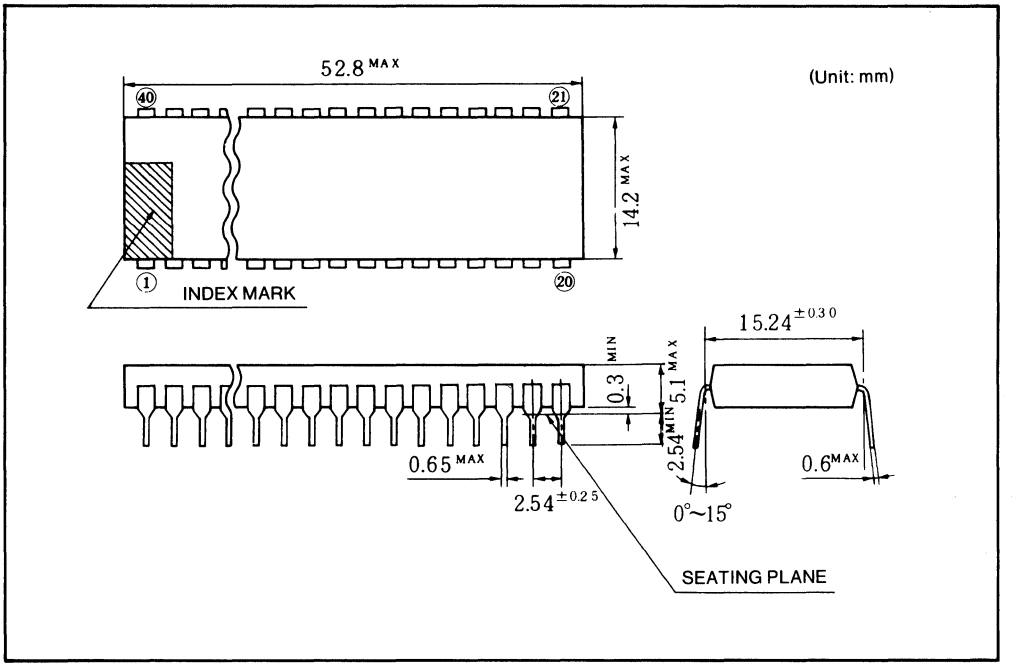
• 24 PIN PLASTIC DIP



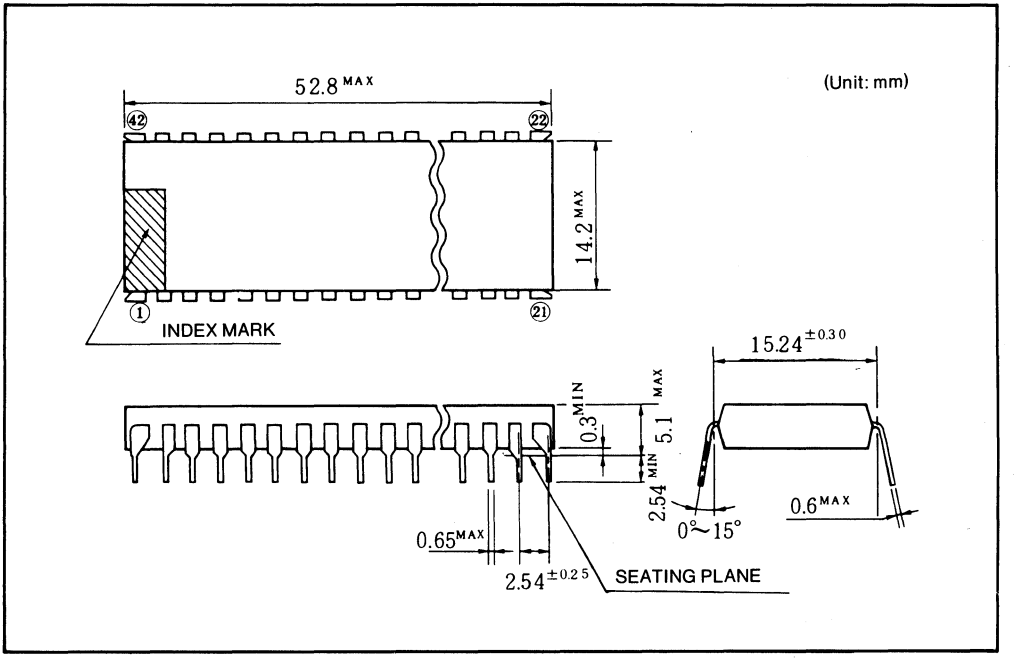
• 28 PIN PLASTIC DIP



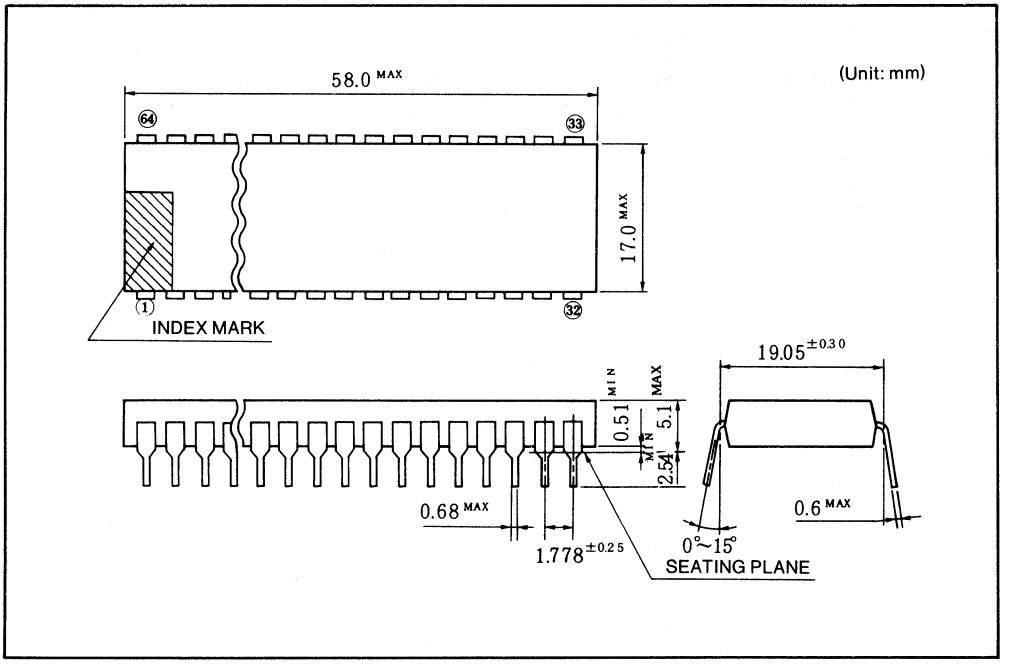
● 40 PIN PLASTIC DIP



● 42 PIN PLASTIC DIP

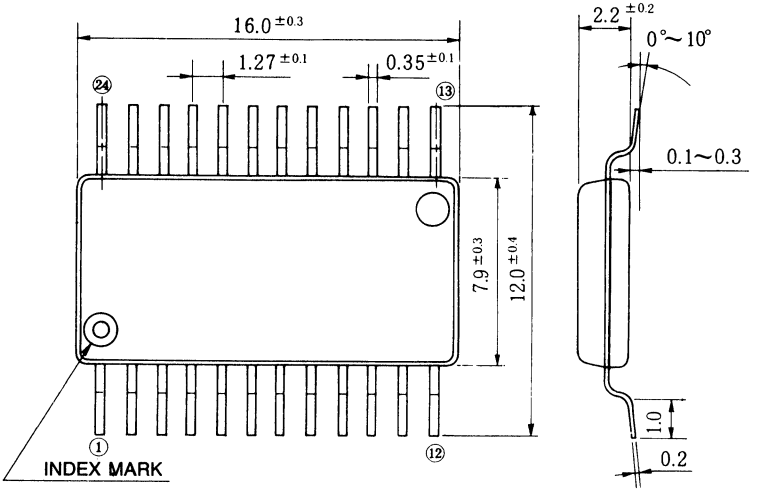


• 64 PIN SHRINK DIP



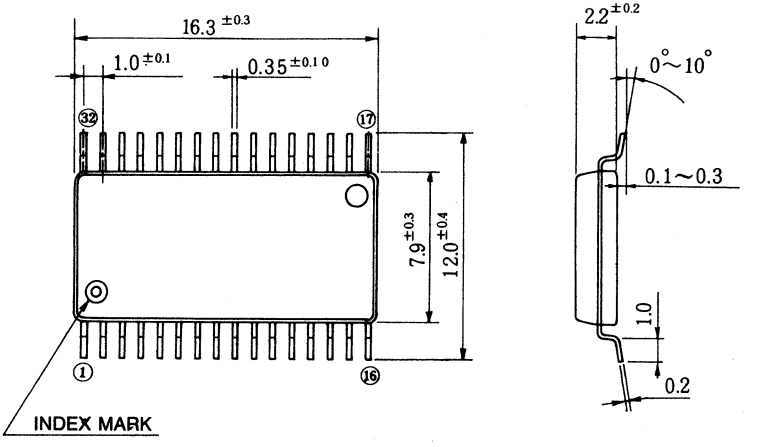
● 24 PIN PLASTIC FLAT

(Unit: mm)



● 32 PIN PLASTIC FLAT

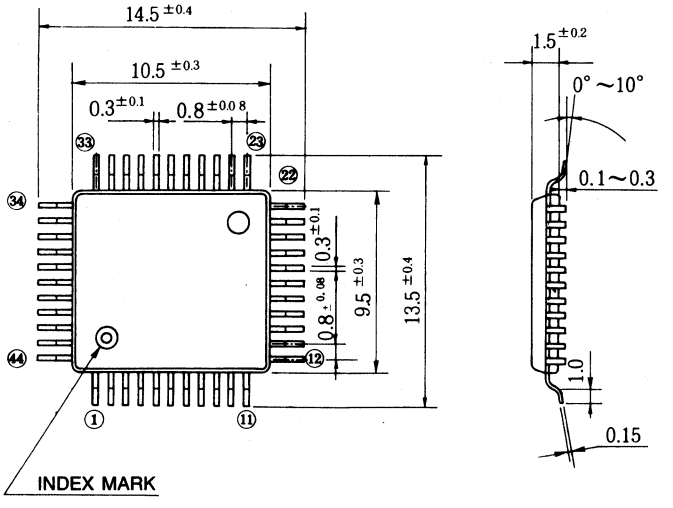
(Unit: mm)



• PACKAGING •

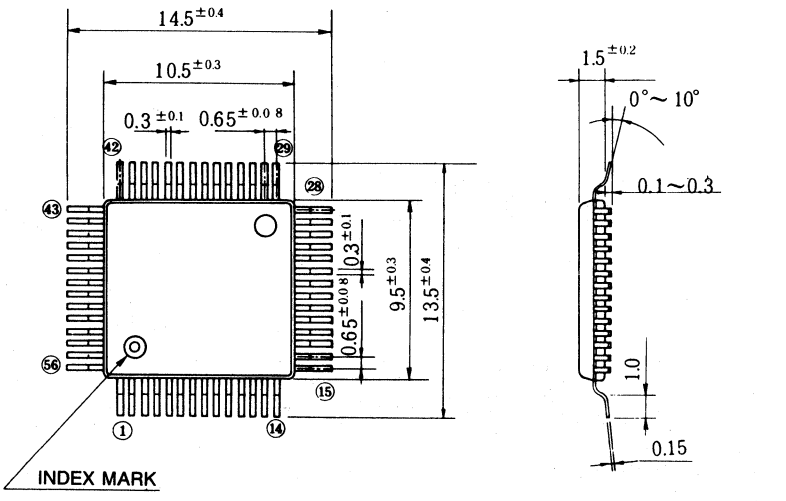
• 44 PIN PLASTIC FLAT

(Unit: mm)

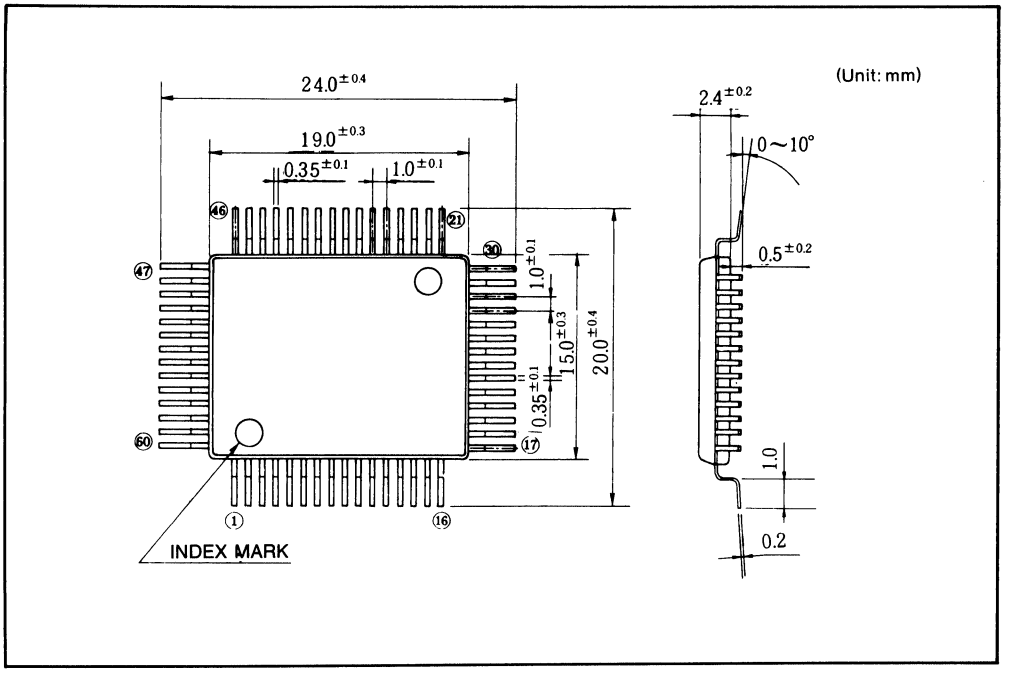


• 56 PIN PLASTIC FLAT

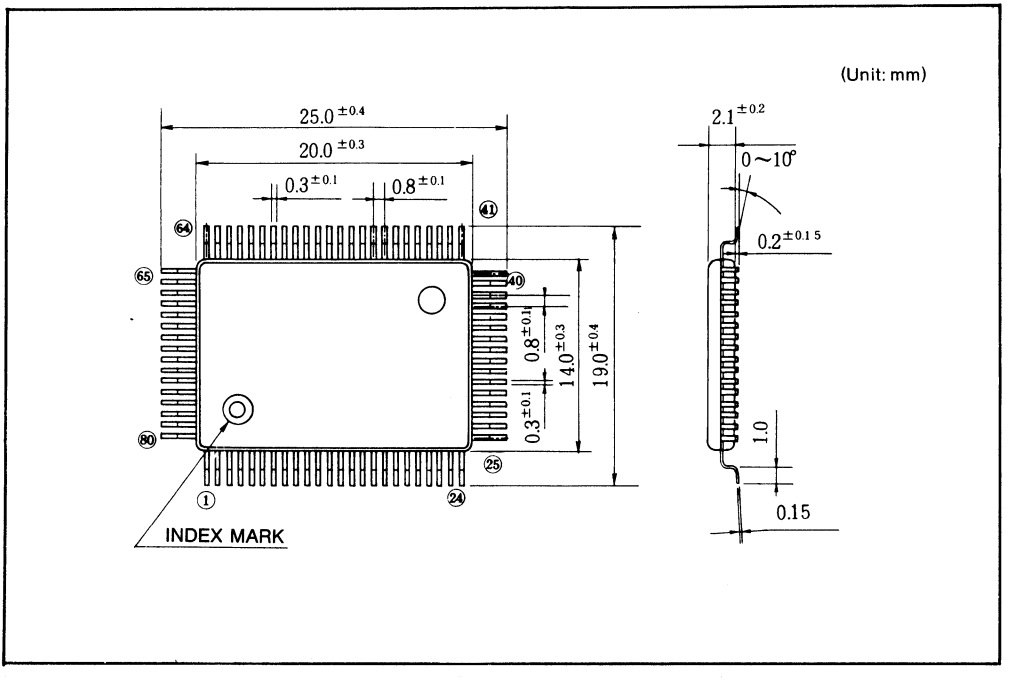
(Unit: mm)



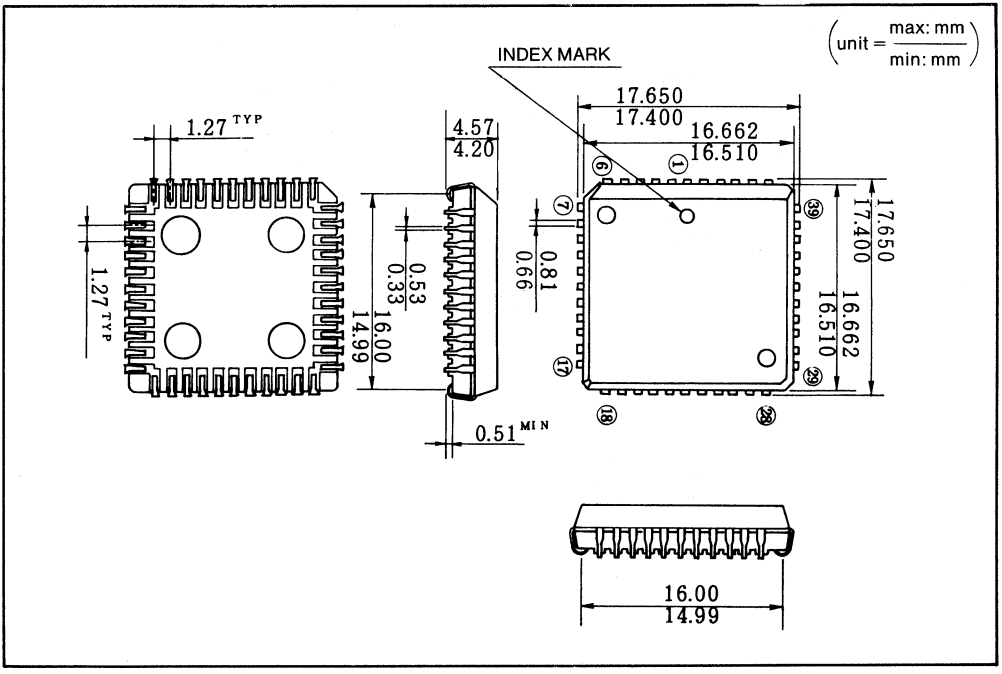
● 60 PIN PLASTIC FLAT



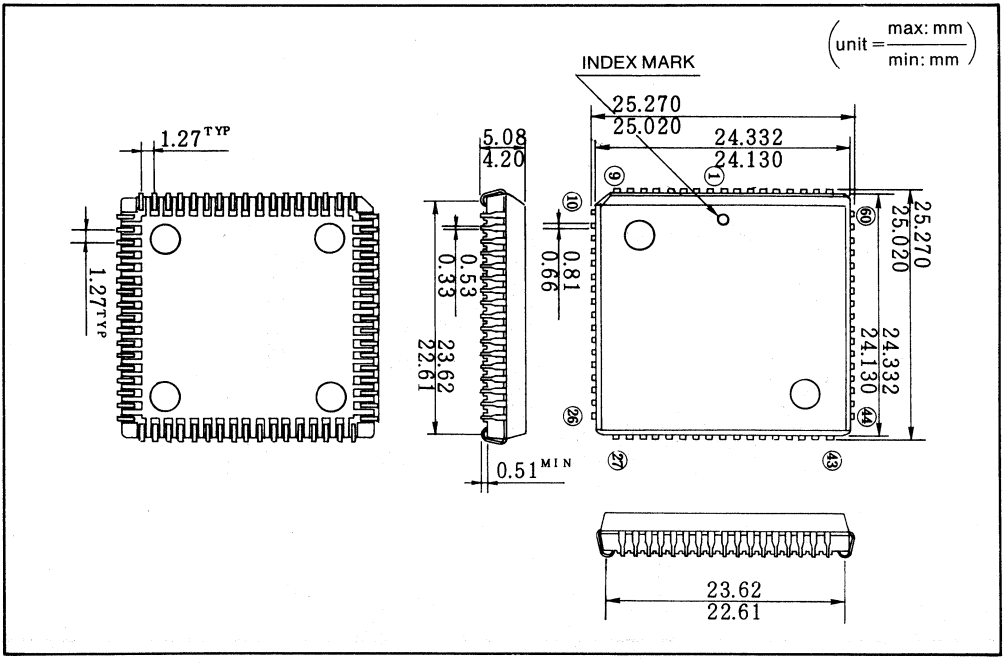
● 80 PIN PLASTIC FLAT



• 44 PIN PLASTIC PLCC

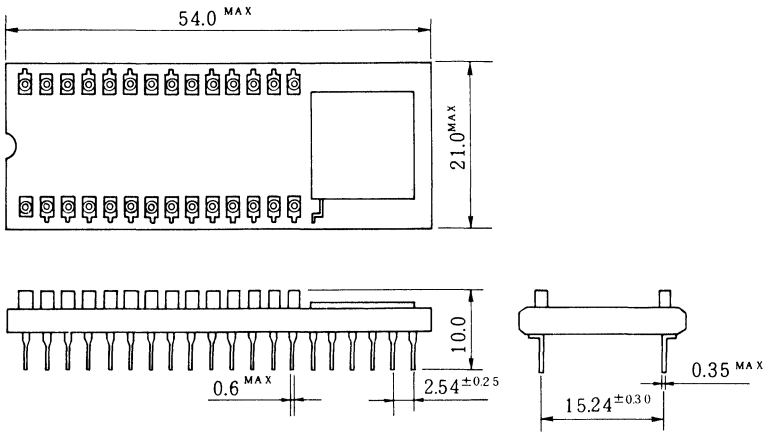


• 68 PIN PLASTIC PLCC



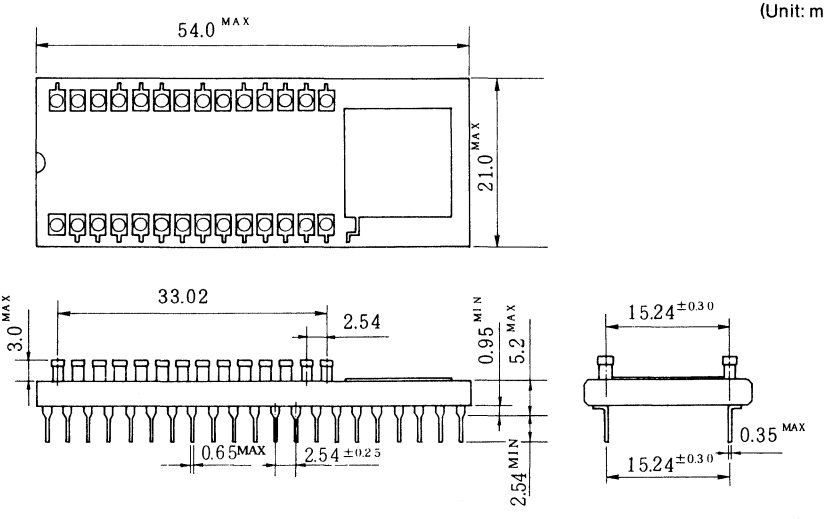
• 40 PIN PIGGY BACK

(Unit: mm)

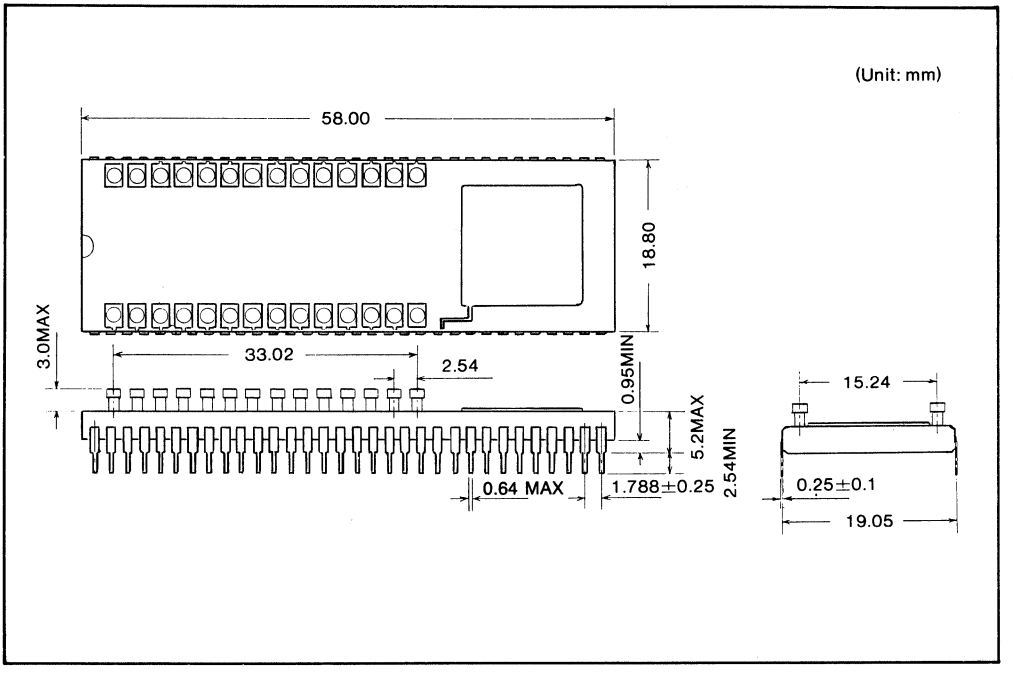


• 42 PIN PIGGY BACK

(Unit: mm)



• 64 PIN SHRINK PIGGY BACK



5

RELIABILITY INFORMATION

RELIABILITY INFORMATION

1. INTRODUCTION

Semiconductor devices play a leading role in the explosive progress of technology. They use some of the most advanced design and manufacturing technology developed to date. With greater integration, diversity and reliability, their applications have expanded enormously. Their use in large scale computers, control equipment, calculators, electronic games and in many other fields has increased at a fast rate.

A failure in electronic banking or telephone switching equipment, for example, could have far reaching effects and can cause incalculable losses. So, the demand, for stable, high quality memory devices is strong.

We, at Oki are fully aware of this demand. So we have adopted a comprehensive quality assurance system based on the concept of consistency in development, manufacturing and sales.

With the increasing demand for improvement in function, capability and reliability, we will expand our efforts in the future. Our quality assurance system and the underlying concepts are outlined briefy below.

2. QUALITY ASSURANCE SYSTEM AND UNDERLYING CONCEPTS

The quality assurance system employed by Oki can be divided into four major stages: device planning, developmental prototype, production prototype, and mass production. This system is outlined in the following block diagram (Fig. 1).

1) Device planning stage

To manufacture devices that meet market demands and satisfy customer needs, we carefully consider functional and failure rate requirements, utilization form, environment and other conditions. Once we determine the proper type, material and structure, we check the design and manufacturing techniques, and the line processing capacity. Then we prepare the development planning and time schedule.

2) Developmental prototype stage

We determine circuits, pattern design, process settings, assembly techniques and structural requirements during this stage. At the same time, we carry out actual prototype reliability testing.

Since device quality is largely determined during the designing stage, Oki pays careful attention to quality confirmation during this stage.

This is how we do it:

(1) After completion of circuit design (or pattern design), personnel from the design, process technology, production technology, installation technology and reliability departments get together for a thorough review to ensure design quality and to anticipate problems that may occur during mass production. Past experience and know-how guide these discussions.

(2) Since many semiconductor memories involve new concepts and employ high level manufacturing technology, the TEG evaluation test is often used during this stage.

Note: TEG (Test Element Group) refers to the device group designed for stability evaluation of MOS transistors, diodes, resistors, capacitors and other circuit component element used in LSI memories.

(3) Prototypes are subjected to repeated reliability and other special evaluation tests. In addition, the stability and capacity of the manufacturing process are checked.

3) Production prototype stage

During this stage, various tests check the reliability and other special features of the production prototype at the mass production factory level. After confirming the quality of a device, we prepare the various standards required for mass production, and then start production. Although reliability and other special tests performed on the production prototype are much the same as those performed on the developmental prototype, the personnel, facilities and production site differ for the two prototypes, necessitating repeated confirmation tests.

4) Mass production

During the mass production stage, careful management of purchased materials, parts and facilities used during the manufacturing process, measuring equipment, manufacturing conditions and environment is necessary to ensure device quality first stipulated during the designing stages. The manufacturing process (including inspection of the completed device) is followed by a lot guarantee inspection to check that the specified quality is maintained under conditions identical to those under which a customer would actually use the device. This lot guarantee inspection is performed in three different forms as shown below.

(1) Group A tests: appearance, labels, dimensions and electrical characteristics inspection

(2) Group B tests: check of durability under thermal and mechanical environmental stresses, and operating life characteristics

(3) Group C tests: performed periodically to check operational life, etc., on a long term basis.

Note: Like the reliability tests, the group B tests conform to the following standards.
MIL-STD-883B, JIS C 7022, EIAJ-IC-121

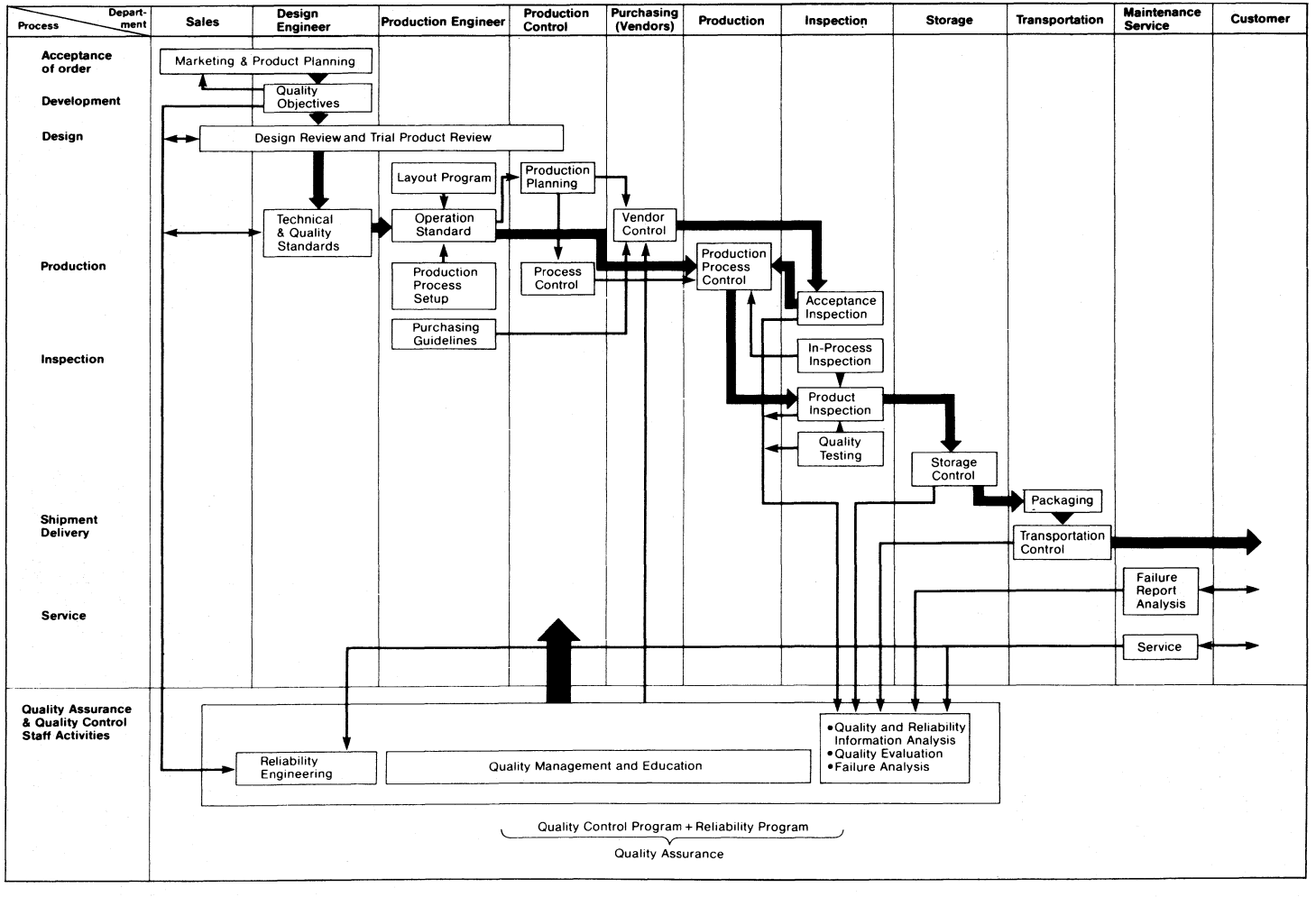


Figure 1 Quality Assurance System

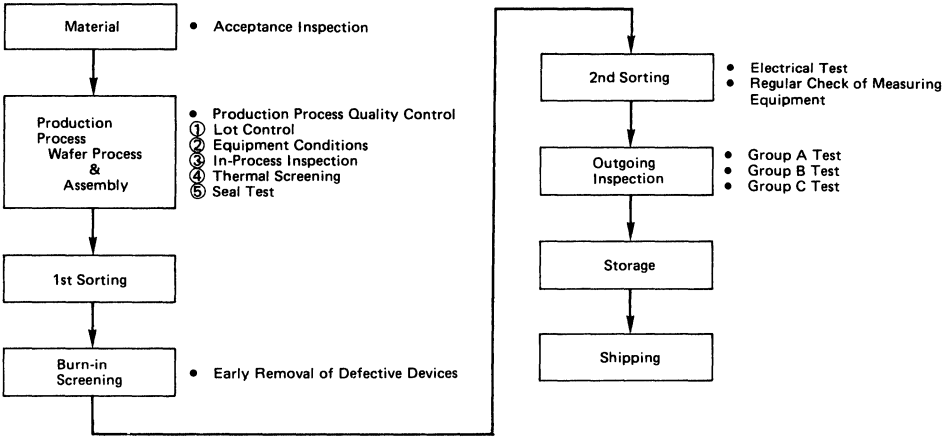


Figure 2 Manufacturing Process

Devices which pass these lot guarantee inspections are stored in a warehouse awaiting shipment to customers. Standards are also set up for handling, storage and transportation during this period, thereby ensuring quality prior to delivery. Figure 2 shows the manufacturing flow of the completed device.

5) At Oki, all devices are subjected to thorough quality checks. If, by chance, a failure does occur after delivery to the customer, defective devices are processed and the problem rectified immediately to minimize the inconvenience to the customer in accordance with the following flowchart.

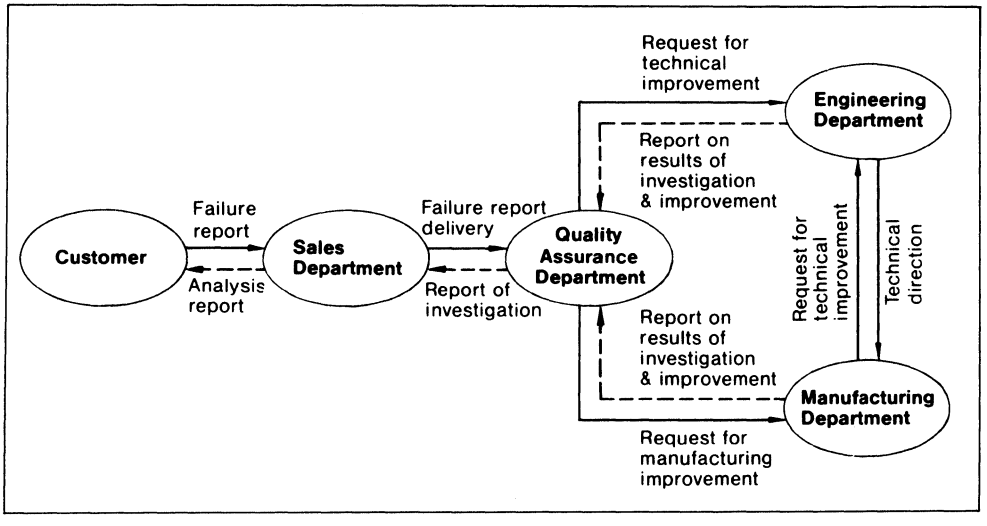
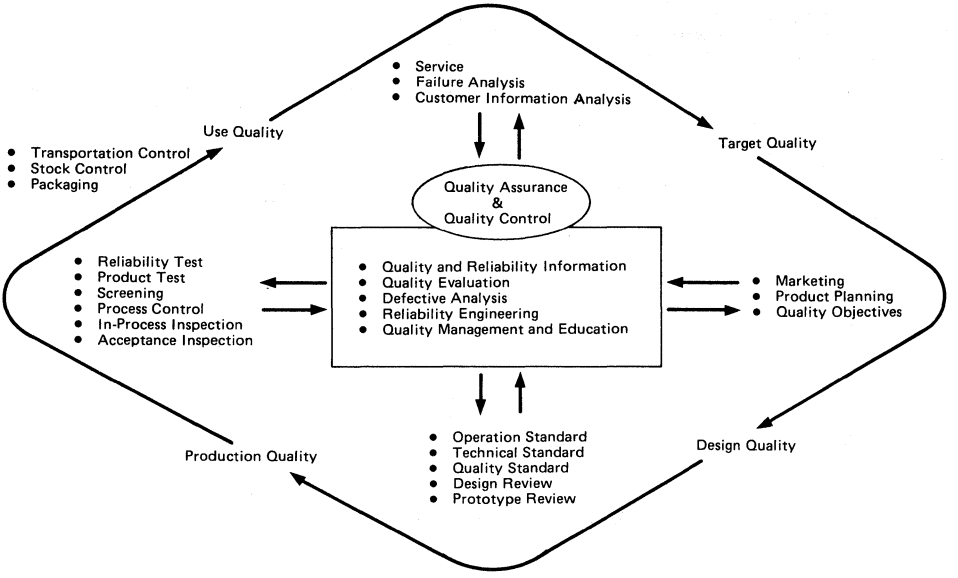


Figure 3 Failure report process



5

3. EXAMPLE OF RELIABILITY TEST RESULTS

We have outlined the quality assurance system and the underlying concepts employed by Oki. Now, we will give a few examples of the reliability tests performed during the developmental and production prototype stages. All reliability tests performed by Oki conform to the following standards.

MIL-STD-883B, JIS C 7022, EIAJ-IC-121

Since these reliability tests must determine performance under actual working conditions in a short period of time, they are performed under severe test conditions. For example, the 125°C high temperature continuous operation test performed for 1000 hours is equivalent to testing device life from 2 to 300 years of use at Ta = 40°C.

By repeating these accelerated reliability tests, device quality is checked and defects analyzed. The resulting information is extremely useful in improving the manufacturing processes. Some of the more common defects in LSI elements and their analysis are described on next page.

MICROCONTROLLER LIFE TEST RESULTS

Test item	Test condition	Part name: MSM80C31/51-XXRS Function: 8 BIT ONE CHIP MICROCONTROLLER Sample size (pcs)	Test hours	Failures	MSM80C35/39/48/49-XXRS 8 BIT ONE CHIP MICROCONTROLLER Sample size (pcs)	Test hours	Failures	MSM83C154-XXRS 8 BIT ONE CHIP MICROCONTROLLER Sample size (pcs)	Test hours	Failures	Referred standard
Operating life test	Ta = 125°C Vcc = 6V	88	2000 (H)	0	88	2000 (H)	0	88	2000 (H)	0	MIL-STD-883C METHOD 1005
Temperature humidity test	Ta = 85°C RH = 85% Vcc = 6V	100	2000 (H)	0	100	2000 (H)	0	100	2000 (H)	0	
Temperature cycling test	−55°C⇄RT⇄150°C (30 min) ↑ (30 min) (5 min)	100	500 (cy)	0	100	300 (cy)	0	100	500 (cy)	0	MIL-STD-883C METHOD 1010
Pressure cooker test	Ta = 121°C RH = 100% 2 atm	50	200 (H)	0	50	200 (H)	0	50	200 (H)	0	

5

Test item	Test condition	Part name: MSM6404-XXJS Function: 4 BIT ONE CHIP MICROCONTROLLER Sample size (pcs)	Test hours	Failures	MSM80C31JS 8 BIT ONE CHIP MICROCONTROLLER Sample size (pcs)	Test hours	Failures	Referred standard
Operating life test	Ta = 125°C Vcc = 6V	88	2000 (H)	0	88	2000 (H)	0	MIL-STD-883C METHOD 1005
Temperature humidity test	Ta = 85°C RH = 85% Vcc = 6V	100	2000 (H)	0	100	2000 (H)	0	
Temperature cycling test	−55°C⇄RT⇄150°C (30 min) ↑ (30 min) (5 min)	100	500 (cy)	0	100	500 (cy)	0	MIL-STD-883C MEJTHOD 1010
Pressure cooker test	Ta = 121°C RH = 100% 2 atm	50	200 (H)	0	50	200 (H)	0	

MICROCONTROLLER ENVIRONMENTAL TEST RESULTS

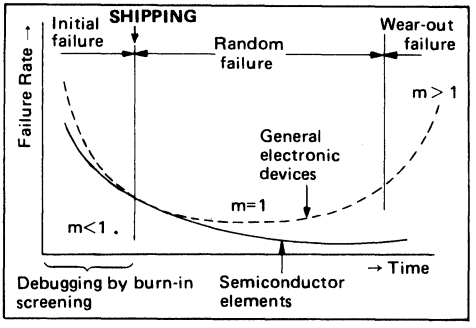
Part name			MSM80C31/51-XXRS		MSM80C35/39/48/49-XXRS		MSM83C154-XXRS		Referred standard
Function			8 BIT ONE CHIP MICROCONTROLLER		8 BIT ONE CHIP MICROCONTROLLER		8 BIT ONE CHIP MICROCONTROLLER		
	Test item	Test condition	Sample size (pcs)	Failures	Sample size (pcs)	Failures	Sample size (pcs)	Failures	
	Soldering Heat Test	260°C 10 SEC	22	0	22	0	22	0	MIL-STD-883C METHOD 2003
	Temperature Cycling Test	−55°C⇄RT⇄150°C (30min) (5min) (30min) 20 cycles	22	0	22	0	22	0	MIL-STD-883C METHOD 1010
	Thermal Shock Test	100°C⇄0°C (5min) (5min) 10 cycles	22	0	22	0	22	0	MIL-STD-883C METHOD 1011
	Lead Integrity — Tensile	500 g 10 SEC	11	0	11	0	11	0	MIL-STD883C METHOD 2004
	Lead Integrity — Bending	250 g 90° BEND 3 TIMES							
	Solderability	230°C 5 SEC	22	0	22	0	22	0	MIL STD883C METHOD 2003

Part name			MSM6404 XXJS		MSM80C31JS		Referred standard
Function			4 BIT ONE CHIP MICROCONTROLLER		8 BIT ONE CHIP MICROCONTROLLER		
	Test item	Test condition	Sample size (pcs)	Failures	Sample size (pcs)	Failures	
Thermal Environmental Test	PRE-Bake	Bake (125°C, 6 hrs)	22	0	22	0	
	Soldering Heat Test	Vapor Phase Reflow (215±2°C, 90+10, −0sec) 2 times					
	Temperature Cycling Test	−55°C⇄RT⇄150°C (30min) (5min) (30min) 20 cycles					MIL-STD-883C METHOD 1010
	Thermal Shock Test	100°C⇄0°c (5min) (5min) 10 cycles					MIL-STD-883C METHOD 1011
Other Test	Solderability	○ Bake (125°C, 24hrs) ○ Immerse into Flux ○ Immerse into Solder (215±2°C 10±1sec)	22	0	22	0	

4. SEMICONDUCTOR MEMORY FAILURES

The life-span characteristics of semiconductor elements in general (not only semiconductor IC devices) are described by the curve shown in the diagram below. Although semiconductor memory failures are similar to those of ordinary integrated circuits, the degree of integration (miniaturization), manufacturing complexity and other circuit element factors influence their incidence.

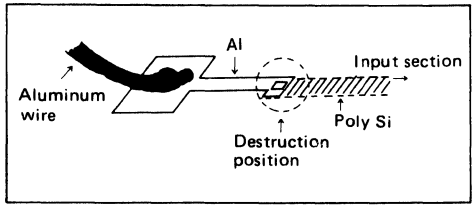
Semiconductor Element Failure Rate Curve



1) Surge Destruction

This is destruction of the input/output stage circuits by external surge currents or static electricity. The accompanying photograph shows a point of contact between aluminum and polysilicon that has been dissolved by a surge current. A hole has formed in the substrate silicon, leading to a short circuit. This kind of failure is traceable in about 30% of defective devices returned to the manufacturer. Despite miniaturization of semiconductor memory component elements (which means the elements themselves are less resistant), these failures usually occur during assembly and other handling operations.

At Oki, all devices are subjected to static electricity intensity tests (under simulated operational conditions) in the development stage to reduce this type of failure. In addition to checking endurance against surge currents, special protective circuits are incorporated in the input and output sections.



2) Oxide Film Insulation Destruction (Pin Holes)

Unlike surge destruction, this kind of failure is caused by manufacturing defects. Locally weakened sections are ruptured when subjected to external electrical stress. Although this problem is accentuated by the miniaturization of circuit elements, it can be resolved by maintaining an ultra-clean manufacturing environment and through 100% burn-in screening.

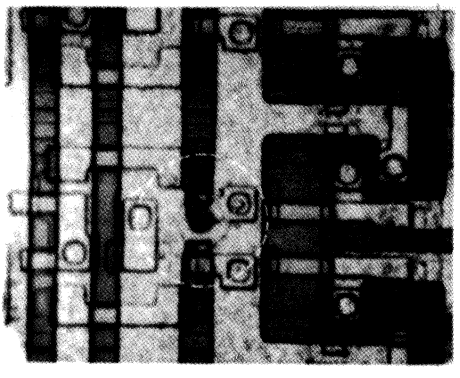
3) Surface Deterioration due to Ionic Impurities

Under some temperature and electric field conditions, charged ionic impurities moving within the oxide film previously resulted in occasional deterioration of silicon surfaces. This problem has been eliminated by new surface stabilization techniques.

4) Photolithographic Defects

Integrated circuits are formed by repeated photographic etching processes. Dust and scratches on the mask (which corresponds to a photographic negative) can cause catastrophic defects. At present, component elements have been reduced in size to the order of 10 cm through miniaturization. However, the size of dust and scratches stays the same. At Oki, a high degree of automation, minimizing human intervention in the process, and unparalleled cleanliness, solves this problem.

Example of surge destruction



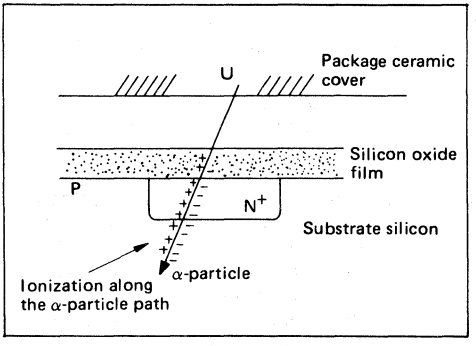
Photolithographic Defect

5) Aluminum Corrosion

Aluminum corrosion is due to electrolytic reactions caused by the presence of water and minute impurities. When aluminum dissolves, lines break. This problem is unique to the plastic capsules now used widely to reduce costs. Oki has carefully studied the possible cause and effect relationship between structure and manufacturing conditions on the one hand, and the generation of aluminum corrosion on the other. Refinements incorporated in Oki LSIs permit superior endurance to even the most severe high humidity conditions.

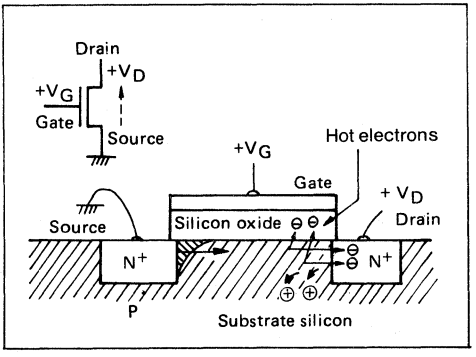
6) Alpha-Particle Soft Failure

This problem occurs when devices are highly miniaturized, such as in 1 megabit RAMs. The inversion of memory cell data by alpha-particle generated by radio-active elements like uranium and thorium (present in minute quantities, measured in ppb) in the ceramic package material causes defects. Since failure is only temporary and normal operation restored quickly, this is referred to as a "soft" failure. At Oki we have eliminated the problem by coating the chip surface of 1 megabit RAMs with a resin which effectively screens out these alpha-particles.



7) Degradation in Performance Characteristics Due to Hot Electrons

With increased miniaturization of circuit elements, internal electric field strength in the channels increases since the applied voltage remains the same at 5V. As a result, electrons flowing in the channels, as shown in the accompanying diagram, tend to enter into the oxide film near the drain, leading to degradation of performance. Although previous low-temperature operation tests have indicated an increase of this failure, we have confirmed by our low-temperature acceleration tests, including checks on test element groups, that no such problem exists in Oki LSIs.



Characteristic deterioration caused by hot electrons

With further progress in the miniaturization of circuit components, failures related to pin hole oxide film destruction and photolithography have increased. To eliminate these defects during manufacturing, Oki has been continually improving its production processes based on reliability tests and information gained from the field. And we subject all devices to high-temperature burn-in screening for 48 to 96 hours to ensure even greater reliability.

DATA SHEETS

OLMS-40 SERIES

6

OKI semiconductor

MSM5840

CMOS 4-BIT SINGLE CHIP MICROCONTROLLER

GENERAL DESCRIPTION

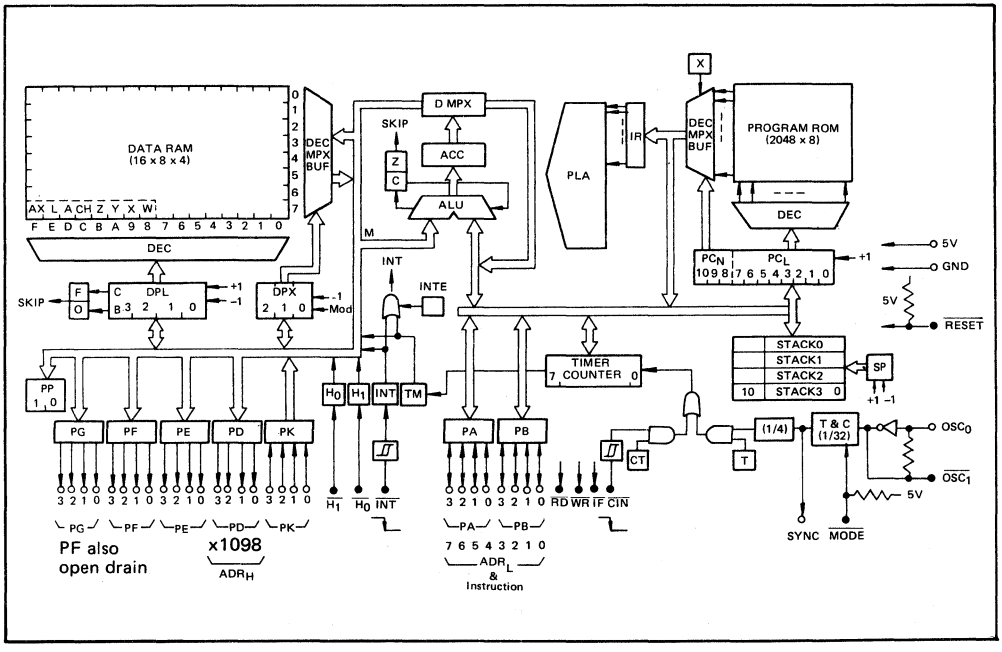
The OKI MSM5840 microcontroller is a low-power, high-performance single chip device implemented in complementary metal oxide semiconductor technology. Integrated within this one chip are 16K bits of mask program ROM, 512 bits of data RAM, 30 Input/Output lines, a programmable timer/counter, and oscillator. Program memory is byte wide and data-paths are organized in 4 bit nibbles. RAM and I/O lines are bit addressable. Up to 4K of external ROM interfaces to the 8 bit bidirectional bus. 98 instructions include binary, BCD, logical operations; bit set, reset, test, 8 bit I/O; relative jumps; multifunctional instructions (increment, modify, skip); 8 bit wide table output; subroutine call and return. 94% of instructions are single byte, single cycle operations.

FEATURES

- Low Power Consumption – 8mW Typical
- 100% Static Logic – 50μW Standby, Typical
- 2K × 8 Internal ROM
- Up to 4K × 8 External ROM
- 128 × 4 Internal RAM
- 30 I/O Lines Incl. 8 Bit Data Bus
- Programmable 8 Bit Timer/Counter
- Self-contained Oscillator
- 98 Instructions
- Expandable Memory and I/O
- 2 Interrupt Levels
- 4 Stack Levels
- Operating Temperature −40° to +85°C
- 3V to 6V Operating V_{DD}
- Battery Powered or Battery Backup
- TTL Compatible (with pullups)
- 7.6μs Cycle Time @4.2MHz (V_{DD} 5V± 10%)

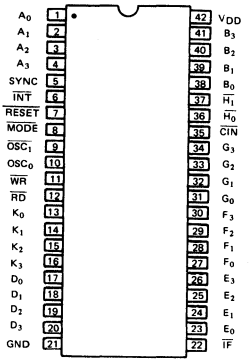
6

FUNCTIONAL BLOCK DIAGRAM

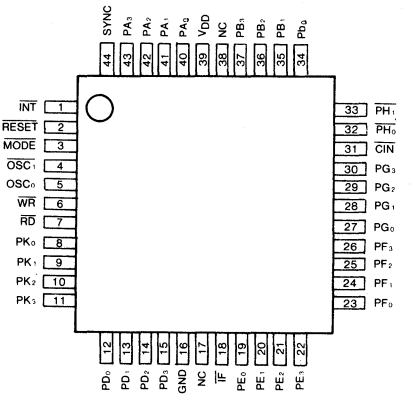


PIN CONFIGURATION (Top View)

42 PIN PLASTIC DIP (RS)



44 Pin Plastic Flat Package



6

PIN DESCRIPTION

Designation	Pin No.	Function
GND	21	Circuit GND potential
V_{DD}	42	Main power source (+5V)
OSC_0	10	Crystal OSC input, external clock input
$\overline{OSC_1}$	9	Crystal OSC input, external clock output (not TTL compatible)
PA, PB	1 to 4 38 to 41	Pseudo-bidirectional ports for 4-bit parallel I/O. Used as a pair for 8-bit I/O. Used to output 8 LSBs of address in external ROM mode. Used to read external instruction during IF.
PD, PE, PE, PG	17 to 20 23 to 34	Output ports for 4-bit parallel output and bit set/reset. Specified by internal port pointer, Bit position specified by set/reset instruction. PD also used for instruction address MSBs in external ROM mode during IF.
PK	13 to 16	4-bit parallel or bit test input port (unlatched)
$\overline{PH}$	36 and 37	2-bit input port with latched memory (negative level sensitive)
$\overline{RESET}$	7	RESET has priority over every other signal. (see MSM5840 user's manual for initialization sequence)
$\overline{MODE}$	8	Used to enable external ROM mode during RESET and also to enable STOP mode during execution (for stepping program)
$\overline{INT}$	6	Negative edge sensitive external interrupt signal associated with EI and DI instructions. Vectors to location 200H.
$\overline{CIN}$	35	Negative edge sensitive external input for counter associated with ECT and DCT instructions. Vectors to location 100H. (same as timer)
SYNC	5	General purpose synchronizing signal output at the beginning of each machine cycle. Used for address strobe during external ROM mode.
$\overline{RD}$	12	Read strobe pulse occuring when port A or B is read (1A, 1B, 1AB)
$\overline{WR}$	11	Write strobe pulse occurring when port A or B is written (OA, OB, OAB, OBS, OTD)
$\overline{IF}$	22	Read strobe pulse occurring during an instruction fetch from external ROM.

FUNCTIONAL DESCRIPTION

Program ROM

The MSM5840 will address up to 4K bytes of program ROM and can have 2K bytes of internal masked ROM, or all ROM may be located externally. External EPROM may be used for program development with conversion to internal ROM occurring after program debug and system checkout and verification. All instructions are byte wide. Only three of the 98 instructions require two bytes of program code. The instructions are routed to a programmed logic array which generates the necessary internal control signals.

Data RAM

Data is organized in 4 bit nibbles. Internal data RAM consists of 128 nibbles, 8 nibbles of which are dedicated registers accessible directly under program control. These are the general purpose registers, W, X, Y and Z, and the 4 save (exchange) registers, CH, A, L, and AX. All other DATA RAM must be addressed indirectly through the DP (data pointer) registers, a seven bit pointer (directly accessible by numerous instructions) consisting of 4 bit DPL register and a 3 bit DP_H register. Any nibble of internal data RAM can be accessed through the DP registers. Some instructions automatically change the contents of the DP registers allowing efficient array processing.

Input/Output Ports

PA, PB – These two ports are pseudo-bidirectional ports which can be used as simple I/O lines or used as either a 4-bit or 8-bit parallel bus. An instruction fetches the external ROM data through these ports by outputting the 8 low order bits of address during SYNC followed by an IF (instruction fetch) cycle. In addition, synchronized data transfers are possible through these ports with the I/O pin signals $\overline{RD}$ and $\overline{WR}$ associated with certain input/output instructions dedicated to these ports. In short, PA and PB can be used as a multiplexed address/instruction/data bus.

PD, PE, PF, PG – These four output ports are addressed indirectly through the TWO BIT port pointer whose contents are changed through certain instructions. These ports are bit (set/reset) addressable. PD is also used for the high order bits of address during an external instruction fetch. PF and PG are open drain outputs and PG is set high by a hardware RESET.

PK is an input port without memory, addressable either as a nibble or bit level input.

PH is a two-bit input port with memory, which can be tested and reset under program control.

External Interrupt

The $\overline{INT}$ pin can be tested under program control or enabled to cause a vectored interrupt to location 200H. It is negative edge sensitive.

Timer/Counter

The timer/counter is an 8-bit counter whose input is selected under program control to be either an external signal (CIN) or an internal square wave of 1/128 the frequency of the OSC_0 input (2 MHz/128 = 15.625 kHz). The timer/counter can be enabled or disabled under program control as can be associated internal interrupt which vectors to location 100H and has higher priority than the external interrupt.

Stack

The stack is an LIFO queue for storing return-from-interrupt and return-from-subroutine address information. It is eleven bits wide and 4 levels deep.

Program Counter (PC)

The program counter is 11 bits wide and loaded under program control.

Accumulator

The accumulator register is the data path focal point of the CPU. Approximately one-half of the instructions involve the accumulator. Its contents are the source and destination for many ALU operations and port operations. CASE statements (computed GOTOs) are possible by using the Jump with Accumulator (JA) instruction.

Flags

The MSM5840 is endowed with the following set of flags.

Z – zero flag	: Indicates that the result of the previous operation was zero
F – all ones	: Indicates a carry from the DP_L register
O – all zeros	: Indicates a borrow from the DP_L register
C – carry	: Indicates a carry from the previous operation
T – timer	: Indicates that the timer/counter is specified as a timer
CT – counter	: Indicates that the timer/counter is specified as a counter
TM – timer flag	: Indicates an overflow of the timer/counter register
INT – interrupt	: Latching memory flag for the external interrupt
INTE – interrupt enable	: Indicates that interrupts have been enabled
H_0 – H_0 memory	: Indicates that an input has been detected on the H_0 input
H_1	: same as H_0 except H_1 input
X	: 0 indicates internal ROM, 1 indicates external ROM. If all external ROM, 0 indicates first bank of 2K.

6

INSTRUCTION SET

	Mnemonic	Description	Instruction Code 7	6	5	4	3	2	1	0	Byte	Cycle
Load, Store, Read, Clear	CLA	Clear Accumulator	0	0	0	1	0	0	0	0	1	1
	CLL	Clear DP_L	0	0	1	0	0	0	0	0	1	1
	CLH	Clear DP_H	0	1	1	0	0	0	0	0	1	1
	LAI	Load Accumulator with Immediate	0	0	0	1	I_3	I_2	I_1	I_0	1	1
	LLI	Load DP_L with Immediate	0	0	1	0	I_3	I_2	I_1	I_0	1	1
	LHI	Load DP_H with Immediate	0	1	1	0	0	I_2	I_1	I_0	1	1
	L	Load Accumulator with Memory	1	0	0	1	0	1	0	0	1	1
	LM	Load Accumulator with Memory then Modify DP_H	1	0	0	1	0	1	I_1	I_0	1	1
	LAL	Load Accumulator with DP_L	0	1	0	1	0	1	0	1	1	1
	LLA	Load DP_L with Accumulator	0	1	0	1	0	1	0	0	1	1
	LAW	Load Accumulator with W Register	1	0	0	0	0	1	0	0	1	1
	LAX	Load Accumulator with X Register	1	0	0	0	0	1	0	1	1	1
	LAY	Load Accumulator with Y Register	1	0	0	0	0	1	1	0	1	1
	LAZ	Load Accumulator with Z Register	1	0	0	0	0	1	1	1	1	1
	SI	Store Accumulator to Memory then Increment DP_L	1	0	0	1	0	0	0	0	1	1
	SMI	Store Accumulator to Memory then Modify DP_H and Increment DP_L	1	0	0	1	0	0	I_1	I_0	1	1
	LWA	Load W Register with Accumulator	1	0	0	0	0	0	0	0	1	1
	LXA	Load X Register with Accumulator	1	0	0	0	0	0	0	1	1	1
	LYA	Load Y Register with Accumulator	1	0	0	0	0	0	1	0	1	1
	LZA	Load Z Register with Accumulator	1	0	0	0	0	0	1	1	1	1
	LPA	Load Port Pointer with Accumulator	0	1	0	1	1	0	0	0	1	1
	LTI	Load Timer with Immediate	0 I_7	1 I_6	1 I_5	0 I_4	1 I_3	0 I_2	0 I_1	0 I_0	2	2
	RTH	Read Timer H	0	1	1	0	1	0	1	0	1	1
	RTL	Read timer L	0	1	1	0	1	0	1	1	1	1
Exchange	XA	Exchange Accumulator with Save Register A	0	1	0	0	1	0	0	1	1	1
	XL	Exchange DP_L with Save Register L	0	1	0	0	1	0	1	0	1	1
	XCH	Exchange DP_H and Carry with Save Register CH	0	1	0	0	1	0	0	0	1	1
	X	Exchange Accumulator with Memory	1	0	0	1	1	0	0	0	1	1
	XM	Exchange Accumulator with Memory then Modify DP_H	1	0	0	1	1	0	I_1	I_0	1	1
	XAX	Exchange Accumulator with Save Register AX	0	1	0	0	1	0	1	1	1	1
Increment/Decrement	INA	Increment Accumulator	0	0	0	0	0	0	0	1	1	1
	INL	Increment DP_L	0	1	0	1	0	1	1	1	1	1
	INM	Increment Memory	0	1	0	1	1	1	0	1	1	1
	INW	Increment W Register (Skip if Zero)	1	0	0	0	1	0	0	0	1	1
	INX	Increment X Register	1	0	0	0	1	0	0	1	1	1
	INY	Increment Y Register	1	0	0	0	1	0	1	0	1	1
	INZ	Increment Z Register	1	0	0	0	1	0	1	1	1	1

INSTRUCTION SET (CONT.)

	Mnemonic	Description	Instruction Code 7	6	5	4	3	2	1	0	Byte	Cycle
Increment/Decrement	DCA	Decrement Accumulator – Skip if Not All Ones	0	0	0	0	1	1	1	1	1	1
	DCL	Decrement DP_L	0	1	0	1	0	1	1	0	1	1
	DCM	Decrement Memory	0	1	0	1	1	1	0	0	1	1
	DCW	Decrement W Register (Skip if All Ones)	1	0	0	0	1	1	0	0	1	1
	DCX	Decrement X Register	1	0	0	0	1	1	0	1	1	1
	DCY	Decrement Y Register	1	0	0	0	1	1	1	0	1	1
	DCZ	Decrement Z Register	1	0	0	0	1	1	1	1	1	1
	DCH	Decrement DP_H – Skip if All Ones and C = Zero	0	1	0	1	1	1	1	1	1	1
Logical	CAO	Complement Accumulator of One	0	1	0	1	0	0	0	0	1	1
	AND	And Accumulator with Memory	0	1	0	0	0	1	0	0	1	1
	OR	Or Accumulator with Memory	0	1	0	0	0	1	0	1	1	1
	EOR	Exclusive or Accumulator with Memory	0	1	0	0	0	1	1	0	1	1
	RAL	Rotate Accumulator Left through Carry	0	1	0	0	0	1	1	1	1	1
Arithmetic	AC	Add Memory to Accumulator with Carry	0	1	0	0	1	1	0	0	1	1
	ACS	Add Memory to Accumulator with Carry, Skip if Carry	0	1	0	0	1	1	0	1	1	1
	AS	Add Memory to Accumulator, Skip if Carry	0	1	0	0	1	1	1	0	1	1
	AIS	Add Immediate to Accumulator, Skip if Carry	0	0	0	0	I_3	I_2	I_1	I_0	1	1
	DAS	Decimal adjust Accumulator in Subtraction	0	1	0	1	1	0	1	0	1	1
	CM	Compare Accumulator with Memory, Skip if Equal	0	1	0	1	1	1	1	0	1	1
	AWS	Add W Register to Accumulator, Skip if Carry	1	0	0	1	1	1	0	0	1	1
	AXS	Add X Register to Accumulator, Skip if Carry	1	0	0	1	1	1	0	1	1	1
	AYS	And Y Register to Accumulator, Skip if Carry	1	0	0	1	1	1	1	0	1	1
	AZS	Add Z Register to Accumulator, Skip if Carry	1	0	0	1	1	1	1	1	1	1
Bit Set/Reset/Test	SPB	Set Port Bit	1	0	1	1	0	0	I_1	I_0	1	1
	RPB	Reset Port Bit	1	0	1	1	0	1	I_1	I_0	1	1
	SMB	Set Memory Bit	1	0	1	1	1	0	I_1	I_0	1	1
	RMB	Reset Memory Bit	1	0	1	1	1	1	I_1	I_0	1	1
	TAB	Test Accumulator Bit (Skip if One)	1	0	1	0	0	0	I_1	I_0	1	1
	TMB	Test Memory Bit	1	0	1	0	0	1	I_1	I_0	1	1
	TKB	Test K Port Bit	1	0	1	0	1	0	I_1	I_0	1	1
	THB	Test H Port Bit	1	0	1	0	1	1	0	I_0	1	1
	TI	Test Interrupt flag	1	0	1	0	1	1	1	1	1	1
	TTM	Test Time flag	1	0	1	0	1	1	1	0	1	1
	TC	Test Carry flag	0	1	0	0	0	0	1	0	1	1
	SC	Set Carry flag	0	1	0	0	0	0	0	0	1	1
	RC	Reset Carry flag	0	1	0	0	0	0	0	1	1	1

INSTRUCTION SET (CONT.)

	Mnemonic	Description	Instruction Code 7	6	5	4	3	2	1	0	Byte	Cycle
Branch/Subroutine	J	Jump	0	0	1	1	0	I_{10}	I_9	I_8	2	2
			I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0		
	JC	Jump in Current Page	1	1	I_5	I_4	I_3	I_2	I_1	I_0	1	1
	JA	Jump with Accumulator	0	1	0	0	0	0	1	1	1	1
	CAL	Call Subroutine	0	0	1	1	1	I_{10}	I_9	I_8	2	2
			I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0		
	RT	Return from Subroutine	0	1	0	1	1	0	0	1	1	2
Input/Output	OBS	Output Byte String	0	1	1	1	0	0	0	0	1	2~17
	OTD	Output Table Data	0	1	1	1	0	0	0	1	1	2
	OA	Output Accumulator to Port A	0	1	1	1	0	0	1	0	1	1
	OB	Output Accumulator to Port B	0	1	1	1	0	0	1	1	1	1
	OP	Output Accumulator to Port P designated Port Pointer	0	1	1	1	0	1	0	0	1	1
	OAB	Output Memory and Accumulator to Ports A and B	0	1	1	1	0	1	0	1	1	1
	OPM	Output Memory to Port P designated Port Pointer	0	1	1	1	0	1	1	0	1	1
	IA	Input Port A in Accumulator	0	1	1	1	1	0	1	0	1	1
	IB	Input Port B in Accumulator	0	1	1	1	1	0	1	1	1	1
	IK	Input Port K in Accumulator	0	1	1	1	1	1	0	0	1	1
	IAB	Input Ports A and B in Memory and Accumulator	0	1	1	1	1	1	0	1	1	1
Control	EI	Enable Interrupt	0	1	0	1	0	0	1	1	1	1
	DI	Disable Interrupt	0	1	0	1	0	0	1	0	1	1
	ET	Enable Timer	0	1	1	0	1	1	1	1	1	1
	DT	Disable Timer	0	1	1	0	1	1	1	0	1	1
	ECT	Enable Counter	0	1	1	1	1	1	1	1	1	1
	DCT	Disable Counter	0	1	1	1	1	1	1	0	1	1
	HLT	Halt	0	1	1	0	1	1	0	1	1	1
	EXP	Exchange Program	0	1	1	0	1	0	0	1	1	1
	NOP	No Operation	0	0	0	0	0	0	0	0	1	1

6

ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	Ta = 25°C	−0.3 to 7	V
Input Voltage	V_I	Ta = 25°C	−0.3 to VDD	V
Operating Voltage PF PG	V_O	Ta = 25°C	−0.3 to 25	V
Storage Temperature	T_{stg}		−55 to +150	°C

Note: Stresses above those listed under ABSOLUTE MAXIMUM RATINGS may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or at any other condition 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.

OPERATING CONDITIONS

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	@1 MHz	3 to 6	V
		@4.2 MHz	4.5 to 5.5	V
Operating Temperature	T_{op}		−40 to +85	°C
Fan Out	N	MOS Load	15	
		TTL Load	1	

D.C. CHARACTERISTICS

(V_{DD} = 5V±10%, Ta = −20° to +70°C)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
High Input Voltage	V_{IH}	–	3.6			V
Low Input Voltage	V_{IL}	–			0.8	V
High Output Voltage (1)	V_{OH}	I_O = −40μA	4.2			V
Low Output Voltage	V_{OL}	I_O = 1.6mA			0.4	V
OSC_o Input Leak Current	I_{IH}	$V_I = V_{DD}/0V$			25	μA
	I_{IL}				−25	
$\overline{RESET}$, $\overline{MODE}$ Leak Current	I_{IH}	$V_I = V_{DD}/0V$			1	μA
	I_{IL}				−50	
Input Leak Current(2)	I_{IH}	$V_I = V_{DD}/0V$			1	μA
	I_{IL}				−1	
PA, PB High Output Current	I_{OH}	V_{OH} = 0.4V			−1	mA
High Output Current(1)	I_{OH}	V_{OH} = 2.5V	−0.25			mA
Low Output Current	I_{OL}	V_{OL} = 0.4V	1.6			mA
PF, PG Output Breakdown Voltage	BV_{OH}	I_O = 10μA	20			V
Input Capacitance	C_I	f = 1MHz Ta = 25°C		5		pF
Output Capacitance	C_O	f = 1MHz Ta = 25°C		7		pF
Current Consumption(3)	I_{DD}	$V_I = V_{DD}/0V$		10	200	μA
	I_{DD}	$V_I = V_{DD}/0V$ f = 4.2MHz		1.6	4	mA

Notes: (1) Except PA, PB (see graphs)
(2) Except OSC_o, $\overline{RESET}$, $\overline{MODE}$
(3) Typical Value of V_{DD} is 5V

A.C. CHARACTERISTICS (INTERNAL ROM MODE)

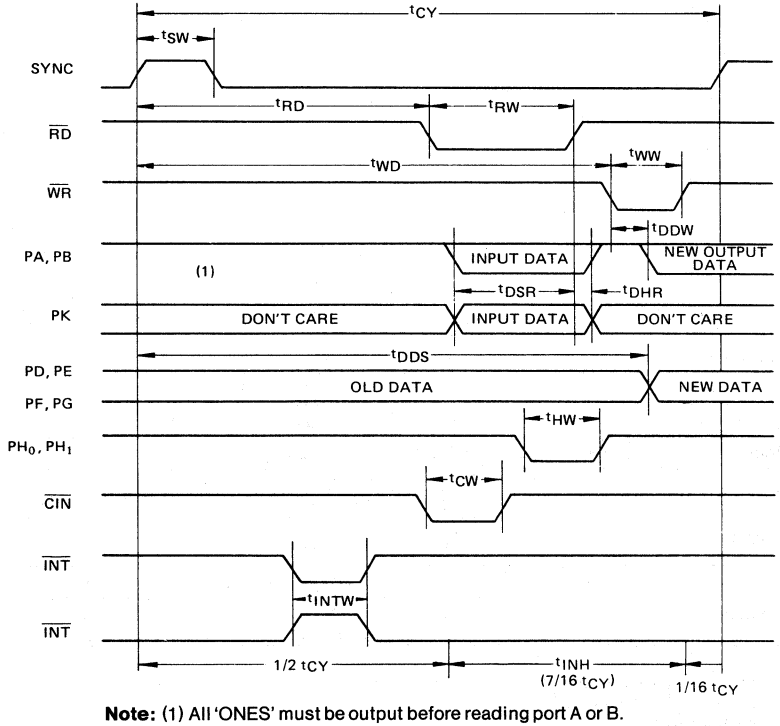
(V_{DD} = 5V±10%, Ta = −40° to +85°C)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Cycle Time	t_{CY}		7.6			μS
Sync Pulse Width	t_{SW}		0.95			μS
$\overline{RD}$ Pulse Width	t_{RW}		1.9			μS
Sync ↑ to $\overline{RD}$ ↓	t_{RD}	C_L = 50pF	1/2 t_{CY} + 0.5			μS
$\overline{WR}$ Pulse Width	t_{WW}		0.95			μS
Sync ↑ to $\overline{WR}$ ↓	t_{WD}	C_L = 50pF	13/16 t_{CY} + 0.5			μS
Port Input Setup Time	t_{DSR}		4/16 t_{CY}			μS
Port Input Hold Time	t_{DHR}		0		0.8	μS
$\overline{WR}$ ↓ to New Data Valid	t_{DDW}	PA, PB C_L = 50pF			0.8	μS
Sync ↑ to New Data Valid	t_{DDS}	PD, PE, PF, PG C_L = 50pF			13/16 t_{CY}+0.5	μS
PH_0, PH_1 Input Pulse Width	t_{HW}	(1)	500			nS
$\overline{CIN}$ Input Pulse Width	t_{CW}		250			nS
INT Input Pulse Width	t_{INTW}	(1)	500			nS

Note: (1) The processor logic will ignore the following events:
1. An INT falling edge occurring during T_{INH} of a T_I instruction.
2. A PH_0 or PH_1 low level occurring only during T_{INH} of a T_{HB} instruction.

6

TIMING CHARTS

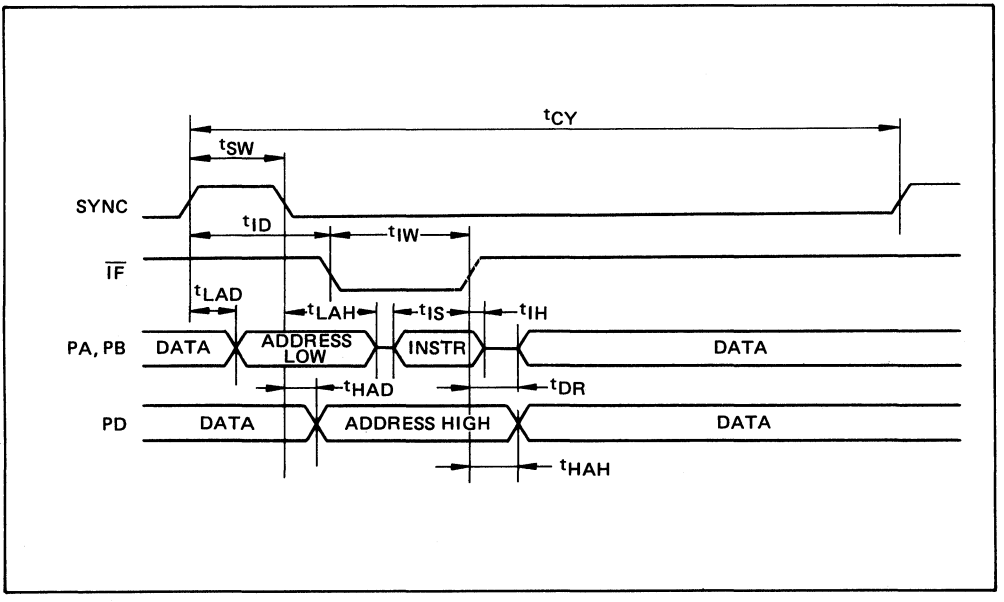


Note: (1) All 'ONES' must be output before reading port A or B.

A.C. CHARACTERISTICS (EXTERNAL ROM MODE)

(V_{DD} = 5V±10%, Ta = −40° to +85°C)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Cycle Time	t_{CY}		7.6			μS
Sync Pulse Width	t_{SW}		0.95			μS
$\overline{IF}$ Pulse Width	t_{IW}		1.425			μS
Sync ↑ to $\overline{IF}$ ↓	t_{ID}	C_L = 50pF	3/16 t_{CY} + 1			μS
Address Low Delay	t_{LAD}	C_L = 50pF			0.8	μS
Address Low Hold	t_{LAH}		1/16 t_{CY}		1/16 t_{CY}+1	μS
Instruction Setup	t_{IS}		1/16 t_{CY}			μS
Instruction Hold	t_{IH}				20	nS
Data Recovery	t_{DR}	C_L = 50pF	0		0.8	μS
Address High Delay	t_{HAD}	C_L = 50pF			0.5	μS
Address High Hold	t_{HAH}		0		0.5	μS

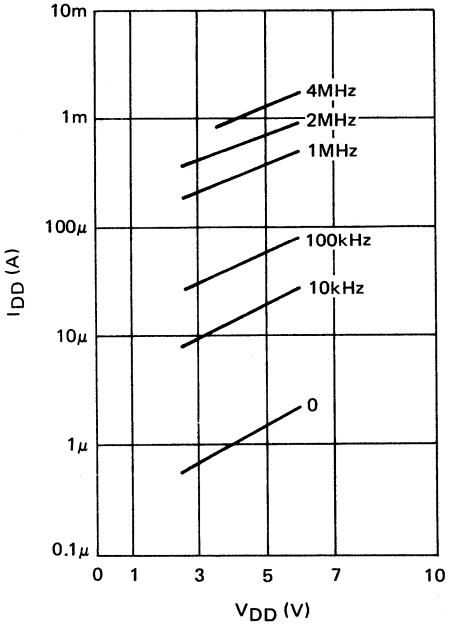


Cycle Dependent Timings	4MHz	2MHz	1MHz	500kHz
1/16t_{CY}	0.5μS	1μS	2μS	4μS
1/16t_{CY}+1	1.5μS	2μS	3μS	5μS
3/16t_{CY}+1	2.5μS	4μS	7μS	13μS
4/16t_{CY}−1	1μS	3μS	7μS	15μS
1/2t_{CY}+1	5μS	9μS	17μS	33μS
7/16t_{CY}	3.5μS	7μS	14μS	28μS
13/16t_{CY}+1	7.5μS	14μS	27μS	53μS

TYPICAL PERFORMANCE CURVES

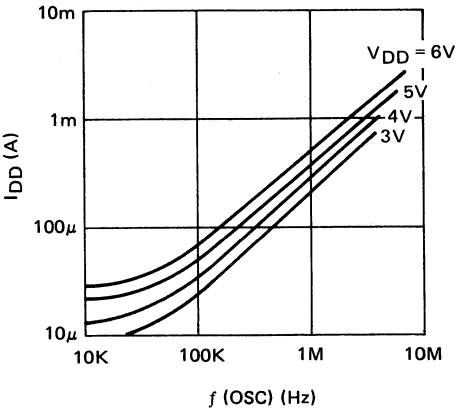
Supply Current vs Supply Voltage

(Ta = 25°C, No Load)



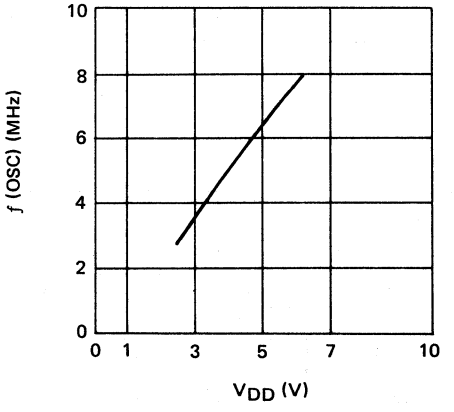
Supply Current vs Oscillator Frequency

(Ta = 25°C, No Load)



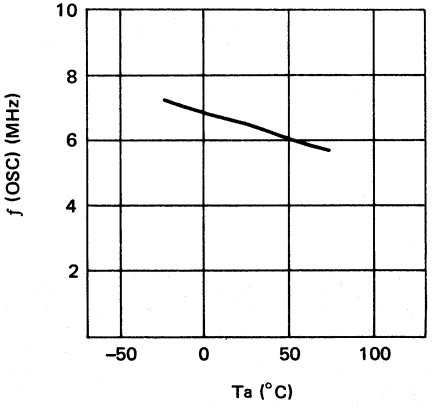
Oscillator Frequency vs Supply Voltage

(Ta = 25°C, C_L = 50pF)

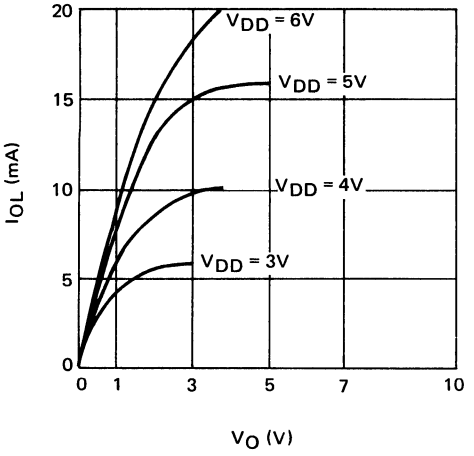


Oscillator Frequency vs Temperature

(C_L = 50pF)

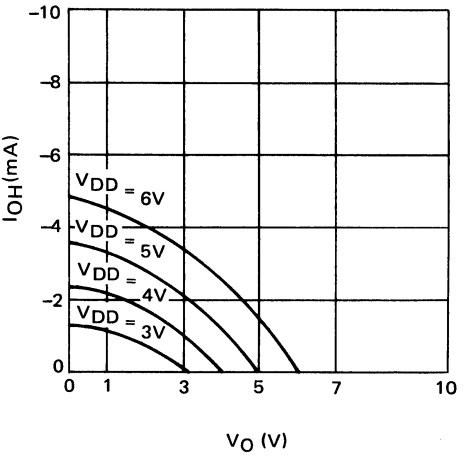


Low Current Out vs Voltage



High Current Out vs Voltage

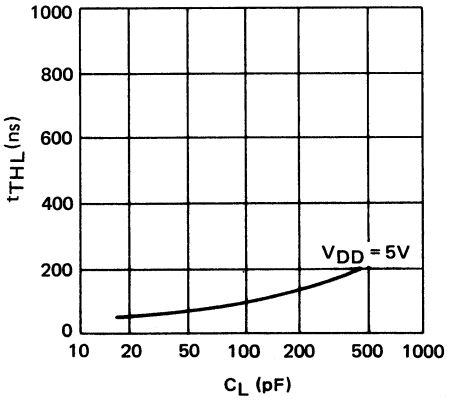
(Ta = 25°C, Except PA, PB)



6

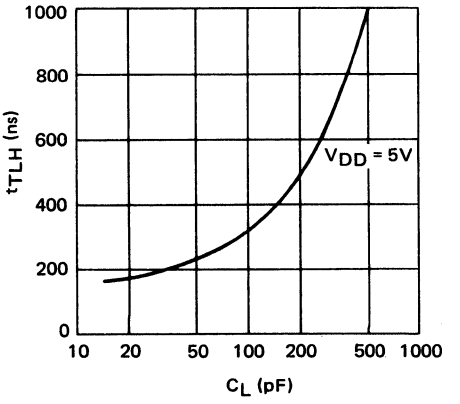
Fall Time vs Load

(Ta = 25°C, PA, PB, PD, PE, $\overline{RD}$, $\overline{WR}$, $\overline{IF}$, SYNC)



Rise Time vs Load

(Ta = 25°C, PD, PE, $\overline{RD}$, $\overline{WR}$, $\overline{IF}$, SYNC)



OKI semiconductor

MSM5842

CMOS 4-BIT SINGLE CHIP MICROCONTROLLER

GENERAL DESCRIPTION

The OKI MSM5842 microcontroller is a low-power, high-performance single chip device implemented in complementary metal oxide semiconductor technology. Integrated with this one chip are 6K bits of mask program ROM, 128 bits of data RAM, 21 Input/Output lines, an 8-bit binary timer/counter, and oscillator. Program memory is byte wide and data paths are organized in 4 bit nibbles. RAM and I/O lines are bit addressable. 52 instructions include binary, BCD, operations; bit set, reset, test; 8-bit I/O; relative jumps; multifunctional instructions (increment, modify, skip); 8-bit wide table output; subroutine call and return. 94% of instructions are single byte, single cycle operations.

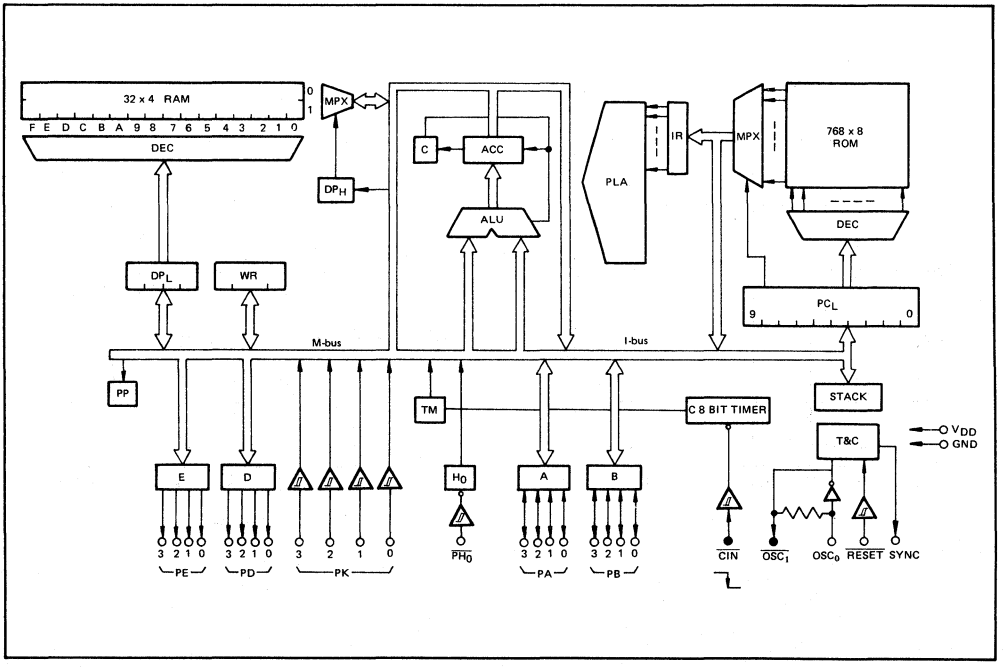
Available in plastic (RS) package.

FEATURES

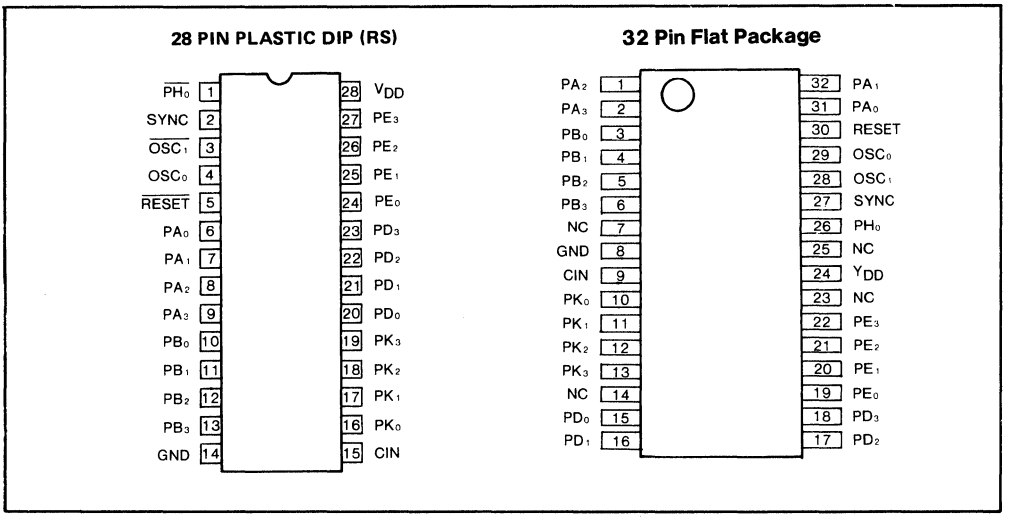
- Low Power Consumption – 7mW Typical
- 100% Static Logic – 100μW Standby
- 768 × 8 Internal ROM
- 32 × 4 Internal RAM
- 21 I/O Lines Incl. 8 Bit Data Bus
- 8 Bit Binary Timer/Counter
- Self-contained Oscillator
- 52 Instructions
- 1 Stack Level
- −20° to +70°C Operating Temperature
- 3V to 6V Operating V_{DD}
- Battery Powered or Battery Backup
- TTL Compatible (with pullups)
- 7.6μs Cycle Time @4.2MHz

6

FUNCTIONAL BLOCK DIAGRAM



PIN CONFIGURATION (Top View)



PIN DESCRIPTION

Designation	Pin No.	Function
GND	14	Circuit GND potential
V_{DD}	28	Main power source (+5V)
OSC_0	4	Crystal OSC input, external clock input
$\overline{OSC_1}$	3	Crystal OSC input, external clock output (not TTL compatible)
PA, PB	6 to 13	Quasi-bidirectional ports for 4 bit parallel I/O. Used as a pair for 8 bit I/O.
PD, PE	20 to 27	Output ports for 4 bit parallel output and bit set/reset. Specified by internal port pointer. Bit position specified by set/reset instruction.
PK	16 to 19	4 bit parallel or bit test input port (unlatched)
$\overline{PH}$	1	1 bit input port with latched memory (negative level sensitive)
$\overline{RESET}$	5	RESET must be low for more than one machine cycle and has priority over every other signal. (see MSM5842 user's manual for initialization sequence)
$\overline{CIN}$	15	Negative edge sensitive external input for timer/counter.
SYNC	2	General purpose synchronizing signal output at the beginning of each machine cycle.

FUNCTIONAL DESCRIPTION

Program ROM

The MSM5842 will address up to 768 bytes of program ROM. All instructions are byte wide. Only three of the 52 instructions require two bytes of program code. The instructions are routed to a programmed logic array which generates the necessary internal control signals.

Data RAM

Data is organized in 4 bit nibbles. Internal data RAM consists of 32 nibbles and one nibble which is a dedicated general purpose register, W, accessible directly under program control. DATA RAM must be addressed indirectly through the DP (data pointer) register, a five bit pointer (directly accessible by numerous instructions) consisting of a 4 bit DP_L register and a 1 bit DP_H register. Any nibble of internal data RAM can be accessed through the DP registers. Some instructions, automatically change the contents of the DP register allowing efficient array processing.

Input/Output Ports

PA, PB – These two ports are pseudo-bidirectional ports which can be used as simple I/O lines or used as either a 4 bit or 8 bit parallel bus.

PD, PE – These two output ports are addressed indirectly through the ONE BIT port pointer whose contents are changed through certain instructions. These ports are bit (set/reset) addressable.

PK is an input port without a Latch circuit, addressable as a nibble input.

PH is a one bit input port with a Latch circuit, which can be tested and reset under program control.

Timer/Counter

The timer/counter is an 8-bit counter whose input is an external signal ($\overline{CIN}$). The TM flag is set when the timer/counter generates a carry.

Stack

The stack is a single register for storing return-from-subroutine address information. It is ten bits wide.

Program Counter (PC)

The program counter is ten bits wide.

Accumulator

The accumulator register is the data path focal point of the CPU. Approximately one-half of the instructions involve the accumulator. Its contents are the source and destination for many ALU operations and port operations. CASE statements (computed GOTOs) are possible by using the Jump with Accumulator (JA) instruction.

Flags

The MSM5842 is endowed with the following set of flags.

Z – zero flag	: Indicates that the result of the previous operation was zero
C – carry	: Indicates a carry from the previous operation
TM – timer flag	: Indicates an overflow of the timer/counter register
H_0 – H_0 memory	: Indicates that an input has been detected on the H_0 input

INSTRUCTION SET

	Mnemonic	Description	Instruction Code 7	6	5	4	3	2	1	0	Byte	Cycle
Load, Store, Read, Clear	CLA	Clear Accumulator	0	0	0	1	0	0	0	0	1	1
	CCL	Clear DP_L	0	0	1	0	0	0	0	0	1	1
	CLH	Clear DP_H	0	1	1	0	0	0	0	0	1	1
	LAI	Load Accumulator with Immediate	0	0	0	1	I_3	I_2	I_1	I_0	1	1
	LLI	Load DP_L with Immediate	0	0	1	0	I_3	I_2	I_1	I_0	1	1
	LHI	Load DP_H with Immediate	0	1	1	0	0	0	0	I_0	1	1
	L	Load Accumulator with Memory	1	0	0	1	0	1	0	0	1	1
	LAL	Load Accumulator with DP_L	0	1	0	1	0	1	0	1	1	1
	LLA	Load DP_L with Accumulator	0	1	0	1	0	1	0	0	1	1
	LAW	Load Accumulator with W Register	1	0	0	0	0	1	0	0	1	1
	SI	Store Accumulator to Memory then Increment DP_L	1	0	0	1	0	0	0	0	1	1
	LWA	Load W Register with Accumulator	1	0	0	0	0	0	0	0	1	1
	LPA	Load Port Pointer with Accumulator	0	1	0	1	1	0	0	0	1	1
	LTI	Load Timer with All Zeros	0	1	1	0	1	0	0	0	1	1

INSTRUCTION SET (CONT.)

	Mnemonic	Description	Instruction Code 7	6	5	4	3	2	1	0	Byte	Cycle
Exchange	X	Exchange Accumulator with Memory	1	0	0	1	1	0	0	0	1	1
Increment/Decrement	INA	Increment Accumulator (Skip if Zero)	0	0	0	0	0	0	0	1	1	1
	INL	Increment DP_L (Skip if Zero)	0	1	0	1	0	1	1	1	1	1
	INM	Increment Memory (Skip if Zero)	0	1	0	1	1	1	0	1	1	1
	INW	Increment W Register (Skip if Zero)	1	0	0	0	1	0	0	0	1	1
	DCA	Decrement Accumulator – Skip if Not All Ones	0	0	0	0	1	1	1	1	1	1
	DCL	Decrement DP_L (Skip if All Ones)	0	1	0	1	0	1	1	0	1	1
	DCM	Decrement Memory (Skip if All Ones)	0	1	0	1	1	1	0	0	1	1
Logical	CAO	Complement Accumulator of One	0	1	0	1	0	0	0	0	1	1
	RAL	Rotate Accumulator Left through Carry	0	1	0	0	0	1	1	1	1	1
Arithmetic	AC	Add Memory to Accumulator with Carry	0	1	0	0	1	1	0	0	1	1
	AS	Add Memory to Accumulator, Skip if Carry	0	1	0	0	1	1	1	0	1	1
	AIS	Add Immediate to Accumulator, Skip if Carry	0	0	0	0	I_3	I_2	I_1	I_0	1	1
	DAS	Decimal adjust Accumulator in Subtraction	0	1	0	1	1	0	1	0	1	1
	CM	Compare Accumulator with Memory, Skip if Equal	0	1	0	1	1	1	1	0	1	1
Bit Set/Reset/Test	SMB	Set Memory Bit	1	0	1	1	1	0	I_1	I_0	1	1
	RMB	Reset Memory Bit	1	0	1	1	1	1	I_1	I_0	1	1
	TAB	Test Accumulator Bit	1	0	1	0	0	0	I_1	I_0	1	1
	TMB	Test Memory Bit	1	0	1	0	0	1	I_1	I_0	1	1
	THB	Test H Port Bit	1	0	1	0	1	1	0	I_0	1	1
	TTM	Test Timer flag	1	0	1	0	1	1	1	0	1	1
	TC	Test Carry flag	0	1	0	0	0	0	1	0	1	1
	SC	Set Carry flag	0	1	0	0	0	0	0	0	1	1
	RC	Reset Carry flag	0	1	0	0	0	0	0	1	1	1
Branch/Subroutine	J	Jump	0 I_7	0 I_6	1 I_5	1 I_4	0 I_3	0 I_2	I_9 I_1	I_8 I_0	2	2
	JC	Jump in Current Page	1	1	I_5	I_4	I_3	I_2	I_1	I_0	1	1
	JA	Jump with Accumulator	0	1	0	0	0	0	1	1	1	1
	CAL	Call Subroutine	0 I_7	0 I_6	1 I_5	1 I_4	1 I_3	0 I_2	I_9 I_1	8 I_0	2	2
	RT	Return from Subroutine	0	1	0	1	1	0	0	1	1	2
Input/Output	OTD	Output Table Data	0	1	1	1	0	0	0	1	1	2
	OA	Output Accumulator to Port A	0	1	1	1	0	0	1	0	1	1
	OB	Output Accumulator to Port B	0	1	1	1	0	0	1	1	1	1
	OP	Output Accumulator to Port P designated Port Pointer	0	1	1	1	0	1	0	0	1	1
	OPM	Output Memory to Port P designated Port Pointer	0	1	1	1	0	1	1	0	1	1
	IA	Input Port A in Accumulator	0	1	1	1	1	0	1	0	1	1
	IB	Input Port B in Accumulator	0	1	1	1	1	0	1	1	1	1
	IK	Input Port K in Accumulator	0	1	1	1	1	1	0	0	1	1
Control	NOP	No Operation	0	0	0	0	0	0	0	0	1	1

6

ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	Ta = 25°C	−0.3 to 7	V
Input Voltage	V_I	Ta = 25°C	−0.3 to VDD	V
Storage Temperature	T_{stg}		−55 to +150	°C

Note: Stresses above those listed under ABSOLUTE MAXIMUM RATINGS may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or at any other condition 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.

OPERATING CONDITIONS

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	1MHz	3 to 6	V
		4.2MHz	4.5 to 5.5	V
Operating Temperature	Top		−40 to 85	°C
Fan Out		MOS Load	40	
		TTL Load	1	

6

D.C. CHARACTERISTICS

(V_{DD} = 5V±10%, Ta = −40° to +85°C)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
High Input Voltage	V_{IH}	–	3.6			V
Low Input Voltage	V_{IL}	–			0.8	V
High Output Voltage (1)	V_{OH}	I_O = −40μA	4.2			V
Low Output Voltage	V_{OL}	I_O = 1.6mA			0.45	V
OSC_O Input Leak Current	I_{IH}	$V_I = V_{DD}/0V$			25	μA
	I_{IL}				−25	
$\overline{RESET}$ Leak Current	I_{IH}	$V_I = V_{DD}/0V$			1	μA
	I_{IL}				−20	
Input Leak Current(2)	I_{IH}	$V_I = V_{DD}/0V$			1	μA
	I_{IL}				−1	
PA, PB High Output Current	I_{OH}	V_{OH} = 0.4V			−1	mA
High Output Current(1)	I_{OH}	V_{OH} = 2.5V	−0.25			mA
Low Output Current	I_{OL}	V_{OL} = 0.45V	1.6			mA
Input Capacitance	C_I	f = 1MHz Ta = 25°C		5		pF
Output Capacitance	C_O	f = 1MHz Ta = 25°C		7		pF
Current Consumption(3)	I_{DD}	$V_I = V_{DD}/0V$		20	200	μA
	I_{DD}	$V_I = V_{DD}/0V$ f=4.2MHz		1.5	4	mA

Notes: (1) Except PA, PB (see graphs)
(2) Except OSC_O, $\overline{RESET}$
(3) Typical Value of V_{DD} is 5V

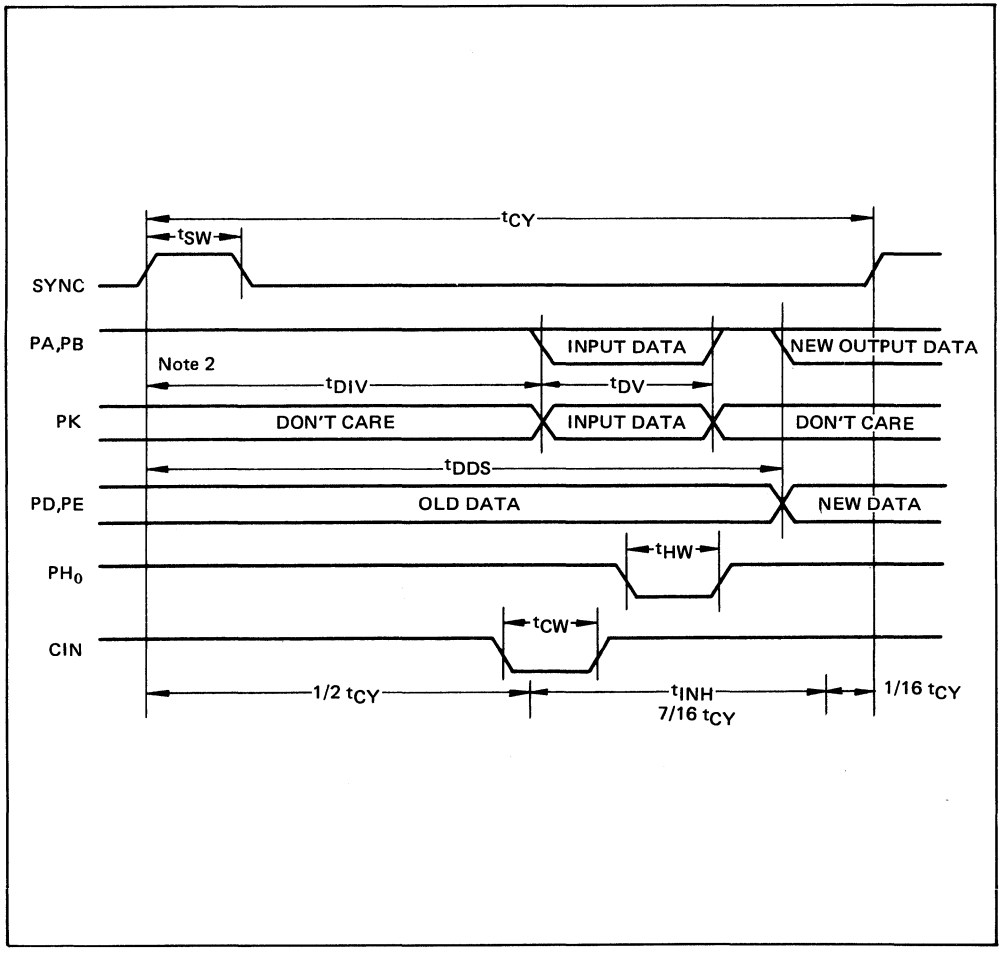
A.C. CHARACTERISTICS

(V_{DD} = 5V±10%, Ta = −40° to +85°C)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Cycle Time	t_{CY}	O_{SC} = 4MHz	7.6			μS
Sync Pulse Width	t_{SW}		0.95			μS
Port Input Invalid Time	t_{DIV}				1/2 t_{CY} + 0.5	μS
Port Input Valid Time	t_{DV}		2			μS
Sync ↑ to New Data Valid	t_{DDS}	PD, PE C_L = 50pF			13/16 t_{CY} + 0.5	μS
PH_0 Input Pulse Width	t_{HW}	(1)	250			nS
$\overline{CIN}$ Input Pulse Width	t_{CW}		250			nS

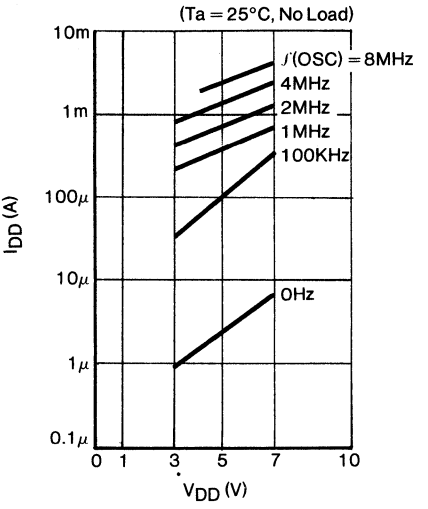
Notes: (1) The processor logic may ignore the following event:
A PH_0 low level occurring only during T_{INH} of a THB instruction.
(2) All 'ONES' must be output before reading port A or B.

TIMING CHARTS

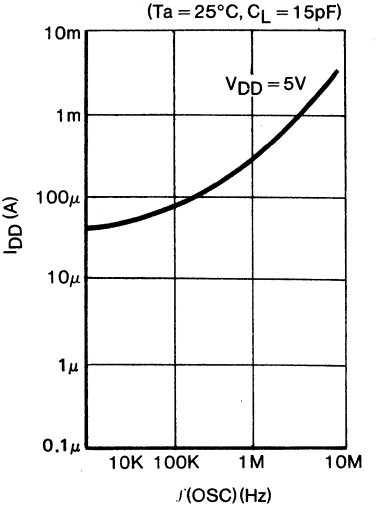


TYPICAL PERFORMANCE CURVES

Supply Current vs Supply Voltage

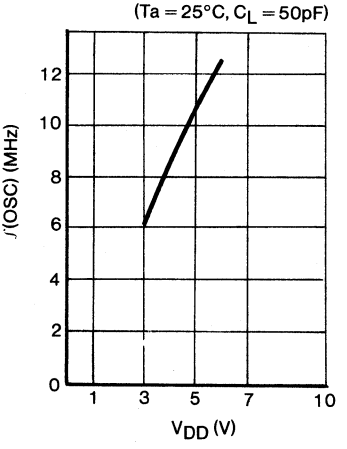


Supply Current vs Oscillator Frequency

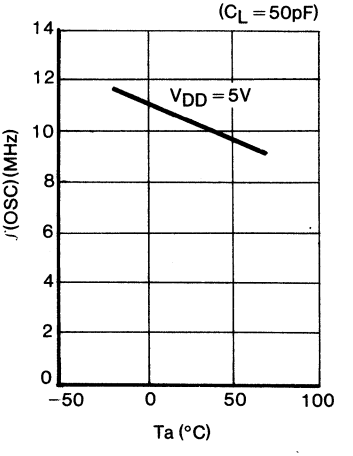


6

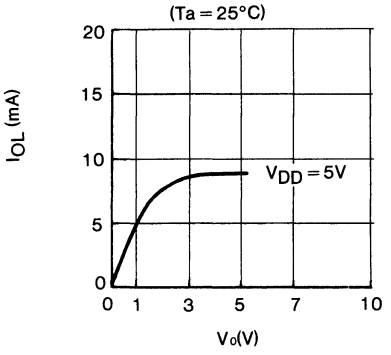
Oscillator Frequency vs Supply Voltage



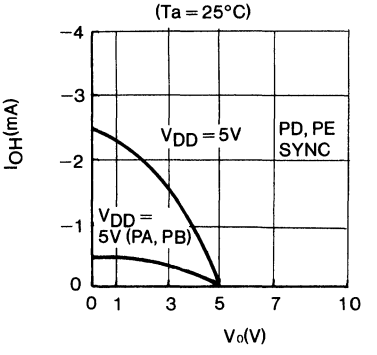
Oscillator Frequency vs Temperature



Low Current Out vs Voltage

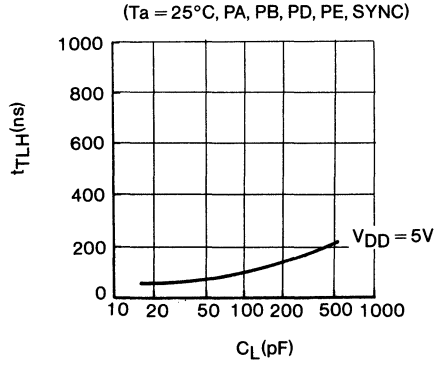


High Current Out vs Voltage

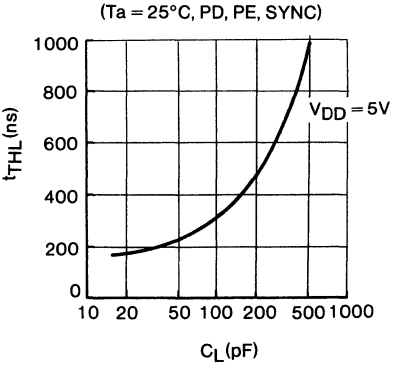


6

Fall Time vs Load



Rise Time vs Load



OKI semiconductor

MSM58421

CMOS 4-BIT SINGLE CHIP MICROCONTROLLER WITH LCD DRIVER

GENERAL DESCRIPTION

The OKI MSM58421 is a low-power, high-performance 4-bit single-chip microcontroller implemented in complementary metal oxide semiconductor technology.

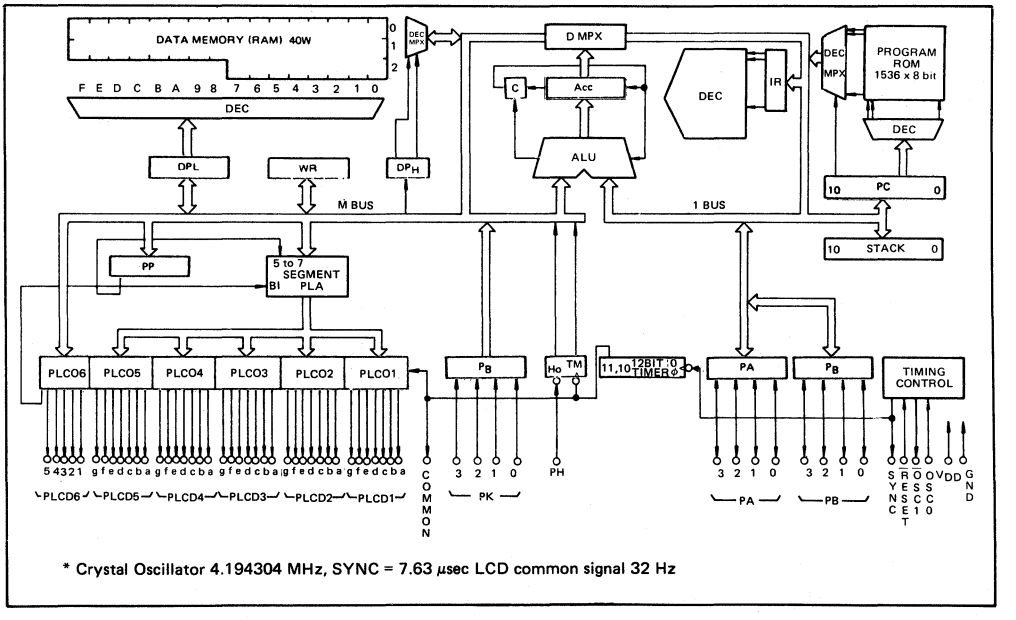
Integrated within this one chip is a 5 digit 7-segment LCD driver and PLA which can change the character font for the 7 segments freely under the control of the mask programmable data.

Also integrated in this chip are mask ROM of 1536 × 8 bits for programming, data RAM of 40 × 4 bits, 13 general-purpose input/output ports, 12-bit timer, and clock oscillator to facilitate easy application to equipment with an LCD display.

FEATURES

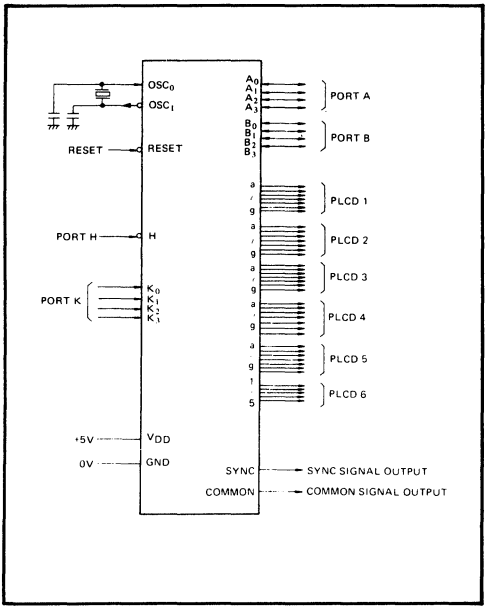
- Low Power Consumption CMOS 4-bit One-Chip Microcomputer
- 100% Static Logic
- 1536 × 8 bits Mask ROM
- 40 × 4 bits Data RAM
- 1 Static Register
- Built-in 12-bit Timer (with 32 Hz Common Output)
- All Input Ports Contain Schmitt Trigger Circuits
- 8-bit Interface Bus
- 52 Instructions
- 94% of the 52 Instructions are 1 Byte and 1 Machine Cycle
- Integrated with 13 Input/Output Ports and 40 Static LCD Driver Circuit
- +5V Single Power Supply, 60-Pin Mold Flat Package
- 7-Segment Character User Programmable Font (32 Words × 7 Segments)
- Various Functions Changeable under Mask Program Control

FUNCTIONAL BLOCK DIAGRAM

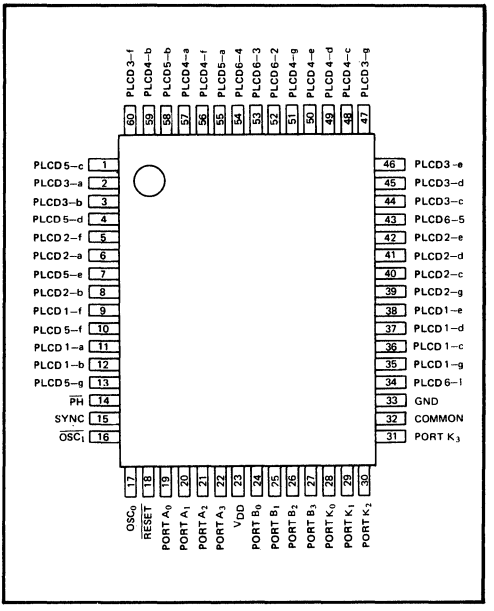


* Crystal Oscillator 4.194304 MHz, SYNC = 7.63 μsec LCD common signal 32 Hz

LOGIC SYMBOL



PIN CONFIGURATION



PIN DESCRIPTION

Designation	Pin No.	Function
GND	33	Circuit GND potential
V_{DD}	23	Main power source (+5V)
OSC_0	17	Crystal OSC input, external clock input
$\overline{OSC_1}$	16	Crystal OSC input, external clock output (not TTL compatible)
PA, PB	19 to 22 24 to 27	Pseudo-bidirectional ports for 4-bits parallel I/O. To input data from these ports, it is necessary to write "1" to them beforehand. When nothing is applied to their terminals, the content of output ports is written in them, so they can also be used as registers. In addition, it is possible to use them for make 8-bit parallel output depending on instructions.
PK	28 to 31	Input ports for 4-bit parallel input with no latching function.
PH	14	Input port with latching function to be set by negative logical signal. That is, this terminal is set at the time when the negative logical signal is applied to it from the outside. It is reset automatically after execution of the test instruction of this port.
$\overline{RESET}$	18	The RESET signal which input has priority over all of other signals and performs the following functions: (1) Resets all bits of the program counter; (2) Resets the timer counter and timer flag; (3) Resets the port pointer; (4) Resets the accumulator; (5) Resets I/O ports PA and PB; (6) Resets the input port PH flag; (7) Initializes the output port PLCD for LCD; (8) Resets the machine cycle to M_1. Since the $\overline{RESET}$ terminal is pulled up to V_{DD} by an internal resistor (approx. 800 kΩ), it is possible to make power ON/reset by connecting it with an external capacitor.

PIN DESCRIPTION (CONT.)

Designation	Pin No.	Function
SYNC	15	General-purpose synchronizing signal output. The signal is output at the beginning of each machine cycle. Output constantly, this signal is used also as clock pulse to external units. The cycle of SYNC becomes 32 times that of the original oscillation (8 μs when the clock pulse is 4 MHz).
PLCD 1 ~ 6	1 to 13 34 to 60	PLCD 1 ~ 6 (Pins 1 ~ 13 and 34 ~ 60) 7-bit and 5-bit parallel output ports, respectively. They are used to direct drive an LCD (static type). Specification of each port is done by the port pointer (PP) as shown in the table below:
COMMON	32	COMMON (Pin 32) COMMON output terminal. This output signal is connected to the common electrode of static type LCD. The frequency of the COMMON signal output is given with the following equation: $f_{(COM)} = f_{(OSC)}/2^{17}$ Where the basic clock $f_{(OSC)}$ is 4.19304 MHz, $f_{(COM)}$ becomes 32 Hz (duty ratio: 50%).

Content of PP				Port Specified
b_3	b_2	b_1	b_0	
x	0	0	0	PLCD1
x	0	0	1	PLCD2
x	0	1	0	PLCD3
x	0	1	1	PLCD4
x	1	0	0	PLCD5
x	1	0	1	PLCD6 1 ~ 4
x	1	1	0	PLCD6 5 and BI
x	1	1	1	——

X: Don't care
BI: 7 Segment Decoder PLA Blank Input
The data of b_0, of the internal 4-bit bus is written to 5 of PLCD6, and that of b_3 to BI.

MASK OPTION TABLE

f(osc)=4.194304MHz

No.	Port K0 28 pin	Port K2 30 pin	Port K3 31 pin	Commom 32 pin	Timer
0	K_0	K_2	K_3	32Hz	12 bit
1	K_0	BUZZER	$\overline{\text{COMMON}}$	32Hz	12 bit
2	K_0	BUZZER	K_3	32Hz	12 bit
3	K_0	BUZZER	K_3	64Hz	11 bit
4	K_0	K_2	K_3	64Hz	11 bit
5	K_0	K_2	K_3	128Hz	10 bit
6	K_0	BUZZER	K_3	128Hz	10 bit
7	K_0	K_2	K_3	256Hz	9 bit
8	K_0	K_2	K_3	512Hz	8 bit
9	K_0	BUZZER	K_3	512Hz	8 bit

f (BUZZER)=2048Hz

ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits	Unit
Power Supply Voltage	V_{DD}	Ta = 25°C	−0.3 to 7	V
Input Voltage	V_I	Ta = 25°C	−0.3 to V_{DD}	V
Power Dissipation	P_D	Ta = 25°C per 1 package	200	mW
Storage Temperature	T_{stg}		−55 to +150	°C

OPERATING RANGE

Parameter	Symbol	Conditions	Limits	Unit
Power Supply Voltage	V_{DD}	$f_{(OSC)}$ = 0 to 4.2 MHz	4 to 6	V
Operating Temperature	T_{OP}	–	−40 to +85	°C
Fan Out (excluding COM, SEC)	N	MOS Load	15	–
		TTL Load	1	

D.C. CHARACTERISTICS

(V_{DD} = 5V±10%, Ta = −40 to +85°C)

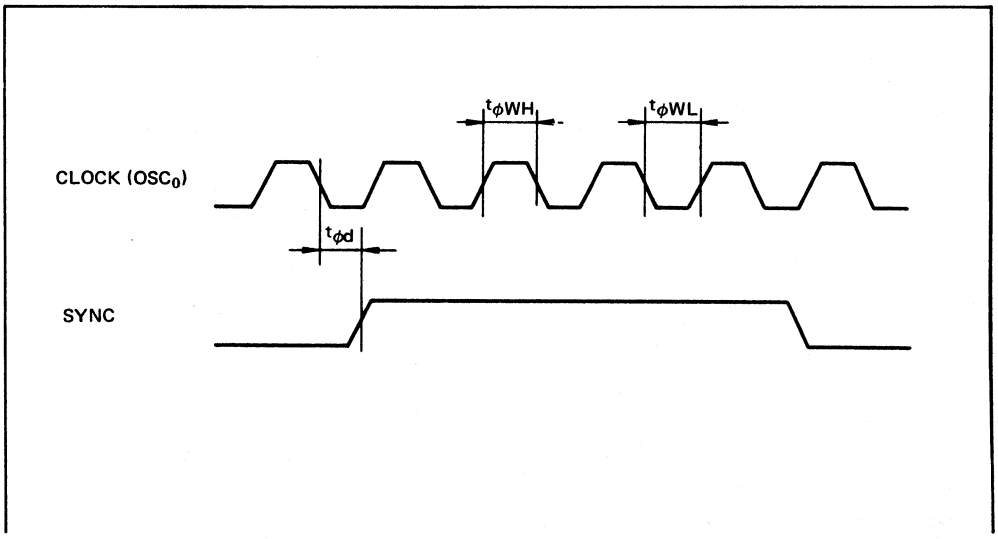
Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
"H" Input Voltage	V_{IH}		3.6			V
"L" Input Voltage	V_{IL}				0.8	V
"H" Output Voltage(1)	V_{OH}	I_O = −80μA	V_{DD} −0.1			V
"H" Output Voltage(2)	V_{OH}	I_O = −20μA	V_{DD} −0.1			V
"H" Output Voltage(3)	V_{OH}	I_O = −40μA	4.2			V
"H" Output Voltage(4)	V_{OH}	I_O = −15μA	4.2			V
"L" Output Voltage(1)	V_{OL}	I_O = 80μA			0.1	V
"L" Output Voltage(2)	V_{OL}	I_O = 20μA			0.1	V
"L" Output Voltage(5)	V_{OL}	I_O = 1.6mA			0.4	V
OSC_0 Input Leak Current	I_{IH}/I_{IL}	$V_I = V_{DD}/V_I$ = 0V			10/−10	μA
Input Current(6)	I_{IH}/I_{IL}	$V_I = V_{DD}/V_I$ = 0V			1/−20	μA
PA PB "H" Output Current	I_{OH}	V_O = 0.4V			−1	mA
"H" Output Current(3)	I_{OH}	V_O = 2.5V	−0.25			mA
"L" Output Current(4)	I_{OL}	V_O = 0.4V	1.6			mA
Input Capacity	C_I	f=1MHz, Ta=25°C		5		pF
Output Capacity	C_O	f=1MHz, Ta=25°C		7		pF
Current Consumption	I_{DD}	f = 4.194304 MHz, at no load		2	5	mA

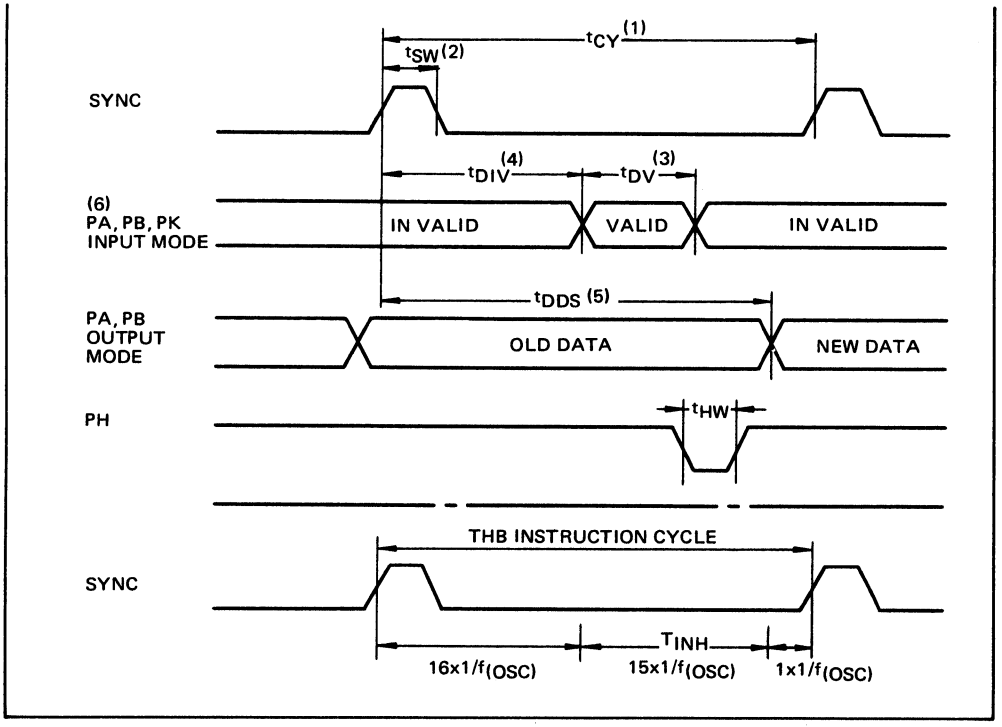
Notes: (1) Applied to COMMON
(2) Applied to SEGMENT
(3) Applied to SYNC
(4) Applied to PA, PB
(5) Applied to SYNC, PA, and PB
(6) Applied to $\overline{RESET}$, PK, and $\overline{PH}$

SWITCHING CHARACTERISTIC

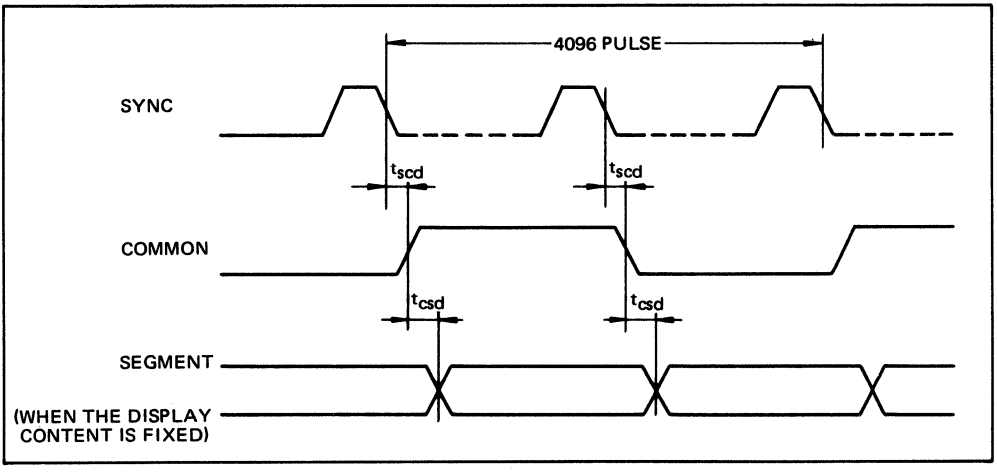
(V_{DD} = 5V±10%, Ta = −40° to +85°C)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
SYNC Delay Time from Clock (OSC_0)	$t_{\phi d}$	C_L = 50pF			800	ns
Clock (OSC_0) Pulse Width	$t_{\phi WH}$ $t_{\phi WL}$		115			ns
Cycle Time	t_{CY}		(1)			μs
SYNC Pulse Width	t_{SW}		(2)			μs
PA PB Data Valid Time PK	t_{DV}	C_L = 50pF	(3)			μs
PA PB Data Invalid Time PK	t_{DIV}	C_L = 50pF			(4)	μs
Data Delay Time	t_{DDS}	C_L = 50pF	500			ns
Port H Set Pulse Width(7)	t_{HW}		500			ns
COMMON Delay Time from SYNC	t_{SCd}	C_L = 50pF			2	μs
SEGMENT Delay Time from COMMON	t_{CSd}	C_L = 50pF			1	μs





Notes: (1) $t_{CV} = 32 \times 1/f_{(OSC)}$
(2) $t_{SW} = 4 \times 1/f_{(OSC)}$
(3) $t_{DV} = 8 \times 1/f_{(OSC)}$
(4) $t_{DIV} = 16 \times 1/f_{(OSC)} + 0.5\,\mu s$
(5) $t_{DDS} = 26 \times 1/f_{(OSC)} + 1\,\mu s$
(6) When data is input from PA or PB, set the contents of PA or PB to "1" prior to reading instruction.
(7) At execution of the THB instruction, any input made during a period of T_{INH} ($15 \times 1/f_{(OSC)}$) shown in the above figure may be neglected.



DESCRIPTION OF TERMINALS

GND (Pin 33)
Circuit grounding potential

V_{DD} (Pin 23)
Main power supply

OSC_0 (Pin 17)
Input of internal oscillation circuit at one side of crystal resonator and ceramic vibrator.

$\overline{OSC_1}$ (Pin 16)
Output of internal oscillation circuit at the other side of crystal resonator and ceramic vibrator (not TTL compatible)

PA, PB (Pins 19 ~ 22 and 24 ~ 27)
These are quasi-bidirectional ports for 4-bit parallel I/O. To input data from these ports, it is necessary to write "1" to them beforehand. When nothing is applied to their terminals, the content of output ports is written in them, so they can also be used as registers. In addition, it is possible to use them to make an 8-bit parallel output depending on instructions.

PK (Pins 28 ~ 31)
Input ports for 4-bit parallel input with no latching function.

$\overline{PH}$ (Pin 14)
Input port with latching function to be set by negative logical signal.
This terminal is set at the time when the negative logical signal is applied to it from the outside. It is reset automatically after execution of the test instruction of this port.

$\overline{RESET}$ (Pin 18)
Reset must be active for greater than 1 machine cycle.
The RESET signal, when input, has priority over all of other signals and performs the following functions:
(1) Resets all bits of the program counter;
(2) Resets the timer counter and timer flag;
(3) Resets the port pointer;
(4) Resets the accumulator;
(5) Resets I/O ports PA and PB;
(6) Resets the input port PH flag;
(7) Initializes the output port PLCD for LCD; and
(8) Resets the machine cycle to M_1.

Since the $\overline{RESET}$ terminal is pulled up to V_{DD} by an internal resistor (approx. 800kΩ), it is possible to activate power ON/reset by connecting it to an external capacitor.

SYNC (Pin 15)
This is a general-purpose synchronizing signal output. The signal is output at the beginning of each machine cycle. Output constantly, this signal is used also as clock pulse to external units.
The cycle of SYNC becomes 32 times that of the original oscillation (8μs when the clock pulse is 4MHz).

PLCD 1 ~ 6 (Pins 1 ~ 13 and 34 ~ 60)
These are 7-bit and 5-bit parallel output ports, respectively. They are used to directly drive an LCD (static type). Specification of each port is done by the port pointer (PP) as shown in the table below;

Content of PP				Port Specified
b_3	b_2	b_1	b_0	
x	0	0	0	PLCD1
x	0	0	1	PLCD2
x	0	1	0	PLCD3
x	0	1	1	PLCD4
x	1	0	0	PLCD5
x	1	0	1	PLCD6 ~ 4
x	1	1	0	PLCD6 5 and BI
x	1	1	1	—

X: Don't Care
BI: 7 Segment Decoder PLA Blank Input
The data of b_0, of the internal 4-bit bus is written to 5 of PLCD6, and that of b_3 to BI.

COMMON (Pin 32)
This is a COMMON output terminal. This output signal is connected to the common electrode of static type LCD. The frequency of the COMMON signal output is given with the following equation:

$$f_{(COM)} = f_{(OSC)}/2^{17}$$

Where the basic clock $f_{(OSC)}$ is 4.19304 MHz, $f_{(COM)}$ becomes 32 Hz (duty ratio: 50%).

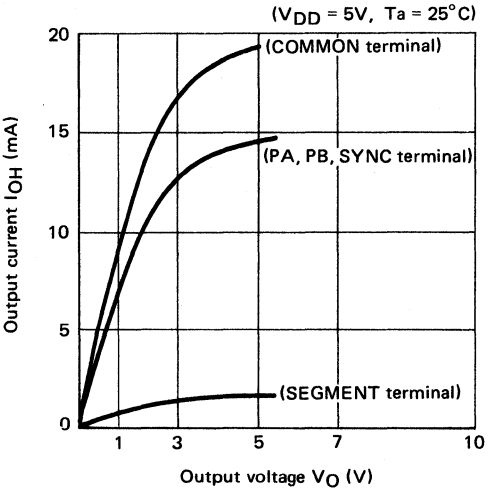
INSTRUCTIONS LIST

Mnemonic	Description	Instruction Code								Byte	Cycle
		7	6	5	4	3	2	1	0		
CLA	Clear Accumulator	0	0	0	1	0	0	0	0	1	1
CLL	Clear DP_L	0	0	1	0	0	0	0	0	1	1
CLH	Clear DP_H	0	1	1	0	0	0	0	0	1	1
LAI	Load Accumulator with Immediate	0	0	0	1	I_3	I_2	I_1	I_0	1	1
LLI	Load DP_L with Immediate	0	0	1	0	I_3	I_2	I_1	I_0	1	1
LHI	Load DP_H with Immediate	0	1	1	0	0	I_0	I_1	I_0	1	1
L	Load Accumulator with Memory	1	0	0	1	0	1	0	0	1	1
LAL	Load Accumulator with DP_L	0	1	0	1	0	1	0	1	1	1
LLA	Load DP_L with Accumulator	0	1	0	1	0	1	0	0	1	1
LAW	Load Accumulator with W Register	1	0	0	0	0	1	0	0	1	1
SI	Store Accumulator to Memory then Increment DP_L	1	0	0	1	0	0	0	0	1	1
LWA	Load W Register with Accumulator	1	0	0	0	0	0	0	0	1	1
LPA	Load Port Pointer with Accumulator	0	1	0	1	1	0	0	0	1	1
LTI	Load Timer with Immediate "0" (Clear Timer & TMF)	0	1	1	0	1	0	0	0	1	1
X	Exchange Accumulator with Memory	1	0	0	1	1	0	0	0	1	1
INA	Increment Accumulator	0	0	0	0	0	0	0	1	1	1
INL	Increment DP_L	0	1	0	1	0	1	1	1	1	1
INM	Increment Memory	0	1	0	1	1	1	0	1	1	1
INW	Increment W Register	1	0	0	0	1	0	0	0	1	1
DCA	Decrement Accumulator	0	0	0	0	1	1	1	1	1	1
DCL	Decrement DP_L	0	1	0	1	0	1	1	0	1	1
DCM	Decrement Memory	0	1	0	1	1	1	0	0	1	1
CAO	Complement Accumulator of One	0	1	0	1	0	0	0	0	1	1
RAL	Rotate Accumulator Left through Carry	0	1	0	0	0	1	1	1	1	1
AC	Add Memory to Accumulator with Carry	0	1	0	0	1	1	0	0	1	1
AS	Add Memory to Accumulator, Skip if Carry	0	1	0	0	1	1	1	0	1	1
AIS	Add Immediate to Accumulator, Skip if Carry	0	0	0	0	I_3	I_2	I_1	I_0	1	1
DAS	Decimal adjust Accumulator in Subtraction	0	1	0	1	1	0	1	0	1	1
CM	Compare Accumulator with Memory	0	1	0	1	1	1	1	0	1	1
SMB	Set Memory Bit	1	0	1	1	1	0	I_1	I_0	1	1
RMB	Reset Memory Bit	1	0	1	1	1	1	I_1	I_0	1	1
TAB	Test Accoumulator Bit	1	0	1	0	0	0	I_1	I_0	1	1
TMB	Test Memory Bit	1	0	1	0	0	1	I_1	I_0	1	1
THB	Test H Port Bit	1	0	1	0	1	1	0	0	1	1
TTM	Test Time flag	1	0	1	0	1	1	1	0	1	1
TC	Test Carry flag	0	1	0	0	0	0	1	0	1	1
SC	Set Carry flag	0	1	0	0	0	0	0	0	1	1
RC	Reset Carry flag	0	1	0	0	0	0	0	1	1	1
J	Jump	0	0	1	1	0	I_{10}	I_9	I_8	2	2
		I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0		
JC	Jump in Current Page	1	1	I_5	I_4	I_3	I_2	I_1	I_0	1	1
JA	Jump with Accumulator	0	1	0	0	0	0	1	1	1	1
CAL	Call Subroutine	0	0	1	1	1	I_{10}	I_9	I_8	2	2
		I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0		
RT	Return from Subroutine	0	1	0	1	1	0	0	1	1	1
OTD	Output Table Data	0	1	1	1	0	0	0	1	1~15	2
OA	Output Accumulator to Port A	0	1	1	1	0	0	1	0	1	1
OB	Output Accumulator to Port B	0	1	1	1	0	0	1	1	1	1
OP	Output Accumulator to Port designated Port Pointer	0	1	1	1	0	1	0	0	1	1
OPM	Output Memory to Port P designated Port Pointer	0	1	1	1	0	1	1	0	1	1
IA	Input Port A in Accumulator	0	1	1	1	1	0	1	0	1	1
IB	Input Port B in Accumulator	0	1	1	1	1	0	1	1	1	1
IK	Input Port K in Accumulator	0	1	1	1	1	1	0	0	1	1
NOP	No Operation	0	0	0	0	0	0	0	0	1	1

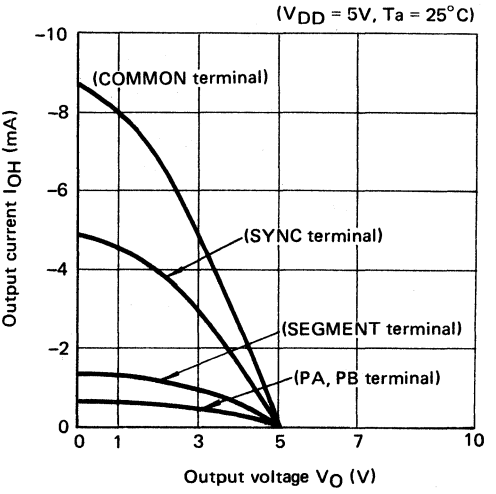
6

TYPICAL PERFORMANCE CURVES

Output Current (I_{OL}) TYP

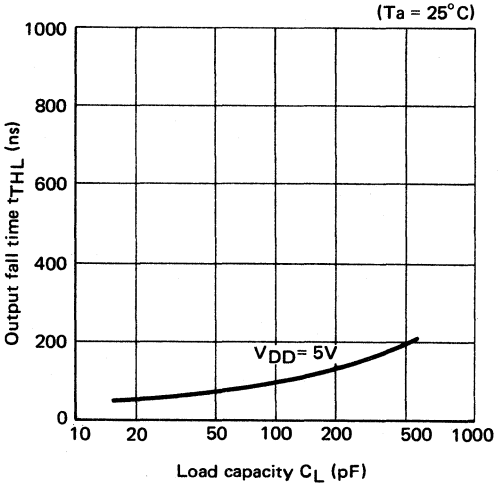


Output Current (I_{OH}) TYP

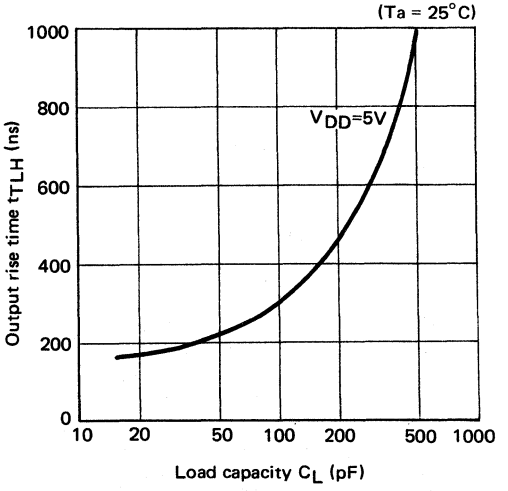


6

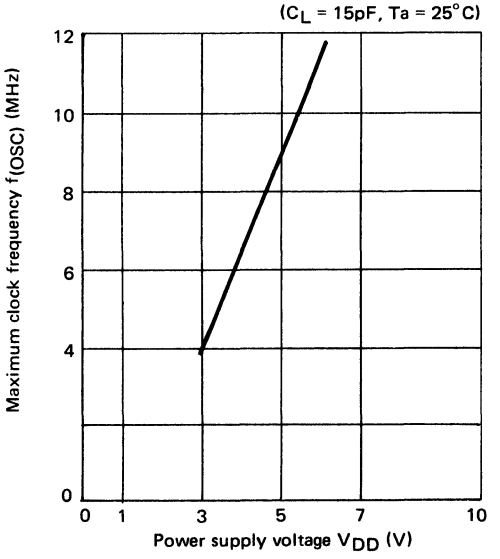
t_{THL} – C_L Characteristic TYP



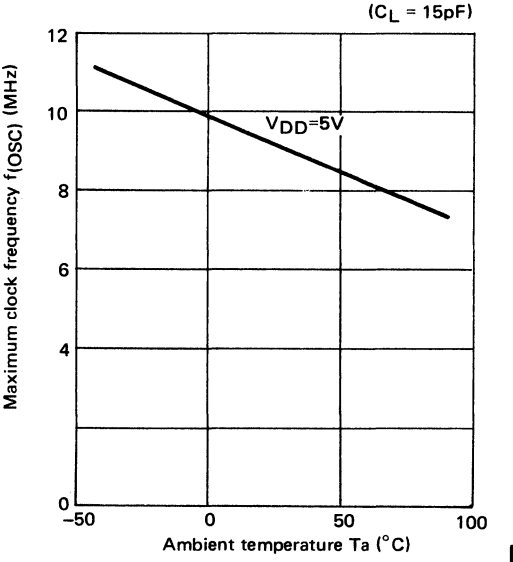
t_{TLH} – C_L Characteristic TYP



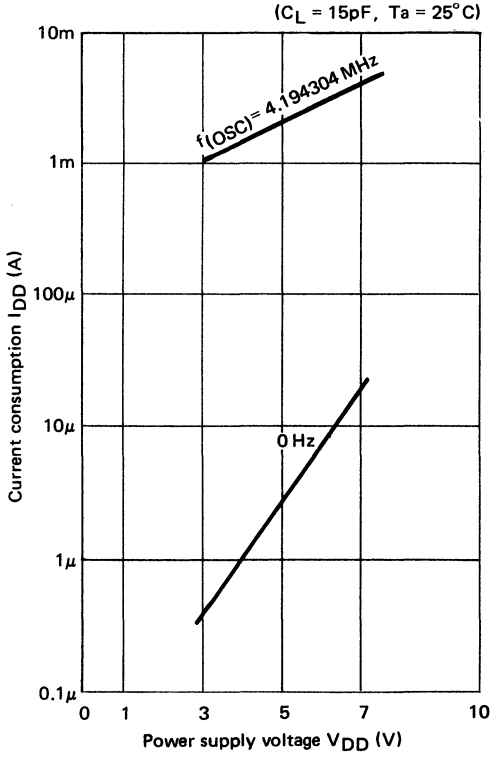
f(OSC) – V_{DD} Characteristic TYP



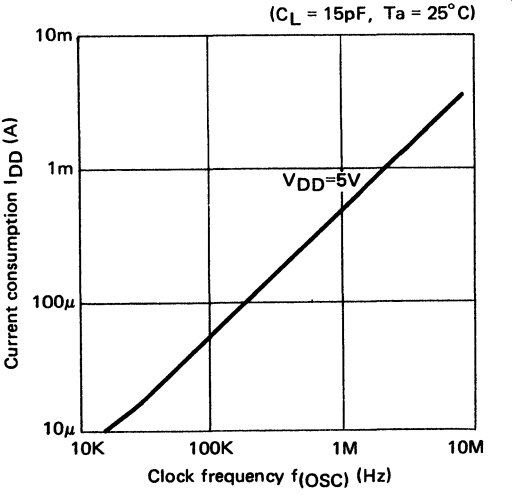
f(OSC) – Ta Characteristic TYP



I_{DD} – V_{DD} Characteristic TYP



I_{DD} – f(OSC) Characteristic TYP



6

MSM58422

CMOS 4-BIT SINGLE CHIP MICROCONTROLLER WITH FLT DRIVER

GENERAL DESCRIPTION

OKI's MSM58422 is a low-power, high-performance 4-bit single-chip microcontroller implemented in complementary metal oxide semiconductor technology.

Integrated within the one chip is a mask ROM of 1536 × 8 bits, RAM of 40 × 4 bits, 10 input/output ports 11-bit timer-counter, clock oscillator, 4-bit parallel arithmetic circuit, 40 static FLT drivers etc.

MSM58422 has an instruction set which consists of 4-bit arithmetic instructions, Boolean (bit) manipulation instructions (bit-set, bit-reset, bit-test), data input/output instructions, and 8-bit code translation (Table data out) instructions.

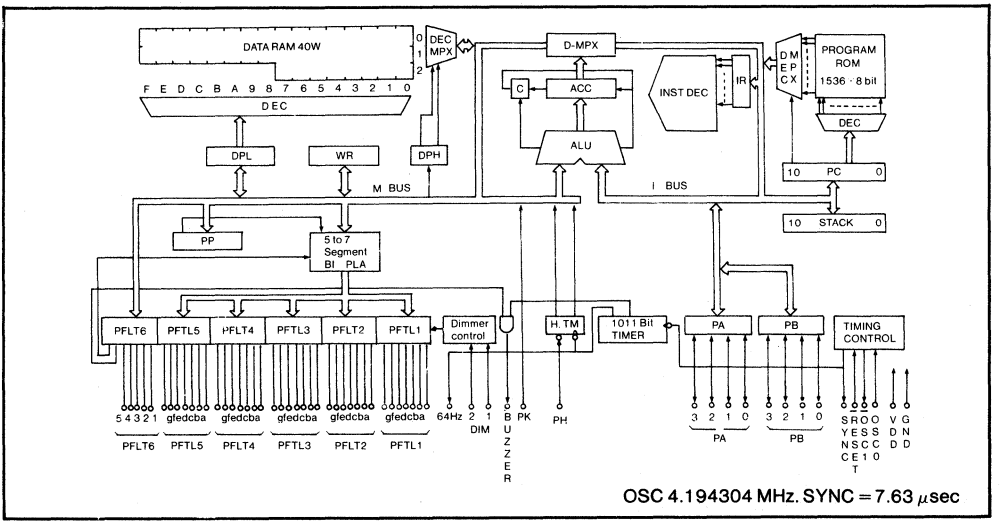
Also the pseudo-bilateral ports are used for connection to the buses of other 8-bit systems.

FEATURES

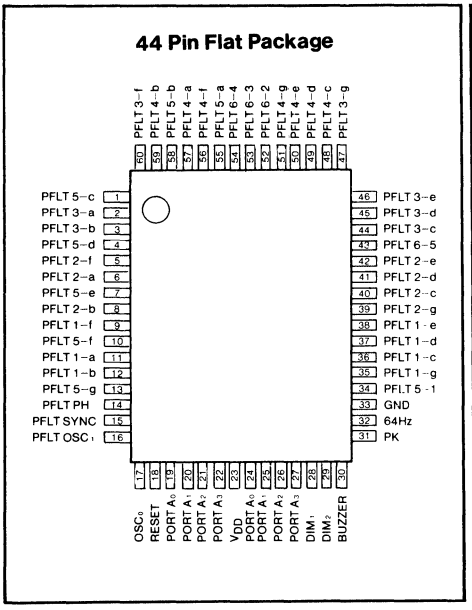
6

- Low Power Consumption CMOS 4-bit One-Chip Microcomputer
- 100% Static Logic
- 1536 × 8 bits MASK ROM
- 8-bit Interface Bus
- 1 Stack Register
- 52 Instructions
- +5V Single Power Supply
- 60-Pin Flat Package
- Built-in 11-bit Timer
- 94% of the 52 Instructions and 1 Byte and 1 Machine Cycle
- Integrated with 10 Input/Output Ports and 40 Static FLT Driver Circuit
- PK and PH Input contain Schmidt Trigger Circuits

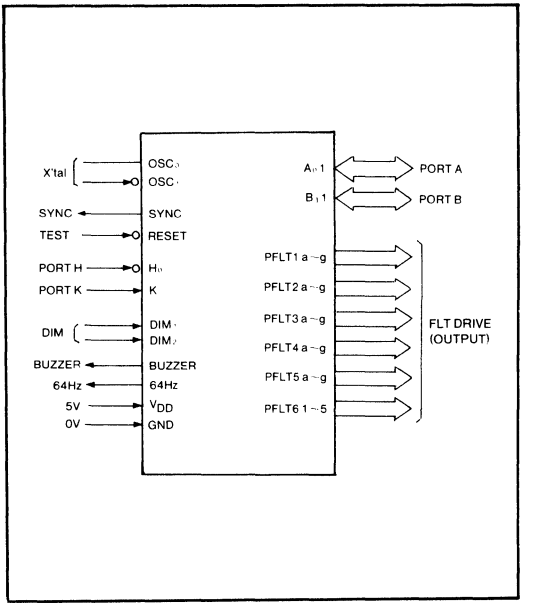
FUNCTIONAL BLOCK DIAGRAM



PIN CONFIGURATION



LOGIC SYMBOL



ABSOLUTE MAXIMUM RATING

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	Ta = 25°C	−0.3 ~ 7	V
Input Voltage	V_I	Ta = 25°C	−0.3 ~ V_{DD}	V
Output Voltage (FLT)	V_O	Ta = 25°C	V_{DD} ~ 30	V
Power Dissipation	P_D	Ta = 25°C per 1 package	200	mW
		Ta = 25°C per 1 FLT	8	
Storage Temperature	T_{stg}	–	−55 ~ +125	°C

OPERATING CONDITIONS

Parameter	Symbol	Condition	Limits	Unit
Supply Voltage	V_{DD}	f(OSC) = 0 to 4.2 MHz	4 ~ 6	V
Operating Temperature	T_{OP}	–	−40 ~ +85	°C
Output Voltage (FLT)	V_O	–	V_{DD} ~ 26	V
Fan Out (excluding FLTs)	N	MOS Load	15	–
		TTL Load	1	

DC CHARACTERISTICS

(V_{DD} = 5V±10%, Ta = −40 ~ +85°C)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
"H" Input Voltage	V_{IH}		3.6			V
"L" Input Voltage	V_{IL}				0.8	V
"H" Output Voltage (1)	V_{OH}	I_O = −15 μA	4.2			V
"H" Output Voltage (2)	V_{OH}	I_O = −40 μA	4.2			V
"L" Output Voltage (3)	V_{OL}	I_O = 1.6 mA			0.4	V
OSC_O Input Leak Current	I_{IH}/I_{IL}	$V_I = V_{DD}/0V$			10/−10	μA
Input Current (4)	I_{IH}/I_{IL}	$V_I = V_{DD}/0V$			1/−20	μA
"H" Output Current (1)	I_{OH}	V_O = 0.4 V			−1	mA
"H" Output Current (5)	I_{OH}	V_O = 2.5 V	−0.25			mA
"H" Output Current (6)	I_{OH}	V_O = 3 V	−1			mA
"L" Output Current (3)	I_{OL}	V_O = 0.4 V	1.6			mA
FLT Output Leak Current	I_{LO}	$V_O = V_{DD}$ − 26 V			−10	μA
Input Capacity	C_I	f = 1 MHz, Ta = 25°C		5		pF
Output Capacity	C_O	f = 1 MHz, Ta = 25°C		7		pF
Current Consumption	I_{DD}	f = 4.2 MHz at no load		2	5	mA

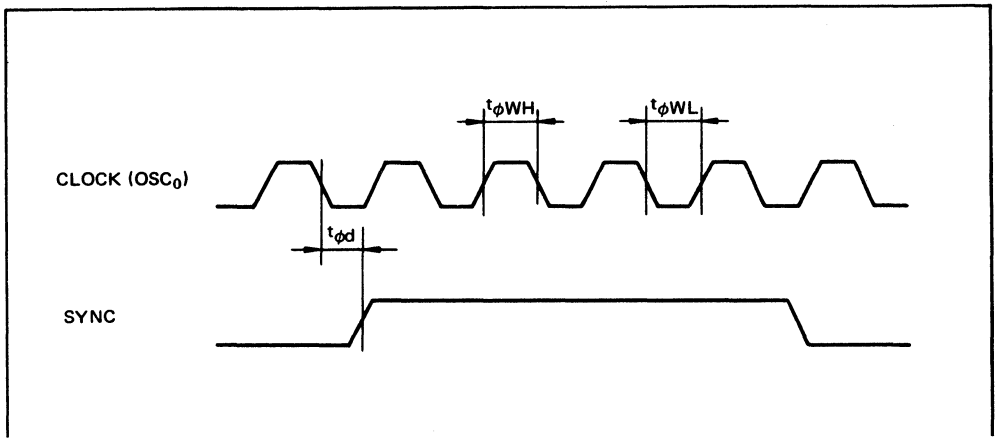
Notes: (1) Applied to PA, PB
(2) Applied to SYNC, BUZZER, 64 Hz and PFTL
(3) Applied to PA, PB, SYNC, BUZZER and 64Hz
(4) Applied to PH, RESET, DIM and PK
(5) Applied to SYNC and 64Hz
(6) Applied to BUZZER and PFLT

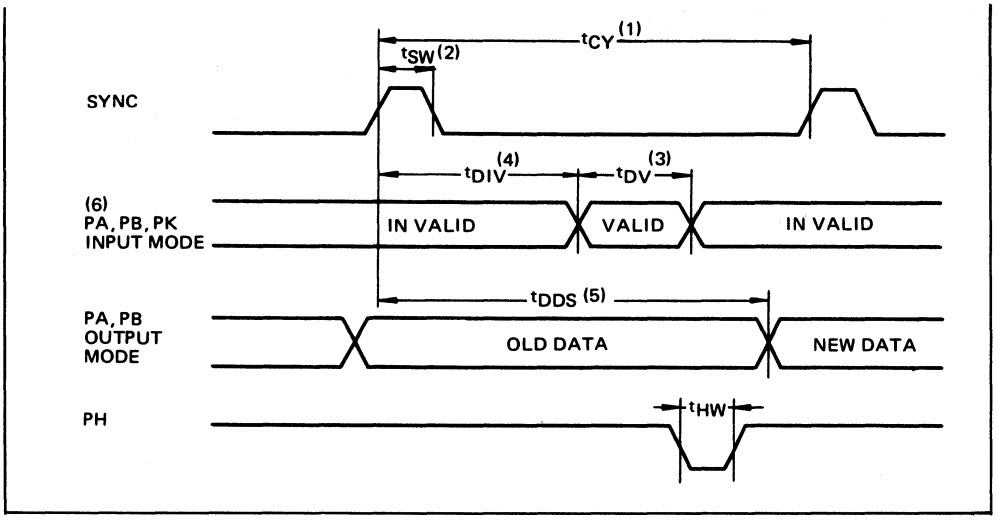
SWITCHING CHARACTERISTICS

(V_{DD} = 5V±10%, Ta = −40 ~ +85°C)

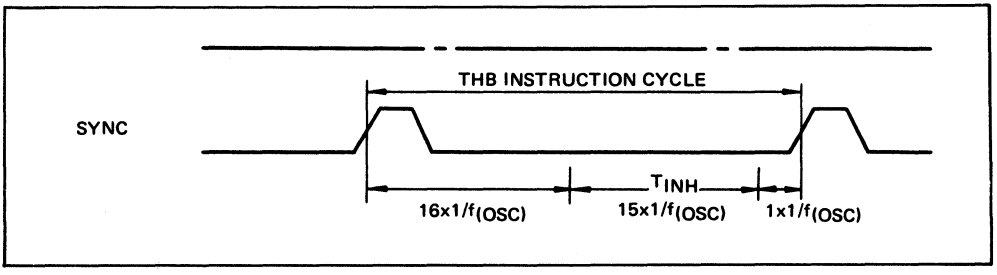
Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
SYNC Delay Time from Clock (OSC_0)	$t_{\phi d}$	C_L = 50pF			800	ns
Clock (OSC_0) Pulse Width	$t_{\phi WH}$ $t_{\phi WL}$		115			ns
Cycle Time	T_{CY}		(1)			μs
SINC Pulse Width	t_{SW}		(2)			μs
PA PB Data Valid Time PK	t_{DV}	C_L = 50pF	(3)			μs
PA PB Data Invalid Time PK	t_{DV}	C_L = 50pF			(4)	μs
Data Delay Time	t_{DDS}	C_L = 50pF			(5)	ns
Port H Set Pulse Width (8)	t_{HW}		500			ns
64Hz Delay Time from SYNC	t_{SFD}	C_L = 50pF			2	μs
BUZZER Delay Time from SYNC	t_{SBD}	C_L = 50pF			2	μs
SEGMENT Delay Time from SYNC	t_{SSD}	C_L = 50pF (8)			2	μs

Notes: (1) $t_{CY} = 32 \times 1/f_{(OSC)}$
(2) $t_{SW} = 4 \times 1/f_{(OSC)}$
(3) $t_{DV} = 8 \times 1/f_{(OSC)}$
(4) $t_{DIV} = 16 \times 1/f_{(OSC)} + 0.5\ \mu s$
(5) $t_{DDS} = 26 \times 1/f_{(OSC)} + 1\ \mu s$

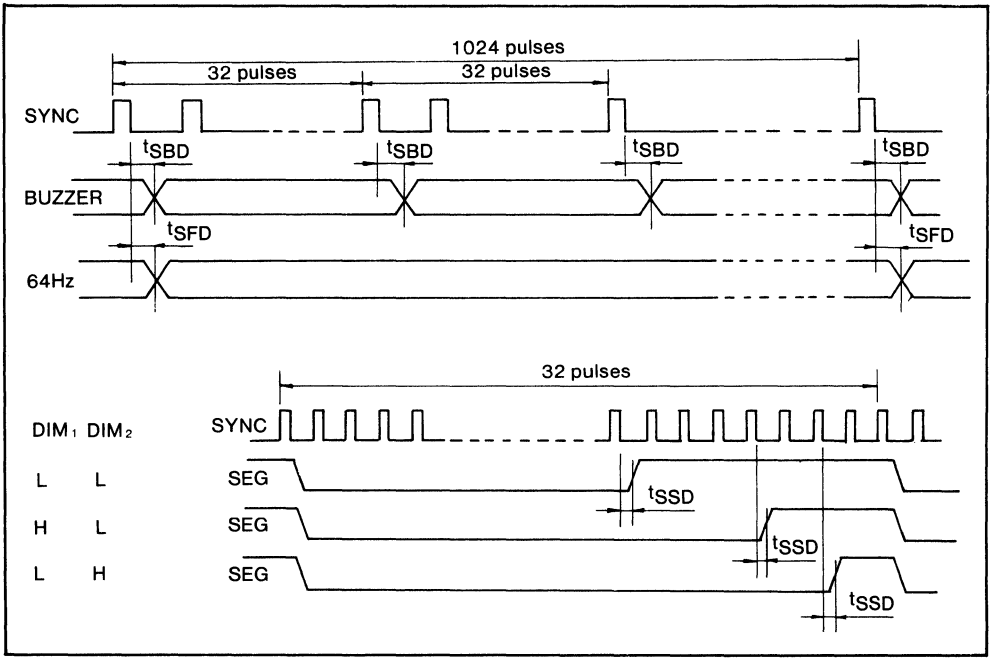




Notes: (6) When data input from PA or PB, set the contents of PA or PB to "1" prior to reading instruction.
(7) Alteration by the instructions relative to output ports PFLT (It is in the case that the outputs of open drain are pulled down to GND by a resistor below 20 kΩ).



Notes: (8) At execution of the THB instruction, any input made during a period of T_{INH} ($15 \times 1/f_{(OSC)}$ shown in the above figure may be neglected.



Notes: (9) The waveform shown above is in lighting up state, in the case that that open-drain output of FLT driver is pulled down to GND by a resistor below 20 kΩ.
DIM_1 and DIM_2 inputs must be in the state specified above.

6

DESCRIPTION OF TERMINALS

GND (Pin 33)
Circuit grounding potential

V_{DD} (Pin 23)
Main power supply

OSC_0 (Pin 17)
Input of the internal oscillation circuit at one side of the crystal resonator and ceramic vibrator.

OSC_1 (Pin 16)
Output of the internal oscillation circuit at the other side of the crystal resonator and ceramic vibrator (not TTL compatible)

PA, PB (Pins 19 ~ 22 and 24 ~ 27)
These are quasi-bidirectional ports for a 4-bit parallel I/O. To input data from these ports, it is necessary to write a "1" to them beforehand. When nothing is applied to their terminals, the content of the output ports is written into them, so they can also be used as registers. In addition, it is possible to use them to make an 8-bit parallel output depending on instructions.

PK (Pin 31)
1-bit input port with no latching function. Contains Schmidt a Schmidt Trigger Circuit.

PH (Pin 14)
Input port with latching function to be set by negative logical signal.

This terminal is set at the time the negative logical signal is applied to it from the outside. It is reset automatically after execution of the test instruction at this port.

RESET (Pin 18)
Reset must be active for greater than 1 machine cycle.

The RESET signal input has priority over all of other signals and performs the following functions:

(1) Resets all bits of the program counter;
(2) Resets the latches of I/O ports PA, PB and output port PFLT6;
(3) Resets the timer flag (TMF);
(4) Resets the accumulator;
(5) Resets the skip F/F circuit;
(6) Resets the machine cycle to MI;
(7) Resets the output port PFLT5–1 to the data of 7 Seg PLA address 0;

Since the RESET terminal is pulled up to V_{DD} by an internal resistor (approx. 800 kΩ), it is possible to activate power ON/reset by connecting it to an external capacitor.

SYNC (Pin 15)
This is a general-purpose synchronizing signal output. The signal is output at the beginning of each machine cycle. Output constantly, this signal is used also as a clock pulse to external units.

One SYNC cycle is 32 times that of the original oscillation (8 μs when the clock pulse is 4 MHz).

PFLT 1 ~ 6 (Pins 1 ~ 13 and 34 ~ 60)
These are 7-bit and 5-bit parallel output ports, respectively. They are used to directly drive an FLT (static type). Specification of each port is accomplished by the port pointer (PP), which is a 4-bit register and is set to the contents of the accumulator by the LPA instruction.

Latching data of each port is output through the logical AND operation with the DIMA signal (later described) and via buffer circuits. (When the data in the latch is 1, DIMA is output. When it is 0, the output is at high impedance.)

Content of PP				Port Specified
b_3	b_2	b_1	b_0	
×	0	0	0	PFLT1
×	0	0	1	PFLT2
×	0	1	0	PFLT3
×	0	1	1	PFLT4
×	1	0	0	PFLT5
×	1	0	1	PFLT6-1 ~ 4
×	1	1	0	PFLT6-5, BI and BZ
×	1	1	1	–

×: Don't care
BI: 7 Segment Decoder PLA Blank Input
BZ: Control Signal output for the buzzer output

The inputs with the latching function of ports PFLT1 ~ 5 are connected to the outputs of the 7 segment decoder PLA and that of PFLT6 directly to the internal buses.

DIM1, DIM2 (Pins 28, 29)
Input terminals for the dimmer control of output ports PFLT1-6.

DIM1	DIM2	DIMA
0	0	1/4 duty
1	0	1/8 duty
0	1	1/16 duty
1	1	1

BUZZER (Pin 30)
This terminal outputs the value of the logical AND operation between latching bit 2 of output port PFLT6 and the timer output (Q5).

By externally connecting a resistor and a transistor, this BUZZER output terminal is used to control an alarm, buzzer etc.

64 Hz (Pin 32)
This is an output terminal, whose frequency is 1/65536 of the OSC0. For example, when the frequency of oscillator is 4.194304 MHz, the frequency of the 64 Hz signal output is given 64 Hz.

This output pulse is used for adjusting the frequency. Its duty is 50%.

INSTRUCTIONS LIST

Mnemonic	Description	Instruction Code								Byte	Cycle
		7	6	5	4	3	2	1	0		
CLA	Clear Accumulator	0	0	0	1	0	0	0	0	1	1
CLL	Clear DP_L	0	0	1	0	0	0	0	0	1	1
CLH	Clear DP_H	0	1	1	0	0	0	0	0	1	1
LAI	Load Accumulator with Immediate	0	0	0	1	I_3	I_2	I_1	I_0	1	1
LLI	Load DP_L with Immediate	0	0	1	0	I_3	I_2	I_1	I_0	1	1
LHI	Load DP_H with Immediate	0	1	1	0	0	I_0	I_1	I_0	1	1
L	Load Accumulator with Memory	1	0	0	1	0	1	0	0	1	1
LAL	Load Accumulator with DP_L	0	1	0	1	0	1	0	1	1	1
LLA	Load DP_L with Accumulator	0	1	0	1	0	1	0	0	1	1
LAW	Load Accumulator with W Register	1	0	0	0	0	1	0	0	1	1
SI	Store Accumulator to Memory then Increment DP_L	1	0	0	1	0	0	0	0	1	1
LWA	Load W Register with Accumulator	1	0	0	0	0	0	0	0	1	1
LPA	Load Port Pointer with Accumulator	0	1	0	1	1	0	0	0	1	1
LTI	Load Timer with Immediate "0" (Clear Timer & TMF)	0	1	1	0	1	0	0	0	1	1
X	Exchange Accumulator with Memory	1	0	0	1	1	0	0	0	1	1
INA	Increment Accumulator	0	0	0	0	0	0	0	1	1	1
INL	Increment DP_L	0	1	0	1	0	1	1	1	1	1
INM	Increment Memory	0	1	0	1	1	1	0	1	1	1
INW	Increment W Register	1	0	0	0	1	0	0	0	1	1
DCA	Decrement Accumulator	0	0	0	0	1	1	1	1	1	1
DCL	Decrement DP_L	0	1	0	1	0	1	1	0	1	1
DCM	Decrement Memory	0	1	0	1	1	1	0	0	1	1
CAO	Complement Accumulator of One	0	1	0	1	0	0	0	0	1	1
RAL	Rotate Accumulator Left through Carry	0	1	0	0	0	1	1	1	1	1
AC	Add Memory to Accumulator with Carry	0	1	0	0	1	1	0	0	1	1
AS	Add Memory to Accumulator, Skip if Carry	0	1	0	0	1	1	1	0	1	1
AIS	Add Immediate to Accumulator, Skip if Carry	0	0	0	0	I_3	I_2	I_1	I_0	1	1
DAS	Decimal adjust Accumulator in Subtraction	0	1	0	1	1	0	1	0	1	1
CM	Compare Accumulator with Memory	0	1	0	1	1	1	1	0	1	1
SMB	Set Memory Bit	1	0	1	1	1	0	I_1	I_0	1	1
RMB	Reset Memory Bit	1	0	1	1	1	1	I_1	I_0	1	1
TAB	Test Accoumulator Bit	1	0	1	0	0	0	I_1	I_0	1	1
TMB	Test Memory Bit	1	0	1	0	0	1	I_1	I_0	1	1
THB	Test H Port Bit	1	0	1	0	1	1	0	0	1	1
TTM	Test Time flag	1	0	1	0	1	1	1	0	1	1
TC	Test Carry flag	0	1	0	0	0	0	1	0	1	1
SC	Set Carry flag	0	1	0	0	0	0	0	0	1	1
RC	Reset Carry flag	0	1	0	0	0	0	0	1	1	1
J	Jump	0	0	1	1	0	I_{10}	I_9	8	2	2
		I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0		
JC	Jump in Current Page	1	1	I_5	I_4	I_3	I_2	I_1	I_0	1	1
JA	Jump with Accumulator	0	1	0	0	0	0	1	1	1	1
CAL	Call Subroutine	0	0	1	1	1	I_{10}	I_9	8	2	2
		I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0		
RT	Return from Subroutine	0	1	0	1	1	0	0	1	1	1
OTD	Output Table Data	0	1	1	1	0	0	0	1	1~15	2
OA	Output Accumulator to Port A	0	1	1	1	0	0	1	0	1	1
OB	Output Accumulator to Port B	0	1	1	1	0	0	1	1	1	1
OP	Output Accumulator to Port designated Port Pointer	0	1	1	1	0	1	0	0	1	1
OPM	Output Memory to Port P designated Port Pointer	0	1	1	1	0	1	1	0	1	1
IA	Input Port A in Accumulator	0	1	1	1	1	0	1	0	1	1
IB	Input Port B in Accumulator	0	1	1	1	1	0	1	1	1	1
IK	Input Port K in Accumulator	0	1	1	1	1	1	0	0	1	1
NOP	No Operation	0	0	0	0	0	0	0	0	1	1

6

MSM5847

CMOS 4 BIT SINGLE CHIP MICROCONTROLLER WITH LCD DRIVER

GENERAL DESCRIPTION

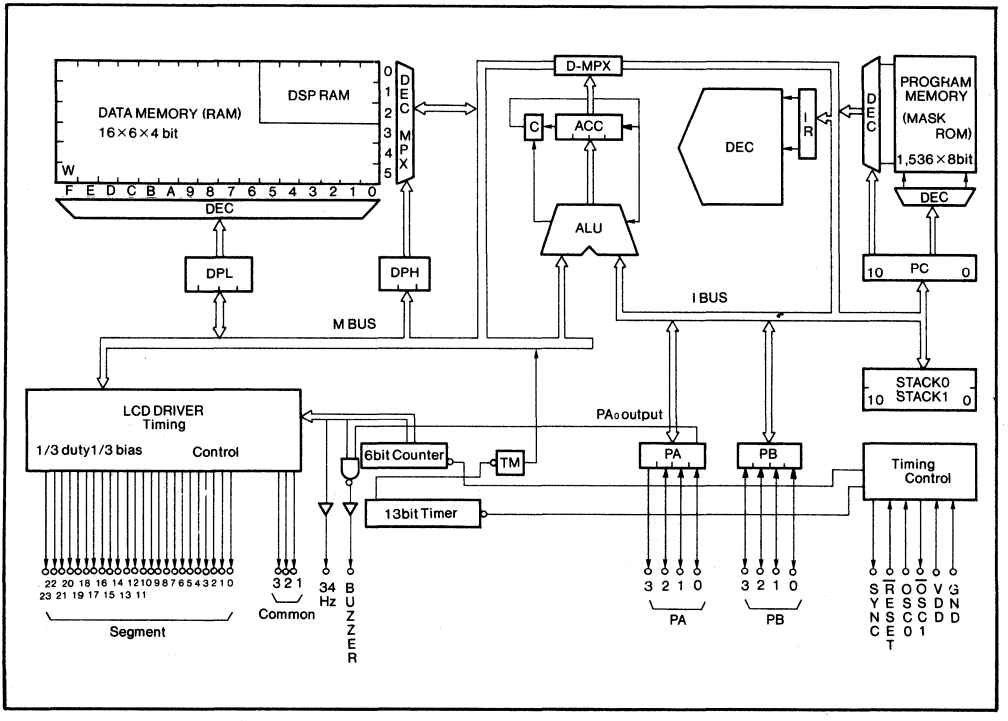
OKI's MSM5847 microcontroller is a low-power, high performance single-chip device implemented in complementary metal oxide semiconductor technology. Integrated onto single chip are 1536 × 8 bits of mask program ROM, 96 × 4 bits of data RAM, 7 input/output lines, 1 output line, a timer, LCD Driver and oscillator. Program memory is byte wide and data paths are organized in 4 bit nibbles. RAM is bit addressable. 43 instructions include binary, logical operations; bit set, reset, test; multifunctional instruction (increment, skip); subroutine call and return. 95% of instructions are single byte, single cycle operations.

FEATURES

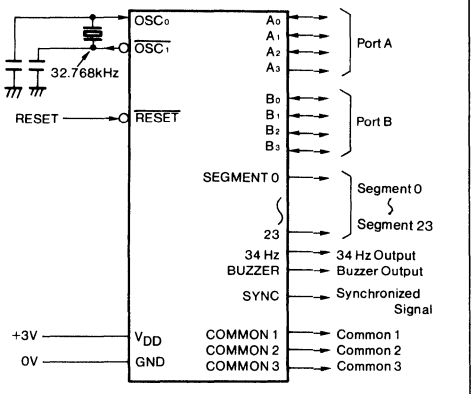
- Low Power Consumption 50 μA Typical
- 1.5K × 8 Internal ROM
- 96 × 4 Internal RAM
- 7 I/O lines, 1 output line including 8 bit data bus
- 13 bit Timer
- Self-contained Oscillator
- 43 Instructions
- 2 Stack Levels
- −20° to +70°C Operating Temperature
- 3 V Operating VDD
- 610μs Cycle Time @32.768 kHz
- 44 pin Flat Package
- Chip Form

6

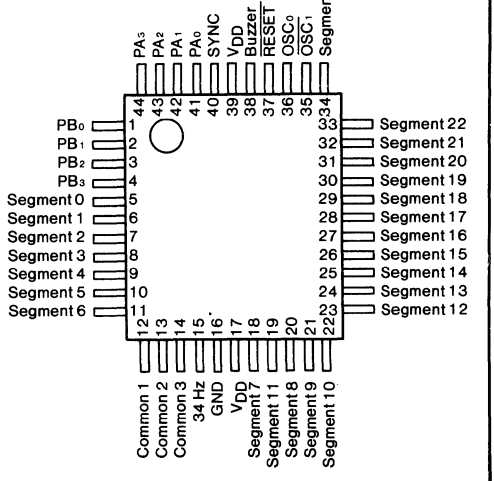
FUNCTIONAL BLOCK DIAGRAM



LOGIC SYMBOL



PIN CONFIGURATION



6

ABSOLUTE MAXIMUM RATING

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	Ta = 25°C	−0.3 ~ 7	V
Input Voltage	V_I		−0.3 ~ V_{DD}	V
Output Voltage	V_O		−0.3 ~ V_{DD}	V
Storage Temperature	Tstg	–	−55 ~ +125	°C

OPERATING CONDITIONS

Parameter	Symbol	Condition	Limits	Unit
Supply Voltage	V_{DD}	f(OSC) = 32.768 kHz	2.7 ~ 3.3	V
Operating Temperature	T_{OP}	–	−20 ~ +70	°C

AC CHARACTERISTICS

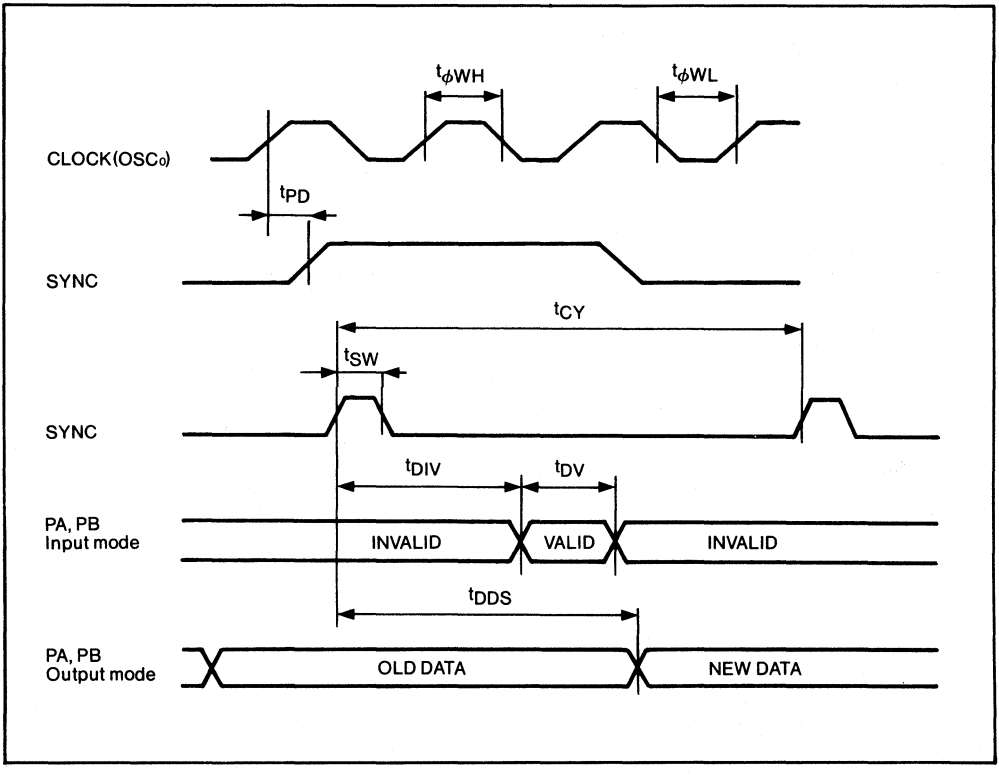
(V_{DD} = 3V±10%, Ta = −20 ~ +70°C)

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
SYNC Delay Time from Clock (OSC_O)	t_ϕ	C_L = 50 pF	–	–	5	μS
Clock (OSC_O) Pulse Width	$t_{\phi WH}$ $t_{\phi WL}$		15	–	–	μs
Cycle Time	t_{CY}	–	Note (1)	–	–	μs
SYNC Pulse Width	t_{SW}	–	Note (2)	–	–	μs
PA, PB Data Valid Time	t_{DV}	–	Note (3)	–	–	μs
PA, PB Data Invalid Time	t_{DIV}	C_L=50pF	–	–	Note (4)	μs
Data Delay Time	t_{DDS}	–	–	–	Note (5)	μs

Note: (1) t_{CY} = 20 × 1/f (OSC)
(2) t_{SW} = 2 × 1/f (OSC)
(3) t_{DV} = 4 × 1/f (OSC)
(4) t_{DIV} = 8 × 1/f (OSC) + 20 μS
(5) t_{DDS} = 17 × 1/f (OSC) + 40 μS

f (OSC) = 32.768kHz
t_{CY} = 610μs
t_{SW} = 61μs
t_{DV} = 122μs
t_{DIV} = 264μs
t_{DDS} = 558.5μs

6



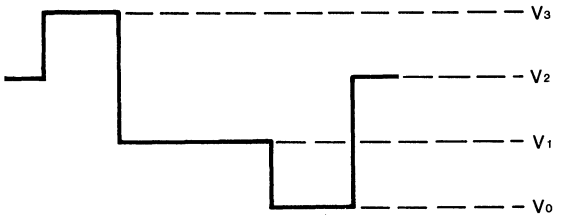
DC CHARACTERISTICS

(V_{DD} = 3V±10%, Ta = −20 ~ +70°C)

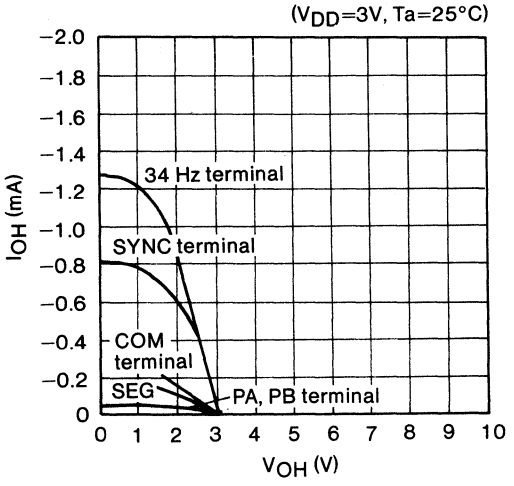
Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
"H" Input Voltage	V_{IH}	–	V_{DD} −0.2	–	–	V
"L" Input Voltage	V_{IL}	–	–	–	0.3	V
*1 Common, Segment Output Voltage	V_0	Applicable to Common 1~3 Segment 0~23	0	–	0.2	V
	V_1		1/3 V_{DD} −0.2	–	1/3 V_{DD} +0.2	V
	V_2		2/3 V_{DD} −0.2	–	2/3 V_{DD} +0.2	V
	V_3		V_{DD} −0.2	–	V_{DD}	V
"H" Output Voltage *2 *3 *4	V_{OH}	I_O = 1μA	V_{DD} −0.2	–	–	V
"L" Output Voltage *2	V_{OL}	I_O = 1μA	–	–	0.2	V
"L" Output Voltage *3 *4	V_{OL}	I_O = 1 mA	–	–	1.0	V
OSC_0 Input Current	I_{IH}/I_{IL}	$V_I=V_{DD}/V_I=0V$	–	–	5/−5	μA
$\overline{RESET}$ Input Current	I_{IH}/I_{IL}	$V_I=V_{DD}/V_I=0V$	–	–	1/−15	μA
Current Consumption	I_{DD}	V_{DD} = 3V f = 32.768kHz, no load	–	50	100	μA

Note: *1 Applicable to $PA_{0\sim2}$, $PB_{0\sim3}$, SYNC and 34 Hz
*2 Applicable to PA_3
*3 Applicable to BUZZER

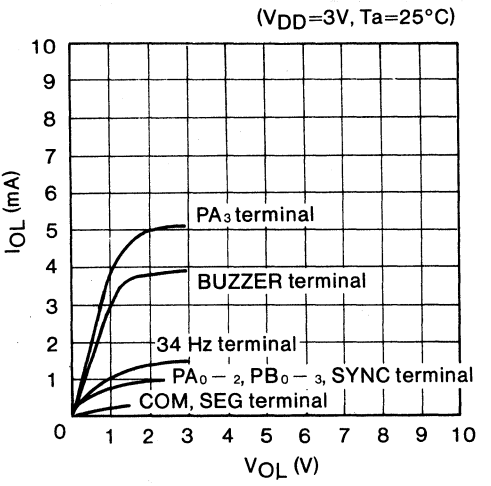
Level of Output Voltage for Common, Segment



TYP. Output Current (I_{OH}) vs Output Voltage (V_{OH}) for High-level State

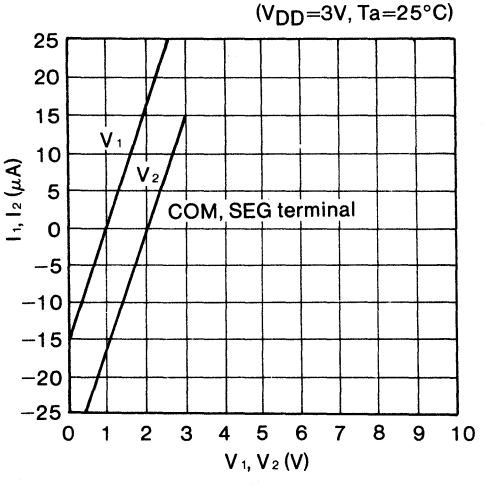


TYP. Output Current (I_{OL}) vs Output Voltage (V_{OL}) for Low-level State

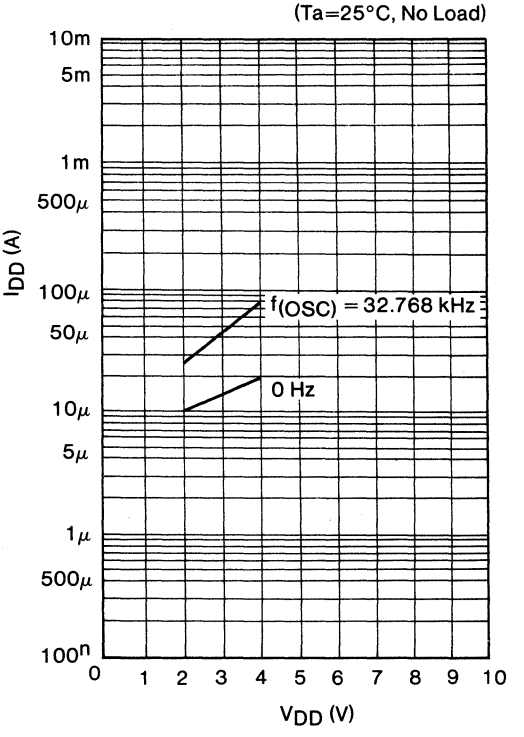


6

TYP. Output Current (I_1, I_2) vs Output Voltage (V_1, V_2) for Middle -level State



Supply Current (I_{DD}) vs Supply Voltage (V_{DD})



OLMS-50/60 SERIES

6

MSM5052

CMOS 4 BIT SINGLE CHIP VERY LOW POWER MICROCONTROLLER WITH TEMPERATURE DETECTION CIRCUIT AND LCD DRIVER

GENERAL DESCRIPTION

The OKI MSS5052 is a low-power and high-performance single-chip microcontroller employing complementary metal oxide semiconductor technology. Integrated onto a single chip are a 4 bit ALU, 18K bits of mask programmable ROM, 248 bits of data RAM, crystal oscillator, voltage doubler, timer, LCD driver, input port, output port and CR oscillator for temperature detection.

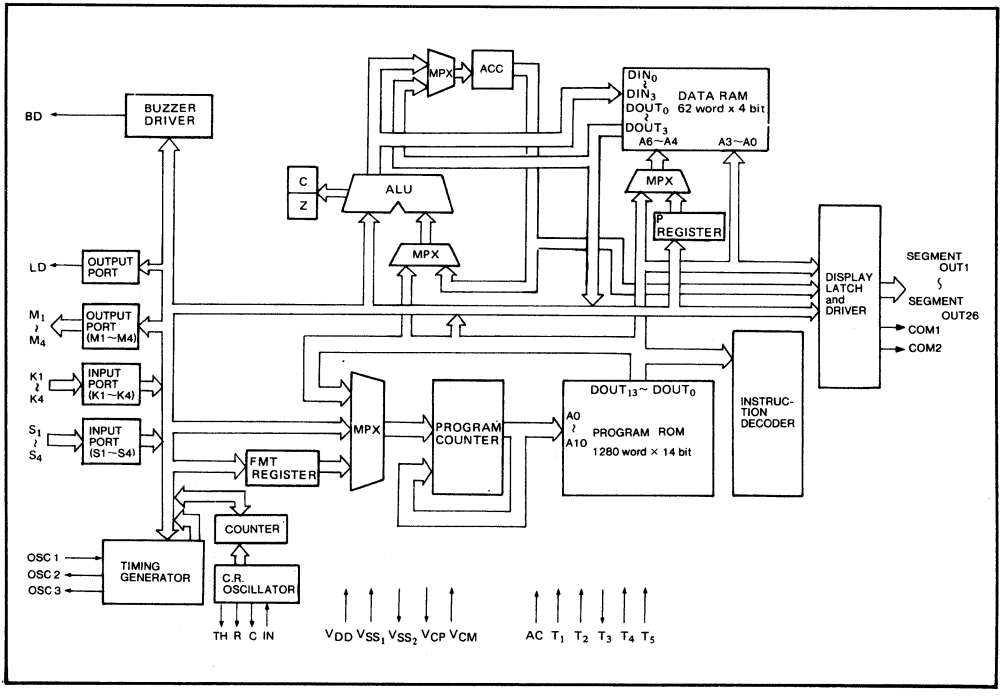
The MSM5052 is widely used in electronic products requiring low power operation, for example, thermometer and clinical thermometer with a time piece.

FEATURES

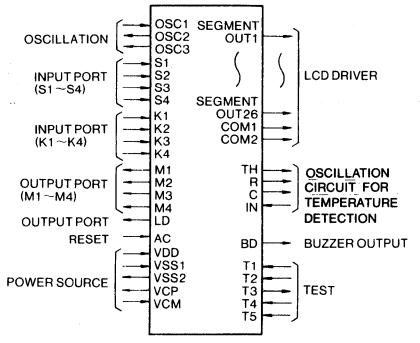
- Low Power Consumption 3 μA Typical
- 1280 × 14 Internal ROM
- 62 × 4 Internal RAM
- 4 × 2 Input Port
- 5 Output Port
- 4 × 4 Key Matrix Input (K1~K4, M1~M4)
- 26 LCD Driver (1/2 Duty, 1/2 Bias, 52 Segment)
- 42 Instructions
- 1.5 V Operating Voltage
- 32.768 kHz Crystal Oscillator
- 122.1 μs Instruction Cycle
- −20 to 75°C Operating Temperature
- 61 pad die

6

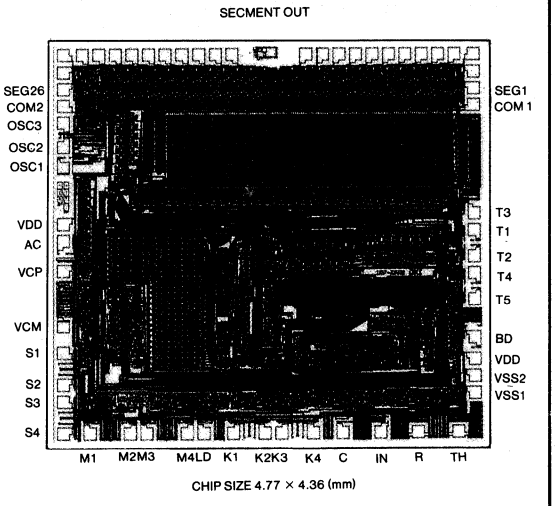
FUNCTIONAL BLOCK DIAGRAM



LOGIC SYMBOL



CHIP PAD LAYOUT



PIN DESCRIPTION

Designation	Function
V_{DD}	Circuit ground potential
V_{SS_1}	Power source (−1.5 V)
V_{SS_2}	Power source for LCD driver (−3.0 V) This terminal is connected to V_{DD} terminal through a 0.1 μF capacitor.
V_{CP}, V_{CM}	Booster capacitor connection terminals V_{CP} terminal is connected to V_{CM} terminal through a 0.1 μF capacitor.
OSC1, OSC3	Input and output terminals of oscillator inverter, 32.768 kHz crystal is connected to these terminals.
T1 ~ T5	Terminals to test internal logic, T1 ~ T3 and T5 are pulled down to V_{SS_1}. T4 is output. Test pins must be normally open.
AC	Terminal to clear internal logic pulled down to V_{SS_1}. After power is turned on, the MSM5052 must be reset by this terminal.
BD	Buzzer output
TH, R, C, IN	Terminal to CR oscillation circuit for temperature detection, fundamental resistor, thermistor, capacitor connection terminal.

FUNCTIONAL DESCRIPTION

A block diagram of the MSM5052 is given on page 91. Each block of logic will be briefly discussed. For more information, please refer to the MSM5052 user's manual.

Program ROM

The MSM5052 addresses up to 1.25 K words of internal mask programmable ROM. Each word consists of 14 bits and all instructions are one word. The instructions are routed to a programmed logic array which generates the signals necessary for control of logic.

Data RAM

Data is organized in 4-bit nibbles. Internal data RAM consists of 62 nibbles.

The RAM is addressed by page address and column address. Normally page address is specified with page register, but direct addressing is available in Page 0.

Column address is directly addressed by operand of various instructions.

ALU

The ALU performs 4-bit parallel operation on RAM and ACC contents, or RAM contents and an immediate digit. It sets or resets the flags (Z, C) depending on the condition.

Program Counter (PC)

The program counter is 11 bits wide and specifies the address of the program ROM.

The PC is incremented by one every execution of the instruction, and then specifies the next instruction to be executed. However, the contents of the PC are rewritten by the execution of the Jump or Branch instruction.

There is no boundary in the ROM, so a Jump or Branch instruction can be put anywhere in the ROM.

Input/Output Port

Input Port (S1 ~ S4)

The input port (S1 ~ S4) is a 4 bit parallel input port. Each pin of the port is pulled down to V_{SS1} by an internal resistor, and the status of the port is fetched by an input instruction.

Input Port (K1 ~ K4)

The input port (K1 ~ K4) is a 4 bit parallel input port. Each pin of the port is pulled down to V_{SS1} by an internal resistor, and the status of the port is fetched by an input instruction.

Output Port (M1 ~ M4)

The output port (M1 ~ M4) is a 4 bit parallel output port. This port consists of data latches and buffers, and the contents of the data latches are rewritten by an output instruction.

Output Port (LD)

The output port (LD) is single output port. This terminal is used for loading of M1 to M4 data.

Display Function

The MSM5052 is provided with a segment output terminal which can directly drive a 1/2 bias, 1/2 duty LCD and the common drive output terminal COM1 and COM2.

The segment drive circuit consists of the display data latch, multiplexer and driver. If the data is sent to the display data latch with a display instruction, the LCD drive waveform is output to the segment drive output terminal.

Time Base

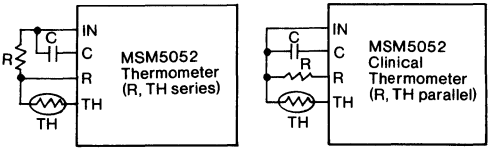
The time base for the CPU is provided by connecting a 32.768 kHz crystal to the OSC1 and OSC3 pins. One machine cycle is 122.1 μs.

A hardware divider up to 1 Hz is provided enabling programs to implement a clock function by counting signals between 16 and 1 Hz.

Temperature Detection Circuit

The temperature detection circuit is composed of an external thermistor, a fundamental resistor, a capacitor and a built-in CR oscillation circuit.

Two types of temperature measurement circuit, as shown below, are available.



ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage 1	$V_{DD} - V_{SS_1}$	Ta = 25°C	−0.3 to +2.0	V
Supply Voltage 2	$V_{DD} - V_{SS_2}$	Ta = 25°C	−0.3 to +4.0	V
Input Voltage	V_{IN_1}	Ta = 25°C	V_{SS_1} −0.3 to +0.3	V
Storage Temperature	Tstg		−55 to 125	°C

OPERATING CONDITIONS

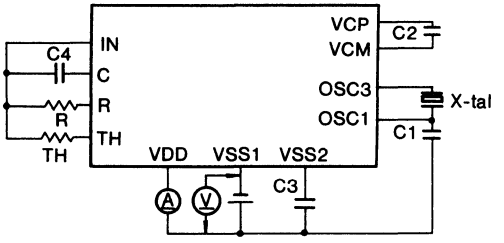
Parameter	Symbol	Limits	Unit
Operating Voltage	$V_{DD} - V_{SS_1}$	1.25 to 1.65	V
Operating Temperature	Topr	−20 to 75	°C

DC CHARACTERISTICS

(V_{DD} =0V, V_{SS_1} = −1.55V, V_{SS_2} = −3.0V, C_I = 30kΩ, Ta = 25°C)

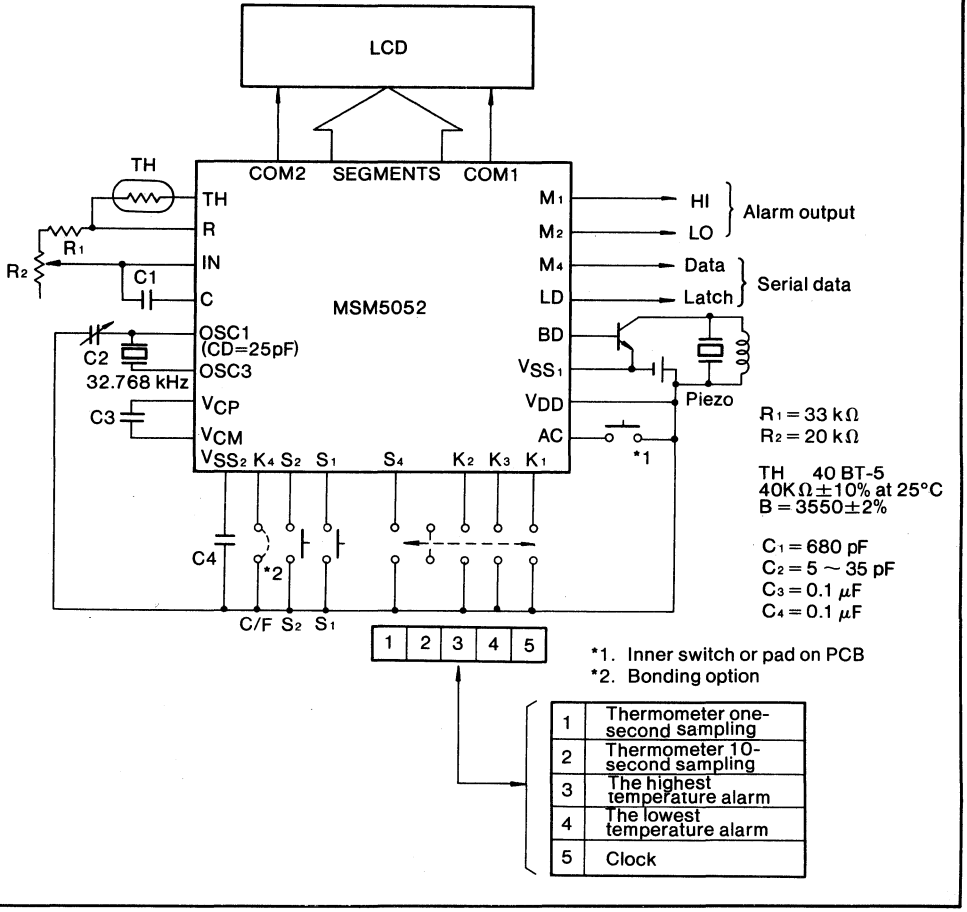
Parameter	Symbol	Condition	Limits			Unit
			Min.	Typ.	Max.	
Power supply current 1	I_{DD_1}	Temperature sampling off	–	3.0	–	μA
Power supply current 2	I_{DD_2}	Temperature sampling on	–	100	–	μA
Oscillation start voltage	$-V_{OSC}$	Within 10 seconds V_{SS_1} terminal	1.45	–	–	V
Output current 1 COM	I_{OH_1}	V_{OH_1} = −0.2V	−4	–	–	μA
	I_{OM_1}	$V_{OM_1} = V_{SS_1} \pm 0.2V$	4/−4	–	–	
	I_{OL_1}	V_{OL_1} = −2.8V	4	–	–	
Output current 2 SEGMENT	I_{OH_2}	V_{OH_2} = −0.2V	−0.4	–	–	μA
	I_{OL_2}	V_{OL_2} = −2.8V	0.4	–	–	
Output current 3 C. R. TH	I_{OH_3}	V_{OH_3} = −0.4 V	−400	–	–	μA
	I_{OL_3}	V_{OL_3} = −1.15V	400	–	–	
Output current 4 M1~M4 LD	I_{OH_4}	V_{OH_4} = −0.4V	−100	–	–	μA
	I_{OL_4}	V_{OL_4} = −1.15V	10	–	–	
Output current 5 BD	I_{OH_5}	V_{OH_5} = −0.4V	−50	–	−500	μA
	I_{OL_5}	V_{OL_5} = −1.15V	4	–	–	
Input current S1~S4 K1~K4	I_{IH_1}	V_{IH_1} = 0V	1	10	100	μA
	I_{IL_1}	V_{IL_1} = −1.55V	–	–	−0.2	
Oscillator built-in capacitor	CD		–	25	–	pF

MEASURING CIRCUIT



C1 : 20 pF
C2, C3 : 0.1 μF
C4 : 3000 pF
X-tal : 32.768 kHz
R, TH : 10 KΩ

TYPICAL APPLICATION



1	Thermometer one-second sampling
2	Thermometer 10-second sampling
3	The highest temperature alarm
4	The lowest temperature alarm
5	Clock

DESCRIPTION OF INSTRUCTIONS

	Mnemonic	Instruction Code														Operation
		13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Arithmetic operation	ADD ACC, AP	0	0	0	0	0	P	0	1	0	0	A				AP ← (AP) + (ACC)
	ADD #D, AP	0	1	1	0	0	P	D				A				AP ← (AP) − D
	ADC AP	0	0	0	0	0	P	0	1	0	1	A				AP ← Decimal adjust {(AP) + (ACC) + (C)}
	SUB ACC, AP	0	0	0	0	1	P	0	1	0	0	A				AP ← (AP) − (ACC)
	SUB #D, AP	0	1	1	0	1	P	D				A				AP ← (AP) − D
	SBC AP	0	0	0	0	1	P	0	1	0	1	A				AP ← Decimal adjust {(AP) − (ACC) − (C)}
	CMP ACC, AP	0	0	0	0	1	P	1	1	1	0	A				(AP) − (ACC)
	CMP #D, AP	0	1	0	1	1	P	D				A				(AP) − D
	INC AP	0	1	1	0	0	P	0	0	0	1	A				AP ← (AP) + 1
	DEC AP	0	1	1	0	1	P	0	0	0	1	A				A ← (AP) − 1
	XOR ACC, AP	0	0	0	0	0	P	0	1	1	1	A				AP ← (AP) ∀ (ACC)
	XOR #D, AP	0	1	1	1	1	P	D				A				AP ← (AP) ∀ D
Bit operation	BIT ACC, AP	0	0	0	0	0	P	1	1	1	0	A				(AP) V $\overline{(ACC)}$
	BIT #D, AP	0	1	0	1	0	P	D				A				(AP) V $\overline{D}$
	BIS ACC, AP	0	0	0	0	0	P	0	1	1	0	A				AP ← (AP) V (ACC)
	BIS #D, AP	0	1	0	0	0	P	D				A				AP ← (AP) V D
	BIC ACC, AP	0	0	0	0	1	P	0	1	1	0	A				AP ← (AP) Λ $\overline{(ACC)}$
	BIC #D, AP	0	1	0	0	1	P	D				A				AP ← (AP) Λ $\overline{D}$
Shift	ASR AP	0	0	0	0	0	P	0	0	1	1	A				→(C) 0 → (AP)→
	ASL AP	0	0	0	0	1	P	0	0	1	1	A				(C) ← (AP) ← 0
Flag operation	CLZ	0	0	0	0	0	0	1	0	1	0	0	0	0	0	Z ← 0
	CLC	0	0	0	0	0	0	1	0	0	1	0	0	0	0	C ← 0
	CLA	0	0	0	0	0	0	1	0	1	1	0	0	0	0	Z ← 0, C ← 0
	SEZ	0	0	0	0	1	0	1	0	1	0	0	0	0	0	Z ← 1
	SEC	0	0	0	0	1	0	1	0	0	1	0	0	0	0	C ← 1
	SEA	0	0	0	0	1	0	1	0	1	1	0	0	0	0	Z ← 1, C ← 1
Data transfer	MOV ACC, AP	1	1	1	1	0	P	0	0	0	0	A				AP ← (ACC)
	MOV ACC, AX	1	1	1	1	0	0	0		X		A				AX ← (ACC)
	MOV #D, AP	0	1	1	1	0	P	D				A				AP ← D
	MOV AP, ACC	1	1	1	1	1	P	0	0	0	0	A				ACC ← (AP)
	MOV AX, ACC	1	1	1	1	1	0	0		X		A				ACC ← (AX)
	CHG AP	1	1	1	0	0	P	0	0	0	0	A				(ACC) ⟷ (AP)
	CHG AX	1	1	1	0	0	0	0		X		A				(ACC) ⟷ (AX)

6

DESCRIPTION OF INSTRUCTIONS (CONT.)

	Mnemonic	Instruction Code														Operation
		13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Jump	JMP adrs	1	0	0	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	PC ← adrs
	JMP @AP	0	0	0	0	0	P	1	1	0	1	A				PC ← (PC) + (AP) + 1
	JMPIO @AP	0	0	0	0	1	P	1	1	0	1	A				PC ← (PC) + [(AP)A7H] + +1
	BEP +n BZE +n	0	0	0	1	1	0	0	1	0	n_4	n_3	n_2	n_1	n_0	PC ← (PC) + n + 1, if Z = 1
	BNE +n BNZ +n	0	0	0	1	1	0	1	1	0	n_4	n_3	n_2	n_1	n_0	PC ← (PC) + n + 1, if Z = 0
	BCS +n BLT +n	0	0	0	1	1	0	0	0	1	n_4	n_3	n_2	n_1	n_0	PC ← (PC) + n + 1, if C = 1
	BCC +n BGE +n	0	0	0	1	1	0	1	0	1	n_4	n_3	n_2	n_1	n_0	PC ← (PC) + n + 1, if C = 0
	BGT +n	0	0	0	1	1	0	1	1	1	n_4	n_3	n_2	n_1	n_0	PC ← (PC) + n + 1, if Z = 0 and C = 0
	BLE +n	0	0	0	1	1	0	0	1	1	n_4	n_3	n_2	n_1	n_0	PC ← (PC) + n + 1, if Z = 1 or C = 1
Input/ Output	INP Port, AP	1	1	0	1	0	P	Port				A				AP ← (Port)
	OUT AP, Port	1	1	0	1	1	P	Port				A				Port ← (AP)
	OUT #D, Port	0	0	0	1	0	0	Port				D				Port ← D
Display	DSP digit, AP	0	0	1	0	0	P	digit				A				digit ← (AP), (ACC)
	DSPF digit, AP	0	0	1	1	0	P	digit				A				digit ← (AP) via table
CPU control	HALT	0	0	0	1	0	0	0	0	0	0	0	0	0	0	Halt CPU
	NOP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	No Operation

6

MSM5054

CMOS 4 BIT SINGLE CHIP VERY LOW POWER MICROCONTROLLER WITH LCD DRIVER

GENERAL DESCRIPTION

The OKI MSM5054 is a low-power, high-performance single-chip microcontroller employing complementary metal oxide semiconductor technology. Integrated onto a single chip are 4-bit of ALU, 14K bits of mask programmable ROM, 248 bits of data RAM, crystal oscillator, voltage doubler, timer, LCD driver, input port and output port.

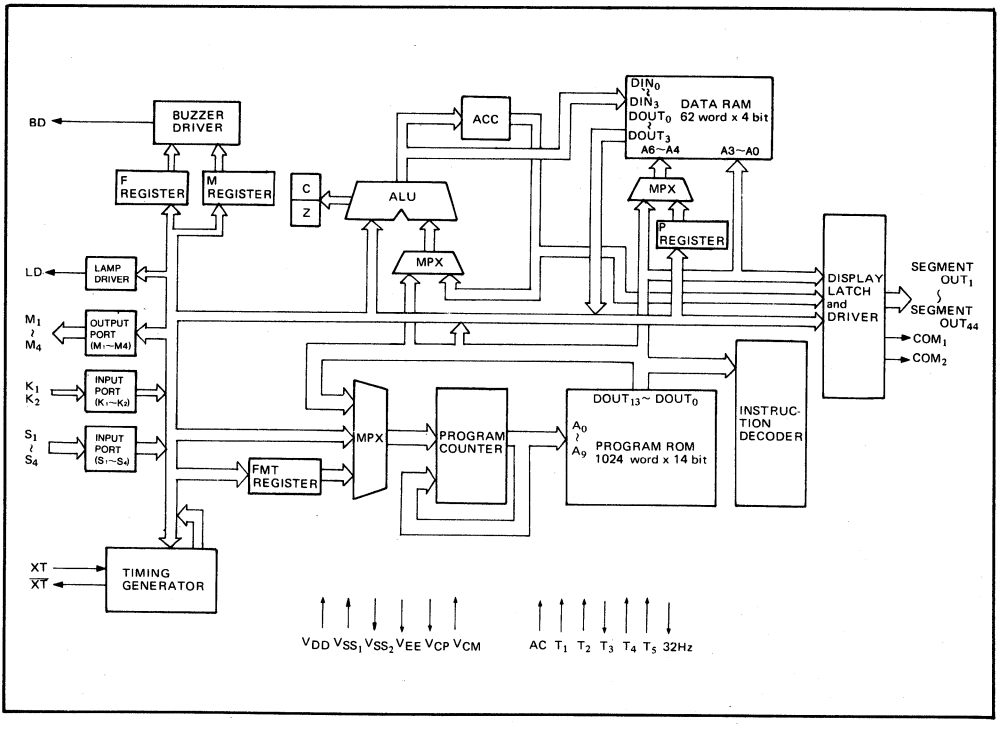
The MSM5054 is widely used in electronic products requiring low power operation, for example, Clocks, Timers and Games.

FEATURES

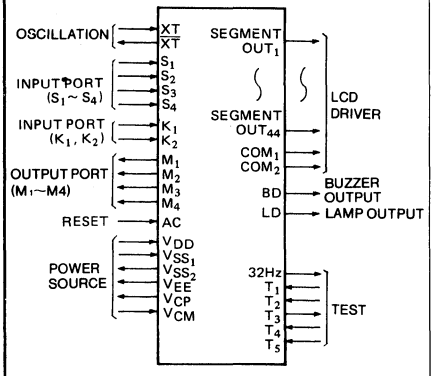
- Low Power Consumption 3 μA Typical
- 1024 × 14 Internal ROM
- 62 × 4 Internal RAM
- 6 Input Port
- 4 Output Port
- 4 × 4 Key Matrix Input (S_1~S_4, M_1~M4)
- 44 LCD Driver (1/2 Duty, 1/2 Bias, 88 Segment)
- 40 Instructions
- 1.5 V or 3 V Operating Voltage (Masking Option)
- 32.768 kHz Crystal Oscillator
- 122.1 μs Instruction Cycle
- −20 to 75°C Operating Temperature
- 74 pad die

6

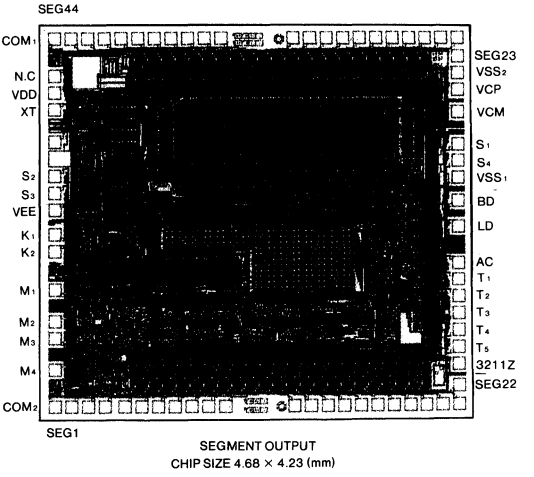
FUNCTIONAL BLOCK DIAGRAM



LOGIC SYMBOL



CHIP PAD LAYOUT



SEGMENT OUTPUT
CHIP SIZE 4.68 × 4.23 (mm)

PIN DESCRIPTION

Designation	Function
V_{DD}	Circuit ground potential
V_{SS1}	Power source (−1.5 V)
V_{SS2}	Power source for LCD driver (−3.0 V) This terminal is connected to V_{DD} terminal through a 0.1 μF capacitor.
V_{EE}	Power source for internal logic (−1.5 to −3.0 V) This terminal is connected to V_{DD} terminal through a 0.1 μF capacitor.
V_{CP}, V_{CM}	Booster capacitor connection terminals V_{CP} terminal is connected to V_{CM} terminal through a 0.1 μF capacitor.
XT, $\overline{XT}$	Input and output terminals of oscillator inverter, 32.768 kHz crystal is connected to these terminals.
$T_1 \sim T_5$	Terminals to test internal logic, $T_1 \sim T_3$ and T_5 are pulled down to V_{SS1}. T_4 is output. Test pins must be normally open.
AC	Terminal to clear internal logic pulled down to V_{SS1}. After power is turned on, the MSM5054 must be reset by this terminal.
BD	Buzzer output
LD	Lamp output

FUNCTIONAL DESCRIPTION

A block diagram of the MSM5054 is given on page 97. Each block of logic will be briefly discussed. For more information, please refer to the MSM5054 user's manual.

Program ROM

The MSM5054 addresses up to 1 K word of internal mask programmable ROM. Each word consists of 14 bits, and all instructions are one word. The instructions are routed to a programmed logic array which generates the signals necessary for control of logic.

Data RAM

Data is organized in 4-bit nibbles. Internal data RAM consists of 62 nibbles.

The RAM is addressed by page address and column address. Normally page address is specified by the page register, but direct addressing is available in Page 0.

Column address is directly addressed by the operand of various instructions.

ALU

The ALU performs 4-bit parallel operation of RAM and ACC contents, or RAM contents and an immediate digit. It sets or resets the flags (Z, C) depending on the condition.

Program Counter (PC)

The program counter is 10 bits wide and specifies the address of the program ROM.

The PC is incremented by one at every execution of the instruction, and then specifies the next instruction to be executed. However, the contents of the PC are rewritten by the execution of the Jump or Branch instruction.

There is no boundary in the ROM, so the Jump or Branch instruction can be put anywhere in the ROM.

Input/Output Port

Input Port (S1 ~ S4)

The input port (S1 ~ S4) is a 4-bit parallel input port. Each pin of the port is pulled down to V_{SS1} by an internal resistor, and the status of the port is fetched by the SWITCH instruction.

Input Port (K1 ~ K4)

The input port (K1 ~ K2) is a 2-bit parallel input port. Each pin of the port is pulled down to V_{SS1} by an internal resistor, and the status of the port is fetched by the KSWITCH instruction.

Output Port (M1 ~ M4)

The output port (M1 ~ M4) is a 4-bit parallel output port. This port consists of data latches and buffers, and the contents of the data latches are rewritten by a matrix instruction.

Display Function

The MSM5054 is provided with a segment output terminal which can directly drive a 1/2 bias, 1/2 duty LCD, and the common drive output terminal COM1 and COM2. The segment drive circuit consists of the display data latch, multiplexer and driver. If the data is sent to the display data latch with a display instruction, the LCD drive waveform is output to the segment drive output terminal.

Time Base

The time base of the CPU is provided by connecting a 32.768 kHz crystal to the XT and $\overline{XT}$ pins. One machine cycle is 122.1 μs.

A hardware divider of up to 1 Hz is provided, enabling programs to implement a clock function by counting signals between 32 and 1 Hz.

ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage 1	$V_{DD} - V_{SS_1}$	Ta = 25°C	−0.3 to +2.0	V
Supply Voltage 2	$V_{DD} - V_{SS_2}$	Ta = 25°C	−0.3 to +4.0	V
Supply Voltage 3	$V_{DD} - V_{EE}$	Ta = 25°C	−0.3 to +4.0	V
Input Voltage	V_{IN_1}	Ta = 25°C	V_{SS_1} −0.3 to +0.3	V
Storage Temperature	Tstg		−55 to 125	°C

OPERATING CONDITIONS

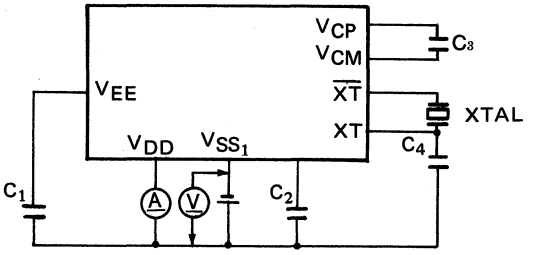
Parameter	Symbol	Limits	Unit
Operating Voltage	$V_{DD} - V_{SS_1}$	1.25 to 1.65	V
Operating Temperature	Topr	−20 to 75	°C

DC CHARACTERISTICS

(V_{DD} =0V, V_{SS_1}, V_{EE} = −1.55V, V_{SS_2} = −3.0V, C_I = 30kΩ, Ta = 25°C)

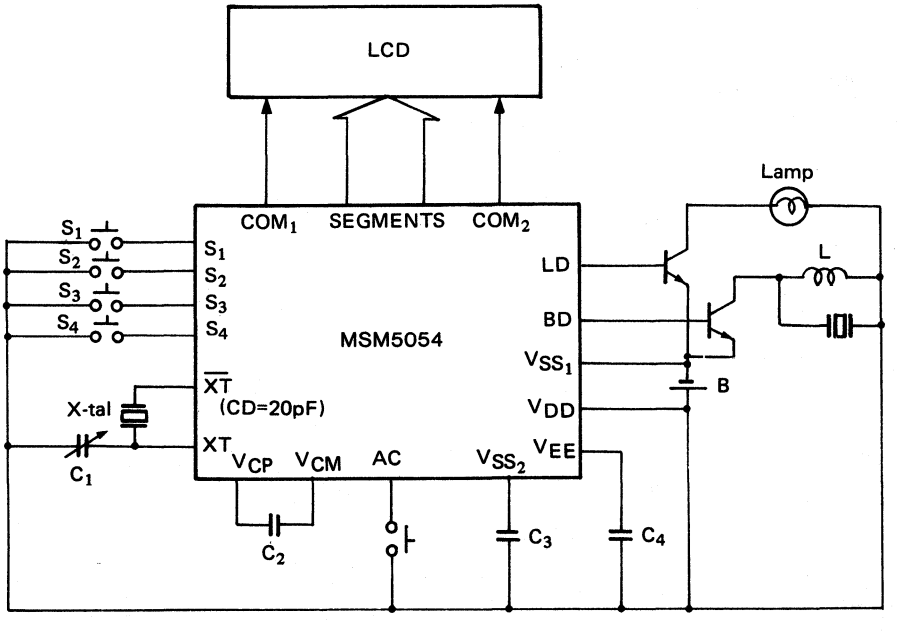
Parameter	Symbol	Condition		Limits			Unit
				Min.	Typ.	Max.	
Power supply current	I_{DD}			−	3.0	−	μA
Oscillation start voltage	$-V_{OSC}$	Within 5 seconds V_{SS_1} terminal		1.45	−	−	V
Output current 1 COM	I_{OH_1}	V_{OH_1} = −0.2V		−4	−	−	μA
	I_{OM_1}	$V_{OM_1} = V_{SS_1} \pm 0.2V$		4/−4	−	−	
	I_{OL_1}	V_{OL_1} = −2.8V		4	−	−	
Output current 2 SEGMENT	I_{OH_2}	V_{OH_2} = −0.2V		−0.4	−	−	μA
	I_{OL_2}	V_{OL_2} = −2.8V		0.4	−	−	
Output current 3 BD	I_{OH_3}	V_{OH_3} = −0.4 V	V_{SS_1} = −1.25V, V_{EE} = −1.25V, V_{SS_2} = −2.4V	−50	−	−500	μA
	I_{OL_3}	V_{OL_3} = −0.8V		2.5	−	7.5	
Output current 4 LD	I_{OH_4}	V_{OH_4} = −0.55V	V_{SS_1} = −1.25V, V_{EE} = −2.4V, V_{SS_2} = −2.4V	−21	−	−83	μA
	I_{OL_4}	V_{OL_4} = −1.15V		1	−	−	
Output current 5 $M_1 \sim M_4$	I_{OH_5}	V_{OH_5} = −0.5V		−100	−	−	μA
	I_{OL_5}	V_{OL_5} = −1.0V		1.5	−	7.5	
Input current 1 $S_1 \sim S_4$	I_{IH_1}	V_{IH_1} = 0V		1	10	50	μA
	I_{IL_1}	V_{IL_1} = −1.55V		−	−	−0.2	
Input current 2 K_1, K_2	I_{IH_2}	V_{IH_2} = 0V		2.5	6	12	μA
	I_{IL_2}	V_{IL_2} = −1.55V		−	−	−0.2	
Oscillator built-in capacitor	CD			−	20	−	pF

MEASURING CIRCUIT



C_1, C_2, C_3 : 0.1μF
C_4 : 30pF
XTAL : 32.768kHz

TYPICAL APPLICATION



C_1 : 5 ~ 35pF
C_2, C_3, C_4 : 0.1μF
B : 1.5V
L : 20mH

DESCRIPTION OF INSTRUCTIONS

	Mnemonic	Instruction Code 13 12 11 10 9 8 7 6 5 4 3 2 1 0	Operation
Arithmetic operation	ADD ACC, AP	0 0 0 0 0 P 0 1 0 0 A	AP ← (AP) + (ACC)
	ADD #D, AP	0 1 1 0 0 P D A	AP ← (AP) + D
	SUB ACC, AP	0 0 0 0 1 P 0 1 0 0 A	AP ← (AP) − (ACC)
	SUB #D, AP	0 1 1 0 1 P D A	AP ← (AP) − D
	ADJUST N, AP	1 1 0 0 0 P $\overline{N}+1$ A	AP ← N adjust {(AP)}
	CMP ACC, AP	0 0 0 0 1 P 1 1 1 0 A	(AP) − (ACC)
	CMP #D, AP	0 1 0 1 1 P D A	(AP) − D
	INC AP	0 1 1 0 0 P 0 0 0 1 A	AP ← (AP) + 1
	DEC AP	0 1 1 0 1 P 0 0 0 1 A	A ← (AP) − 1
	XOR ACC, AP	0 0 0 0 0 P 0 1 1 1 A	AP ← (AP) ∀ (ACC)
	XOR #D, AP	0 1 1 1 1 P D A	AP ← (AP) ∀ D
Bit operation	BIT ACC, AP	0 0 0 0 0 P 1 1 1 0 A	(AP) V ($\overline{ACC}$)
	BIT #D, AP	0 1 0 1 0 P D A	(AP) V $\overline{D}$
	BIS ACC, AP	0 0 0 0 0 P 0 1 1 0 A	AP V (ACC)
	BIS #D, AP	0 1 0 0 0 P D A	(AP) V D
	BIC ACC, AP	0 0 0 0 1 P 0 1 1 0 A	AP ∧ ($\overline{ACC}$)
	BIC #D, AP	0 1 0 0 1 P D A	AP ∧ $\overline{D}$
Shift	ASR AP	0 0 0 0 0 P 0 0 1 1 A	→(C) 0 → (AP)→
	ASL AP	0 0 0 0 1 P 0 0 1 1 A	(C) ← (AP) ← 0
Flag operation	CLZ	0 0 0 0 0 0 1 0 1 0 0 0 0 0	Z ← 0
	CLC	0 0 0 0 0 0 1 0 0 1 0 0 0 0	C ← 0
	CLA	0 0 0 0 0 0 1 0 1 1 0 0 0 0	Z ← 0, C ← 0
	SEZ	0 0 0 0 1 0 1 0 1 0 0 0 0 0	Z ← 1
	SEC	0 0 0 0 1 0 1 0 0 1 0 0 0 0	C ← 1
	SEA	0 0 0 0 1 0 1 0 1 1 0 0 0 0	Z ← 1, C ← 1
Data transfer	MOV ACC, AP	1 1 1 1 0 P 0 0 0 0 A	AP ← (ACC)
	MOV ACC, AX	1 1 1 1 0 0 0 X A	AX ← (ACC)
	MOV #D, AP	0 1 1 1 0 P D A	AP ← D
	MOV AP, ACC	1 1 1 1 1 P 0 0 0 0 A	ACC ← (AP)
	MOV AX, ACC	1 1 1 1 1 0 0 X A	ACC ← (AX)

6

DESCRIPTION OF INSTRUCTIONS (CONT.)

	Mnemonic	Instruction Code 13 12 11 10 9 8 7 6 5 4 3 2 1 0	Operation
Jump	JMP Adrs	1 0 0 0 a_9 a_8 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	PC ← Adrs
	JMP @AP	0 0 0 0 0 P 1 1 0 1 A	PC ← (PC) + (AP) + 1
	JMPIO @AP	0 0 0 0 1 P 1 1 0 1 A	PC ← (PC) + {(AP)∧7H } +1
	BEQ +n BZE +n	0 0 0 1 1 0 0 1 0 n_4 n_3 n_2 n_1 n_0	PC ← (PC)+n+1, if Z=1
	BNE +n BNZ +n	0 0 0 1 1 0 1 1 0 n_4 n_3 n_2 n_1 n_0	PC ← (PC)+n+1, if Z=0
	BCS +n	0 0 0 1 1 0 0 0 0 n_4 n_3 n_2 n_1 n_0	PC ← (PC)+n+1, if C=1
	BCC +n	0 0 0 1 1 0 1 0 0 n_4 n_3 n_2 n_1 n_0	PC ← (PC)+n+1, if C=0
Input/Output	SWITCH AP	1 1 0 1 0 P 0 0 0 1 A	AP ← INPUT PORT (S_1 ~ S_4)
	KSWITCH AP	1 1 0 1 0 P 0 0 1 0 A	AP ← INPUT PORT (K_1 ~ K_2)
	MATRIX AP	1 1 0 1 1 P 0 0 1 0 A	OUTPUT PORT (M_1 ~ M_4) ←(AP)
	MATRIX Mn	0 0 0 1 0 0 0 0 1 0 M_4 M_3 M_2 M_1	OUTPUT PORT (M_1~M_4) ← Mn (n=1, 2, 3, 4)
	BUZZER freq., sound	0 0 0 1 0 0 1 1 0 0 b_3 b_2 b_1 b_0	Freq ← freq, Mreg ← sound Buzzer start
	BSO	0 0 0 1 0 0 1 1 0 0 0 0 0 0	Buzzer stop
	LAMP ON/OFF	0 0 0 1 0 0 0 0 0 1 0 0 b_1 b_0	LD ON/OFF
Display	DSP digit, AP	0 0 1 0 0 P digit A	Digit ← (AP), (ACC)
	FORMAT AP	1 1 0 1 1 P 0 0 1 1 A	FMT reg. ← (AP)
	FORMAT N	0 0 0 1 0 0 0 0 1 1 N	FMT reg. ← N
	DSPF digit, AP	0 0 1 1 0 P digit A	Digit ← (AP) via table
CPU Control & Others	HALT	0 0 0 1 0 0 0 0 0 0 0 0 0 0	Halt
	INTENAB 32/16	0 0 0 1 0 0 1 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 1 0	Enable timer
	INTDSAB 32/16	0 0 0 1 0 0 1 0 1 1 0 1 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 1	Disable timer
	INTMODE AP	1 1 0 1 0 P 0 1 0 0 A	AP ← Interrupt mode
	PAGE A0	1 1 0 1 1 0 0 1 0 1 A	Preg ← (A0)
	PAGE N	0 0 0 1 0 0 0 1 0 1 N	Preg ← N
	RATE AP	1 1 0 1 0 P 1 0 0 1 A	AP ← DIVIDER (8 Hz~1 Hz)
	RSTRATE	0 0 0 1 0 0 1 0 0 0 1 0 0 0	DIVIDER (8 Hz~1 Hz) ← 0
	BACKUP ON/OFF	0 0 0 1 0 0 0 0 0 1 b_3 b_2 0 0	Backup ON/OFF
	NOP	0 0 0 0 0 0 0 0 0 0 0 0 0 0	No operation

MSM5055

CMOS 4 BIT SINGLE CHIP VERY LOW POWER MICROCONTROLLER WITH LCD DRIVER

GENERAL DESCRIPTION

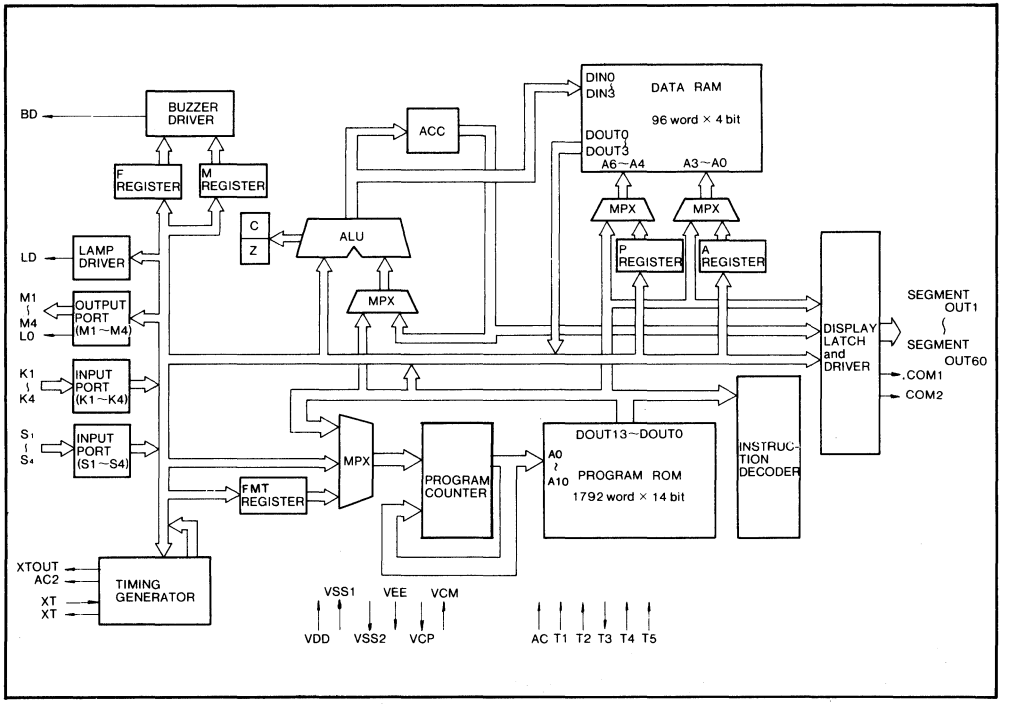
The OKI MSM5055 is a low-power, high-performance single-chip microcontroller employing complementary metal oxide semiconductor technology. Integrated onto a single chip are 4 bits of ALU, 25K bits of mask programmable ROM, 384 bits of data RAM, crystal oscillator, voltage doubler, timer, LCD driver, input port, output port and interface circuit for voice LSI (MSM6212).

The MSM5055 is widely used in electronic products requiring low power operation, for example, multi-functioned watches, voice synthesizer watches and games.

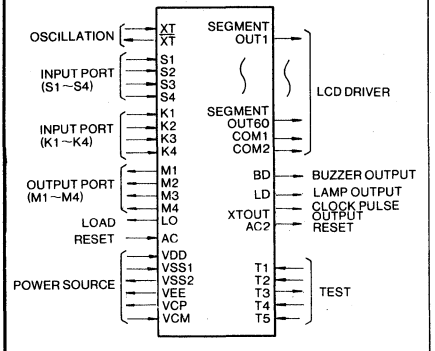
FEATURES

- Low Power Consumption 3 μA Typical
- 1792 × 14 Internal ROM
- 96 × 4 Internal RAM
- 4 × 2 Input Port
- 4 × 1 Output Port
- 4 × 4 Key Matrix Input (K1 ~ K4, M1 ~ M4)
- 60 LCD Driver
  (1/2 Duty, 1/2 Bias, 120 Segment)
- 42 Instructions
- 1.5 V or 3 V Operating Voltage
  (Masking Option)
- 32.768 kHz Crystal Oscillator
- 122.1 μS Instruction Cycle
- −20 to 75°C Operating Temperature
- 94 pad die

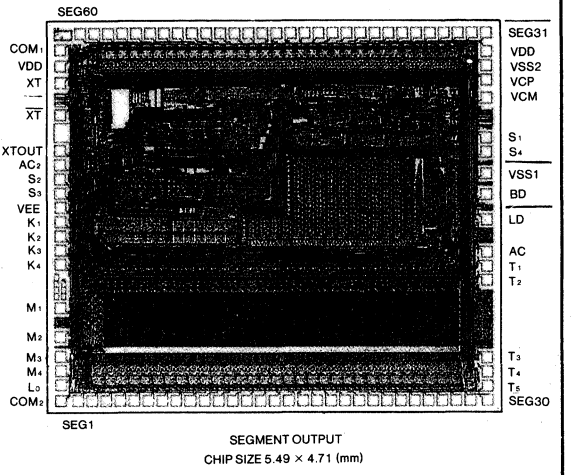
FUNCTIONAL BLOCK DIAGRAM



LOGIC SYMBOL



CHIP PAD LAYOUT



SEGMENT OUTPUT
CHIP SIZE 5.49 × 4.71 (mm)

6

PIN DESCRIPTION

Designation	Function
V_{DD}	Circuit ground potential
V_{SS_1}	Power source (−1.5 V)
V_{SS_2}	Power source for LCD driver (−3.0 V) This terminal is connected to V_{DD} terminal through a 0.1 μF capacitor.
V_{EE}	Power source for internal logic (−1.5 to −3.0 V) This terminal is connected to V_{DD} terminal through a 0.1 μF capacitor.
V_{CP}, V_{CM}	Booster capacitor connection terminals V_{CP} terminal is connected to V_{CM} terminal through a 0.1 μF capacitor.
XT, $\overline{XT}$	Input and output terminals of oscillator inverter, 32.768 kHz crystal is connected to these terminals.
T1 ~ T5	Terminals to test internal logic, T1 ~ T3 and T5 are pulled down to V_{SS_1}. T4 is output. Test pins must be normally open.
AC	Terminal to clear internal logic pulled down to V_{SS_1}. After power is turned on, the MSM5055 must be reset by this terminal.
BD	Buzzer output
LD	Lamp output
LO	Load data terminal of M_1 to M_4
AC2	Reset terminal for external circuit
XT OUT	Clock output for external circuit

FUNCTIONAL DESCRIPTION

A block diagram of the MSM5055 is given on page 104. Each block of logic will be briefly discussed. For more information, please refer to the MSM5055 user's manual.

Program ROM

The MSM5055 addresses up to 1 K word of internal mask programmable ROM. Each word consists of 14 bits, and all instructions are one word. The instructions are routed to a programmed logic array which generates the signals necessary for control of logic.

Data RAM

Data is organized in 4-bit nibbles. Internal data RAM consists of 62 nibbles.

The RAM is addressed by page address and column address. Normally page address is specified by the page register, but direct addressing is available in Page 0.

Column address is directly addressed by the operand of various instructions.

ALU

The ALU performs 4-bit parallel operation of RAM and ACC contents, or RAM contents and an immediate digit. It sets or resets the flags (Z, C) depending on the condition.

Program Counter (PC)

The program counter is 10 bits wide and specifies the address of the program ROM.

The PC is incremented by one at every execution of the instruction, and then specifies the next instruction to be executed. However, the contents of the PC are rewritten by the execution of the Jump or Branch instruction.

There is no boundary in the ROM, so the Jump or Branch instruction can be put anywhere in the ROM.

Input/Output Port

Input Port (S1 ~ S4)

The input port (S1 ~ S4) is a 4-bit parallel input port. Each pin of the port is pulled down to V_{SS1} by an internal resistor, and the status of the port is fetched by the SWITCH instruction.

Input Port (K1 ~ K4)

The input port (K1 ~ K2) is a 2-bit parallel input port. Each pin of the port is pulled down to V_{SS1} by an internal resistor, and the status of the port is fetched by the KSWITCH instruction.

Output Port (M1 ~ M4)

The output port (M1 ~ M4) is a 4-bit parallel output port. This port consists of data latches and buffers, and the contents of the data latches are rewritten by a matrix instruction.

Display Function

The MSM5055 is provided with a segment output terminal which can directly drive a 1/2 bias, 1/2 duty LCD, and the common drive output terminal COM1 and COM2. The segment drive circuit consists of the display data latch, multiplexer and driver. If the data is sent to the display data latch with a display instruction, the LCD drive waveform is output to the segment drive output terminal.

Time Base

The time base of the CPU is provided by connecting a 32.768 kHz crystal to the XT and $\overline{XT}$ pins. One machine cycle is 122.1 μs.

A hardware divider of up to 1 Hz is provided, enabling programs to implement a clock function by counting signals between 32 and 1 Hz.

ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage 1	$V_{DD} - V_{SS_1}$	Ta = 25°C	−0.3 to +2.0	V
Supply Voltage 2	$V_{DD} - V_{SS_2}$	Ta = 25°C	−0.3 to +4.0	V
Supply Voltage 3	$V_{DD} - V_{EE}$	Ta = 25°C	−0.3 to +4.0	V
Input Voltage	V_{IN_1}	Ta = 25°C	V_{SS_1} −0.3 to +0.3	V
Storage Temperature	Tstg		−55 to 125	°C

OPERATING CONDITIONS

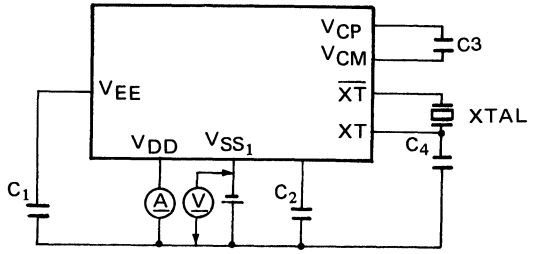
Parameter	Symbol	Limits	Unit
Operating Voltage	$V_{DD} - V_{SS_1}$	1.25 to 1.65	V
Operating Temperature	Topr	−20 to 75	°C

DC CHARACTERISTICS

($VDD = 0V$, V_{SS_1}, $V_{EE} = -1.55V$, $V_{SS_2} = -3.0V$, $C_I = 30k\Omega$, Ta = 25°C)

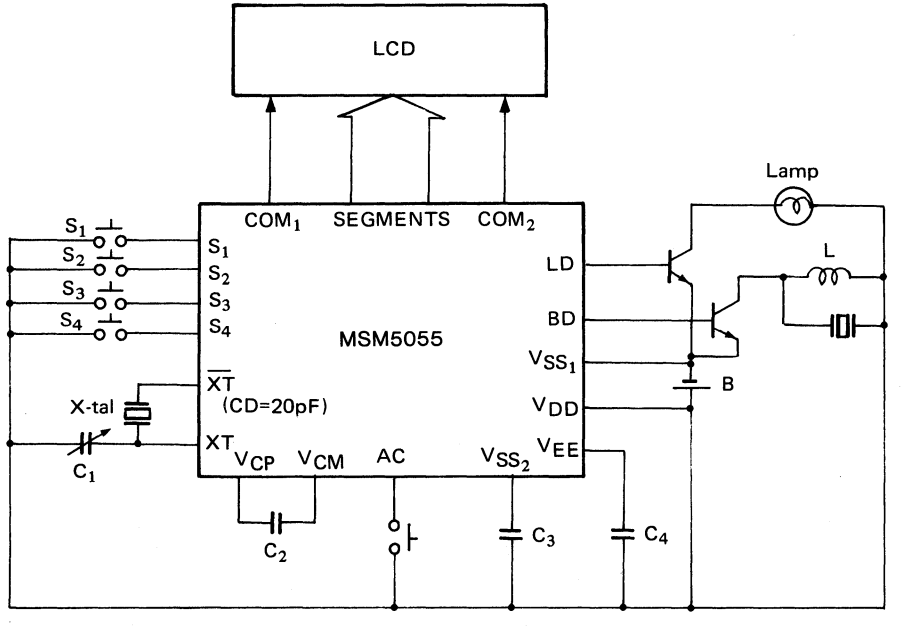
Parameter	Symbol	Conditions		Limits			Unit
				Min.	Typ.	Max.	
Power supply current	I_{DD}			−	3.0	−	μA
Oscillation start voltage	$-V_{OSC}$	Within 5 seconds V_{SS_1} terminal		1.45	−	−	V
Output current 1 COM	I_{OH_1}	$V_{OH_1} = -0.2V$		−4	−	−	μA
	I_{OM_1}	$V_{OM_1} = V_{SS_1} \pm 0.2V$		4/−4	−	−	
	I_{OL_1}	$V_{OL_1} = -2.8V$		4	−	−	
Output current 2 SEGMENT	I_{OH_2}	$V_{OH_2} = -0.2V$		−0.4	−	−	μA
	I_{OL_2}	$V_{OL_2} = -2.8V$		0.4	−	−	
Output current 3 AC2 LOAD, XTOUT	I_{OH_3}	$V_{OH_3} = -0.5V$		−10	−	−	μA
	I_{OL_3}	$V_{OL_3} = -1.05V$		10	−	−	
Output current 4 M1~M4	I_{OH_4}	$V_{OH_4} = -0.5V$		−100	−	−	μA
	I_{OL_4}	$V_{OL_4} = -1.0V$		1.5	−	12.7	
Output current 5 LD	I_{OH_5}	$V_{OH_5} = -0.55V$	$V_{SS_1} = -1.25V$ $V_{EE} = -2.4V$ $V_{SS_2} = -2.4V$	−21.6	−	−83	μA
	I_{OL_5}	$V_{OL_5} = -1.15V$		1	−	−	
Output current 5 BD	I_{OH_6}	$V_{OH_6} = -0.4V$	$V_{SS_1} = VEE$ $= -1.25V$ $V_{SS_2} = -2.4V$	−50	−	−	μA
	I_{OH_6}	$V_{OL_6} = -0.8V$		−	5	−	
Input current 1 $S_1 \sim S_4$	I_{IH_1}	$V_{IH_1} = 0V$		1	10	50	μA
	I_{IL_1}	$V_{IL_1} = -1.55V$		−	−	−0.2	
Input current 2 $K_1 \sim K_4$	I_{IH_2}	$V_{IH_2} = 0V$		2.5	6	12	μA
	I_{IL_2}	$V_{IL_2} = -1.55V$		−	−	−0.2	
Oscillator built-in capacitor	CD			−	20	−	pF

MEASURING CIRCUIT



C1 . C2 . C3 : 0.1 μF
C4 : 30 pF
X-tal : 32.768 kHz

TYPICAL APPLICATION



C1 : 5 ~ 35pF
C2, C3, C4 : 0.1μF
B : 1.5V
L : 20mH

DESCRIPTION OF INSTRUCTIONS

	Mnemonic	Instruction Code														Operation
		13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Arithmetic operation	ADD ACC, AP	0	0	0	0	0	P	0	1	0	0	A				AP ← (AP) + (ACC)
	ADD #D, AP	0	1	1	0	0	P	D				A				AP ← (AP) + D
	SUB ACC, AP	0	0	0	0	1	P	0	1	0	0	A				AP ← (AP) − (ACC)
	SUB #D, AP	0	1	1	0	1	P	D				A				AP ← (AP) − D
	ADJUST N, AP	1	1	0	0	0	P	$\overline{N}+1$				A				AP ← N adjust {(AP)}
	CMP ACC, AP	0	0	0	0	1	P	1	1	1	0	A				(AP) − (ACC)
	CMP #D, AP	0	1	0	1	1	P	D				A				(AP) − D
	INC AP	0	1	1	0	0	P	0	0	0	1	A				AP ← (AP) + 1
	DEC AP	0	1	1	0	1	P	0	0	0	1	A				A ← (AP) − 1
	XOR ACC, AP	0	0	0	0	0	P	0	1	1	1	A				AP ← (AP) $\forbar{\vee}$ (ACC)
	XOR #D, AP	0	1	1	1	1	P	D				A				AP ← (AP) $\forbar{\vee}$ D
Bit operation	BIT ACC, AP	0	0	0	0	0	P	1	1	1	0	A				(AP) V $(\overline{ACC})$
	BIT #D, AP	0	1	0	1	0	P	D				A				(AP) V $\overline{D}$
	BIS ACC, AP	0	0	0	0	0	P	0	1	1	0	A				AP V (ACC)
	BIS #D, AP	0	1	0	0	0	P	D				A				(AP) V D
	BIC ACC, AP	0	0	0	0	1	P	0	1	1	0	A				AP $\wedge$ $(\overline{ACC})$
	BIC #D, AP	0	1	0	0	1	P	D				A				AP $\wedge$ $\overline{D}$
Shift	ASR AP	0	0	0	0	0	P	0	0	1	1	A				⌊(C) 0 → (AP)⌋
	ASL AP	0	0	0	0	1	P	0	0	1	1	A				(C) ← (AP) ← 0
Flag operation	CLZ	0	0	0	0	0	0	1	0	1	0	0	0	0	0	Z ← 0
	CLC	0	0	0	0	0	0	1	0	0	1	0	0	0	0	C ← 0
	CLA	0	0	0	0	0	0	1	0	1	1	0	0	0	0	Z ← 0, C ← 0
	SEZ	0	0	0	0	1	0	1	0	1	0	0	0	0	0	Z ← 1
	SEC	0	0	0	0	1	0	1	0	0	1	0	0	0	0	C ← 1
	SEA	0	0	0	0	1	0	1	0	1	1	0	0	0	0	Z ← 1, C ← 1
Data transfer	MOV ACC, AP	1	1	1	1	0	P	0	0	0	0	A				AP ← (ACC)
	MOV ACC, AX	1	1	1	1	0	0	0	X			A				AX ← (ACC)
	MOV #D, AP	0	1	1	1	0	P	D				A				AP ← D
	MOV AP, ACC	1	1	1	1	1	P	0	0	0	0	A				ACC ← (AP)
	MOV AX, ACC	1	1	1	1	1	0	0	X			A				ACC ← (AX)
Jump	JMP adrs	1	0	0	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	PC ← adrs
	JMP @AP	0	0	0	0	0	P	1	1	0	1	A				PC ← (PC) + (AP) + 1
	JMPIO @AP	0	0	0	0	1	P	1	1	0	1	A				PC ← (PC) + {(AP) $\wedge$ 7H} +1
	BEQ +n BZE +n	0	0	0	1	1	0	0	1	0	n_4	n_3	n_2	n_1	n_0	PC ← (PC)+n+1, if Z=1

DESCRIPTION OF INSTRUCTIONS (CONT.)

	Mnemonic	Instruction Code 13 12 11 10 9 8 7 6 5 4 3 2 1 0	Operation
Jump	BNE +n BNZ +n	0 0 0 1 1 0 1 1 0 n_4 n_3 n_2 n_1 n_0	PC ← (PC)+n+1, if Z=0
	BCS +n	0 0 0 1 1 0 0 0 0 n_4 n_3 n_2 n_1 n_0	PC ← (PC)+n+1, if C=1
	BCC +n	0 0 0 1 1 0 1 0 0 n_4 n_3 n_2 n_1 n_0	PC ← (PC)+n+1, if C=0
Input/Output	SWITCH AP	1 1 0 1 0 P 0 0 0 1 A	AP ← INPUT PORT (S1 ~ S4)
	KSWITCH AP	1 1 0 1 0 P 0 0 1 0 A	AP ← INPUT PORT (K1 ~ K4)
	MATRIX AP	1 1 0 1 1 P 0 0 1 0 A	OUTPUT PORT (M1 ~ M4) ←(AP)
	MATRIX Mn	0 0 0 1 0 0 0 0 1 0 M_4 M_3 M_2 M_1	OUTPUT PORT (M1~M4) ← Mn (n=1, 2, 3, 4)
	XTCP ON/OFF	0 0 0 1 0 0 1 0 0 0 0 0 b_1 b_0	XTOUT ON/OFF
	FREQ N	0 0 0 1 0 0 1 1 0 1 N	Freq ← N
	BUZZER sound	0 0 0 1 0 0 1 1 0 0 b_3 b_2 1 0	Mreg ← sound, Buzzer start
	BSO	0 0 0 1 0 0 1 1 0 0 0 0 0 0	Buzzer stop
	LAMP ON/OFF	0 0 0 1 0 0 0 0 0 1 0 0 b_1 b_0	LD ON/OFF
Display	DSP digit, AP	0 0 1 0 0 P digit A	Digit (Low part) ← (AP), (ACC)
	DSPH digit, AP	0 0 1 0 1 P digit A	Digit (High part) ← (AP), (ACC)
	FORMAT AP	1 1 0 1 1 P 0 0 1 1 A	FMT reg. ← (AP)
	FORMAT N	0 0 0 1 0 0 0 0 1 1 N	FMT reg. ← N
	DSPF digit, AP	0 0 1 1 0 P digit A	Digit (Low part) ← (AP) via table
	DSPFH digit, AP	0 0 1 1 1 P digit A	Digit (High part) ← (AP) via table
CPU Control & Others	HALT	0 0 0 1 0 0 0 0 0 0 0 0 0 0	Halt
	INTENAB 32/16	0 0 0 1 0 0 1 0 1 1 1 0 0 0	Enable timer
		0 0 0 1 0 0 0 1 0 0 0 0 1 0	
	INTDSAB 32/16	0 0 0 1 0 0 1 0 1 1 0 1 0 0	Disable timer
		0 0 0 1 0 0 0 1 0 0 0 0 0 1	
	INTMODE AP	1 1 0 1 0 P 0 1 0 0 A	AP ← Interrupt mode
	PAGE A0	1 1 0 1 1 0 0 1 0 1 A	Preg ← (A0)
	PAGE N	0 0 0 1 0 0 0 1 0 1 N	Preg ← N
	ADRS AP	1 1 0 1 1 P 0 1 1 0 A	Areg ← (AP)
	ADRS N	0 0 0 1 0 0 0 1 1 0 N	Areg ← N
	RATE AP	1 1 0 1 0 P 1 0 0 1 A	AP ← DIVIDER (8 Hz~1 Hz)
	RSTRATE	0 0 0 1 0 0 1 0 0 0 1 0 0 0	DIVIDER (8 Hz~1 Hz) ← 0
	BACKUP ON/OFF	0 0 0 1 0 0 0 0 0 1 b_3 b_2 0 0	Backup ON/OFF
	NOP	0 0 0 0 0 0 0 0 0 0 0 0 0 0	No operation

6

MSM5056

CMOS 4BIT SINGLE CHIP VERY LOW POWER MICROCONTROLLER WITH LCD DRIVER

GENERAL DESCRIPTION

The OKI MSM5056 is a low-power, high-performance single-chip microcontroller employing complementary metal oxide semiconductor technology. Integrated onto a single chip are a 4-bit ALU, 25K bits of mask programmable ROM, 360 bits of data RAM, crystal oscillator, voltage doubler, timer, LCD driver, input port, output port and overcharge protection circuit for connection to a solar cell.

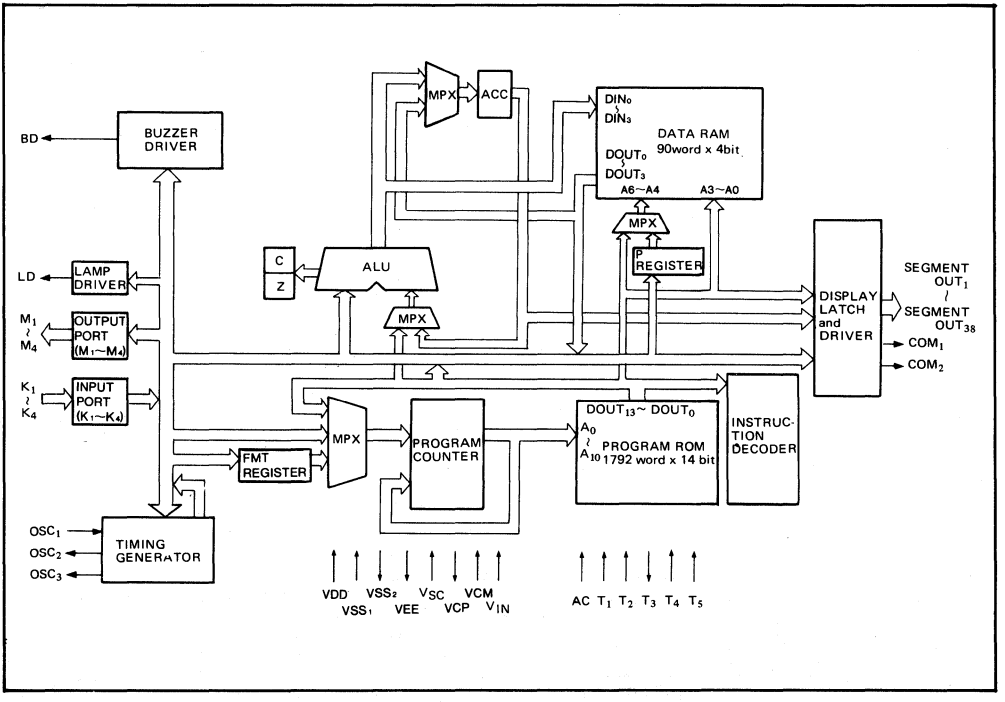
The MSM5056 is widely used in electronic products requiring low power operation, for example, solar calculator watches and games.

FEATURES

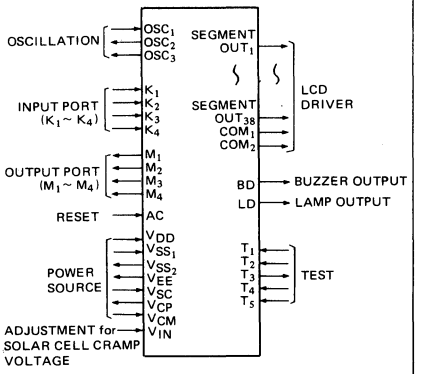
- Low Power Consumption 3 μA Typical
- 1792 × 14 Internal ROM
- 90 × 4 Internal RAM
- 4 Input Port
- 4 Output Port
- 4 × 4 Key Matrix Input (K_1~K_4, M_1~M_4)
- 38 LCD Driver
  (1/2 Duty, 1/2 Bias, 88 Segment)
- 42 Instructions
- 1.5 V Operating Voltage
  (The solar cell can be connected.)
- 32.768 kHz Crystal Oscillator
- 122.1 μs Instruction Cycle
- −20 to 75°C Operating Temperature
- 68 pad die

6

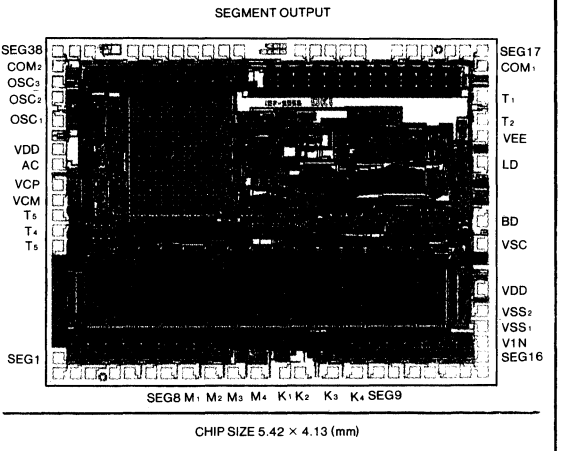
FUNCTIONAL BLOCK DIAGRAM



LOGIC SYMBOL



CHIP PAD LAYOUT



PIN DESCRIPTION

Designation	Function
V_{DD}	Circuit ground potential
V_{SS_1}	Power source (−1.5 V)
V_{SC}	Solar cell connection terminal
V_{SS_2}	Power source for LCD driver (−3.0 V) This terminal is connected to V_{DD} terminal through a 0.1 μF capacitor.
V_{EE}	Power source for internal logic (−1.5 to −3.0 V) This terminal is connected to V_{DD} terminal through a 0.1 μF capacitor.
V_{CP}, V_{CM}	Booster capacitor connection terminals V_{CP} terminal is connected to V_{CM} terminal through a 0.1 μF capacitor.
XT, $\overline{XT}$	Input and output terminals of oscillator inverter, 32.768 kHz crystal is connected to these terminals.
$T_1 \sim T_5$	Terminals to test internal logic, $T_1 \sim T_3$ and T_5 are pulled down to V_{SS_1}. T_4 is output. Test pins must be normaly open.
AC	Terminal to clear internal logic pulled down to V_{SS_1}. After power is turned on, the MSM5056 must be reset by this terminal.
BD	Buzzer output
LD	Lamp output
V_{IN}	Adjustment for solar cell cramp voltage This terminal is connected to V_{SS_1} terminal through 50 ~ 200 kΩ resistor.

FUNCTIONAL DESCRIPTION

A block diagram of the MSM5056 is given on page 111. Each block of logic will be briefly discussed. For more information, please refer to the MSM5056 user's manual.

Program ROM

The MSM5056 will address up to 1.75 K words of internal mask programmable ROM. Each word consists of 14 bits and all instructions are one word. The instructions are routed to a programmed logic array which generates the signals necessary for control of logic.

Data RAM

Data is organized in 4 bit nibbles. Internal data RAM consists of 90 nibbles.

The RAM is addressed by page address and column address. Normally page address is specified with the page register, but direct addressing is available at Page 0.

Column address is directly addressed by the operands of various instructions.

ALU

The ALU performs 4-bit parallel operation of RAM and AC contents, or RAM contents and an immediate digit. It sets or resets the flags (Z, C) depending on the condition.

Program Counter (PC)

The program counter is an 11-bit wide counter and specifies the address of the program ROM.

The PC is incremented by one at every execution of an instruction, and then specifies the next instruction to be executed. However, the contents of the PC are rewritten by the execution of a Jump or Branch instruction.

There is no boundary in the ROM, so a Jump or Branch instruction can be put anywhere in the ROM.

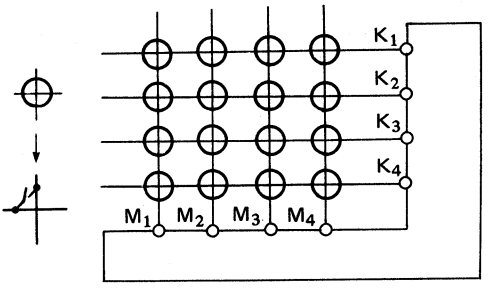
Input/Output Port

Input Port (K1 ~ K2)

The input port (K1 ~ K4) is a 4-bit parallel input port. Each pin of the port is pulled down to V_{SS1} by an internal resistor, and the status of the port can be fetched by an input instruction.

Output Port (M1 ~ M4)

The output port (M1 ~ M4) is a 4-bit parallel output port. This port consists of data latches and buffers. The contents of data latches are rewritten by an output instruction. A key matrix is used in combination with K1 to K4.



Display Function

The MSM5056 is provided with a segment output terminal which can directly drive a 1/2 bias, 1/2 duty LCD and common drive output terminals. COM1 and COM2. The segment drive circuit consists of the display data latch, multiplexer and driver. If the data is sent to the display data latch with the display instruction, the LCD drive waveform is output to the segment drive output terminal.

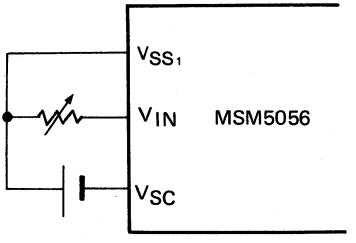
Time Base

Time base of the CPU is provided by connecting a 32.768 kHz crystal to the OSC1 and OSC3 pin. One machine cycle is 122.1 μs.

A hardware divider up to 1 Hz is provided enabling programs to implement and a clock function by counting signals between 16 and 1 Hz.

Solar Cell Overcharge Protection Circuit

When a solar cell is connected to prolong the usefull life of the battery, a resistor is inserted between the V_{IN} pin and V_{SS1} to adjust the overcharge protection voltage.



ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage 1	$V_{DD} - V_{SS_1}$	Ta = 25°C	−0.3 to +3.0	V
Supply Voltage 2	$V_{DD} - V_{SC}$	Ta = 25°C	−0.3 to +3.5	V
Supply Voltage 3	$V_{DD} - V_{SS_2}$	Ta = 25°C	−0.3 to +6.0	V
Supply Voltage 4	$V_{DD} - V_{EE}$	Ta = 25°C	−0.3 to +6.0	V
Input Voltage	V_{IN_1}	Ta = 25°C	V_{SS_1} −0.3 to +0.3	V
Storage Temperature	Tstg		−55 to 125	°C

OPERATING CONDITIONS

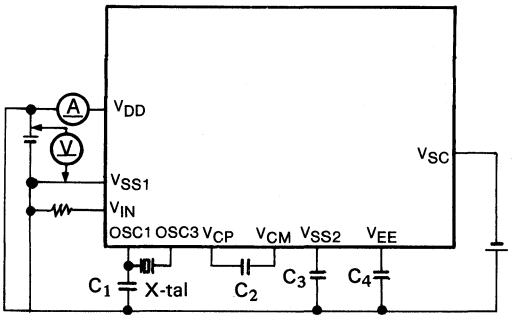
Parameter	Symbol	Limits	Unit
Operating Voltage	$V_{DD} - V_{SS_1}$	1.25 to 1.65	V
Operating Temperature	Topr	−20 to 75	°C

DC CHARACTERISTICS

(V_{DD} =0V, V_{SS_1}, V_{EE} = −1.55V, V_{SS_2} = −3.0V, C_I = 30kΩ, C_G = 30pF, Ta = 25°C)

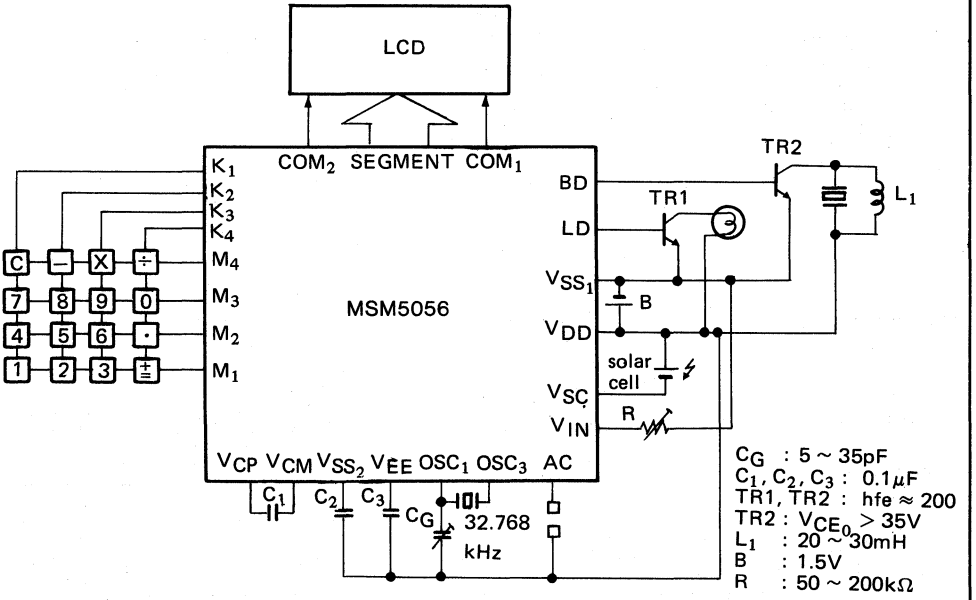
Parameter	Symbol	Condition		Limits Min.	Limits Typ.	Limits Max.	Unit
Operating voltage 1	$-V_{SS_1}$	V_{SS_1} terminal		1.25	1.55	2.0	V
Operating voltage 2	$-V_{SC}$	V_{SC} terminal		0	2.0	3.0	V
Power supply current	I_{DD}	V_{SS_1} terminal		−	3.0	−	μA
Oscillation start voltage	$-V_{OSC}$	Within 10 seconds V_{SS_1} terminal		1.45	−	−	V
Output current 1 COM	I_{OH_1}	$V_{OH_1} = -0.2V$		−4	−	−	μA
	I_{OM_1}	$V_{OM_1} = V_{SS_1} \pm 0.2V$		4/−4	−	−	
	I_{OL_1}	$V_{OL_1} = -2.8V$		4	−	−	
Output current 2 SEGMENT	I_{OH_2}	$V_{OH_2} = -0.2V$		−0.4	−	−	μA
	I_{OL_2}	$V_{OL_2} = -2.8V$		0.4	−	−	
Output current 3 $M_1 \sim M_4$	I_{OH_3}	V_{SS_1}, V_{EE} −1.25V	V_{OH_3} −0.4 V	−100	−	−	μA
	I_{OL_3}	V_{SS_2} −2.3V	V_{OL_3} −0.85V	3	−	8	
Output current 4 BD	I_{OH_4}	$V_{OH_4} = -0.4V$		−50	−100	−200	μA
	I_{OL_4}	$V_{OL_4} = -1.15V$		3	10	30	
Input current $K_1 \sim K_4$	I_{IH_1}	$V_{IN} = -0V$		5	10	15	μA
	I_{IL_1}	$V_{IN} = -1.55V$		−	−	−0.2	
Oscillator built-in capacitor	CD			−	20	−	pF
Solar battery cramp resistor	R_{IN}	$V_{SS_1} = -1.8V$ V_{IN} terminal		50	−	200	kΩ

MEASURING CIRCUIT



C_1 : 30pF
C_2, C_3, C_4 : 0.1μF
X-tal : 32.768kHz

TYPICAL APPLICATION



C_G : 5 ~ 35pF
C_1, C_2, C_3 : 0.1μF
TR1, TR2 : hfe ≈ 200
TR2 : V_{CE_0} > 35V
L_1 : 20 ~ 30mH
B : 1.5V
R : 50 ~ 200kΩ

DESCRIPTION OF INSTRUCTIONS

	Mnemonic	Instruction Code														Operation
		13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Arithmetic operation	ADD ACC, AP	0	0	0	0	0	P	0	1	0	0	A				AP ← (AP) + (ACC)
	ADD #D, AP	0	1	1	0	0	P	D				A				AP ← (AP) − D
	ADC AP	0	0	0	0	0	P	0	1	0	1	A				AP ← Decimal adjust {(AP) + (ACC) + (C)}
	SUB ACC, AP	0	0	0	0	1	P	0	1	0	0	A				AP ← (AP) − (ACC)
	SUB #D, AP	0	1	1	0	1	P	D				A				AP ← (AP) − D
	SBC AP	0	0	0	0	1	P	0	1	0	1	A				AP ← Decimal adjust {(AP) − (ACC) − (C)}
	CMP ACC, AP	0	0	0	0	1	P	1	1	1	0	A				(AP) − (ACC)
	CMP #D, AP	0	1	0	1	1	P	D				A				(AP) − D
	INC AP	0	1	1	0	0	P	0	0	0	1	A				AP ← (AP) + 1
	DEC AP	0	1	1	0	1	P	0	0	0	1	A				A ← (AP) − 1
	XOR ACC, AP	0	0	0	0	0	P	0	1	1	1	A				AP ← (AP) ∀ (ACC)
	XOR #D, AP	0	1	1	1	1	P	D				A				AP ← (AP) V D
Bit operation	BIT ACC, AP	0	0	0	0	0	P	1	1	1	0	A				(AP) V $\overline{(ACC)}$
	BIT #D, AP	0	1	0	1	0	P	D				A				(AP) V $\overline{D}$
	BIS ACC, AP	0	0	0	0	0	P	0	1	1	0	A				AP ← (AP) V (ACC)
	BIS #D, AP	0	1	0	0	0	P	D				A				AP ← (AP) V D
	BIC ACC, AP	0	0	0	0	1	P	0	1	1	0	A				AP ← (AP) Λ $\overline{(ACC)}$
	BIC #D, AP	0	1	0	0	1	P	D				A				AP ← (AP) Λ $\overline{D}$
Shift	ASR AP	0	0	0	0	0	P	0	0	1	1	A				⌊(C) 0 → (AP)⌋
	ASL AP	0	0	0	0	1	P	0	0	1	1	A				(C) ← (AP) ← 0
Flag operation	CLZ	0	0	0	0	0	0	1	0	1	0	0	0	0	0	Z ← 0
	CLC	0	0	0	0	0	0	1	0	0	1	0	0	0	0	C ← 0
	CLA	0	0	0	0	0	0	1	0	1	1	0	0	0	0	Z ← 0, C ← 0
	SEZ	0	0	0	0	1	0	1	0	1	0	0	0	0	0	Z ← 1
	SEC	0	0	0	0	1	0	1	0	0	1	0	0	0	0	C ← 1
	SEA	0	0	0	0	1	0	1	0	1	1	0	0	0	0	Z ← 1, C ← 1
Data transfer	MOV ACC, AP	1	1	1	1	0	P	0	0	0	0	A				AP ← (ACC)
	MOV ACC, AX	1	1	1	1	0	0	0		X		A				AX ← (ACC)
	MOV #D, AP	0	1	1	1	0	P	D				A				AP ← D
	MOV AP, ACC	1	1	1	1	1	P	0	0	0	0	A				ACC ← (AP)
	MOV AX, ACC	1	1	1	1	1	0	0		X		A				ACC ← (AX)
	CHG AP	1	1	1	0	0	P	0	0	0	0	A				(ACC) ⟷ (AP)
	CHG AX	1	1	1	0	0	0	0		X		A				(ACC) ⟷ (AX)

6

DESCRIPTION OF INSTRUCTIONS (CONT.)

	Mnemonic	Instruction Code														Operation
		13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Jump	JMP adrs	1	0	0	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	PC ← adrs
	JMP @AP	0	0	0	0	0	P	1	1	0	1	A				PC ← (PC) + (AP) + 1
	JMPIO @AP	0	0	0	0	1	P	1	1	0	1	A				PC ← (PC) + {(AP)∧7H} + +1
	BEQ +n BZE +n	0	0	0	1	1	0	0	1	0	n_4	n_3	n_2	n_1	n_0	PC ← (PC) + n + 1, if Z = 1
	BNE +n BNZ +n	0	0	0	1	1	0	1	1	0	n_4	n_3	n_2	n_1	n_0	PC ← (PC) + n + 1, if Z = 0
	BCS +n BLT +n	0	0	0	1	1	0	0	0	1	n_4	n_3	n_2	n_1	n_0	PC ← (PC) + n + 1, if C = 1
	BCC +n BGE +n	0	0	0	1	1	0	1	0	1	n_4	n_3	n_2	n_1	n_0	PC ← (PC) + n + 1, if C = 0
	BGT +n	0	0	0	1	1	0	1	1	1	n_4	n_3	n_2	n_1	n_0	PC ← (PC) + n + 1, if Z = 0 and C = 0
	BLE +n	0	0	0	1	1	0	0	1	1	n_4	n_3	n_2	n_1	n_0	PC ← (PC) + n + 1, if Z = 1 or C = 1
Input/ Output	INP Port, AP	1	1	0	1	0	P	Port				A				AP ← (Port)
	OUT AP, Port	1	1	0	1	1	P	Port				A				Port ← (AP)
	OUT #D, Port	0	0	0	1	0	0	Port				D				Port ← D
Display	DSP digit, AP	0	0	1	0	0	P	digit				A				digit ← (AP), (ACC)
	DSPF digit, AP	0	0	1	1	0	P	digit				A				digit ← (AP) via table
CPU control	HALT	0	0	0	1	0	0	0	0	0	0	0	0	0	0	Halt CPU
	NOP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	No Operation

6

MSM6051

CMOS 4BIT HIGH PERFORMANCE SINGLE CHIP VERY LOW POWER MICROCONTROLLER WITH LCD DRIVER

GENERAL DESCRIPTION

OKI's MSM6051 is a low-power and high-performance single-chip microcontroller employing complementary metal oxide semiconductor technology. Integrated onto a single chip are a 4-bit ALU, 35K bits of mask programmable ROM, 480 bits of data RAM, crystal oscillator, voltage doubler, timer, LCD driver, input port and output port.

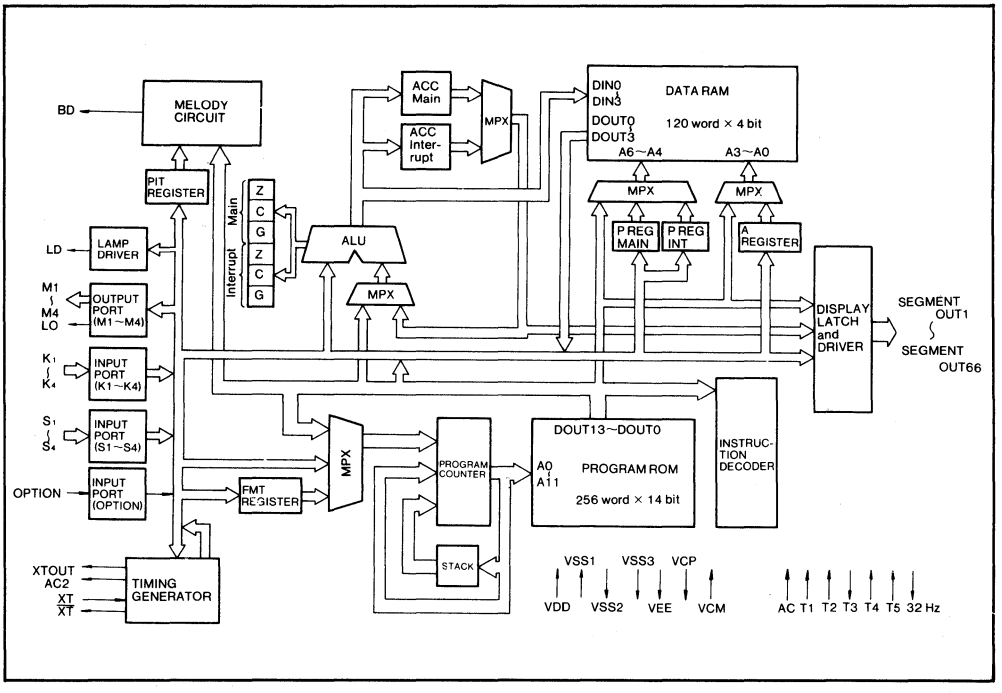
The MSM6051 is widely used in electronic products requiring low power operation, for example, stopwatches with lap time memory, calculator watches and handy terminals.

FEATURES

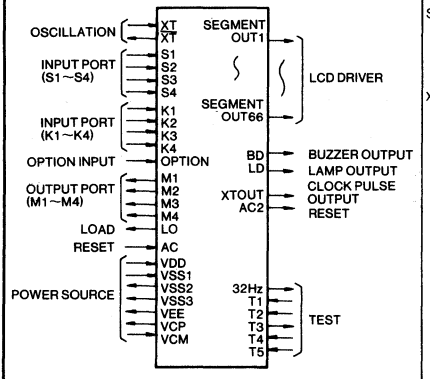
- Low Power Consumption 3 μA Typical
- 2560 × 14 Internal ROM
- 120 × 4 Internal RAM
- 9 Input Port
- 4 Output Port
- 4 × 4 Key Matrix Input (K1~K4, M1~M4)
- 66 LCD Driver (including 3 common) (1/3 Duty, 1/3 Bias, 189 Segment)
- 59 Instructions
- 1.5 V or 3 V Operating Voltage (Masking Option)
- 32.768 kHz crystal Oscillator
- 91.5 μs Instruction Cycle
- −20 to 75°C Operating Temperature
- 101 pad die

6

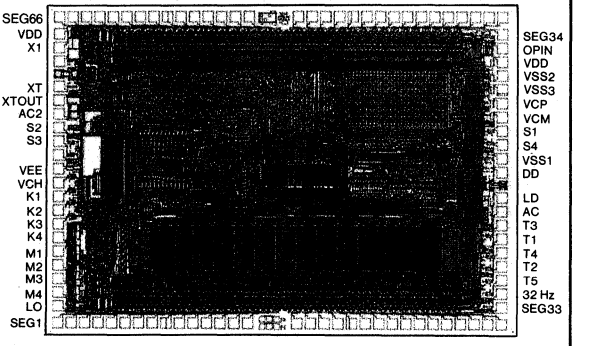
FUNCTIONAL BLOCK DIAGRAM



LOGIC SYMBOL



CHIP PAD LAYOUT



SEGMENT OUTPUT
CHIP SIZE 5.85 × 4.10 (mm)

6

PIN DESCRIPTION

Designation	Function
V_{DD}	Circuit ground potential
V_{SS_1}	Power source (−1.5 V)
V_{SS_2}	Power source for LCD driver (−3.0 V) This terminal is connected to V_{DD} terminal through a 0.1 μF capacitor.
V_{SS_3}	Power source for LCD driver (−4.5 V) This terminal is connected to V_{DD} terminal through a 0.1 μF capacitor.
V_{EE}	Power source for internal logic (−1.5 to −3.0 V) This terminal is connected to V_{DD} terminal through a 0.1 μF capacitor.
V_{CP}, V_{CM}	Booster capacitor connection terminals V_{CP} terminal is connected to V_{CM} terminal through a 0.1 μF capacitor.
XT, $\overline{XT}$	Input and output terminals of oscillator inverter, 32.768 kHz crystal is connected to these terminals.
$T_1 \sim T_5$	Terminals to test internal logic, $T_1 \sim T_3$ and T_5 are pulled down to V_{SS_1}. T_4 is output. Test pins must be normally open.
AC	Terminal to clear internal logic pulled down to V_{SS_1}. After power is turned on, the MSM6051 must be reset by this terminal.
BD	Buzzer output
LD	Lamp output
LO	Load data terminal of M_1 to M_4.
AC2	Reset terminal for external circuit.
XT OUT	Clock output for external circuit.

FUNCTIONAL DESCRIPTION

A block diagram of the MSM6051 is given on page 118. Each block of logic will be briefly discussed. For more information, please refer to the MSM6051 user's manual.

Program ROM

The MSM6051 addresses up to 2.5 K words of internal mask programmable ROM. Each word consists of 14 bits, and all instructions are one word. The instructions are routed to a programmed logic array which generates the signals necessary for control of logic.

Data RAM

Data is organized in 4 bit nibbles. Internal data RAM consists of 120 nibbles.

The RAM is addressed by page address and column address. Normally page address is specified the with page register, but direct addressing is available at Page 0.

Column address is directly addressed by operand of various instructions.

ALU

The ALU performs 4-bit parallel operation of RAM and ACC contents, or RAM contents and an immediate digit. It sets or resets the flags (Z, C, G) depending on the condition.

Program Counter (PC)

The program counter is 12 bits wide and specifies the address of the program ROM.

The PC is incremented by one at every execution of the instruction, and then specifies the next instruction to be executed. However, the contents of the PC are rewritten by the execution of a Jump, Call or Branch instruction.

There is no boundary in the ROM, so a Jump or Branch instruction can be put anywhere in the ROM.

Stack

The MSM6051 has a 3 level stack apart from data RAM. The contents of the PC are loaded into the stack when a call instruction is executed or an interrupt is generated.

Input/Output Port

Input Port (S1 ~ S4)

The input port (S1 ~ S4) is a 4-bit parallel input port. Each pin of the port is pulled down to V_{SS_1} by an internal resistor, and the status of the port is fetched by a SWITCH instruction.

Input Port (K1 ~ K4)

The input port (K1 ~ K4) is a 4-bit parallel input port. Each pin of the port is pulled down to V_{SS_1} by an internal resistor, and the status of the port is fetched by a KSWITCH instruction.

Input Port (OPIN)

The input port (OPIN) is single input port. OPIN is pulled down to V_{SS_1} by an internal power save circuit, and the status of the port is fetched by an input instruction.

Output Port (M1 ~ M4)

The output port (M1 ~ M4) is a 4-bit parallel output port. This port consists of data latches and buffers, and the contents of data latches are rewritten by a matrix instruction.

Display Function

The MSM6051 is provided with the segment output terminal which can directly drive a 1/3 bias, 1/3 duty LCD, and the common drive output terminals COM1, COM2 and COM3.

The segment drive circuit consists of the display data latch, multiplexer and driver. If the data is sent to the display data latch with the display instruction, the LCD drive waveform is output to the segment drive output terminal.

Time Base

The time base of the CPU is provided by connecting 32.768 kHz crystal to the XT and $\overline{XT}$ pin. One machine cycle is 91.5 μs.

A hardware divider up to 1 Hz is provided enabling programs to implement a clock function by counting signals between 32 and 1 Hz.

Also, a 1/100 second digit counting function is provided as a hardware feature to make for easy implementation of a stopwatch function.

ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage 1	$V_{DD} - V_{SS1}$	Ta = 25°C	−2.0 to +0.3	V
Supply Voltage 2	$V_{DD} - V_{SS2}$	Ta = 25°C	−4.0 to +0.3	V
Supply Voltage 3	$V_{DD} - V_{SS3}$	Ta = 25°C	−6.0 to +0.3	V
Supply Voltage 4	$V_{DD} - V_{EE}$	Ta = 25°C	−4.0 to +0.3	V
Input Voltage	V_{IN1}	Ta = 25°C	V_{SS1} −0.3 to +0.3	V
Storage Temperature	Tstg		−55 to 125	°C

OPERATING CONDITIONS

Parameter	Symbol	Limits	Unit
Operating Voltage	$V_{DD} - V_{SS1}$	1.25 to 1.65	V
Operating Temperature	Topr	−20 to 75	°C

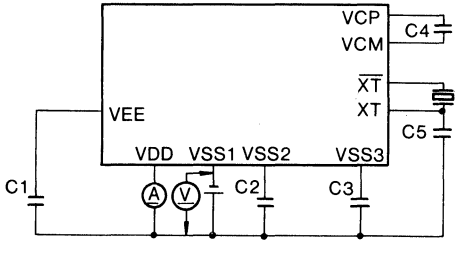
DC CHARACTERISTICS

(V_{DD} =0V, V_{SS1}, V_{EE} = −1.55V, V_{SS2} = −3.0V, V_{SS3} = −4.5V, C_I = 30kΩ, Ta = 25°C)

Parameter	Symbol	Conditions		Limits Min.	Limits Typ.	Limits Max.	Unit
Power supply current	I_{DD}	V_{SS} terminal		–	3.0	–	μA
Oscillation start voltage	$-V_{OSC}$	Within 5 seconds V_{SS1} terminal		1.45	–	–	V
Output current 1 COM	I_{OH1}	$V_{OH1} = -0.2V$		−4	–	–	μA
	I_{OMH1}	$V_{OMH1} = V_{SS1} \pm 0.2$		4/−4	–	–	
	I_{OML1}	$V_{OML1} = V_{SS2} \pm 0.2$		4/−4	–	–	
	I_{OL1}	$V_{OL1} = -4.3V$		4	–	–	
Output current 2 SEGMENT	I_{OH2}	$V_{OH2} = -0.2V$		−4	–	–	μA
	I_{OMH2}	$V_{OMH2} = V_{SS1} \pm 0.2$		4/−4	–	–	
	I_{OML2}	$V_{OML2} = V_{SS2} \pm 0.2$		4/−4	–	–	
	I_{OL2}	$V_{OL2} = -4.3V$		4	–	–	
Output current 3 AC2 LOAD XTOUT	I_{OH3}	$V_{OH3} = -0.5$ V		−10	–	–	μA
	I_{OL3}	$V_{OL3} = -1.15V$		10	–	–	
Output current 4 $M_1 \sim M_4$	I_{OH4}	$V_{OH4} = -0.5V$		−100	–	–	μA
	I_{OL4}	$V_{OL4} = -1.0V$		2	–	10	
Output current 5 LD	I_{OH5}	$V_{OH5} = -0.55V$	$V_{SS1} = -1.25V$, $V_{EE} = -2.0V$, $V_{SS2} = -2.0V$	−12.5	−25	−83	μA
	I_{OL5}	$V_{OL5} = -0.85V$		1	–	–	
Output current 6 BD	I_{OH6}	$V_{OH6} = -0.55V$	$V_{SS1} = -1.25V$, $V_{EE} = -2.0V$, $V_{SS2} = -2.0V$	−17	−30	−62	μA
	I_{OL6}	$V_{OL6} = -0.85V$		–	5	–	

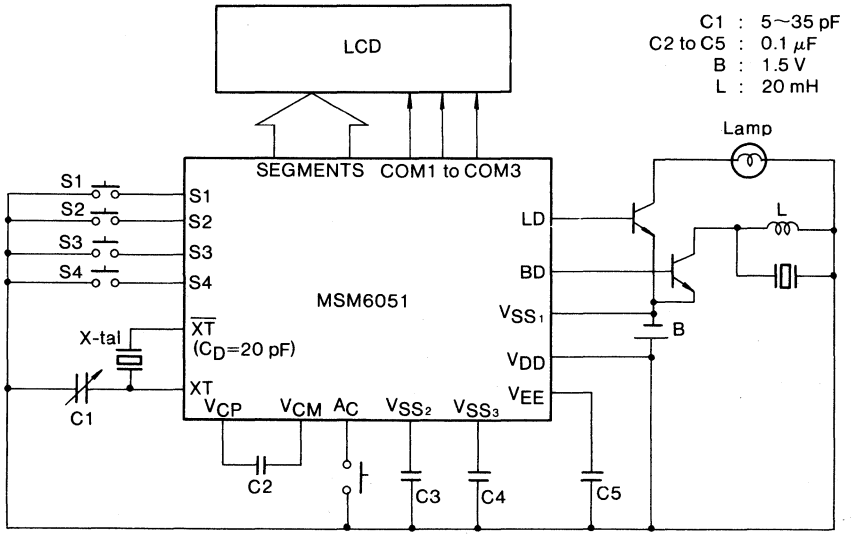
Parameter	Symbol	Conditions	Limits			Unit
			Min.	Typ.	Max.	
Input current 1 S_1~S_4	I_{IH_1}	V_{IH_1} = 0V	2	20	100	μA
	I_{IL_1}	V_{IL_1} = −1.55V	–	–	−0.2	
Input current 2 K_1~K_4	I_{IH_2}	V_{IH_2} = 0V	5	13	26	μA
	I_{IL_2}	V_{IL_2} = −1.55V	–	–	−0.2	
Input current 3 OPIN	I_{IH_3}	V_{IH_3} = 0V	–	30	45	μA
	I_{IL_3}	V_{IL_3} = −1.55V	–	–	−0.2	
Oscillator built-in capacitance	CD		–	20	–	pF

MEASURING CIRCUIT



C1 . C2 . C3 . C4 : 0.1 μF
C5 : 30 pF
X-tal : 32.768 kHz

TYPICAL APPLICATION



C1 : 5~35 pF
C2 to C5 : 0.1 μF
B : 1.5 V
L : 20 mH

DESCRIPTION OF INSTRUCTIONS

	Mnemonic	Instruction Code														Operation
		13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Arithmetic operation	ADD ACC, AP	0	0	0	0	0	P	0	1	0	0	A				AP ← (AP) + (ACC)
	ADD #D, AP	0	1	1	0	0	P	D				A				AP ← (AP) + D
	ADC AP	0	0	0	0	0	P	0	1	0	1	A				AP ← Decimal adjust {(AP) + (ACC) + (C)}
	ADCN AP	1	1	0	0	0	P	N				A				AP ← N adjust {(AP) + (C)}
	SUB ACC, AP	0	0	0	0	1	P	0	1	0	0	A				AP ← (AP) − (ACC)
	SUB #D, AP	0	1	1	0	1	P	D				A				AP ← (AP) − D
	SBC AP	0	0	0	0	1	P	0	1	0	1	A				AP ← Decimal adjust {(AP) − (ACC) − (C)}
	SBCN AP	1	1	0	0	1	P	N				A				AP ← N adjust {(AP) − (C)}
	CMP ACC, AP	0	0	0	0	1	P	1	1	1	0	A				(AP) − (ACC)
	CMP #D, AP	0	1	0	1	1	P	D				A				(AP) − D
	INC AP	1	1	1	0	0	P	0	0	0	0	A				AP ← (AP) + 1
	INC AX	1	1	1	0	0	0	0	X			A				AX ← (AX) + 1
	DEC AP	1	1	1	0	1	P	0	0	0	0	A				AP ← (AP) − 1
	DEC AX	1	1	1	0	1	0	0	X			A				AX ← (AX) − 1
	XOR ACC, AP	0	0	0	0	0	P	0	1	1	1	A				AP ← (AP) $\forall$ (ACC)
	XOR #D, AP	0	1	1	1	1	P	D				A				AP ← (AP) $\forall$ D
Bit operation	BIT ACC, AP	0	0	0	0	0	P	1	1	1	0	A				(AP) V ($\overline{ACC}$)
	BIT #D, AP	0	1	0	1	0	P	D				A				(AP) V $\overline{D}$
	BIS ACC, AP	0	0	0	0	0	P	0	1	1	0	A				(AP) V (ACC)
	BIS #D, AP	0	1	0	0	0	P	D				A				(AP) V D
	BIC ACC, AP	0	0	0	0	1	P	0	1	1	0	A				(AP) $\wedge$ ($\overline{ACC}$)
	BIC #D, AP	0	1	0	0	1	P	D				A				(AP) $\wedge$ $\overline{D}$
Rotate/Shift	ROR AP	0	0	0	0	0	P	0	0	1	0	A				⌊→ (C) → (AP) ⌋
	ROL AP	0	0	0	0	1	P	0	0	1	0	A				⌊ (C) ← (AP) ← ⌋
	ASR AP	0	0	0	0	0	P	0	0	1	1	A				⌊→ (C) 0 → (AP) ⌋
	ASL AP	0	0	0	0	1	P	0	0	1	1	A				(C) ← (AP) ← 0
Flag operation	CLZ	0	0	0	0	0	0	1	0	1	0	0	0	0	0	Z ← 0
	CLC	0	0	0	0	0	0	1	0	0	1	0	0	0	0	C ← 0
	CLG	0	0	0	0	0	0	1	0	0	0	0	0	0	0	G ← 0
	CLA	0	0	0	0	0	0	1	0	1	1	0	0	0	0	Z ← 0, C ← 0, G ← 0
	SEZ	0	0	0	0	1	0	1	0	1	0	0	0	0	0	Z ← 1
	SEC	0	0	0	0	1	0	1	0	0	1	0	0	0	0	C ← 1

DESCRIPTION OF INSTRUCTIONS (CONT.)

	Mnemonic	Instruction Code 13 12 11 10 9 8 7 6 5 4 3 2 1 0	Operation
	SEG	0 0 0 0 1 0 1 0 0 0 0 0 0 0	G ← 1
	SEA	0 0 0 0 1 0 1 0 1 1 0 0 0 0	Z ← 1, C ← 1, G ← 1
Data transfer	MOV ACC, AP	1 1 1 1 0 P 0 0 0 0 A	AP ← (ACC)
	MOV ACC, AX	1 1 1 1 0 0 0 X A	AX ← (ACC)
	MOV #D, AP	0 1 1 1 0 P D A	AP ← D
	MOV AP, ACC	1 1 1 1 1 P 0 0 0 0 A	ACC ← (AP)
	MOV AX, ACC	1 1 1 1 1 0 0 X A	ACC ← (AX)
Sub-routine	CALL adrs	1 0 1 1 a_9 a_8 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	STACK ← (PC), PC ← adrs
	RET	0 0 0 0 0 0 1 1 0 0 0 0 0 0	PC ← (STACK) + 1
	RTI	0 0 0 0 1 0 1 1 0 0 0 0 0 0	PC ← (STACK) + 1, or PC ← (STACK) ACC ↔ ACC', Z ↔ Z', C ↔ C' G ↔ G', Preg ↔ P'reg
Jump	JMP adrs	1 0 a_{11} a_{10} a_9 a_8 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	PC ← Adrs
	JMP @AP	0 0 0 0 0 P 1 1 0 1 A	PC ← (PC) + (AP) + 1
	JMPIO @AP	0 0 0 0 1 P 1 1 0 1 A	PC ← (PC) + {(AP) ∧ 7H} +1
	BEQ +n BZE +n	0 0 0 1 1 0 0 1 0 n_4 n_3 n_2 n_1 n_0	PC ← (PC)+n+1, if Z=1
	BEQ −n BZE −n	0 0 0 1 1 1 0 1 0 n_4 n_3 n_2 n_1 n_0	PC ← (PC) −n, if Z=1
	BNE +n BNZ +n	0 0 0 1 1 0 1 1 0 n_4 n_3 n_2 n_1 n_0	PC ← (PC) +n + 1, if Z=0
	BNE −n BNZ −n	0 0 0 1 1 1 1 1 0 n_4 n_3 n_2 n_1 n_0	PC ← (PC) −n, if Z=0
	BCS +n	0 0 0 1 1 0 0 0 0 n_4 n_3 n_2 n_1 n_0	PC ← (PC)+n+1, if C=1
	BCS −n	0 0 0 1 1 1 0 0 0 n_4 n_3 n_2 n_1 n_0	PC ← (PC) −n, if C=1
	BCC +n	0 0 0 1 1 0 1 0 0 n_4 n_3 n_2 n_1 n_0	PC ← (PC)+n+1, if C=0
	BCC −n	0 0 0 1 1 1 1 0 0 n_4 n_3 n_2 n_1 n_0	PC ← (PC)−n, if C=0
	BGT +n	0 0 0 1 1 0 0 0 1 n_4 n_3 n_2 n_1 n_0	PC ← (PC)+n+1, if G=1
	BGT −n	0 0 0 1 1 1 0 0 1 n_4 n_3 n_2 n_1 n_0	PC ← (PC)−n, if G=1
	BLE +n	0 0 0 1 1 0 1 0 1 n_4 n_3 n_2 n_1 n_0	PC ← (PC)+n+1, if G=0
	BLE −n	0 0 0 1 1 1 1 0 1 n_4 n_3 n_2 n_1 n_0	PC ← (PC)−n, if G=0
	BGE +n	0 0 0 1 1 0 0 1 1 n_4 n_3 n_2 n_1 n_0	PC ← (PC)+n+1, if Z=1 or G=1
	BGE −n	0 0 0 1 1 1 0 1 1 n_4 n_3 n_2 n_1 n_0	PC ← (PC)−n, if Z=1 or G=1
	BLT +n	0 0 0 1 1 0 1 1 1 n_4 n_3 n_2 n_1 n_0	PC ← (PC)+n+1, if Z=0 and G=0
	BLT −n	0 0 0 1 1 1 1 1 1 n_4 n_3 n_2 n_1 n_0	PC ← (PC)−n, if Z=0 and G=0
Input/ Output	SWITCH AP	1 1 0 1 0 P 0 0 0 1 A	AP ← INPUT PORT (S1 ~ S4)
	KSWITCH AP	1 1 0 1 0 P 0 0 1 0 A	AP ← INPUT PORT (K1 ~ K4)

6

DESCRIPTION OF INSTRUCTIONS (CONT.)

	Mnemonic	Instruction Code														Operation
		13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Input/Output	MATRIX AP	1	1	0	1	1	P	0	0	1	0	A				OUTPUT PORT (M1~M4) ← (AP)
	MATRIX Mn	0	0	0	1	0	0	0	0	1	0	M_4	M_3	M_2	M_1	OUTPUT PORT (M1~M4) ← Mn (n=1, 2, 3, 4)
	XTCP ON/OFF	0	0	0	1	0	0	1	0	0	0	0	0	b_1	b_0	XTOUT ON/OFF
	PITCH N	0	0	0	1	0	0	1	1	0	0	N				PIT reg. ← N
	MSA adrs	1	0	1	0	1	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Stack ← (PC), PC ← adrs, Melody start
	MSO	0	0	0	1	0	0	1	0	1	0	0	0	0	0	Melody stop
	LAMP ON/OFF	0	0	0	1	0	0	0	0	0	1	0	0	b_1	b_0	LD ON/OFF
Display	DSP digit, AP	0	0	1	0	0	P	digit				A				Digit (Low part) ← (AP), (ACC)
	DSPH digit, AP	0	0	1	0	1	P	digit				A				Digit (High part) ← (AP), (ACC)
	FORMAT AP	1	1	0	1	1	P	0	0	1	1	A				FMT reg. ← (AP)
	FORMAT N	0	0	0	1	0	0	0	0	1	1	N				FMT reg. ← N
	DSPF digit, AP	0	0	1	1	0	P	digit				A				Digit (Low part) ← (AP) via table
	DSPFH digit, AP	0	0	1	1	1	P	digit				A				Digit (High part) ← (AP) via table
CPU Control & Others	HALT	0	0	0	1	0	0	0	0	0	0	0	0	0	0	Halt
	INTENAB 32/16/2	0	0	0	1	0	0	1	0	1	1	1	0	0	0	Enable timer interrupt
		0	0	0	1	0	0	0	1	0	0	b_3	0	b_1	0	
	INTDSAB 32/16/2	0	0	0	1	0	0	1	0	1	1	0	1	0	0	Disable timer interrupt
		0	0	0	1	0	0	0	1	0	0	0	b_2	0	b_0	
	ACTIVATE	0	0	0	1	0	0	1	0	0	1	0	0	0	0	Activate main routine
	INTMODE AP	1	1	0	1	0	P	0	1	0	0	A				AP ← Interrupt mode
	KENAB	0	0	0	1	0	0	0	1	1	1	1	0	0	0	Enable INPUT PORT (K1~K4)
	KDSAB	0	0	0	1	0	0	0	1	1	1	0	1	0	0	Disable INPUT PORT (K1~K4)
	STATUS AP	1	1	0	1	0	P	1	0	1	0	A				AP ← Status
	PAGE A0	1	1	0	1	1	0	0	1	0	1	A				Preg ← (A0)
	PAGE N	0	0	0	1	0	0	0	1	0	1	N				Preg ← N
	ADRS AP	1	1	0	1	1	P	0	1	1	0	A				Areg ← (AP)
	ADRS N	0	0	0	1	0	0	0	1	1	0	N				Areg ← N
	RATE AP	1	1	0	1	0	P	1	0	0	1	A				AP ← DIVIDER (8 Hz~1 Hz)
	RSTRATE	0	0	0	1	0	0	1	0	0	0	1	0	0	0	DIVIDER (8 Hz~1 Hz) ← 0
	FLAGIN AP	1	1	0	1	0	P	1	1	1	0	A				AP ← S1 on flag, S2 on flag
	S1RATE AP	1	1	0	1	0	P	1	1	0	0	A				AP ←S1reg.
	S2RATE AP	1	1	0	1	0	P	1	1	0	1	A				AP ←S2reg.
	BACKUP ON/OFF	0	0	0	1	0	0	0	0	0	1	b_3	b_2	0	0	Backup ON/OFF
	NOP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	No operation

MSM6351

CMOS 4BIT HIGH PERFORMANCE AND VERY LOW POWER SINGLE CHIP MICROCONTROLLER WITH LCD DRIVER

GENERAL DESCRIPTION

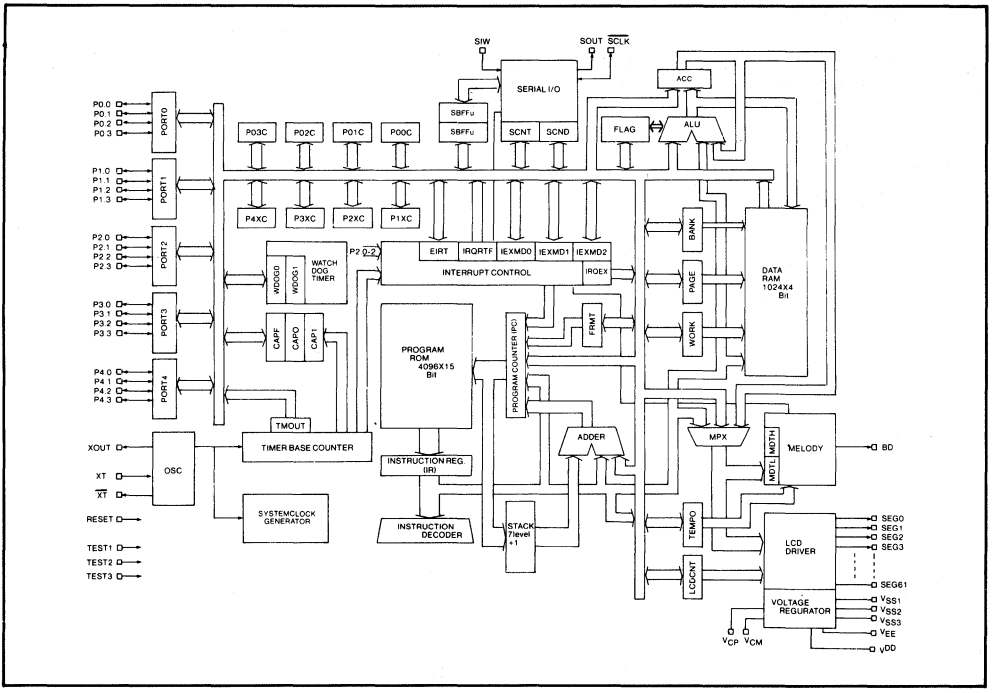
OKI's MSM6351 is a low-power, high-performance single-chip microcontroller employing silicon gate CMOS technology. Integrated onto a single chip are 4-bit ALU, 61K bits of mask programmable ROM, 4096 bits of RAM, 20 bits of I/O port, serial I/O port, time-base counter, LCD driver, 3 interrupts, crystal oscillator and voltage tripler.

The MSM6351 is widely used in electronic products requiring low power consumption, a large number of LCD drivers and a large size of memory.

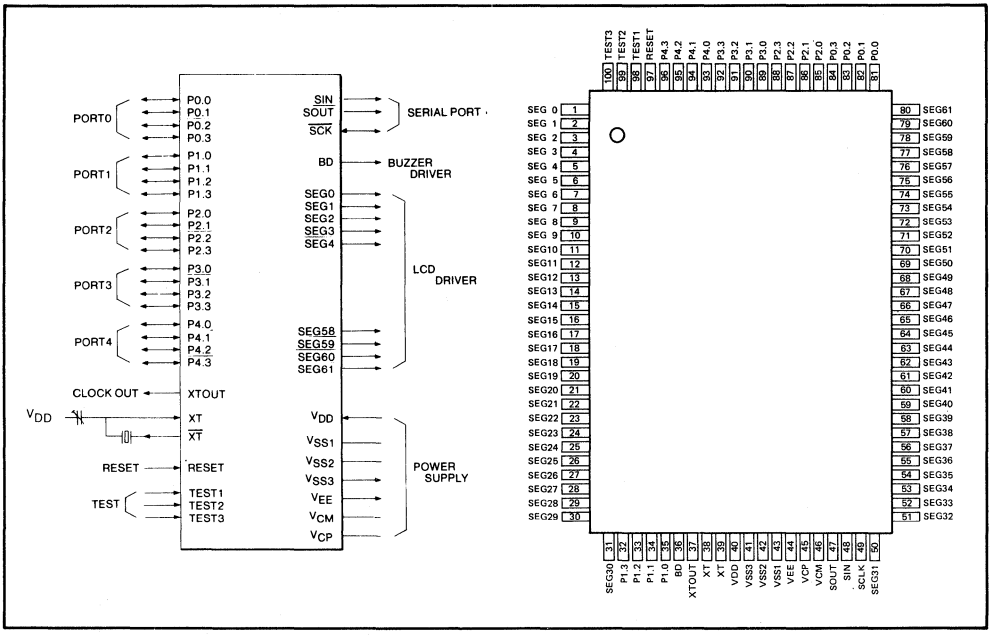
FEATURES

- Low Power Consumption 3 μA Typical
- 4096 × 15 Internal ROM
- 1024 × 4 Internal RAM
- 20 Input/Output Ports
- 62 LCD Drivers
  1/3 Duty, 1/3 Bias or 1/4 Duty 1/3 Bias
  (Selectable by software)
- Serial I/O Port
  8 bits or 5 bits data frame mode
  Asynchronous
  receiver/transmitter mode
  Internal or external clock mode
- 15 stages Time Base Counter
- Watch Dog Timer
- Capture function by external trigger signal
- 3 Interrupt Sources
  Real time interrupt
  External interrupt
  Serial I/O port interrupt
- Melody Circuit
- 65 Instructions
- Sub-routine Nesting: 7 levels
- 32.768 kHz Crystal Oscilator
- Machine Cycle: 61.0 μsec.
- Power Supply: 1.5V or 3.0V (selectable by mask option)
- 100 pad die or 100 Pin Flat Package
- Silicon gate CMOS Process

FUNCTIONAL BLOCK DIAGRAM



LOGIC SYMBOL



PIN DESCRIPTION

Designation	Function			
P0.0	PORT 0	• 4-bit I/O port 0 The input (*)/output, the existence (*)/absence of pull-down resistance, and the HALT function release enable/disable condition can be selected for each bit.		Capture trigger signal
P0.1				
P0.2				
P0.3				
P1.0 ~ P1.3	PORT 1	• 4-bit I/O port 1	The input (*)/output, the existence (*)/absence of pull-down resistance, and the HALT function release enable/disable condition can be selected for each bit.	
P2.0	PORT 2	• 4-bit I/O port 2		External interrupt signal
P2.1				
P2.2				
P2.3				
P3.0 ~ P3.3	PORT 3	• 4-bit I/O port 3		
P4.0 ~ P4.3	PORT 4	• 4-bit I/O port 4		
XTOUT	• Oscillator clock output The oscillator clock is output when XTF (forth bit of port P00C) is set to "1".			
XT	• Oscillator connection terminal			
XT				
RESET	• Reset input Input with a pull-down resistance, the system is reset when "1" is input.			
TEST 1	• Test input Input with a pull-down resistance. Generally used when the resistance is open.			
TEST 2				
TEST 3				
SIN	• Circuit configuration			
SOUT	• Serial port data input the circuit is set at high impedance level when "1" is set to HZOUT (the forth bit of port P4XC) or when no data is transmitted.			
SCK	• Serial port clock input and output the input and output (*) are switched from each other the serial port control register SCNT. In the output mode, the serial clock frequency can be selected from the demultiplied signal (1/1, 1/2 or 1/4 of the original base clock) (see note 1).			
BD	• Melody output (buzzer drive output) (not available in the MSM6353.)			
SEG0 ~ SEG61	• LCD drive output LCD drive output with the 1/3 bias, or the 1/3 or 1/4 dusty system. The duty can be switched by LCD control register LCDCNT. A maximum of 177 segments can be displayed with the 1/3 duty and 232 with the 1/4 duty.			

Designation	Function
V_{DD}	• 0V power supply terminal
V_{SS1}	• –1.5V power terminal (for the 1.5V specification)
V_{SS2}	• –3.0V power terminal (for the 3.0V specification)
V_{SS3}	• –4.5V power terminal (not available in the MSM6353)
V_{EE}	• Internal logic power terminal
V_{CM}	• Internal power generator capacitor connecting terminal
V_{CP}	

Note 1: The base clock refers to an oscillator output signal demultiplied into 1/1, 1/2, 1/4, or 1/8 of the output frequency (with the masking option).

FUNCTIONAL DESCRIPTION

A block diagram of MSM6351 is given on page 128. Each block of logic will be briefly discussed. For further information, please refer to MSM6351/53 user's manual.

Programmable ROM

The programmable ROM has a capacity of 4096 words, each of which is 15 bits long. It is provided with the address space of 000 to FFFH.

In the MSM6351, the programmable ROM is not only used for programming but also used to save the following items:

1) LCD indicator segment conversion table
2) Melody tone data

The program instructions are all mde up of one word, thus the ROM can save up to 4096 instructions in it.

Data RAM

The data RAM has a capacity of 1024 words, each consisting of four bits. It is provided with the space for address between 000H and 3FFH. Data is organized in 4 bit nibble.

Page Register (PAGE)

The page register specifies one of 16 pages in each bank of the data RAM. It is used in the addressing bank mode or in the paging mode.

Bank Register (BANK)

The bank register specifies one of four banks in the data RAM. It is used together with the page register in the bank paging mode.

Working Specification Register

The working specification register specifies one of 16 pages in bank 0 of the data RAM. It is used in the addressed working specification mode.

Operational Section

The operational section consists of the ALU, accumulator (ACC), and conditional flags C, Z and G.

This operational section performs four-bit computation of the contents of the data RAM with the contents of ACC or the immediate data fetched into the instruction words.

This computation is mainly performed with the data RAM which functions as a register. The resultant data of the computation are input to the data RAM or to ACC (for operation other than comparison).

Program Counter (PC)

The program counter (PC) generates addresses for the programmable ROM.

The addresses for programming are changed according to the instructions executed. These addresses are incremented by one each time the instruction is executed.

When an interrupt is generated, the current address is stored in the STACK. The address is set to 400H, 401H or 402H depending on the type of the interrupt (see Fig. 3.5.1).

These addresses are set to the start addresses of each interrupt routine.

In the MSM6351, the PC also gives LCD indicator "segment conversion table" or melody "tone data" addresses to the programmable ROM.

The output data of the programmable ROM whose address is specified by the PC is fetched into the instruction register (IR). If the output data is an instruction, it is decoded by the instruction decoder. Then, control signals to each section are generated.

Ports

The MSM6351/6353 handle the I/O ports, flags and registers collectively as ports. Therefore, each of the I/O ports, flags and registers are selected by specifying their own addresses.

All of these ports are accessed by the INP and OUT instructions.

PORT NAMES, ADDRESSES AND THEIR CONTENTS

Port name	Address	Bit 3	Bit 2	Bit 1	Bit 0	Access mode (*)
PORT0	00	P0.3	P0.2	P0.1	P0.0	R/W
PORT1	01	P1.3	P1.2	P1.1	P1.0	R/W
PORT2	02	P2.3	P2.2	P2.1	P2.0	R/W
PORT3	03	P3.3	P3.2	P3.1	P3.0	R/W
PORT4	04	P4.3	P4.2	P4.1	P4.0	R/W
P00C	05	XTF	HRE00	HZ00	DIR00	R/W
P01C	06	BUF	HRE01	HZ01	DIR01	R/W
P02C	07	–	HRE02	HZ02	DIR02	R/W
P03C	08	–	HRE03	HZ03	DIR03	R/W
P1XC	09	–	HRE1X	HZ1X	DIR1X	R/W
P2XC	0A	HZSOUT	HRE2X	HZ2X	DIR2X	R/W
P3XC	0B	EISIO	HRE3X	HZ3X	DIR3X	R/W
P4XC	0C	5/8	HRE4X	HZ4X	DIR4X	R/W
SBF	0D	(L) d_3	d_2	d_1	d_0	R/W
		(u) d_7	d_6	d_5	d_4	
SCNT	0E	CLKSL1	CLKSL0	MODE	LSB/MSB	R/W
SCND	0F	SIOEND	STPErr	ENRC	ENTR	BIT: 3 and 2; R 1 and 0; R/W
IRQEX	10	–	IRQP22	IRQP21	IRQP20	R
EIRT	11	EI256Hz	EI32Hz	EI16Hz	EI1Hz	R/W
IRQRT	12	IRQ256Hz	IRQ32Hz	IRQ16Hz	IRQ1Hz	R
IEXM0	13	–	EIP20	L/E0	P/N0	R/W
IEXM1	14	–	EIP21	L/E1	P/N1	R/W
IEXM2	15	–	EIP22	L/E2	P/N2	R/W
TMOUT	16	15th	14th	13th	12th	R
CAPRT	17	128Hz / 32Hz	256Hz / 64Hz	512Hz / 128Hz	1KHz / 256Hz	R
		–	CAPMD	CAP1F	CAP0F	Bity 2: R/W Other bits: R
FLAG	18	MSTART	G	Z	C	Bity 3: R Other bits: R/W
WDOG	19	W_{03}	W_{02}	W_{01}	W_{00}	R/W
		W_{13}	W_{12}	W_{11}	W_{10}	
FRAMT	1A	F_3	F_2	F_1	F_0	R/W
LCDCT	1B	AIION	Drty 3/4	FLM1	FLM0	R/W

6

PORT NAMES, ADDRESSES AND THEIR CONTENTS (Continued)

Port name	Address	Bit 3	Bit 2	Bit 1	Bit 0	Access mode (*)
TEMPO	1C	P_3	P_2	P_1	P_0	R/W
BANK	1D	b_3	b_2	b_1	b_0	R/W
PAGE	1E	P_3	P_2	P_1	P_0	R/W
WORK	1F	w_2	w_2	w_1	w_0	R/W

Note: • In access mode (*), R denotes "readable" bits and W "writable" bits.
• Bits marked with denote bits which are not present.

I/O Ports

The MSM6351/6353 are provided with five ports; PORT0 to PORT4. Each port consists of four bits.

Ports are controlled by the I/O port control register. The register controls PORT1 to PORT4 port by port and controls PORT0 in bit units.

Each port is accessed by the OUT and INP instruction.

Time Base Counter

The MSM6351/6353 have their built-in time base counters consisting of a 15-stage binary counter. System base clock $\phi 0$ is input to the time base counter clock.

Capture Circuit

The MSM6351/6353 are provided with a capture function that fetches the 1KHz to 128Hz outputs at stages 5 to 8 of the time base counter or the 256 Hz to 32Hz outputs at stages 7 to 10 when P0.0 or P0.1 of I/O port 0 is set at the "H" level.

Watchdog Timer

The MSM6351/6353 have their built-in watchdog timer to prevent any program runway occurrence. The time may be set with two types of setting time: 250ms and 2s.

Serial Port (SERIAL I/O)

The MSM6351/6353 have their built-in serial port. It is used for asynchronous data communications. A data length of five or eight bits can be selected. Either internal or external clock can be selected as the driving clock. At the end of data transfer, a serial port inerrupt can be generated.

The serial port registers and their functions are described in the following sections.

Melody Output Circuit (built in the MSM6351) (MEMODY)

The melody output circuit automatically outputs melody or buzzer sound. It is built in the MSM6351.

The melody circuit initiates its operation by the MSA instruction. Automatically fetching the musical note data defined in the program ROM, the MSA instruction outputs the melody from buzzer driving output terminal BD.

Liquid Crystal Display Circuit (LCD DRIVER)

The MSM6351 has its built-in liquid crystal display circuit that can drive the liquid crystal (LCD).

The liquid crystal display consists of the display data register for writing the data to indicate and the display driver. After data is written in the display data register with an display instruction, the display driver automatically fetches data from the display data register to output the driving waveform.

Interruption Controller (INTERRUPT CONTROL)

There are four types of interruptions as follows:

1) Real-time interruption — Interruption with the time base counter output
2) External interruption — Interruption from PORT2
3) Serial port interruption — Interruption by terminating serial port data transfer
4) Melody interruption — Interruption by requesting melody data (not available in the MSM6353)

ABSOLUTE MAXIMUM RATING (Target Specification)

V_{DD} = 0V (V_{SS2} = Battery Voltage)

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	Ta = 25°C	−6.0 ~ +0.3	V
Input Voltage	V_{IN}		V_{SS2}−0.3 ~ +0.3	V
Output Voltage*1	V_{01}		V_{SS2}−0.3 ~ +0.3	V
Output Voltage*2	V_{02}		−6.0 ~ +0.3	V
Storage Temperature	T_{STG}		−55 ~ +125	°C

*1 Normal Output
*2 When LCD Driver in use

OPERATING CONDITION (Target Specification)

Parameter	Symbol	Limits	Units
Operating Voltage	$-V_{SS2}$	2.6 ~ 3.3	V
Operating Temperature	Topr	−20 to 60	°C

6

DC CHARACTERISTICS 3V Li Battery (Target Specification)

V_{DD} = 0V, V_{SS1} = −1.5V, V_{SS2} = −3.0V, V_{SS3} = −4.5V, f = 32,768Hz, Ta = 25°C*1

Parameter	Symbol	Donditions	Rating			Unit	Terminal Applied
			Min.	Typ.	Max.		
Current Consumption	I_{DD}	*3	—	3.0	—	μA	
Voltage for Oscillation Start	$-V_{OSC}$	Within 2 sec.	—	—	2.4	V	
Output Current 1 (Common Segment)	$-I_{OH1}$	V_{OH1} = −0.2V	4	—	—	μA	SEG0 ~ SEG61
	$\|I_{OMH1}\|$	V_{OMH1} = V_{SS1}±0.2V	4	—	—		
	$\|I_{OML1}\|$	V_{OML1} = V_{SS2}±0.2V	4	—	—		
	I_{OL1}	V_{OL1} = −4.3V	4	—	—		
Output Current 2	$-I_{OH2}$	V_{OH2} = −0.5V	500	—	—	μA	PORT0~PORT4*2 S_{OUT}, $S\overline{CLK}$ X_{TOUT}
	I_{OL2}	V_{OL2} = −2.5V	500	—	—		
Output Current 3	$-I_{ON3}$	V_{ON4} = −0.5V V_{SS2} = 3.0V	7	—	—	μA	BD
	I_{OL3}	V_{ON4} = −2.5V V_{SS2} = 3.0V	20	—	—		
Input Current 1	$-I_{IN1}$	V_{IN1} = 0V I/O input, with pull down	150	300	600	μA	PORT0~PORT4
Input Current 2	$\|I_{I2}\|$	V_{IN2} = 0V, −3V I/O input, without pull down	—	—	1	μA	PORT0~PORT4 S_{IN}, $S\overline{CLK}$
Input Current 3	$-I_{IH3}$	V_{IH3} = 0V with pull down	—	25	—	μA	RESET
Input Voltage	$-V_{IH}$		—	—	0.5	V	All input terminals
	$-V_{IH}$		2.5	—	—		

*1. When 3V battery with halver is used.

*2. PORT0 = P0.0 ~ P0.3, PORT1 = P1.0 ~ P1.3, PORT2 = P2.0 ~ P2.3, PORT 3 = P3.0 ~ P3.3, PORT4 = P4.0 ~ P4.3

*3. This value changes depending on the soft duty (HALT to HALT)

6

INSTRUCTION LIST

	Mnemonic	Instruction code																Description	Machine cycle	Page
		14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
Operation instruction	ADD ACC, REG1	0	0	0	0	0	0	P	0	1	0	0	r_3	r_2	r_1	r_0		(rP), (ACC), (Z), (C) ← (rP) + (ACC)	1	
	ADD #i, REG1	0	0	1	1	0	0	P	i_3	i_2	i_1	i_0	r_3	r_2	r_1	r_0		(rP), (ACC), (Z), (C) ← (rP) + i	1	
	ADC REG1	0	0	0	0	0	0	P	0	1	0	1	r_3	r_2	r_1	r_0		(rP), (ACC), (Z), (C) ← decimal adj [(rP) + (ACC) + (C)]	1	
	ADCN REG1	0	1	1	0	0	0	P	N_3	N_2	N_1	N_0	r_3	r_2	r_1	r_0		(rP), (ACC), (Z), (C) ← N adjust [(rP) + (C)]	1	
	SUB ACC, REG1	0	0	0	0	0	1	P	0	1	0	0	r_3	r_2	r_1	r_0		(rP), (ACC), (Z), (C) ← (rP) − (ACC)	1	
	SUB #i, REG1	0	0	1	1	0	1	P	i_3	i_2	i_1	i_0	r_3	r_2	r_1	r_0		(rP), (ACC), (Z), (C) ← (rP) − i	1	
	SBC REG1	0	0	0	0	0	1	P	0	1	0	1	r_3	r_2	r_1	r_0		(rP), (ACC), (Z), (C) ← decimal adj [(rP) − (ACC) − (C)]	1	
	SBCN REG1	0	1	1	0	0	1	P	N_3	N_2	N_1	N_0	r_3	r_2	r_1	r_0		(rP), (ACC), (Z), (C) ← N adjust [(rP) − (C)]	1	
	CMP ACC, REG1	0	0	0	0	0	1	P	1	1	1	0	r_3	r_2	r_1	r_0		(Z), (G) ← (rP) − (ACC)	1	
	CMP #i, REG1	0	0	1	0	1	1	P	i_3	i_2	i_1	i_0	r_3	r_2	r_1	r_0		(Z), (G) ← (rP) − i	1	
	OMC REG1	0	0	0	0	0	0	P	0	0	0	1	r_3	r_2	r_1	r_0		(rP), (ACC), (Z), (C) ← (rP) + 1	1	
	INCD REG2	1	0	0	0	b_1	b_0	0	P_3	P_2	P_1	P_0	r_3	r_2	r_1	r_0		(rPb), (ACC), (Z), (C) ← (rPb) + 1	1	
	DEC REG1	0	0	0	0	0	1	P	0	0	0	1	r_3	r_2	r_1	r_0		(rP), (ACC), (Z), (C) ← (rP) − 1	1	
	DECD REG2	1	0	0	1	b_1	b_0	0	P_3	P_2	P_1	P_0	r_3	r_2	r_1	r_0		(rPb), (ACC), (Z), (C) ← (rPb) − 1	1	
Bit operation instruction	BIT ACC, REG1	0	0	0	0	0	0	P	1	1	1	0	r_3^3	r_2	r_1^1	r_0		(Z) ← $\overline{(rP_3)} \wedge (ACC_3) \vee \overline{(rP_2)} \wedge (ACC_2) \vee \overline{(rP_1)} \wedge (ACC_1) \vee \overline{(rP_0)} \wedge (ACC_0)$	1	
	BIT #i, REG1	0	0	1	0	1	0	P	i_3	i_2	i_1	i_0	r_3	r_2	r_1	r_0		(Z) ← $\overline{(rP_3)} \wedge i_3 \vee \overline{(rP_2)} \wedge i_2 \vee \overline{(rP_1)} \wedge i_1 \vee \overline{(rP_0)} \wedge i_0$	1	
	BIS ACC, REG1	0	0	0	0	0	0	P	0	1	1	0	r_3	r_2	r_1	r_0		(rP), (ACC), (Z) ← (rP) ∨ (ACC)	1	
	BIS #i, REG1	0	0	1	0	0	0	P	i_3	i_2	i_1	i_0	r_3	r_2	r_1	r_0		(rP), (ACC), (Z) ← (rP) ∨ i	1	
	BIC ACC, REG1	0	0	0	0	0	1	P	0	1	1	0	r_3	r_2	r_1	r_0		(rP), (ACC), (Z) ← (rP) ∧ $\overline{(ACC)}$	1	
	BIC #1, REG1	0	0	1	0	0	1	P	i_3	i_2	i_1	i_0	r_3	r_2	r_1	r_0		(rP), (ACC), (Z) ← (rP) ∧ $\overline{i}$	1	
	XOR ACC, REG1	0	0	0	0	0	0	P	0	1	1	1	r_3	r_2	r_1	r_0		(rP), (ACC), (Z) ← (rP) ∨ (ACC)	1	
	XOR #i, REG1	0	0	1	1	1	1	P	i_3	i_2	i_1	i_0	r_3	r_2	r_1	r_0		(rP), (ACC), (Z) ← (rP) ∀ i	1	

INSTRUCTION LIST (Continued)

	Mnemonic	Instruction code																Description	Machine cycle	Page
		14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
Rotate instruction	ROR REG1	0	0	0	0	0	0	P	0	0	1	0	r_3	r_2	r_1	r_0	(Z), (ACC) ← [→ (C) → (rP) →]	1	
	ROL REG1	0	0	0	0	0	1	P	0	0	110	r_3	r_2	r_1	r_0	r	(Z), (ACC) ← [← (C) ← (rP) ←]	1	
	ASR REG1	0	0	0	0	0	0	P	0	0	1	1	r_3	r_2	r_1	r_0	(Z), (ACC) ← [O → (rP) → (C)]	1	
	ASL REG1	0	0	0	0	0	1	P	0	0	1	1	r_3	r_2	r_1	r_0	(Z), (ACC) ← [(C) ← (rP) ← O]	1	
Flag operation instruction	CLG	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	(G) ← O	1	
	CLC	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	(C) ← O	1	
	CLZ	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	(Z) ← O	1	
	CLA	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	(Z), (C), (G) ← O	1	
	SEG	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	(G) ← 1	1	
	SEC	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	(C) ← 1	1	
	SEZ	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	(Z) ← 1	1	
	SEA	0	0	0	0	0	1	0	1	0	1	1	0	0	0	0	(Z), (C), (G) ← 1	1	
Date transfer instruction	MOV ACC, REG1	0	0	0	0	0	0	P	1	1	1	1	r_3	r_2	r_1	r_0	(rP) ← (ACC)	1	
	MOVD ACC, REG2	1	0	1	0	b_1	b_0	0	P_3	P_2	P_1	P_0	r_3	r_2	r_1	r_0	(rPb) ← (ACC)	1	
	MOV #i, REG1	0	0	1	1	1	0	P	i_3	i_2	i_1	i_0	r_3	r_2	r_1	r_0	(rP), (ACC), (Z) ← i	1	
	MOV REG1, ACC	0	0	0	0	0	1	P	1	1	1	1	r_3	r_2	r_1	r_0	(ACC), (Z) ← (rP)	1	
	MOVD REG2, ACC	1	0	1	1	b_1	b_0	0	P_3	P_2	P_1	P_0	r_3	r_2	r_1	r_0	(ACC), (Z) ← (rPb)	1	
	EXG REG1	0	0	0	0	0	1	P	0	0	0	0	r_3	r_2	r_1	r_0	(rP) ↔ (ACC)	1	
	EXGD REG2	0	1	1	1	b_1	b_0	0	P_3	P_2	P_1	P_0	r_3	r_2	r_1	r_0	(rPb) ↔ (ACC)	1	
Sub-routine instruction	CALL adrs	1	1	1	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	(STACK) o (PC), (PC) ← a_{11} ~ a_0 (SP) ← (SP) + 1	1	
	RET	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	(PC) ← (STACK) + 1 (SP) ← (SP) − 1	1	
	RT1	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	(PC) ← (STACK) + 1 (SP) ← (SP) − 1 (at INT routine)	1	
Jump instruction	JMP adrs	1	1	0	a_{11}	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	(PC) ← a_{11} ~ a_0	1	
	JMP a REG1	0	0	0	0	0	0	P	1	1	0	1	r_3	r_2	r_1	r_0	(PC) ← (PC) + (rP) + 1	1	
	JMPIO a REG1	0	0	0	0	0	1	P	1	1	0	1	r_3	r_2	r_1	r_0	(PC) ← (PC) + 7 A (rP) + 1	1	

INSTRUCTION LIST (Continued)

	Mnemonic	Instruction code 14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Description	Machine cycle	Page
Jump instruction	BGT n	0	0	0	0	1	1	P	0	0	1	n_4	n_3	n_2	n_1	n_0	if (G) = 1 then (PC) ← (PC) + N else (PC) ← (PC) + 1	1	
	BLE n	0	0	0	0	1	1	P	1	0	1	n_4	n_3	n_2	n_1	n_0	if (G) = 0 then (PC) ← (PC) + N else (PC) ← (PC) + 1	1	
	BCS n	0	0	0	0	1	1	P	0	0	0	n_4	n_3	n_2	n_1	n_0	if (C) = 1 then (PC) ← (PC) + N else (PC) ← (PC) + 1	1	
	BCC n	0	0	0	0	1	1	P	1	0	0	n_4	n_3	n_2	n_1	n_0	if (C) = 0 then (PC) ← (PC) + N else (PC) ← (PC) + 1	1	
	BEQ n (BZE n)	0	0	0	0	1	1	P	0	1	0	n_4	n_3	n_2	n_1	n_0	if (Z) = 1 then (PC) ← (PC) + N else (PC) ← (PC) + 1	1	
	BNE n (BNZ n)	0	0	0	0	1	1	P	1	1	0	n_4	n_3	n_2	n_1	n_0	if (Z) = 0 then (PC) ← (PC) + N else (PC) ← (PC) + 1	1	
	BGE n	0	0	0	0	1	1	P	0	1	1	n_4	n_3	n_2	n_1	n_0	if [(G) = 1 or (Z) = 1] then (PC) ← (PC) + N else (PC) → (PC) + 1	1	
	BLT n	0	0	0	0	1	1	P	1	1	1	n_4	n_3	n_2	n_1	n_0	if [(G) = 0 or (Z) = 0] then (PC) ← (PC) + N else (PC) ← (PC) + 1	1	
																	When P = 0 in bit 8, N = n + 1; when P = 1, N = −n		
Melody start	MSA adrs*	0	0	0	0	1	0	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Specifies the first address of note data. ($E00_H$ ~ FFF_H)	2	
Display instruction	DSP dig, REG1*	0	0	0	1	0	0	P	dig_3	dig_2	dig_1	dig_0	r_3	r_2	r_1	r_0	digit (Lower Part) ← (rP), (ACC)	1	
	DSPH dig, REG1*	0	0	0	1	0	1	P	dig_3	dig_2	dig_1	dig_0	r_3	r_2	r_1	r_0	digit (High Part) ← (rP), (ACC)	1	
	DSPF dig, REG1*	0	0	0	1	1	0	P	dig_3	dig_2	dig_1	dig_0	r_3	r_2	r_1	r_0	digit (Low Part) ← (rP) via Table	2	
	DSPFH dig, REG1*	0	0	0	1	1	1	P	dig_3	dig_2	dig_1	dig_0	r_3	r_2	r_1	r_0	digit (High Part) ← (rP) via Table	2	
Input/output instruction	OUT REG1, PORT	0	1	0	1	0	y_4	P	y_3	y_2	y_1	y_0	r_3	r_2	r_1	r_0	(PORT y) ← (rP)	1	
	OUT #i, PORT	0	1	0	1	1	y_4	0	y_3	y_2	y_1	y_0	i_3	i_2	i_1	i_0	(PORT y) ← i	1	
	INP PORT, REG1	0	1	0	0	0	y_4	P	y_3	y_2	y_1	y_0	r_3	r_2	r_1	r_0	(rP), (ACC) ← (PORT y)	1	
CPU control	NOP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	No Operation	1	
	HALT	0	0	0	0	0	1	0	0	1	1	1	0	0	0	0	Halt CPU	1	

Note: Instructions marked with an asterisk (*) are available for MSM6351 only.

OKI semiconductor

MSM6052

CMOS 4BIT SINGLE CHIP LOW POWER MICROCONTROLLER FOR TELEPHONE

GENERAL DESCRIPTION

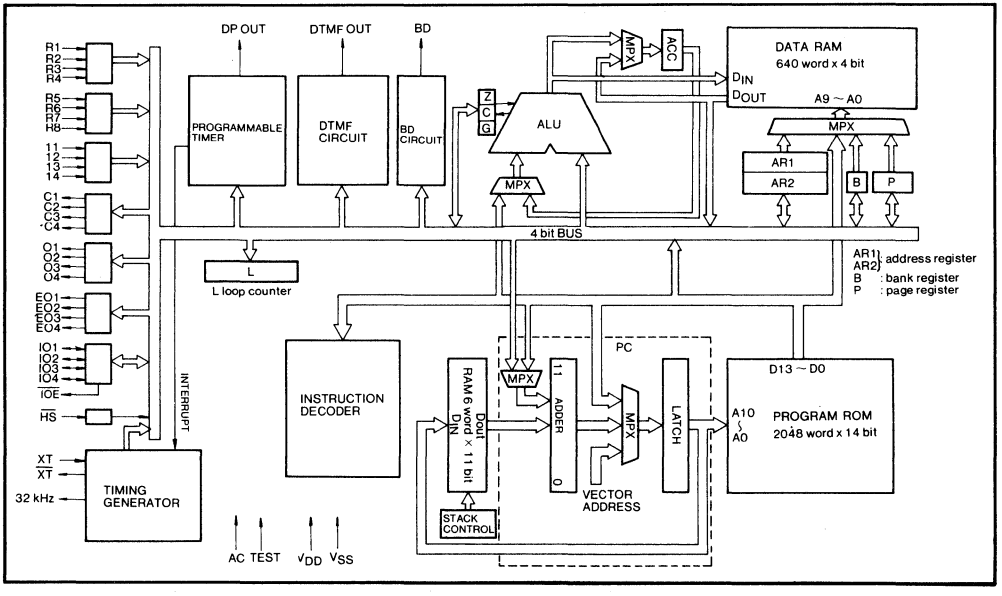
The OKI MSM6052 is low-power, high-performance single-chip 4-bit microcontroller employing complementary metal oxide semiconductor technology, especially designed for use in sophisticated telephone sets. Integrated onto a single chip are 4 bits of ALU, 28K bits of mask programmable ROM, 2560 bits of data RAM, programmable timer, oscillator, 12-bits of input port, 12-bits of output port and 4-bits of input/output port. In addition to these units, a DTMF generator is provided.

With the MSM6052, sophisticated telephone sets become feasible through a single chip instead of the conventional 3-chip configuration.

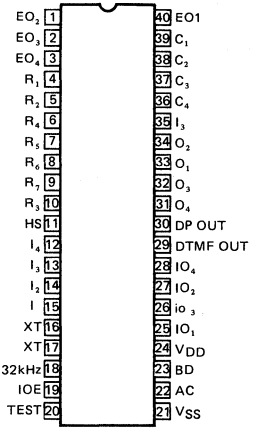
FEATURES

- Low Power Consumption 0.3mA Typical @3V (DTMF output off)
- 2048 × 14 Internal ROM
- 640 × 4 Internal RAM
- 3 × 4 Input Port
- 3 × 4 Output Port
- 1 × 4 Input/Output Port
- DTMF Generator
- Buzzer Sound Output
- 4-Bit Programmable Timer Applicable for Output of Dial Pulse
- Interrupt by Progammable Timer
- 5 Level Stack
- Power Down Mode
- 52 Instructions
- Instructions Useful for Data Management (Data Search and Block Data Transfer)
- 2.5 to 6.0V Operating Voltage
- 3.58 MHz Oscillator
- 17.9 μs Instruction Cycle
- −20 to 75°C Operating Temperature
- 28 Pin DIP or 40 Pin DIP

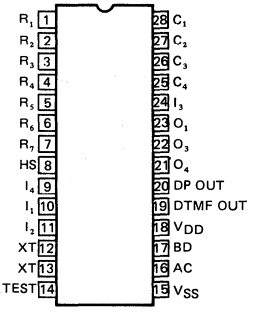
FUNCTIONAL BLOCK DIAGRAM



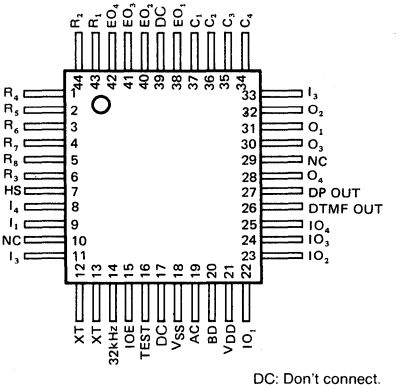
PIN CONFIGURATION



(a) 40 Lead Plastic DIP



(b) 28 Lead Plastic DIP

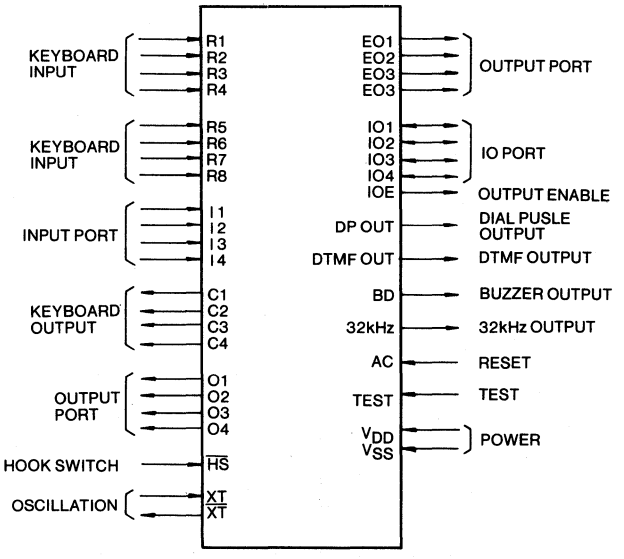


DC: Don't connect.

(c) 44 Lead Plastic Flat Package
(NC pin must not be connected to any signal.)

6

LOGIC SYMBOL



PIN DESCRIPTION

Designation	Function
V_{DD}	Power source
V_{SS}	Circuit ground potential
AC	Terminal to clear internal logic, pulled down to V_{SS}. After power is turned on, the MSM6052 must be reset by this terminal.
TEST	Terminal to test internal logic, pulled down to V_{SS}. This terminal must be open in normal operation.
XT, $\overline{XT}$	Input and output terminals of oscillator inverter. 3.58 MHz ceramic resonator is connected to these terminals.
$\overline{HS}$	Input terminal connected to the hook switch, pulled up tp V_{DD}.
DP OUT	Output terminal of dial pulse. Dial pulse rate (10 pps or 20 pps) and Make Break ratio (40% or 33 %) can be selected by software.
DTMF OUT	Output terminal of DTMF signal
BD	Output terminal of buzzer sound
32 kHz	Output terminal of 32 kHz clock
$R_1 \sim R_4$ $R_5 \sim R_8$	Input port pulled down to V_{SS}.
$I_1 \sim I_4$	Input port having clocked pull-down resistor to V_{SS}. Only when this port is accessed, pull-down resistors are connected to this port.
$C_1 \sim C_4$ $O_1 \sim O_4$	Output port
$IO_1 \sim IO_4$	Tri-state bidirectional port
IOE	Output terminal When $IO_1 \sim IO_4$ is accessed, input completion signal (when read) or load signal (when written) is output from IOE terminal.

FUNCTIONAL DESCRIPTION

A block diagram of the MSM6052 is given on page 129. Each block of logic will be briefly discussed. For more information, please refer to the MSM6052 user's manual.

Program ROM

The MSM6052 will address up to 2 K words of internal mask programmable ROM. Each word consists of 14-bits and all instructions are one word. The instructions are routed to a programmed logic array which generates the signals necessary for control of logic.

Data RAM

Data is organized in 4-bit nibbles. Internal data RAM consists of 640 nibbles.

All locations are addressed by 10-bit address registers (AR_1, AR_2), 2-bit bank register (B), 4-bit page register (P) or a part of the instruction's operand.

ALU

The ALU performs a 4-bit parallel operation on RAM and ACC contents, or on RAM contents and an immediate digit. It sets or resets the three flags (Z, C, G) depending on the condition.

Program Counter (PC)

The program counter is an 11-bit wide counter that specifies the address of program ROM.

The PC is incremented by one at every execution of the instruction, and then specifies the next instruction to be executed. However, the contents of the PC are rewritten by the execution of a Jump, Call or Branch instruction.

As there is no boundary in the ROM, and a Jump, Call or Branch instruction can be put anywhere in the ROM.

Stack

The MSM6052 has a 5 level stack apart from the data RAM. The contents of the PC are loaded into stack when a Call instruction is executed or an interrupt is generated. Nesting of subroutines within subroutines can continue up to 4 times, including the interrupt.

- **Input Port**

Port (R1 ~ R4)

4-bit input port. Each pin of the port is pulled down to V_{SS} by an internal resistor, and status of the port is fetched by an input instruction.

Port (R5 ~ R8)

4-bit input port. Each pin of the port is pulled down to V_{SS} by an internal resistor, and the status of the port is fetched by an input instruction.

Port (I1 ~ I4)

4-bit input port. Each pin of the port is pulled down to V_{SS} by an internal resistor and transistor. Only when it is desired to fetch status of the port, input current flows through these pins. Status of the port is fetched by an input instruction.

- **Output Port**

Port (C1 ~ C4)

4-bit output port. These ports consist of data latches and buffers, and the contents of the data latches are rewritten by an output instruction.

Port (O1 ~ O4)

4-bit output port. This port consists of data latches and buffers, and the contents of the data latches are rewritten by an output instruction.

Electrical characteristics of O3 and O4 are different from those of O1 and O2. O3 and O4 of the ports are used as $\overline{\text{XMIT}}$ $\overline{\text{MUTE}}$ and $\overline{\text{MUTE}}$ normally.

Port (EO1 ~ EO4)

4-bit output port. This port consists of data latches and buffers, and the contents of data latches are rewritten by an output instruction.

- **Input/Output Port**

Port (IO1 ~ IO4)

4-bit bidirectional port. This port consists of data latches, output buffers and input buffers. The contents of the data latches are rewritten by an output instruction, and status of the port is fetched by an input instruction.

Address Registers (AR1, AR2)

The address registers are used to specify the 10-bit address of data RAM, when a data search instruction (RDAR) or block data transfer instruction (MVAR) is executed.

This register is an up/down counter, and is incremented or decremented by 1 with execution of the instruction.

Timing Generator

By connecting a 3.58 MHz ceramic resonator to the XT and $\overline{\text{XT}}$ terminal, the timing generator generates a basic timing signal to control the MSM6052.

The MSM6052 can operate in 2 modes, normal operating mode and power down mode. STOP instruction is used to place the MSM6052 in the power down mode. The oscillation stops and all functions are stopped. However, the contents of RAM and all registers are maintained.

Programmable Timer

The programmable timer consists of a 4-bit down counter and a 1/100 prescaler.

Any of a 7990.1 Hz clock, 1997.5 Hz clock and 998.8 Hz clock is input to the 1/100 prescaler. Output of the 1/100 prescaler decrements the 4-bit down counter by 1.

When the contents of the 4-bit down counter is decremented to 0, the programmable timer generates an interrupt.

This programmable timer can be used as a dial pulse generator. The dial pulse rate (10 pps, 20 pps) and Make/Break ratio (40%, 33%) of the dial pulse which the programmable timer generates are selectable.

DTMF Circuit

DTMF circuit is used to generate a DTMF signal. 12 kinds of DTMF signal (0 to 9, #, *) can be output by an output instruction.

BD Circuit

The BD circuit generates the square wave which can be used as the confirmation sound, warning sound and so on. 15 kinds of sound (4.66 to 0.82 kHz) are output by an output instruction specifying the frequency.

ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	Ta = 25°C	−0.3 to 7.0	V
Input Voltage	V_I	Ta = 25°C	−0.3 to V_{DD}+0.3	V
Output Voltage	V_O	Ta = 25°C	−0.3 to V_{DD}+0.3	V
Storage Temperature	Tstg	–	−55 to 125	°C
Power Dissipation	P_D	Ta = 25°C	200	mW

OPERATING CONDITIONS

Parameter	Symbol	Limits	Unit
Operating Voltage	V_{DD}	2.5 to 6.0	V
Memory Retension Voltage	V_{DDM}	1.2 to 6.0	V
Operating Temperature	Topr	−20 to 75	°C

DC CHARACTERISTICS

(VDD = 3V, Ta = −20 to 75°C)

Parameter	Symbol	Conditions		Limits			Unit
				Min.	Typ.	Max.	
"H" Input Voltage	V_{IH}	V_{DD}=3V		2.2	–	–	V
		V_{DD}=6V		4.4	–	–	V
"L" Input Voltage	V_{IL}	V_{DD}=3V		–	–	0.8	V
		V_{DD}=6V		–	–	1.6	V
"H" Output Current (1)	I_{OH1}	O_3, O_4 DP OUT	V_{OH}=2.6V	−200	–	–	μA
"L" Output Current (1)	I_{OL1}		V_{OL}=0.4V	500	–	–	μA
"H" Output Current (2)	I_{OH2}	$C_1 \sim C_4$	V_{OH}=2.6V	−1	–	–	mA
"L" Output Current (2)	I_{OL2}		V_{OL}=0.4V	10	–	–	μA
"H" Output Current (3)	I_{OH3}	O_1, O_2, BD	V_{OH}=2.6V	−20	–	–	μA
"L" Output Current (3)	I_{OL3}		V_{OL}=0.4V	10	–	–	μA

DC CHARACTERISTICS (CONT.)

Parameter	Symbol	Conditions		Limits			Unit
				Min.	Typ.	Max.	
"H" Output Current (4)	I_{OH4}	$IO_1 \sim IO_4$ IOE $EO_1 \sim EO_4$	V_{OH}= 2.6V	−150	–	–	μA
"L" Output Current (4)	I_{OL4}		V_{OL}= 0.4V	300	–	–	μA
"H" Output Current (5)	I_{OH5}	32 kHz	V_{OH}=2.6V	−40	–	–	μA
"L" Output Current (5)	I_{OL5}		V_{OL}=0.4V	25	–	–	μA
Pull-up Resistance	R_{UP}	$\overline{HS}$		17	–	150	kΩ
Pull down Resistance (1)	Rdwon 1	$R_1 \sim R_8$		33	–	300	kΩ
Pull down Resistance (2)	Rdwon 2	$I_1 \sim I_4$, AC, TEST		10		100	kΩ
Input Leak Current	I_{IL}	$IO_1 \sim IO_4$	$0 \leq V_{IN} \leq V_{DD}$ V_{DD}= 2.5 to 6.0V	–	–	±2	μA
Current Consumption (1)	I_{DDP}	DTMF output off	V_{DD}= 3V	–	0.3	0.6	mA
			V_{DD}= 6V	–	1.2	2.4	mA
Current Consumption (2)	I_{DDT}	DTMF output on	V_{DD}= 3V	–	1.2	2.4	mA
			V_{DD}=6V	–	3.5	7.0	mA
Memory retention Current	I_{DDM}	ON HOOK V_{DD}=2.5V	Ta=25°C	–	0.01	0.2	μA
			Ta=−20to75°C	–	–	2	μA

6

AC CHARACTERISTICS

(V_{DD} = 3V, Ta = −20 to 75°C)

Parameter	Symbol	Conditions		Limits			Unit
				Min.	Typ.	Max.	
Key Input Time	T_{KIN}	V_{DD}=2.5 to 6.0V		33	–	–	ms
Tone Output Voltage	V_{OUT}	Row only R_L=1 kΩ	V_{DD}=2.5V	150	250	350	mV rms
			V_{DD}=4.0V	200	350	570	
			V_{DD}=6.0V	300	480	850	
High/Low Level Ratio	dB_{CR}	V_{DD}=2.5 to 6.0V		1	2	3	dB
Distortion Ratio	%DIS	R_L=1 kΩ		–	1	5	%
Rise/Fall Time (1)	t_{TLH1}	O_3, O_4, DP OUT C_L=50 pF		–	–	0.5	μS
	t_{THL1}			–	–	0.5	
Rise/Fall Time (2)	t_{TLH2}	$C_1 \sim C_4$ C_L=50 pF		–	–	0.5	μS
	t_{THL2}			–	–	10	
Rise/Fall Time (3)	t_{TLH3}	O_1, O_2, BD, 32 kHz C_L=50 pF		–	–	5	μS
	t_{THL3}			–	–	10	
Rise/Fall Time (4)	t_{TLH4}	$IO_1 \sim IO_4$, IOE, $EO_1 \sim EO_4$ C_L=50 pF		–	–	1	μS
	t_{THL4}			–	–	1	

DESCRIPTION OF INSTRUCTIONS

	Mnemonic	Instruction Code 13 12	11 10 9 8	7 6 5 4	3 2 1 0	Operation
Arithmetic and logic	ADD ACC, AP	0 0	0 0 0 P	0 1 0 0	A	AP ← (AP) + ACC
	ADD #D, AP	0 1	1 0 0 P	D	A	AP ← (AP) + D
	ADC AP	0 0	0 0 0 P	0 1 0 1	A	AP ← (AP) + ACC + C
	SUB ACC, AP	0 0	0 0 1 P	0 1 0 0	A	AP ← (AP) − ACC
	SUB #D, AP	0 1	1 0 1 P	D	A	AP ← (AP) − D
	SBC AP	0 0	0 0 1 P	0 1 0 1	A	AP ← (AP) − ACC − C
	CMP ACC, AP	0 0	0 0 1 P	1 1 1 0	A	(AP) − ACC
	CMP #D, AP	0 1	0 1 1 P	D	A	(AP) − D
	XOR ACC, AP	0 0	0 0 0 P	0 1 1 1	A	AP ← (AP) ∀ ACC
	XOR #D, AP	0 1	1 1 1 P	D	A	AP ← (AP) ∀ D
Bit operation	BIT ACC, AP	0 0	0 0 0 P	1 1 1 0	A	(AP) V $\overline{ACC}$
	BIT #D, AP	0 1	0 1 0 P	D	A	(AP) V $\overline{D}$
	BIS ACC, AP	0 0	0 0 0 P	0 1 1 0	A	AP ← (AP) V ACC
	BIS #D, AP	0 1	0 0 0 P	D	A	AP ← (AP) V D
	BIC ACC, AP	0 0	0 0 1 P	0 1 1 0	A	AP ← (AP) Λ $\overline{ACC}$
	BIC #D, AP	0 1	0 0 1 P	D	A	AP ← (AP) Λ $\overline{D}$
Rotate	ROR AP	0 0	0 0 0 P	0 0 1 0	A	⌊(AP) → C⌋
	ROL AP	0 0	0 0 1 P	0 0 1 0	A	⌊(AP) ← C⌋
	ASR AP	0 0	0 0 0 P	0 0 1 1	A	0 → (AP) → C
	ASL AP	0 0	0 0 1 P	0 0 1 1	A	C ← (AP) ← 0
Flag operation	SEZ	0 0	0 0 1 0	1 0 1 0	0 0 0 0	Z ← 1
	CLZ	0 0	0 0 0 0	1 0 1 0	0 0 0 0	Z ← 0
	SEC	0 0	0 0 1 0	1 0 0 1	0 0 0 0	C ← 1
	CLC	0 0	0 0 0 0	1 0 0 1	0 0 0 0	C ← 0
	SEG	0 0	0 0 1 0	1 0 0 0	0 0 0 0	G ← 1
	CLG	0 0	0 0 0 0	1 0 0 0	0 0 0 0	G ← 0
	SEA	0 0	0 0 1 0	1 0 1 1	0 0 0 0	Z ← 1, C ← 1, G ← 1
	CLA	0 0	0 0 0 0	1 0 1 1	0 0 0 0	Z ← 0, C ← 0, G ← 0
Data transfer	MOV ACC, AP	1 1	1 1 0 1	0 0 0 0	A	AP ← ACC
	MOV ACC, AX	1 1	1 1 0 0	X	A	AX ← ACC
	MOV #D, AP	0 1	1 1 0 P	D	A	AP ← D
	MOV AP, ACC	1 1	1 1 1 1	0 0 0 0	A	ACC ← (AP)
	MOV AX, ACC	1 1	1 1 1 0	X	A	ACC ← (AX)
	CHG AP	1 1	1 0 0 1	0 0 0 0	A	(AP) ⟷ ACC

6

DESCRIPTION OF INSTRUCTIONS (CONT.)

	Mnemonic	Instruction Code																Operation
		13	12	11	10	9	8	7	6	5	4	3	2	1	0			
Data transfer	CHG AX	1	1	1	0	0	0			X				A				(AX) ⟷ ACC
	RDAR	1	1	0	0	0	0	0	0	0	0	0	0	0	0			ACC ← (AR_1)
	RDAR + (–)	1	1	0	0	0	0	0	0	1	D/I	0	0	0	0			ACC ← (AR_1), AR_1 ← $AR_1 \pm 1$
	RDAR + (–), Z	1	1	0	0	0	0	0	1	0	D/I	0	0	0	0			ACC ← (AR_1) if (AR_1)=0 then PC ← PC + 1 else AR_1 ← $AR_1 \pm 1$, repeat
	RDAR + (–), N	1	1	0	0	0	0	1	0	0	D/I	0	0	0	0			ACC ← (AR_1) if (AR_1)≠0 then PC ← PC + 1 else AR_1 ← $AR_1 \pm 1$, repeat
	RDAR + (–), Z, L	1	1	0	0	1	0	0	1	0	D/I	0	0	0	0			ACC ← (AR_1), L ← L – 1 if (AR_1)=0 or L=0 then PC← PC+1 else AR_1 ← $AR_1 \pm 1$, repeat
	RDAR + (–), N, L	1	1	0	0	1	0	1	0	0	D/I	0	0	0	0			ACC ← (AR_1), L ← L – 1 if (AR_1)≠0 or L=0 then PC ← PC + 1 else AR_1 ← $AR_1 \pm 1$, repeat
	MVAR	1	1	0	1	0	0	0	0	0	0	0	0	0	0			AR_2 ← (AR_1)
	MVAR + (–)	1	1	0	1	0	0	0	0	1	D/I	0	0	0	0			AR_2 ← (AR_1), AR_1 ← $AR_1 \pm 1$, AR_2 ← $AR_2 \pm 1$
	MVAR + (–), Z	1	1	0	1	0	0	0	1	0	D/I	0	0	0	0			AR_2 ← (AR_1), if (AR_1)=0 then PC ← PC +1 else AR_1 ← $AR_1 \pm 1$, AR_2 ← $AR_2 \pm 1$, repeat
	MVAR + (–), N	1	1	0	1	0	0	1	0	0	D/I	0	0	0	0			AR_2 ← (AR_1) if (AR_1)≠0 then PC ← PC +1 else AR_1 ← $AR_1 \pm 1$, AR_2 ← $AR_2 \pm 1$, repeat
	MVAR + (–), L	1	1	0	1	1	0	0	0	0	D/I	0	0	0	0			AR_2 ← (AR_1), L ← L – 1 if L=0 then PC ← PC +1 else AR_1 ← $AR_1 \pm 1$, AR_2 ← $AR_2 \pm 1$, repeat
	MVAR + (–), Z, L	1	1	0	1	1	0	0	1	0	D/I	0	0	0	0			AR_2 ← (AR_1), L ← L –1 if (AR_1)=0 or L=0 then PC ← PC +1 else AR_1 ← $AR_1 \pm 1$, AR_2 ← $AR_2 \pm 1$, repeat
	MVAR + (–), N, L	1	1	0	1	1	0	1	0	0	D/I	0	0	0	0			AR_2 ← (AR_1), L ← L –1 if (AR_1)≠0 or L=0 then PC ← PC +1 else AR_1 ← $AR_1 \pm 1$, AR_2 ← $AR_2 \pm 1$, repeat
Sub routine	CALL adrs	1	0	1	a_{10}	a_9	a_8	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0			STACK ← (PC), PC ← adrs
	RET	0	0	0	0	0	0	1	1	0	0	0	0	0	0			PC ← (STACK) + 1
	RTI	0	0	0	0	1	0	1	1	0	0	0	0	0	0			PC ← (STACK) or PC ← (STACK) +1

DESCRIPTION OF INSTRUCTIONS (CONT.)

	Mnemonic	Instruction Code 13 12	11 10 9 8	7 6 5 4	3 2 1 0	Operation
Jump	JMP adrs	1 0	0 a_{10} a_9 a_8	a_7 a_6 a_5 a_4	a_3 a_2 a_1 a_0	PC ← adrs
	JMP @AP	0 0	0 0 0 P	1 1 0 1	A	PC ← (PC) + (AP) + 1
	JMPIO @AP	0 0	0 0 1 P	1 1 0 1	A	PC ← (PC) + {(AP) ∧ 7H} +1
Branch	BEQ n (BZE n)	1 1	1 0 1 P	0 1 0 n_4	n_3 n_2 n_1 n_0	if Z=1 then PC←PC − n or PC ← PC + n +1 else ← PC ← PC + 1
	BNE n (BNZ n)	1 1	1 0 1 P	1 1 0 n_4	n_3 n_2 n_1 n_0	if Z=0 then PC←PC − n or PC ← PC + n + 1 else PC ← PC +1
	BCS n	1 1	1 0 1 P	0 0 0 n_4	n_3 n_2 n_1 n_0	if C=1 then PC ← PC −n or PC ← PC + n + 1 else PC ← PC +1
	BCC n	1 1	1 0 1 P	1 0 0 n_4	n_3 n_2 n_1 n_0	if C=0 then PC ← PC − n or PC ← PC+n+1 else PC ← PC +1
	BGT n	1 1	1 0 1 P	0 0 1 n_4	n_3 n_2 n_1 n_0	if G=1 then PC ← PC − n or PC ← PC + n + 1 else PC ← PC +1
	BLE n	1 1	1 0 1 P	1 0 1 n_4	n_3 n_2 n_1 n_0	if G=0 then PC ← PC − n or PC ← PC +n+1 else PC ← PC +1
	BGE n	1 1	1 0 1 P	0 1 1 n_4	n_3 n_2 n_1 n_0	if G=1 or Z=1 then PC ← PC − n or PC ← PC+n+1, else PC ← PC +1
	BLT n	1 1	1 0 1 P	1 1 1 n_4	n_3 n_2 n_1 n_0	if G=0 and Z=0 then PC ← PC −n or PC ← PC+n+1 else PC ← PC+1
Input/ Output	IN PORT, AP	0 0	0 1 0 P	P_L	A	AP ← (PORT)
	OUT AP, PORT	0 0	1 0 P_H P	P_L	A	PORT ← (AP)
	OUT #D, PORT	0 0	1 1 P_H 0	P_L	D	PORT ← D
CPU control and others	STOP	0 0	1 1 1 0	0 0 0 0	0 0 0 0	Stop system clock
	HALT	0 0	1 1 1 0	0 0 0 1	0 0 0 0	Halt CPU
	ACT	0 0	1 1 1 0	0 0 1 0	0 0 0 0	Activate CPU
	EI	0 0	1 1 1 0	0 1 1 0	1 0 0 0	Enable timer interrupt
	DI	0 0	1 1 1 0	0 1 1 0	0 1 0 0	Disable timer interrupt
	ET	0 0	1 1 1 0	0 1 1 0	0 0 1 0	Enable timer activate
	DT	0 0	1 1 1 0	0 1 1 0	0 0 0 1	Disable timer activate
	EC	0 0	1 1 1 0	0 1 1 1	1 0 0 0	Enable output port (C_1~C_4)
	DC	0 0	1 1 1 0	0 1 1 1	0 1 0 0	Disable output port (C_1~C_4)

6

DESCRIPTION OF INSTRUCTIONS (CONT.)

	Mnemonic	Instruction Code																Operation
		13	12	11	10	9	8	7	6	5	4	3	2	1	0			
CPU control and others	OM	0	0	1	1	1	0	0	1	1	1	0	0	1	0	Set I/O port (IO_1~IO_4) to output mode		
	IM	0	0	1	1	1	0	0	1	1	1	0	0	0	1	Set I/O port (IO_1~IO_4) to input mode		
	RST	0	0	1	1	1	0	1	0	0	1	0	0	0	0	Reset divider		
	NOP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	No operation		

MSM6352

CMOS 4BIT SINGLE CHIP LOW POWER MICROCONTROLLER FOR TELEPHONE

GENERAL DESCRIPTION

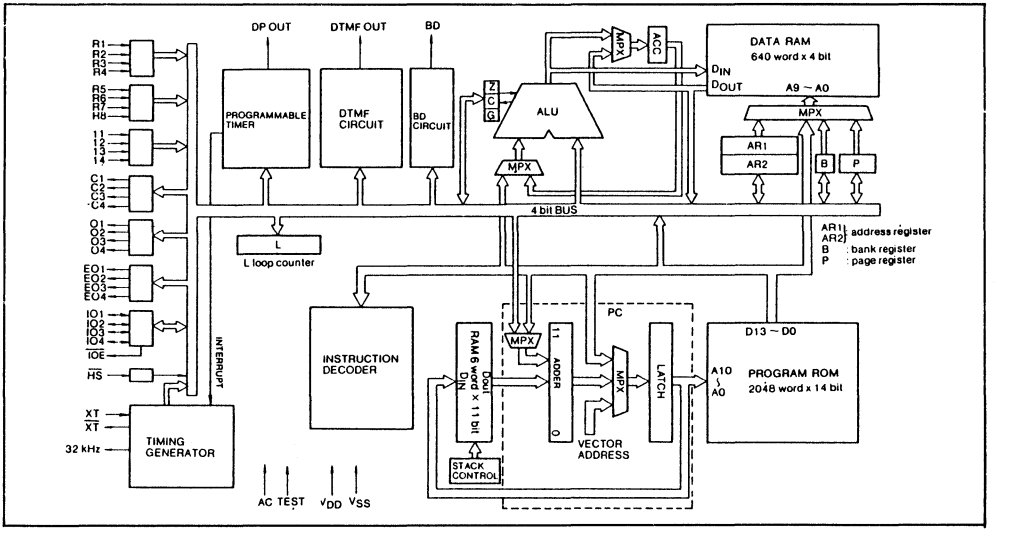
The OKI MSM6352 is a low-power, high-performance single-chip 4-bit microcontroller employing complementary metal oxide semiconductor technology, especially designed for use in sophisticated telephone sets. Integrated onto a single chip are a 4-bit ALU, 28K bits of mask programmable ROM, 2560 bits of data RAM, programmable timer, oscillator, 12-bits of input port, 12-bits of output port and 4-bits of input/output port. In addition to these units, a DTMF generator is provided.

With the MSM6352, sophisticated telephone sets become feasible through a single chip instead

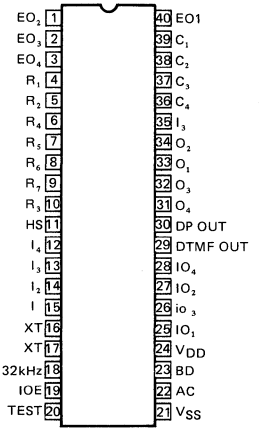
FEATURES

- Low Power Consumption 1.8mA Typical @3V (DTMF output off)
- 2048 × 14 Internal ROM
- 640 × 4 Internal RAM
- 3 × 4 Input Port
- 3 × 4 Output Port
- 1 × 4 Input/Output Port
- DTMF Generator (Single Tone Mode or Dual Tone Mode)
- Buzzer Sound Output
- 4 Bits Programmable Timer Applicable for Output of Dial Pulse
- Watch Dog Timer
- On Hook Dialing and Off Hook Dialing Function
- Interrupt
  Programmable Timer-Interrupt
  Real Time Interrupt
- 5 Level Stack
- Power Down Mode
- 52 Instruction Set
- Instructions Useful for Data Management (Data Search and Block Data Transfer)
- 2.0 to 5.5V (2.2 to 5.5V at TONE MODE) Operating Voltage
- Low Voltage Detector
- 3.58 MHz Oscillator
- 17.9 μs Instruction Cycle
- −20 to 75°C Operating Temperature
- 28 Pin DIP, 40 Pin DIP or 44 Pin FLAT
- Software Compatibility with MSM6052

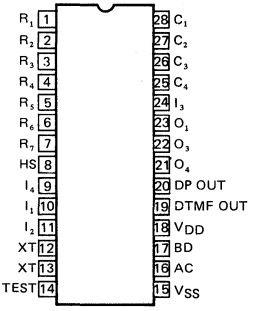
FUNCTIONAL BLOCK DIAGRAM



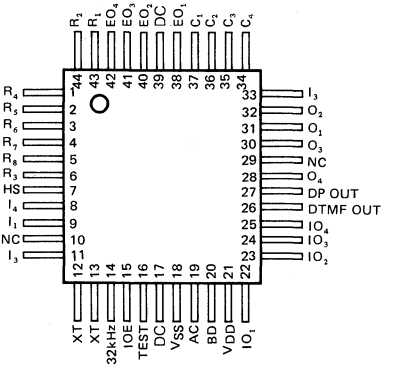
PIN CONFIGURATION



(a) 40 Lead Plastic DIP



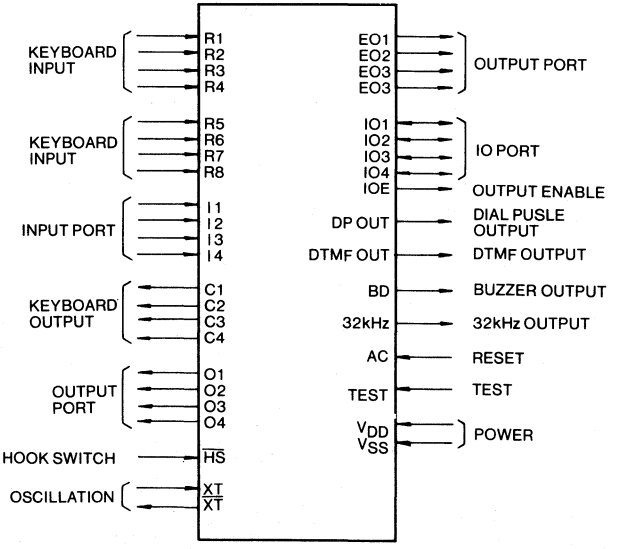
(b) 28 Lead Plastic DIP



DC: Don't connect.

(c) 44 Lead Plastic Flat Package
(NC pin must not be connected to any signal.)

LOGIC SYMBOL



PIN DESCRIPTION

Designation	Function
V_{DD}	Pource source
V_{SS}	Circuit ground potential
AC	Terminal to clear internal logic, pulled down to V_{SS}. After power is turned on, the MSM6052 must be reset by this terminal.
TEST	Terminal to test internal logic, pulled down to V_{SS}. This terminal must be open in normal operation.
XT, $\overline{XT}$	Input and output terminals of oscillator inverter. 3.58 MHz ceramic resonator is connected to these terminals.
$\overline{HS}$	Input terminal connected to the hook switch, pulled up tp V_{DD}.
DP OUT	Output terminal of dial pulse. Dial pulse rate (10 pps or 20 pps) and Make Break ratio (40% or 33 %) can be selected by software.
DTMF OUT	Output terminal of DTMF signal
BD	Output terminal of buzzer sound
32 kHz	Output terminal of 32 kHz clock
$R_1 \sim R_4$ $R_5 \sim R_8$	Input port pulled down to V_{SS}.
$I_1 \sim I_4$	Input port having clocked pull-down resistor to V_{SS}. Only when this port is accessed, pull-down resistors are connected to this port.
$C_1 \sim C_4$ $O_1 \sim O_4$	Output port
$IO_1 \sim IO_4$	Tri-state bidirectional port
IOE	Output terminal When $IO_1 \sim IO_4$ is accessed, input completion signal (when read) or load signal (when written) is output from IOE terminal.

FUNCTIONAL DESCRIPTION

A block diagram of the MSM6352 is given on page 149. Each block of logic will be briefly discussed. For more information, please refer to the MSM6352 user's manual.

Program ROM

The MSM6352 will address up to 1 K words of internal mask programmable ROM. Each word consists of 14-bits and all instructions are one word. The instructions are routed to a programmed logic array which generales the signals necessary for control of logic.

Data RAM

Data is organized in 4-bit nibbles. Internal data RAM consists of 640 nibbles.

All locations are addressed by 10-bit address registers (AR_1, AR_2), 2-bit bank register (B), 4-bit page register (P) or a part of the instruction's operand.

Program Counter (PC)

The PC is an 11-bit counter to specify the ROM's address. The PC is normally incremented by one by every execution of the instruction, and then specifies the next instruction to be executed. However, Jump, Conditional branch, and Subroutine instructions are exceptions.

When the JMP adrs or CALL adrs instruction is executed, all of the PC contents are rewritten, so jump can be done to any address of the ROM.

Bank Register (B)

The bank register is a 2 bits register which specifies the bank of the RAM. Read/Write operation is performed by the Input/Output instruction.

Page Register (P)

The page register is a 4 bits register which specifies the page of the RAM. Read/Write operation is performed by the Input/Output instruction.

Address Register 1 (AR_1)

The address register AR_1 is a 10 bits register which specifies the RAM's address. This register is used by the RDAR instruction or the MVAR instruction. Read/Write operation is performed by the Input/Output instruction.

Address Register 2 (AR_2)

The address register AR_2 is a 10 bits register which specifies the RAM's address. This register is used by the MVAR instruction. Read/Write operation is performed by the Input/Output instruction.

Loop Counter (L)

The loop counter is a 10 bits down counter which specifies a number of words of the data to be searched or to be moved by the RDAR instruction or the MVAR instruction. Its contents can be rewritten by the output instruction.

ALU, Conditional Flag, ACC

(a) ALU

The ALU performs 4-bits parallel operation of the RAM contents and ACC contents, or the RAM contents and an immediate data. The arithmetic, logic, comparison and rotate operations can be done.

(b) Conditional flag

The zero flag (Z), carry flag (C) and greater flag (G) are provided. These flags are set or reset depending on the operation result and referred to by the conditional branch instruction.

The flag operation instruction enables these flags to be set or reset individually or allogether.

(c) ACC

The ACC is 4-bits register for arithmetic, and equipped with data transfer instruction between ACC and RAM.

Stack

The stack consists of a RAM of 5 words x 11 bits. It is used to save the PC contents when the sub-routine is called or a timer interrupt is generated, and 5-level nesting can be done including a timer interrupt. The PC contents saved in the stack is popped to the PC by the RETURN instruction.

Interrupt

MSM6352 has two kinds of interrupt as below.

- Realtime interrupt
- Programmabletimer interrupt

Stop Mode

Stop mode is established by the execution of the STOP instruction. In the stop mode, oscillation of system clock stops and all operations are suspended, but the RAM contents and all register contents are maintained.

Halt Mode

Halt mode is established by the execution of the HALT instruction and the execution of program of main routine is suspended. In the halt mode, all RAM contents and register contents are maintained.

Timer Activation and Realtime Interrupt Circuit

The timer activation and realtime interrupt circuit are to release HALT mode (timer activiation) and to generate interrupt (Realtime interrupt) at the talling edge of the 31.21 Hz clock obtained by dividing the 3.579545 MHz system clock by 114688. The timer activation and realtime interrupt circuit can be used for the generation of timer activation and realtime interrupt by setting or resetting the mode setting flag (TMF).

Divider Circuit

The 3 stage binary divider circuit to which 31.21 Hz clock is supplied is provided. The divider circuit's contents can be read by the input instruction (IN2, AP), and at the same time HS input port data is also read. The divider circuit can be reset by the RST instruction.

Programmable Timer and Programmable Timer Interrupt

The programmable timer is used for dial pulse output or timer interrupt generation. This timer consists of control resistor PTL, 1/100 divider circuit, 4 bit presettable down counter PTC, interrupt flag IRQF, interrupt enable flag EIF, selection flag EOF of off and on-hook dialing made, and dial pulse phase selection flag DPE, 1/100 divider circuit, PTC, IRQF, EIF, EOF and DPF are reset at system reset.

DTMF Output Circuit

The DTMF output circuit is to generate DTMF tone signal and is controlled by DTMF and TONE register. Rewriting the contents of the output latch for DTMF circuit by output instructions, 12 kinds of dual or single tones can be output to the DTMF output port. The tone output frequency is selected by the DTMF register.

BD Circuit

The BD circuit generates the square wave which can be used as the confirmation sound, warning sound and so on. 15 kinds of sound (4.66 to 0.82 kHz) are output by an output instruction specifying the frequency.

Watchdog Timer WDT

The watchdog timer is to generate the system reset signal to recover from system ran away trouble.

Input Port ($R_1 \sim R_4$)

$R_1 \sim R_4$ is 4-bits input port, which status is fetched by the input instruction. The port is pulled down to the low level (V_{SS}) by resistor, so it can be used as keyboard input port.

Input ($R_5 \sim R_8$)

$R_5 \sim R_8$ is 4-bits input port, which status is fetched by the input instruction. The port is pulled down to the low level (V_{SS}) by resistor, so it can be used as keyboard input port.

Input Port ($I_1 \sim I_4$)

$I_1 \sim I_4$ is 4-bits input port, which status can be fetched by the input instruction. It is pulled down to low level (V_{SS}) by register via transistor only when it is desired to fetch the port status or input signal is low level.

As input current is restricted, it can be used being fixed at high level (V_{DD}).

$\overline{HS}$ Input Pin

It is one bit input pin, which status can be fetched by the input instruction. It is pulled up to high level (V_{DD}) by resistor. It is used as a hook switch input pin.

Output Port ($C_1 \sim C_4$)

$C_1 \sim C_4$ is 4-bits output port. The contents of the output latch can be rewritten by the output instruction. The low level is output at each output pin after the system is reset. When the HS input pin is open or at high level, the low level is output to each output pin irrespective of the contents of the output latch.

The outputs of $C_1 \sim C_4$ are all CMOS output.

Output Port ($O_1 \sim O_4$)

$O_1 \sim O_4$ is 4-bit output port. The contents of the output latch can be rewritten by the output instruction.

Output latch of O_1 and O_2 are reset and O_3 and O_4 are set at system reset.

Each output from $O_1 \sim O_4$ port is C-MOS.

Output Port $EO_1 \sim EO_4$)

$EO_1 \sim EO_4$ is 4-bits output port. The contents of the output latch can be rewritten by the output instruction. The level is output at each output pin after the system is reset. Each output of $EO_1 \sim EO_4$ is CMOS output.

Input/Output Port ($IO_1 \sim IO_4$)

$IO_1 \sim IO_4$ is 4-bits input/output port. Fetching of the port status and rewriting of the output latch contents can be done by the input/output instruction.

Each Output of $IO_1 \sim IO_4$ is CMOS at output mode.

IOE Output Pin

It is one bit output pin. A load signal is output at this pin when the output latch's ($IO_1 \sim IO_4$) contents are rewritten.

DTMF Output Pin

It is output pin to output DTMF signals. Start and stop of the DTMF output are done by the output instruction.

DP Output Pin

It is an output pin for dial pulse output. Start and top of the dial pulse output can be done by the output instruction.

The DP OUT pin output is C-MOS output.

BD Output Pin

It is output pin for the buzzer output. The buzzer output can be started and stopped by the output instruction. Output of BD port is CMOS output.

32 kHz Output Pin

It is an output pin to output 31.960 kHz clock (duty: 50%) which is obtained by dividing the 3.579545 MHz system clock by 112. This clock keep outputting as long as system clock oscilation is executed. Output of 32 kHz pin is CMOS out-out.

XT, $\overline{XT}$ Pins

These are input and output pins of the oscillator inverter, and the oscillator circuit is provided with the built in feed back resistor. By connecting to them oscillation of system clock status. 3.579545 MHz ceramic resonator and capacitors.

ELECTRIC CHARACTERISTICS

• Absolute Maximum Ratings

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	Ta = 25°C	−30 ~ 7	V
Input Voltage	V_I	Ta = 25°C	−0.3 ~ V_{DD} + 0.3	V
Output Voltage	V_O	Ta = 25°C	−0.3 ~ V_{DD} + 0.3	V
Power Dissipation	P_D	Ta = 25°C	200 max.	mW
Storage Temperature	Tstg	—	−55 ~ +125	°C

• Operating Conditions

Parameter	Symbol	Conditions	Limits	Unit
Operating Voltage	V_{DD}	Pulse Mode f_{OSC} = 3.58 MHz	2.0 ~ 5.5	V
Memory Retension Voltage	V_{DDM}	—	1.2 ~ 5.5	V
Operating Temperature	Topr	—	−20 ~ +75	°C

Note: Operating conditions for tone mode is V_{DD} = 2.2 ~ 5.5V.

• DC Characteristics

(Ta = −20 ~ +75°C)

Parameter	Symbol	Conditions		Supply Voltage	Min.	Typ.	Max.	Unit	Survey Circuit
"H" Output Current (1)	I_{OH_1}	O_3, O_4	V_{OH} = 2.6V	3.0 V	−0.2	–	–	mA	1
"L" Output Current (1)	I_{OL_1}	DP OUT	V_{OL} = 0.4V	3.0V	0.5	–	–	mA	
"H" Output Current (2)	I_{OH_2}	$C_1 \sim C_4$	V_{OH} = 2.6V	3.0V	−1.0	–	–	mA	
"L" Output Current (2)	I_{OL_2}		V_{OL} = 0.4V	3.0V	10	–	–	μA	
"H" Output Current (3)	I_{OH_3}	O_1, O_2	V_{OH} = 2.6V	3.0V	−2.0	–	–	μA	
"L" Output Current (3)	I_{OL_3}	BD	V_{OL} = 0.4V	3.0V	10	–	–	μA	
"H" Output Current (4)	I_{OH_4}	$IO_1 \sim IO_4$ IOE $EO_1 \sim EO_4$	V_{OH} = 2.6V	3.0V	−150	–	–	μA	
"L" Output Current (4)	I_{OL_4}		V_{OL} = 0.4V	3.0V	300	–	–	μA	
"H" Output Current (5)	I_{OH_5}	32kHz	V_{OH} = 2.6V	3.0V	−40	–	–	μA	
"L" Output Current (5)	I_{OL_5}		V_{OL} = 0.4V	3.0V	25	–	–	μA	
"H" Input Voltage	V_{IH}	–		3.0V	2.2	–	–	V	2
				5.5V	40	–	–		
"L" Input Voltage	V_{IL}	–		3.0V	–	–	0.8	V	
				5.5V	–	–	1.4		
"H" Input Current (1)	I_{IH_1}	HS	V_{IH} = 5.5V	5.5V	–	–	2	μA	3
"L" Input Current (1)	I_{IL_1}		V_{IL} = 0V	3.0V	−2.0	–	−180	μA	
				5.5V	−40	–	−360		
"H" Input Current (2)	I_{IH_2}	$R_1 \sim R_3$	V_{IH} = 5.5V	5.5V	20	–	180	μA	
			V_{IH} = 3.0V	3.0V	10	–	90		
"L" Input Current (2)	I_{IL_2}		V_{IL} = 0V	5.5V	–	–	−2	μA	
"H" Input Current (3)	I_{IH_3}	$I_1 \sim I_4$ AC, TEST	V_{IH} = 5.5V	5.5V	60	–	600	μA	
			V_{IH} = 3.0V	3.0V	30	–	300		
"L" Input Current (3)	I_{IL_3}		V_{IL} = 0V	5.5V	–	–	−2	μA	
"H" Input Current (4)	I_{IH_4}	$IO_1 \sim IO_4$	V_{IH} = 5.5V	5.5V	–	–	2	μA	
"L" Input Current (4)	I_{IL_4}		V_{IL} = 0V	5.5V	–	–	−2	μA	
Current Consumption (1)	I_{DDP}	Tone output off With no load		2.5V	–	0.25	0.5	mA	4
				5.0V	–	1.5	2.4		
Current Consumption (2)	I_{DDT}	Tone output on With on load		2.5V	–	1.3	2.4	mA	
				5.0V	–	4.2	6.8		
Current Consumption (3)	I_{DDM}	ON HOOK, Ta = 25°C With no load		2.5V	–	–	0.2	μA	

6

• AC Characteristics

(Ta = –20 ~ +75°C)

Parameter	Symbol	Conditions	Supply Voltage	Min.	Typ.	Max.	Unit	Survey Circuit
Cycle Time	t_{CY}	f = 3.579545MHz	3.0V	–	17.9	–	μs	
Tone Output	V_{OUT}	Row side only R_L = 1kΩ	2.2V	–	180	–	mV rms	
			4.0V	–	260	–		
			5.5V	–	330	–		
High/Low Level Ratio	dB_{CR}	–	3.0V	1	2	3	dB	
			5.5V	1	2	3		
Distorsion Ratio	$\%d_{IS}$	R_L = 1kΩ	3.0V	–	–	5	%	
			5.5V	–	–	5		
Switch Input Time	t_{KIN}		–	3.3	–	–	ms	5
Rise/Fall Time (1)	t_{TLH_1}	Q_3, Q_4, DP OUT C_L = 50pF	3.0V	–	–	0.5	μs	
	t_{THL_1}		3.0V	–	–	0.5		
Rise/Fall Time (2)	t_{TLH_2}	$C_1 \sim C_4$ C_L = 50pF	3.0V	–	–	0.5	μs	
	t_{THL_2}		3.0V	–	–	10		
Rise/Fall Time (3)	t_{TLH_3}	Q_1, Q_2, BD, 32kHz C_L = 50pF	3.0V	–	–	5	μs	
	t_{THL_3}		3.0V	–	–	10		
Rise/Fall Time (4)	t_{TLH_4}	$IO_1 \sim IO_4$, IOE $EO_1 \sim EO_4$ C_L = 50pF	3.0V	–	–	1	μs	
	t_{THL_4}		3.0V	–	–	1		

• DTMF Tone Output Freqeuncy

	Standard Frequency (Hz)	Output Freqeuncy (Hz)	Deviation (%)
R1	697	699.1	+0.30
R2	770	766.2	–0.49
R3	852	847.4	–0.54
R4	941	948.0	+0.74
C1	1209	1215.9	+0.57
C2	1336	1331.7	–0.32
C3	1477	1471.9	–0.35

f_{OSC} = 3.579545MHz

	Mnemonic	Instruction Code 13 12	11 10 9 8	7 6 5 4	3 2 1 0		ACC	Zero	Carry	Greater	Description
Arithmetic operation	ADD ACC, AP	0 0	0 0 0 P	0 1 0 0	A	$A=0_H\sim F_H$, P=0 or 1	*	*	*	–	AP ← (AP) + ACC
	ADD #D, AP	0 1	1 0 0 P	D	A	$D=0_H\sim F_H$ $A=0_H\sim F_H$, P=0 or 1	*	*	*	–	AP ← (AP) + D
	ADC AP	0 0	0 0 0 P	0 1 0 1	A	$A=0_H\sim F_H$, P=0 or 1	*	*	*	–	AP ← (AP) + ACC + C
	SUB ACC, AP	0 0	0 0 1 P	0 1 0 0	A	$A=0_H\sim F_H$, P=0 or 1	*	*	*	–	AP ← (AP)– ACC
	SUB #D, AP	0 1	1 0 1 P	D	A	$D=0_H\sim F_H$, $A=0_H\sim F_H$, P=0 or 1	*	*	*	–	AP ← (AP) – D
	SBC AP	0 0	0 0 1 P	0 1 0 1	A	$A=0_H\sim F_H$, P=0 or 1	*	*	*	–	AP ← (AP) – ACC – C
	CMP ACC, AP	0 0	0 0 1 P	1 1 1 0	A	$A=0_H\sim F_H$, P=0 or 1	–	*	–	*	(AP) – ACC
	CMP #D, AP	0 1	0 1 1 P	D	A	$D=0_H\sim F_H$, $A=0_H\sim F_H$, P=0 or 1	–	*	–	*	(AP) – D
	XOR ACC, AP	0 0	0 0 0 P	0 1 1 1	A	$A=0_H\sim F_H$, P=0 or 1	*	*	–	–	AP ← (AP) ∀ ACC
	XOR #D, AP	0 1	1 1 1 P	D	A	$D=0_H\sim F_H$, $A=0_H\sim F_H$, P=0 or 1	*	*	–	–	AP ← (AP) ∀ D
Bit operation	BIT ACC, AP	0 0	0 0 0 P	1 1 1 0	A	$A=0_H\sim F_H$, P=0 or 1	–	*	–	–	(AP) ∨ $\overline{ACC}$
	BIT #D, AP	0 1	0 1 0 P	D	A	$D=0_H\sim F_H$, $A=0_H\sim F_H$, p=0 or 1	–	*	–	–	(AP) ∨ $\overline{D}$
	BIS ACC, AP	0 0	0 0 0 P	1 1 1 0	A	$A=0_H\sim F_H$, P=0 or 1	*	*	–	–	AP ← (AP) ∨ ACC
	BIS #D, AP	0 1	0 0 0 P	D	A	$D=0_H\sim F_H$, $A=0_H\sim F_H$, P=0 or 1	*	*	–	–	AP ← (AP) ∨ D
	BIC ACC; AP	0 0	0 0 1 P	0 1 1 0	A	$A=0_H\sim F_H$, P=0 or 1	*	*	–	–	AP ← (AP) ∧ $\overline{ACC}$
	BIC #D, AP	0 1	0 0 1 P	D	A	$D=0_H\sim F_H$, $A=0_H\sim F_H$, P=0 or 1	*	*	–	–	AP ← (AP) ∧ $\overline{D}$
Rotate	ROR AP	0 0	0 0 0 P	0 0 1 0	A	$A=0_H\sim F_H$, P=0 or 1	*	*	*	–	↳ (AP) → C ⌋
	ROL AP	0 0	0 0 1 P	0 0 1 0	A	$A=0_H\sim F_H$, P=0 or 1	*	*	*	–	⌊ (AP) ← C ↲
	ASR AP	0 0	0 0 0 P	0 0 1 1	A	$A=0_H\sim F_H$, P=0 or 1	*	*	*	–	0 → (AP) → C
	ASL AP	0 0	0 0 1 P	0 0 1 1	A	$A=0_H\sim F_H$, P=0 or 1	*	*	*	–	C ← (AP) ← 0
Flag operation	SEZ	0 0	0 0 1 0	1 0 1 0	0 0 0 0		–	1	–	–	Z ← 1
	CLZ	0 0	0 0 0 0	1 0 1 0	0 0 0 0		–	0	–	–	Z ← 0
	SEC	0 0	0 0 1 0	1 0 0 1	0 0 0 0		–	–	1	–	C ← 1
	CLC	0 0	0 0 0 0	1 0 0 1	0 0 0 0		–	–	0	–	C ← 0
	SEG	0 0	0 0 1 0	1 0 0 0	0 0 0 0		–	–	–	1	G ← 1
	CLG	0 0	0 0 0 0	1 0 0 0	0 0 0 0		–	–	–	0	G ← 0
	SEA	0 0	0 0 1 0	1 0 1 1	0 0 0 0		–	1	1	1	Z ← 1, C ← 1, G ← 1
	CLA	0 0	0 0 0 0	1 0 1 1	0 0 0 0		–	0	0	0	Z ← 0, C ← 0, G ← 0
Data transfer	MOV ACC. AP	1 1	1 1 0 1	0 0 0 0	A	$A=0_H\sim F_H$	–	–	–	–	AP ← ACC
	MOV ACC, AX	1 1	1 1 0 0	X	A	$X=0_H\sim F_H$ $A=0_H\sim F_H$	–	–	–	–	AX ← ACC
	MOV #D, AP	0 1	1 1 0 P	D	A	$D=0_H\sim F_H$ $A=0_H\sim F_H$, P=0 or 1	*	*	–	–	AP ← D
	MOV AP, ACC	1 1	1 1 1 1	0 0 0 0	A	$A=0_H\sim F_H$	*	*	–	–	ACC ← (AP)
	MOV AX, ACC	1 1	1 1 1 0	X	A	$X=0_H\sim F_H$ $A=0_H\sim F_H$	*	*	–	–	ACC ← (AX)
	CHG AP	1 1	1 0 0 1	0 0 0 0	A	$A=0_H\sim F_H$	*	–	–	–	(AP) ↔ ACC
	CHG AX	1 1	1 0 0 0	X	A	$X=0_H\sim F_H$ $A=0_H\sim F_H$	*	–	–	–	(AX) ↔ ACC

	Mnemonic	Instruction Code 13 12	11 10 9 8	7 6 5 4	3 2 1 0		ACC	Zero	Carry	Greater	Description
Data transfer	RDAR	1 1	0 0 0 0	0 0 0 0	0 0 0 0		*	*	–	–	ACC ← (AR_1)
	RDAR +(–)	1 1	0 0 0 0	0 0 1 D/I	0 0 0 0	D/I = 0 or 1	*	*	–	–	ACC ← (AR_1), AR_1 ← AR_1 ±1
	RDAR +(–), Z	1 1	0 0 0 0	0 1 0 D/I	0 0 0 0	D/I = 0 or 1	0	1	–	–	ACC ← (AR_1) if (AR_1)=0 then PC ← PC+1 else AR_1 ← AR_1 ±1, repeat
	RDAR +(–), N	1 1	0 0 0 0	1 0 0 D/I	0 0 0 0	D/I = 0 or 1	*	0	–	–	ACC ← (AR_1) if (AR_1)=0 then PC ← PC+1 else AR_1 ← AR_1 ±1, repeat
	RDAR +(–),Z,L	1 1	0 0 1 0	0 1 0 D/I	0 0 0 0	D/I = 0 or 1	*	*	–	–	ACC ← (AR_1), L ← L–1 if (AR_1)=0 or L=0 then PC ← PC+1 else AR_1 ← AR_1 ±1, repeat
	RDAR +(–),N,L	1 1	0 0 1 0	1 0 0 D/I	0 0 0 0	D/I = 0 or 1	*	*	–	–	ACC ← (AR_1), L ← L–1 if (AR_1)=0 or L=0 then PC ← PC+1 else AR_1 ← AR_1 ±1, repeat
	MVAR	1 1	0 1 0 0	0 0 0 0	0 0 0 0		–	*	–	–	(AR_2) ← (AR_1)
	MVAR +(–)	1 1	0 1 0 0	0 0 1 D/I	0 0 0 0	D/I = 0 or 1	–	*	–	–	(AR_2) ← (AR_1), AR_1←AR_1 ±1, AR_2←AR_2 ±1
	MVAR +(–), Z	1 1	0 1 0 0	0 1 0 D/I	0 0 0 0	D/I = 0 or 1	–	1	–	–	(AR_2) ← (AR_1) if (AR_1)=0 then PC ← PC+1 else AR_1←AR_1 ±1, AR_2←AR_2 ±1, repeat
	MVAR +(–), N	1 1	0 1 0 0	1 0 0 D/I	0 0 0 0	D/I = 0 or 1	–	0	–	–	(AR_2) ← (AR_1) if (AR_1)=0 then PC ← PC+1 else AR_1←AR_1 ±1, AR_2←AR_2 ±1, repeat
	MVAR +(–), L	1 1	0 1 1 0	0 0 0 D/I	0 0 0 0	D/I = 0 or 1	–	*	–	–	(AR_2) ← (AR_1), L ← L–1 if L=0 then PC ← PC+1 else AR_1←AR_1 ±1, AR_2←AR_2 ±1, repeat
	MVAR +(–),Z,L	1 1	0 1 1 0	0 1 0 D/I	0 0 0 0	D/I = 0 or 1	–	*	–	–	(AR_2) ← (AR_1), L ← L–1 if (AR_1)=0 or L=0 then PC ← PC+1 else AR_1←AR_1 ±1, AR_2←AR_2 ±1, repeat
	MVAR +(–),N,L	1 1	0 1 1 0	1 0 0 D/I	0 0 0 0	D/I = 0 or 1	–	*	–	–	(AR_2) ← (AR_1), L ← L–1 if (AR_1)=0 or L=0 then PC ← PC+1 else AR_1←AR_1 ±1, AR_2←AR_2 ±1, repeat
Subroutine	CALL adrs	1 0	1 a_{10} a_9 a_8	a_7 a_6 a_5 a_4	a_3 a_2 a_1 a_0	a_{10}~a_0=000H~7FFH	–	–	–	–	STACK ← (PC), PC ← adrs
	RET	0 0	0 0 0 0	1 1 0 0	0 0 0 0		–	–	–	–	PC ← (STACK) + 1
	RTI	0 0	0 0 1 0	1 1 0 0	0 0 0 0		–	–	–	–	PC ← (STACK) or PC ← (STACK) + 1
Jump	JMP adrs	1 0	0 a_{10} a_9 a_8	a_7 a_6 a_5 a_4	a_3 a_2 a_1 a_0	a_{10}~a_0=000H~7FFH	–	–	–	–	PC ← adrs
	JMP @ AP	0 0	0 0 0 P	1 1 0 1	A	A=0H~FH, P=0 or 1	–	–	–	–	PC ← (PC) + (AP) + 1
	JMPIO @ AP	0 0	0 0 1 P	1 1 0 1	A	A=0H~FH, P=0 or 1	–	–	–	–	PC ← (PC) + [(AP) A 7] + 1
Branch	BEQ n (BZE n)	1 1	1 0 1 P	0 1 0 n_4	n_3 n_2 n_1 n_0	n_4~n_0=00H~1FH P=0 or 1	–	–	–	–	if Z=1 then PC←PC–n or PC←PC+n+1 else PC←PC+1
	BNE n (BNZ n)	1 1	1 0 1 P	1 1 0 n_4	n_3 n_2 n_1 n_0	n_4~n_0=00H~1FH P=0 or 1	–	–	–	–	if Z=0 then PC←PC–n or PC←PC+n+1 else PC←PC+1
	BCS n	1 1	1 0 1 P	0 0 0 n_4	n_3 n_2 n_1 n_0	n_4~n_0=00H~1FH P=0 or 1	–	–	–	–	if C=1 then PC←PC–n or PC←PC+n+1 else PC←PC+1
	BCC n	1 1	1 0 1 P	1 0 0 n_4	n_3 n_2 n_1 n_0	n_4~n_0=00H~1FH P=0 or 1	–	–	–	–	if C=0 then PC←PC–n or PC←PC+n+1 else PC←PC+1
	BGT n	1 1	1 0 1 P	0 0 1 n_4	n_3 n_2 n_1 n_0	n_4~n_0=00H~1FH P=0 or 1	–	–	–	–	if G=1 then PC←PC–n or PC←PC+n+1 else PC←PC+1
	BLE n	1 1	1 0 1 P	1 0 1 n_4	n_3 n_2 n_1 n_0	n_4~n_0=00H~1FH P=0 or 1	–	–	–	–	if G=0 then pC←PC–n or PC←PC+n+1 else PC←PC+1
	BGE n	1 1	1 0 1 P	0 1 1 n_4	n_3 n_2 n_1 n_0	n_4~n_0=00H~1FH P=0 or 1	–	–	–	–	if Z=1 or G=1 then PC←PC–n or PC+n+1 else PC←PC+1
	BLT n	1 1	1 0 1 P	1 1 1 n_4	n_3 n_2 n_1 n_0	n_4~n_0=00H~1FH P=0 or 1	–	–	–	–	if Z=0 then PC←PC–n or PC←PC+n+1 and G=0 else PC←PC+1
Input/Output	IN PORT, AP	0 0	0 1 0 P	P_L	A	P_L=0H~FH A=0H~FH, P=0 or 1	*	*	–	–	AP ← (PORT)
	OUT AP, PORT	0 0	1 0 P_H P	P_L	A	P_L=0H~FH, P_H=0 or 1 A=0H~FH, P=0 or 1	–	–	–	–	PORT ← (AP)
	OUT #D, PORT	0 0	1 1 P_H 0	P_L	D	P_L=0H~FH, P_H=0 or 1 D=0H~FH	–	–	–	–	PORT ← D

	Mnemonic	Instruction Code														ACC	Zero	Carry	Greater	Description
		13	12	11	10	9	8	7	6	5	4	3	2	1	0					
CPU Control & Others	STOP	0	0	1	1	1	0	0	0	0	0	0	0	0	0	–	–	–	–	Stop system clock
	HALT	0	0	1	1	1	0	0	0	0	1	0	0	0	0	–	–	–	–	Halt CPU
	ACT	0	0	1	1	1	0	0	0	1	0	0	0	0	0	–	–	–	–	Activate CPU
	EI	0	0	1	1	1	0	0	1	1	0	1	0	0	0	–	–	–	–	Enable timer interrupt
	DI	0	0	1	1	1	0	0	1	1	0	0	1	0	0	–	–	–	–	Disable timer interrupt
	ET	0	0	1	1	1	0	0	1	1	0	0	0	1	0	–	–	–	–	Enable timer activate
	DT	0	0	1	1	1	0	0	1	1	0	0	0	0	1	–	–	–	–	Disable timer activate
	EC	0	0	1	1	1	0	0	1	1	1	1	0	0	0	–	–	–	–	Enable output port ($C_1 \sim C_4$)
	DC	0	0	1	1	1	0	0	1	1	1	0	1	0	0	–	–	–	–	Disable output port ($C_1 \sim C_4$)
	OM	0	0	1	1	1	0	0	1	1	1	0	0	1	0	–	–	–	–	Set I/O port ($IO_1 \sim IO_4$) to output mode
	IM	0	0	1	1	1	0	0	1	1	1	0	0	0	1	–	–	–	–	Set I/O port ($IO_1 \sim IO_4$) to input mode
	RST	0	0	1	1	1	0	1	0	0	1	0	0	0	0	–	–	–	–	Reset divider
	SELINT	0	0	1	1	1	0	1	1	0	0	1	0	0	0	–	–	–	–	Select interrupt mode
	SELHR	0	0	1	1	1	0	1	1	0	0	0	1	0	0	–	–	–	–	Select timer activation mode
	EHLH	0	0	1	1	1	0	1	1	0	0	0	0	1	0	–	–	–	–	Enable hook switch low to high transmission
	DHLH	0	0	1	1	1	0	1	1	0	0	0	0	0	1	–	–	–	–	Disable hook switch low to high transmission
	SELN	0	0	1	1	1	0	1	0	1	1	0	0	1	0	–	–	–	–	Select negative phase
	SELP	0	0	1	1	1	0	1	0	1	1	0	0	0	1	–	–	–	–	Select positive phase
	EO	0	0	1	1	1	0	1	0	1	1	1	0	0	0	–	–	–	–	Enable on-hook dial
	DO	0	0	1	1	1	0	1	0	1	1	0	1	0	0	–	–	–	–	Disable on-hook dial
	NOP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	–	–	–	–	No operation

OLMS-64 SERIES

6

OKI semiconductor

MSM6404

HIGH-SPEED 4-BIT SINGLE CHIP MICROCONTROLLER

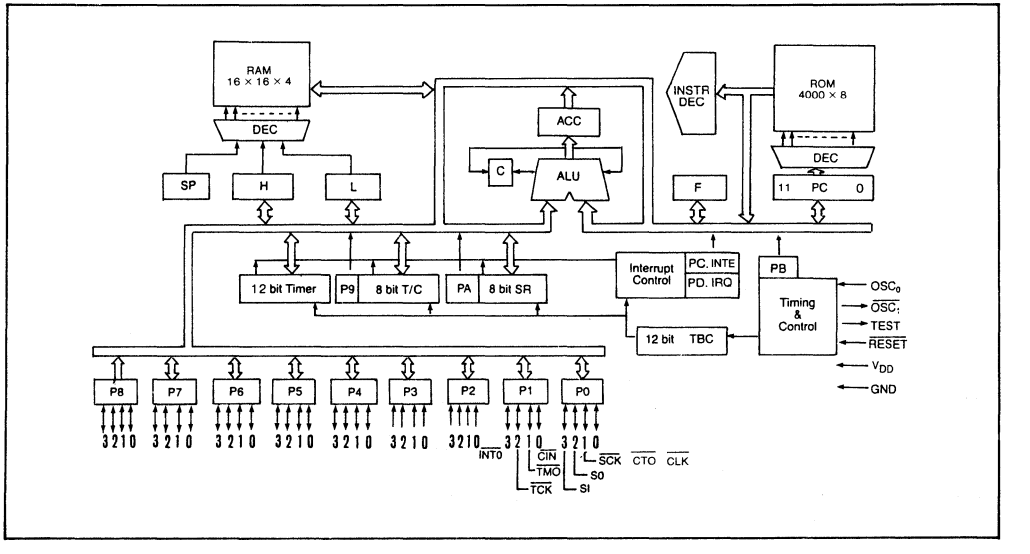
GENERAL DESCRIPTION

The OKI MSM6404 microcontroller is a low power, high-performance single-chip device implemented in complementary metal oxide semiconductor technology. 32K bits of mask program ROM, 1024 bits of data RAM, 36 Input/Output lines, a programmable timer/event-counter, and oscillator are integrated onto one chip. Program memory is byte wide and data-paths are organized in 4 bit nibbles. RAM and I/O lines are bit addressable. 122 instructions include binary, BCD operations; bit set, reset, test; 8 bit I/O; relative jumps; multifunctional instructional (increment, modify, skip) 8 bit wide table output; subroutine call and return.

FEATURES

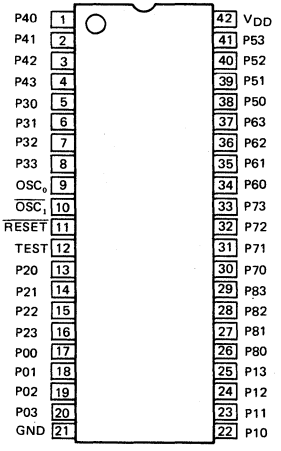
- 4000 × 8 MASK ROM
  An evaluation board is available for up to 8k × 8.
- 256 × 4 RAM (including the stack area)
- 9 × 4 Ports, 36 I/O lines
  4 lines for input ports having a latch, and the other 32 lines for bit operation are available.
- Three built-in counters
  12-bit time-base counter
  12-bit programmable timer
  8-bit high-speed programmable timer/event counter
- Built-in 8-bit serial I/O register (with 3-bit counter)
- Five interrupts with five priority levels (4 internal, 1 external)
- 32 stack levels (in RAM)
- LED direct drive available (8mA x 5 ports at the same time)
- Power down features
- Instruction execution time
  952 ns 4.2 MHz clock
- Instruction systems suitable for control
- 122 instructions
- Mask option
  P60-63 for input port
- Full static operation
- Low power consumption
  TYP 0.4μW at V_{DD}=2V
  TYP 5μW at V_{DD}=5V 0Hz clock
- 5V single power supply, 42-pin DIP or 44-pin FLAT

BLOCK DIAGRAM

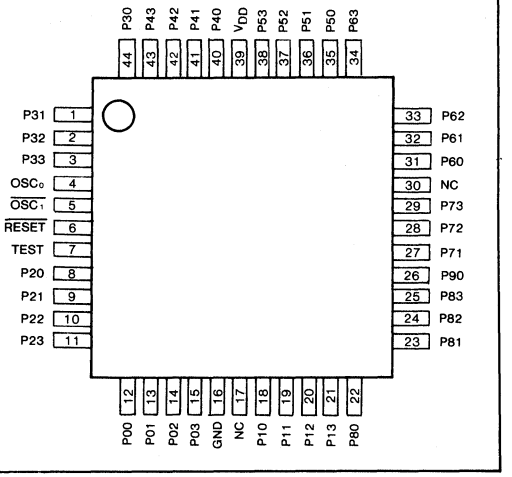


PIN CONFIGURATION

(Top View) 42-pin Plastic DIP



(Top View) 44-pin Plastic Flat Package



PIN DESCRIPTION

Pin name	Input/output	Function		When reset
P00 P01 / $\overline{SCK}$ P02 / SO P03 / SI	Input/output	4-bit input/output port. P01 to P03 are also used as serial interface terminals.		"1"
P10 / $\overline{CIN}$ P11 / $\overline{TMO}$ P12 / $\overline{TCK}$ P13	Input/output	4-bit input port with latch. Built-in pull up register for all bit input.		"1"
P20 / $\overline{INT}$ P21 P22 P23	Input	4-bit input port with a latch. P20 is shared with INT input. (Fall trigger input) P21 ~ 23 are level input. Built-in pull up register for all bit input.		The latch is reset.
P30–33	Input/output	4-bit input/output port		"1"
P40–43	Input/output	4-bit input/output port	8-bit output port	"0"
P50–53	Input/output	4-bit input/output port		"0"
P60–63	Input/output	4-bit input/output port *1		"0"
P70–73	Input/output	4-bit input/output port		"0"
P80–83	Input/output	4-bit input/output port		"0"
OSC_0 $\overline{OSC_1}$	Input/output	X'tal connection terminal for system clock oscillation		Oscillation wave
TEST	Output	(Test terminal for Maker)		Pulse output
$\overline{RESET}$	Input	System reset input terminal		
VDD GND		Power source voltage supply		

Note: When each port is used for output, it is possible to drive one TTL (one input).
*1 Can be made as a port dedicated to input (mask option).

INSTRUCTION LIST

	Mnemonic		Description	Code	Byte	Cycle
Load, Push, Pop	LAI	n	A ← n	9n	1	1
	LLI	n	L ← n	8n	1	1
	LHLI	nn	HL ← nn	15nn	2	2
	LMI	nn	M(w) ← nn	14nn	2	2
	LAL		A ← L	21	1	1
	LLA		L ← A	2D	1	1
	LAH		A ← H	22	1	1
	LHA		H ← A	2E	1	1
	LAM		A ← M	38	1	1
	LMA		M ← A	2F	1	1
	LAM+		A ← M, L ← L+1, Skip if L=0	24	1	1
	LAM−		A ← M, L ← L−1, Skip if L=F	25	1	1
	LMA+		M ← A, L ← L+1, Skip if L=0	26	1	1
	LMA−		M ← A, L ← L−1, Skip if L=F	27	1	1
	LAMM	n_2	A ← M, H ← H V n_2	39–3B	1	1
	LAMD	mm	A ← Md	10mm	2	2
	LMAD	mm	Md ← A	11mm	2	2
	LMTD	mm	Md(w) ← T (M(w), A), T=ROM table	19mm	2	3
	LMCT		M(w) ← CT	3E59	2	2
	LCTM		CT ← M(w)	3E51	2	2
	LMSR		M(w) ← SR	3E5A	2	2
	LSRM		SR ← M(w)	3E52	2	2
	LTMM		TM ← (M(w), A)	3E50	2	2
	PUSH		ST ← C, A, H, L, SP ← SP−4	1C	1	3
	POP		C, A, H, L ←, ST SP ← SP+4	1D	1	3
Exchange	X		A ⟷ M	28	1	1
	XM	n_2	A ⟷ M, H ← H V n_2	29–2B	1	1
	X+		A ⟷ M, L ← L+1, Skip if L=0	3C	1	1
	X−		A ⟷ M, L ← L−1, Skip if L=F	2C	1	1
Increment/ Decrement	INA		A ← A+1, Skip if A=0	30	1	1
	INM		M ← M+1, Skip if M=0	33	1	1
	INL		L ← L+1, Skip if L=0	31	1	1

6

INSTRUCTION LIST (Continued)

	Mnemonic		Description	Code	Byte	Cycle
Increment/Decrement	INH		H ← H+1, Skip if M = 0	32	1	1
	INMD	mm	Md ← Md+1, Skip if Md = 0	12mm	2	2
	DCA		A ← A−1, Skip if A = F	34	1	1
	DCM		M ← M−1, Skip if M = F	37	1	1
	DCL		L ← L−1, Skip if L = F	35	1	1
	DCH		H ← H−1, Skip if H = F	36	1	1
	DCMD	mm	Md ← Md−1, Skip if Md = F	13mm	2	2
Arithmetic	ADS		A ← A+M, Skip if Cy =1	02	1	1
	ADCS		A, C ← A+M+C, Skip if Cy=1	01	1	1
	ADC		A, C ← A+M+C	03	1	1
	AIS	n	A ← A+n, Skip if Cy =1	3E4n	2	2
	DAA		A ← A+6	06	1	1
	DAS		A ← A+10	0A	1	1
	AND		A ← A ∨ M	0D	1	1
	OR		A ← A ∨ M	05	1	1
	EOR		A ← A ∨ M	04	1	1
	CMA		A ← $\overline{A}$	0B	1	1
	CIA		A ← $\overline{A}$+1	0C	1	1
	RAL		Rotate Left with C	0E	1	1
	RAR		Rotate Right with C	0F	1	1
	TC		Skip if C =1	09	1	1
	SC		C ← 1	07	1	1
	RC		C ← 0	08	1	1
Compare	CAI	n	Skip if A =n	3E0n	2	2
	CLI	n	Skip if L =n	3E2n	2	2
	CPI	p, n	Skip if P_p=n	17pn	2	2
	CMI	n	Skip if M =n	3E1n	2	2
	CAM		Skip if A =M	16	1	1
Bit operation	TAB	n_2	Skip if Abit (n_2) =1	54–57	1	1
	RAB	n_2	Abit (n_2) ← 0	64-67	1	1
	SAB	n_2	Abit (n_2) ← 1	74–77	1	1
	TMB	n_2	Skip if Mbit (n_2) =1	58–5B	1	1
	RMB	n_2	Mbit (n_2) ← 0	68–6B	1	1

INSTRUCTION LIST (Continued)

	Mnemonic		Description	Code	Byte	Cycle
Bit operation	SMB	n	Mbit (n_2) ←1	78–7B	1	1
	TFB	n_2	Skip if Fbit (n_2) =1	5C–5F	1	1
	RFB	n_2	Fbit (n_2) ←0	6C-6F	1	1
	SFB	n_2	Fbit (n_2) ←1	7C–7F	1	1
	TPB	n_2	Skip if Pbit (n_2) =1	50–53	1	1
	RPB	n_2	Pbit (n_2) ←0	60–63	1	1
	SPB	n	Pbit (n_2) ←1	70–73	1	1
	TPBD	p n_2	Skip if $P_p$bit (n_2) =1	3D $p_{0\sim3}$	2	2
	RPBD	p n_2	$P_p$bit (n_2) =0	3D $p_{4\sim7}$	2	2
	SPBD	p n_2	$P_p$bit (n_2) =1	3D $p_{8\sim B}$	2	2
Interrupt	MEI		MEIF ←1	3E60	2	2
	MDI		MEIF ←0	3E61	2	2
	EITB		EITBF ←1	3DC9	2	2
	EITM		EITMF ←1	3DCA	2	2
	EICT		EICTF ←1	3DCB	2	2
	EIEX		EIEXF ←1	3DC8	2	2
	DITB		EITBF ←0	3DC5	2	2
	DITM		EITMF ←0	3DC6	2	2
	DICT		EICTF ←0	3DC7	2	2
	DIEX		EIEXF ←0	3DC4	2	2
	TITB		Skip If EITBF =1	3DC1	2	2
	TITM		Skip If EITMF =1	3DC2	2	2
	TICT		Skip If EICTF =1	3DC3	2	2
	TIEX		Skip If EIEXF =1	3DC0	2	2
	TQEX		Skip If IRQEX =1	3D20	2	2
	TQTB		Skip If IRQTB =1	3DD0	2	2
	TQTM		Skip If IRQTM =1	3DD1	2	2
	TQCT		Skip If IRQCT =1	3DD2	2	2
	TQSR		Skip If IRQSR =1	3DD3	2	2
	RQEX		IRQ EX ←0	3D24	2	2
	RQTB		IRQ TB ←0	3DD4	2	2
	RQTM		IRQ TM ←0	3DD5	2	2
	RQCT		IRQ CT ←0	3DD6	2	2
	RQSR		IRQ SR ←0	3DD7	2	2

INSTRUCTION LIST (Continued)

	Mnemonic		Description	Code	Byte	Cycle
Shift resistor / Counter	ECT		CTF ←1 (start)	3DBB	2	2
	ESR		SRF ←1 (start)	3DBA	2	2
	DCT		CTF ←0 (stop)	3DB7	2	2
	DSR		SRF ←0 (stop)	3DB6	2	2
	TCT		Skip If CTF =1	3DB3	2	2
	TSR		Skip If SRF =1	3DB2	2	2
Branch	JCP	a_6	PC ←a_6	C0~FF	1	1
	JP	a_{12}	PC ←a_{12}	4a_{12}	2	2
	CZP	a	ST ←PC+1, PC ←2a, SP ←SP−4	Ba	1	4
	CAL	a_{12}	ST ←PC+2, PC ←a_{12}, SP ←SP−4	Aa_{12}	2	4
	OPT		P_5, P_4 ←T (M(w), A), T=ROM table	18	1	3
	RT		PC ←ST, SP ←SP+4	IE	1	4
	RTS		PC ←ST, SP ←SP+4, Skip unconditional	IF	1	4
	JA		PC ←(PC ←A)+1	IA	1	1
	JM		PC ←(M(w), A)	IB	1	2
Input/Output	IP		A ←P	20	1	1
	IPD	p	A ←Pp	3DpD	2	2
	OP		P ←A	23	1	1
	OPD	p	Pp ←A	3DpC	2	2
CPU control	NOP		No Operation	00	1	1
	HALT		Halt CPU	3DB8	2	2
	STOP		Stop Clock	3DB9	2	2

ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	Ta = 25°C	−0.3 to 7	V
Input Voltage	V_I		−0.3 to V_{DD}	V
Output Voltage	V_O		−0.3 to V_{DD}	V
Power Dissipation	P_D	Ta = 25°C per package	200 max.	mW
		Ta = 25°C per output	50 max.	mW
Storage Temperature	Tstg	–	−55 to +150	°C

OPERATING RANGE

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	$f_{(OSC)} \leqq 1$ MHz	3 to 6	V
		$f_{(OSC)} \leqq 4.2$ MHz	4.5 to 5.5	V
Data-Hold Voltage	V_{DDH}	f (OSC) = 0 Hz	2 to 6	V
Operating Temperature	T_{OP}	–	−40 to +85	°C
Fan Out	N	MOS Load	15	–
		TTL Load	1	

DC CHARACTERISTICS

(V_{DD} = 5V±10%, Ta = −40 to +85°C)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
"H" Input Voltage*1,*2	V_{IH}	–	2.4	–	V_{DD}	V
"H" Input Voltage*3,*4	V_{IH}	–	3.6	–	V_{DD}	V
"L" Input Voltage	V_{IL}	–	−0.3	–	0.8	V
"H" Output Voltage*1,*5	V_{OH}	I_O = −15μA	4.2	–	–	V
"L" Output Voltage*1	V_{OL}	I_O = 1.6mA	–	–	0.4	V
"L" Output Voltage*5	V_{OL}	I_O = 15μA	–	–	0.4	V
"L" Output Voltage*6	V_{OL}	I_O = 8mA	–	1	2	V
Input Current*3	I_{IH}/I_{IL}	$V_I = V_{DD}/0V$	–	–	15/−15	μA
Input Current*2,*4	I_{IH}/I_{IL}	$V_I = V_{DD}/0V$	–	–	1/−30	μA
"H" Output Current*1	I_{OH}	V_O = 2.4V	−0.1	–	–	mA
"H" Output Current*1	I_{OH}	V_O = 0.4V	–	–	−1.2	mA
Input Capacity	C_I	f=1MHz Ta=25°C	–	5	–	pF
Output Capacity	C_O		–	7	–	
Current Dissipation (when stop condition)	I_{DDS}	V_{DD}=2V, no load Ta=25°C	–	0.2	5	μA
		No load	–	1	100	μA
Current Dissipation	I_{DD}	Quarts oscillation f=4 MHz, no load	–	6	12	mA

*1 Applied to P0, P1, P3, P4, P5, P6, P7 and P8

*2 Applied to P2 *3 Applied to OSC_0 *4 Applied to $\overline{RESET}$ *5 Applied to $\overline{OSC_1}$

*6 In using LED, total output current should be within the limit of Power dissipitation in "Absolute Maximum Rating."

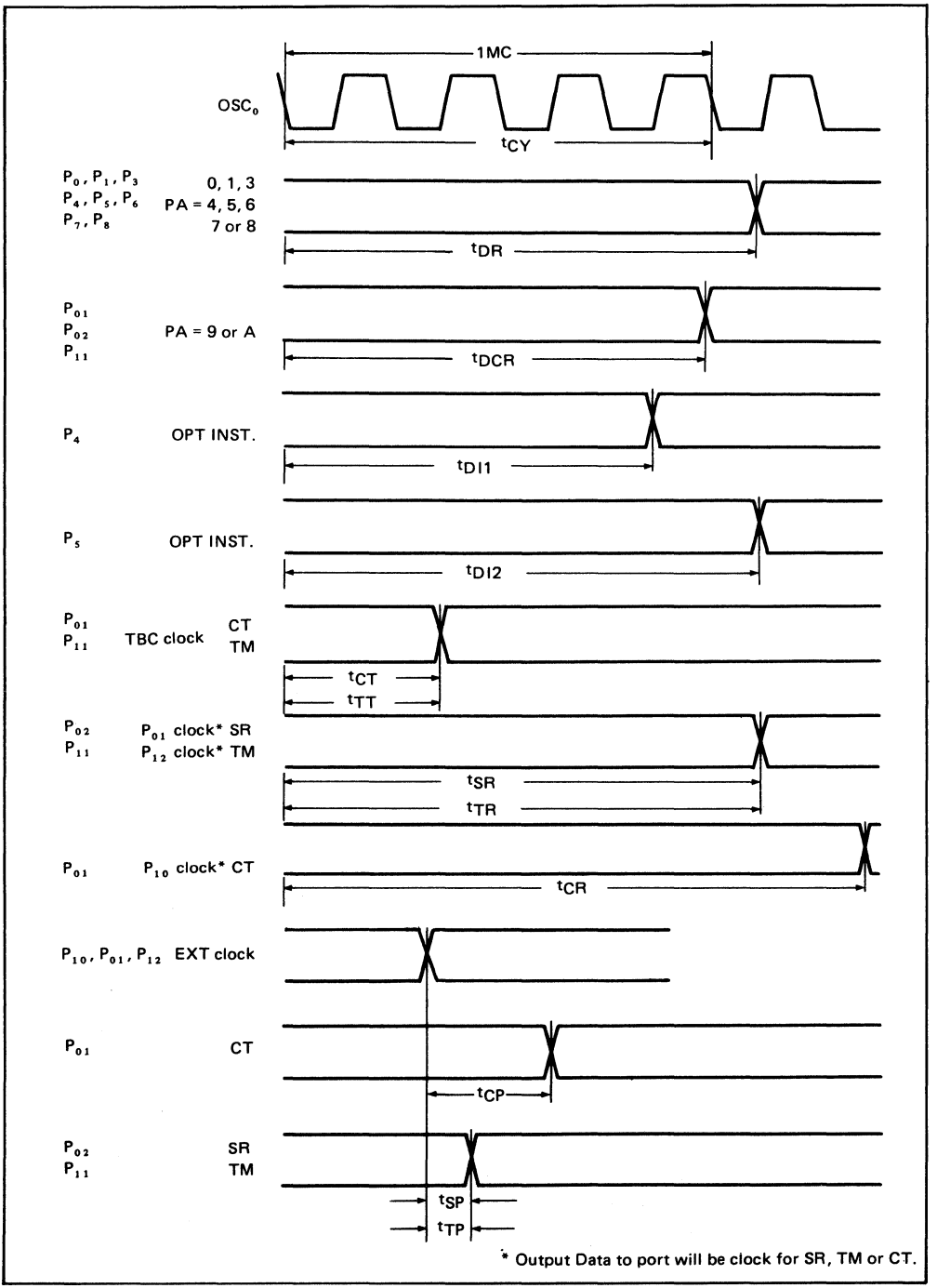
AC CHARACTERISTICS

(V_{DD} = 5V±10%, Ta = −40° to +85°C)

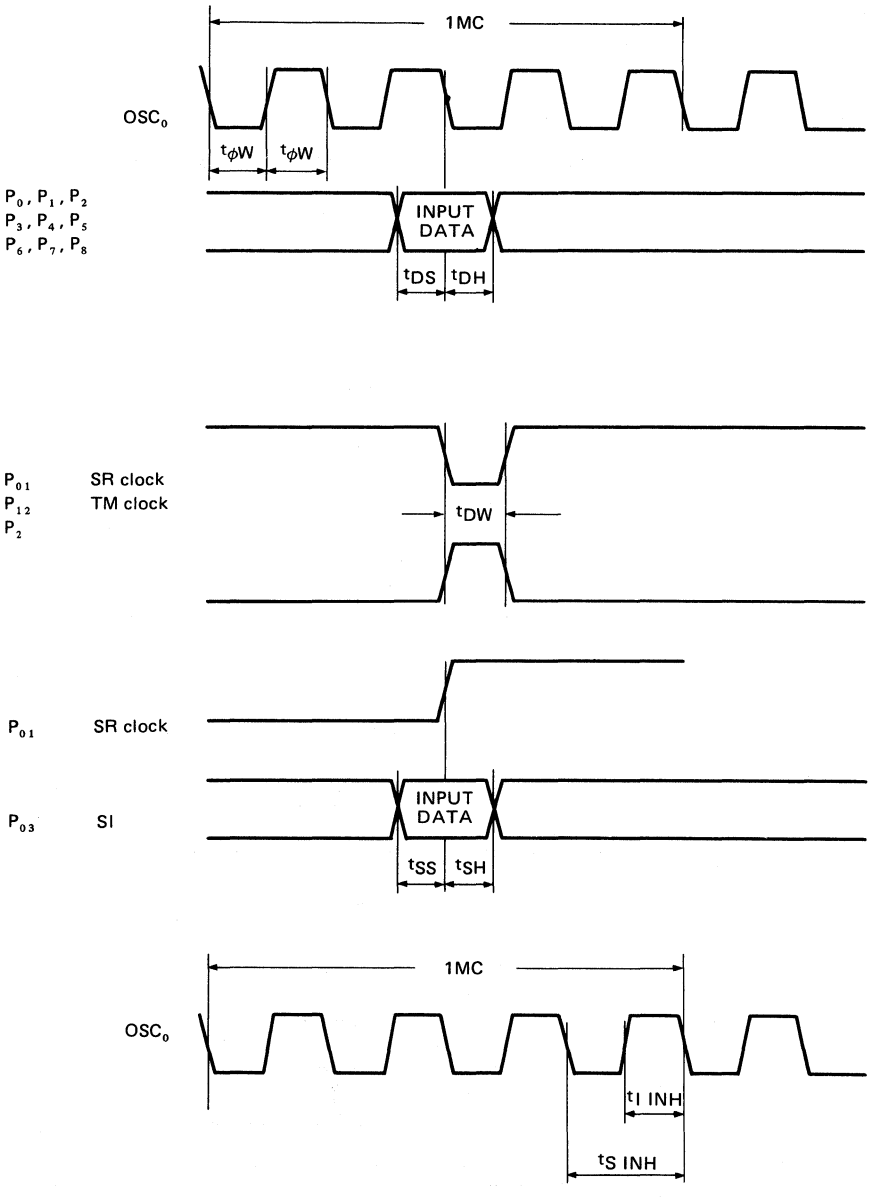
Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Clock Pulse Width Clock (OSC)	$t_{\phi W}$	–	119	–	–	nS
Cycle Time	t_{CY}	–	952	–	–	nS
Input Data Setup Time	t_{DS}	–	120	–	–	nS
Input Data Hold Time	t_{DH}	–	120	–	–	nS
Input Data, Input Clock Pulse Width	t_{DW}	–	120	–	–	nS
SR Data Setup Time	t_{SS}	–	120	–	–	nS
SR Data Hold Time	t_{SH}	–	120	–	–	nS
Data Delay Time	t_{DR}	C_L=15pF	–	–	t_{CY}+300	nS
Data Delay Time at Mode Switching	t_{DCR}	C_L=15pF	–	–	7/8 t_{CY}+300	nS
Data Delay Time at OPT Instruction	t_{DI1}	C_L=15pF	–	–	6/8 t_{CY}+300	nS
Data Delay Time at OPT Instruction	t_{DI2}	C_L=15pF	–	–	7/8 t_{CY}+300	nS
CT/TM Data Delay Time using TBC Clock	t_{CT}/t_{TT}	C_L=15pF	–	–	2/8 t_{CY}+360	nS
SR/TM Data Delay Time using PR Clock	t_{SR}/t_{TR}	C_L=15pF	–	–	t_{CY}+480	nS
CT Data Delay Time using PR Clock	t_{CR}	C_L=15pF	–	–	10/8 t_{CY}+480	nS
CT Data Delay Time using External Clock	t_{CP}	C_L=15pF	–	–	2/8 t_{CY}+360	nS
SR/TM Data Delay Time using External Clock	t_{SP}/t_{TP}	C_L=50pF	–	–	360	nS
SR Clock Invalid Time	t_{SINH}	–	2/8 t_{CY}	–	–	nS
INT Invalid Time	t_{IINH}	–	1/8 t_{CY}	–	–	nS

TIMING CHARTS

Output Condition



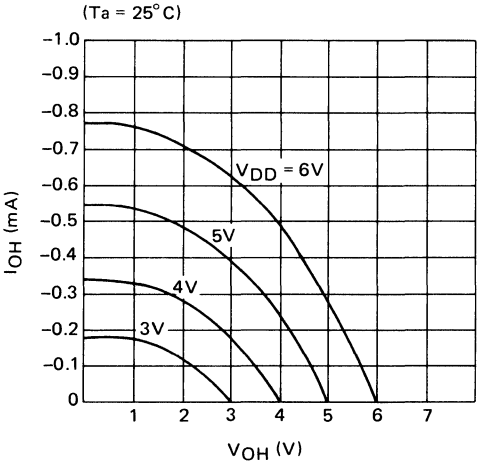
Input Condition



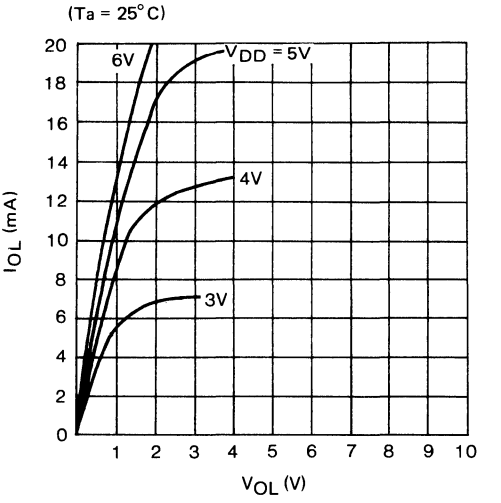
$t_{S\ INH}$: P_{01} (SR clock) INH period during LMSR INST.
(Note: P_{01} is used for clock of SR)

$t_{I\ INH}$: P_{20} (interrupt) INH period during RPB and RPBD INST.

TYP. Current vs Voltage for High State Output (I_{OH}) (V_{OH})

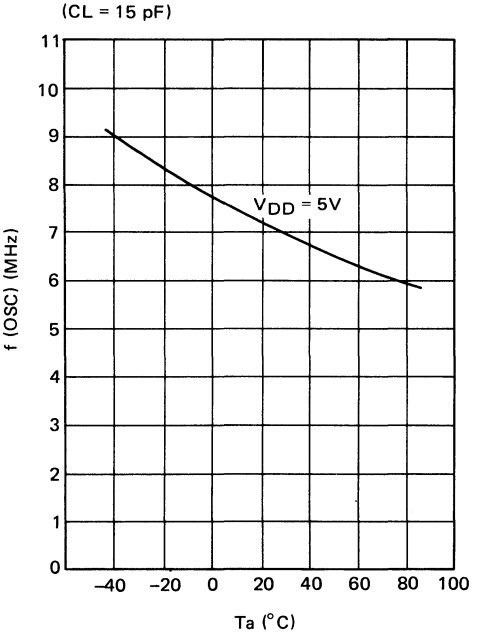


TYP. Current vs. Voltage for Low State Output (I_{OL}) (V_{OL})

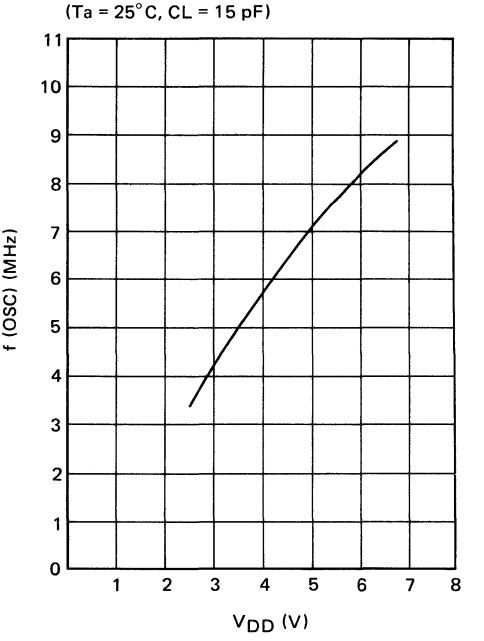


6

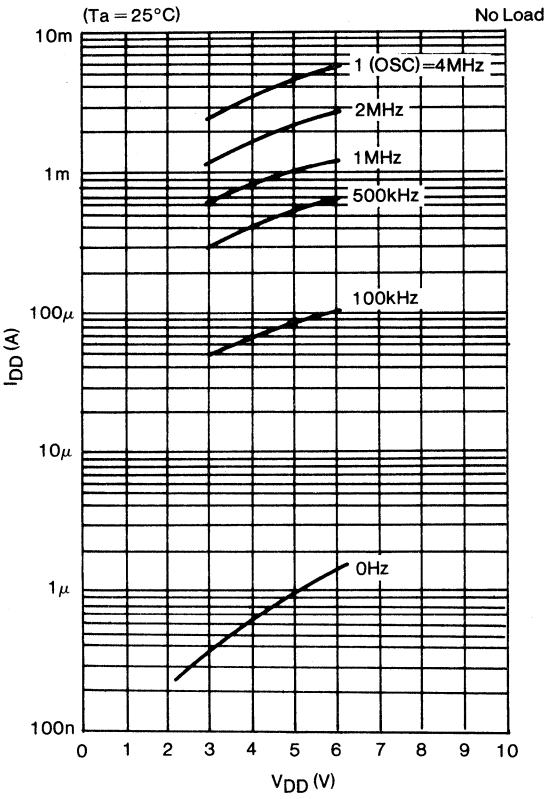
TYP. Maximum Oscillator Frequency vs Temperature f(OSC) (Ta)



TYP. Maximum Oscillator Frequency vs. Supply Voltage f(OSC) (V_{DD})



TYP. Supply Current vs. Supply Voltage
(I_{DD}) (V_{DD})



MSM6404VS

MSM6404 PIGGY BACK

GENERAL DESCRIPTION

The MSM6404VS is a device whose built-in ROM is replaced by external EPROM using the piggy-back method.

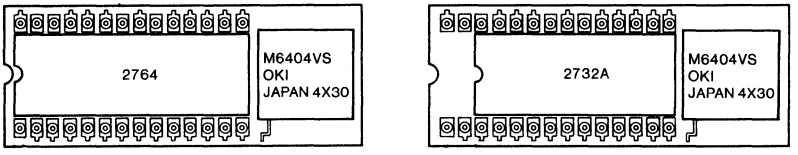
FEATURES

- Supply Voltage: 5V ± 5%
- Frequency: DC ~ 4.2 MHz
- Operating Temperature: 0 ~ 70°C

Note: There are a few differences in the electrical characteristics of this chip and the evaluation chip. Please refer to next page for the detail.

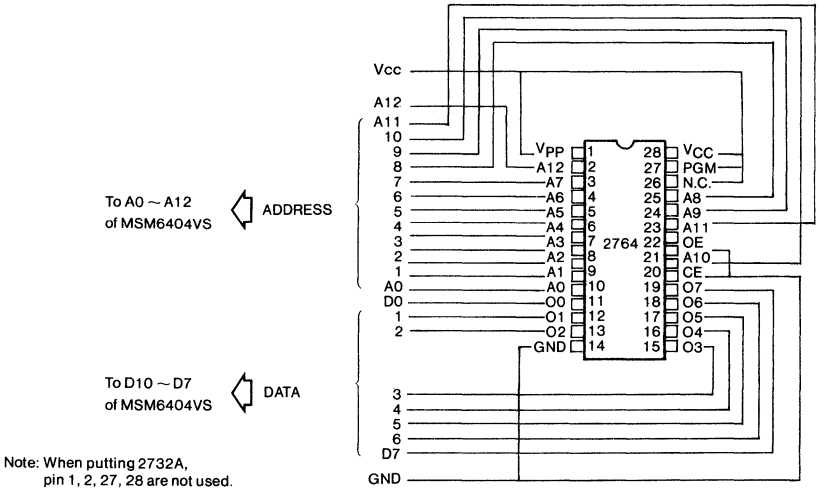
PUTTING METHOD OF ROM

Please refer to drawing below.



PIN CONFIGURATION

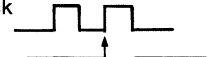
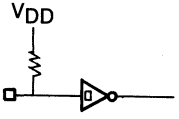
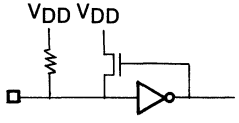
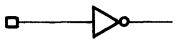
Pin Connection between MSM6404VS and EPROM



Note: When putting 2732A, pin 1, 2, 27, 28 are not used.

6

DIFFERENCES BETWEEN MSM6404 AND MSM6404VS (PIGGY-BACK)

Item	6404	6404VS (Piggy-Back)
1. Port initialization during reset	Port P0, 1, 3 are set to "1" and port 2, 4, 5, 6, 7, 8 are reset to "0" directly by signal put into the RESET.	Port P0, 1, 3 are set to "1" and port 2, 4, 5, 6,7, 8 are reset to "0" during reset cycle being executed.
2. Timer Operation	After being reset, timer stops counting until data are set in it.	It is undecidable whether the timer starts counting or not after being reset. Therefore, the timer should be initialized by software.
3. Shift registor	Serial Out F/F (SOF/F) is set to "0" after being reset.	It is undecidable whether Serial Out F/F (SOF/F) is set to "0" or "1" after being reset. Therefore Serial Out F/F should be initialized by software.
4. Port input/output timing	Internal clock  Data are input at this moment	Internal clock  Data are input at this moment
	Internal clock Synchronized with falling edge  Data are output at this moment	Internal clock  Data are output at this moment
5. Port input/output (maracteris-tics)	TTL FO=1 (I_{OL} = 1.6 mA 0.4V)	LSTTL FO=1 (I_{OL} = 0.4 mA 0.4V)
	P20~3  V_{DD}	P20~3  V_{DD} V_{DD}
	TTL compatible input P00~P83  (Except P20~3)	CMOS input P00~83  (Except P20~3)
Available ROM capacity	4K byte	Accessible up to 8K byte
IPL call instruction	Not available	Available

MSM6408

HIGH-SPEED 4-BIT SINGLE CHIP MICROCONTROLLER

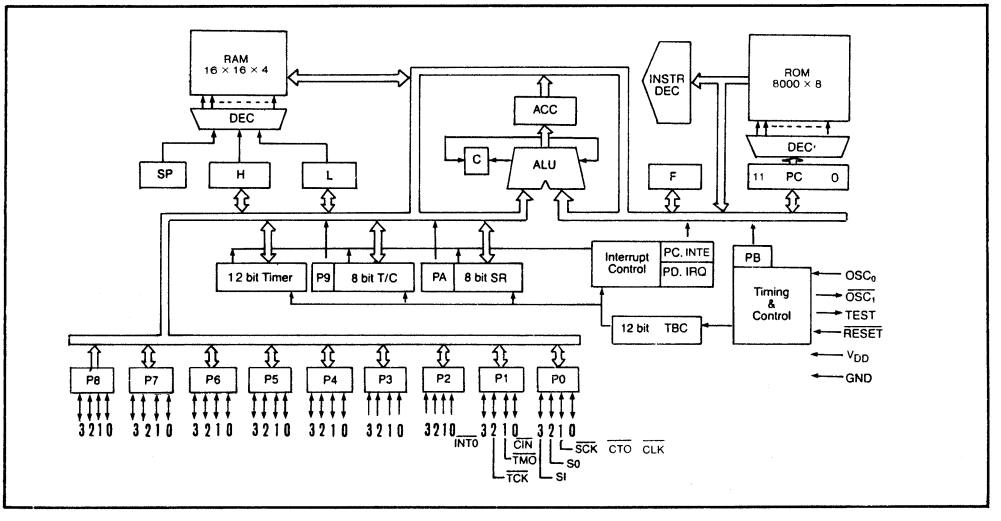
GENERAL DESCRIPTION

The OKI MSM6408 microcontroller is a low power, high-performance single-chip device implemented in complementary metal oxide semiconductor technology. 64K bits of mask program ROM, 1024 bits of data RAM, 36 Input/Output lines, a programmable timer/event-counter, and oscillator are integrated onto one chip. Program memory is byte wide and data-paths are organized in 4-bit nibbles. RAM and I/O lines are bit addressable. 122 instructions include binary, BCD operations; bit set, reset, test; 8-bit I/O; relative jumps; multifunctional instructional (increment, modify, skip) 8-bit wide table output; subroutine call and return.

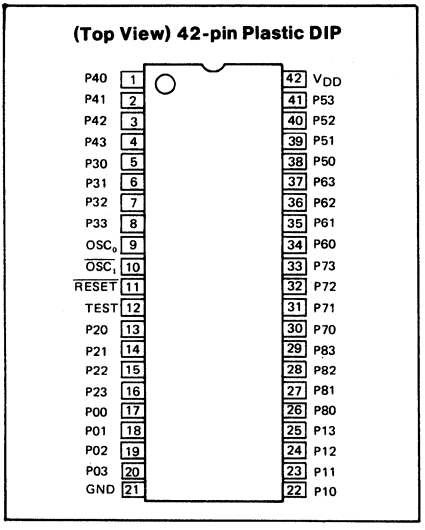
FEATURES

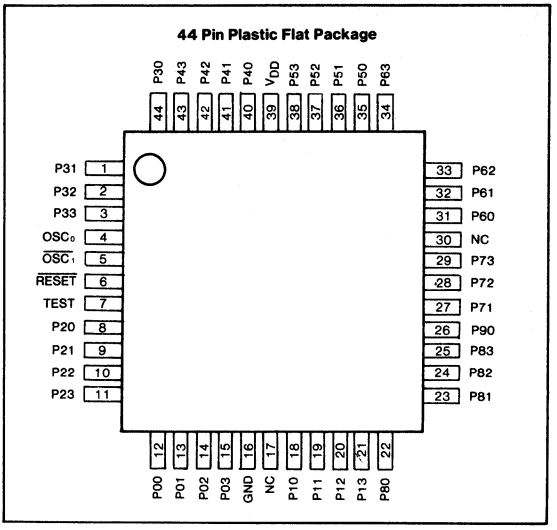
- 8096 × 8 MASK ROM
  An evaluation board is available for up to 8K × 8.
- 256 × 4 RAM (including the stack area)
- 9 × 4 Ports, 36 I/O lines
  4 lines for input ports having a latch, and the other 32 lines for bit operation are available.
- Three built-in counters
  12-bit time-base counter
  12-bit programmable timer
  8-bit high-speed programmable timer/event counter
- Built-in 8-bit serial I/O register (with 3-bit counter)
- Five interrupts with five priority levels
  (4 internal, 1 external)
- 32 stack levels (in RAM)
- LED direct drive available (8mA x 5 ports at the same time)
- Power down features
- Instruction execution time
  1.0 μs 4.0 MHz clock
- Instruction systems suitable for control
- 122 instructions
- Mask option
  P60-63 for input port
- Full static operation
- Low power consumption
  TYP 0.4μW at V_{DD}=2V
  TYP 5μW at V_{DD}=5V 0Hz clock
- 5V single power supply, 42-pin DIP or 44-pin FLAT

BLOCK DIAGRAM



PIN CONFIGURATION





PIN DESCRIPTION

Pin name	Input/output	Function	When reset
P00 P01 / $\overline{SCK}$ P02 / SO P03 / SI	Input/output	4-bit input/output port. P01 to P03 are also used as serial interface terminals.	"1"
P10 / $\overline{CIN}$ P11 / $\overline{TMO}$ P12 / $\overline{TCK}$ P13	Input/output	4-bit input port with latch. Built-in pull up register for all bit input.	"1"
P20 / $\overline{INT}$ P21 P22 P23	Input	4-bit input port with a latch. P20 is shared with INT input. (Fall trigger input) P21 ~ 23 are level input. Built-in pull up register for all bit input.	The latch is reset.
P30–33	Input/output	4-bit input/output port	"1"
P40–43	Input/output	4-bit input/output port (8-bit output port)	"0"
P50–53	Input/output	4-bit input/output port (8-bit output port)	"0"
P60–63	Input/output	4-bit input/output port *1	"0"
P70–73	Input/output	4-bit input/output port	"0"
P80–83	Input/output	4-bit input/output port	"0"
OSC_0 $\overline{OSC_1}$	Input/output	X'tal connection terminal for system clock oscillation	Oscillation wave
TEST	Output	(Test terminal for Maker)	Pulse output
$\overline{RESET}$	Input	System reset input terminal	
VDD GND		Power source voltage supply	

Note: When each port is used for output, it is possible to drive one TTL (one input).
*1 Can be made as a port dedicated to input (mask option).

INSTRUCTION LIST

	Mnemonic		Description	Code	Byte	Cycle
Load, Push, Pop	LAI	n	A ← n	9n	1	1
	LLI	n	L ← n	8n	1	1
	LHLI	nn	HL ← nn	15nn	2	2
	LMI	nn	M(w) ← nn	14nn	2	2
	LAL		A ← L	21	1	1
	LLA		L ← A	2D	1	1
	LAH		A ← H	22	1	1
	LHA		H ← A	2E	1	1
	LAM		A ← M	38	1	1
	LMA		M ← A	2F	1	1
	LAM+		A ← M, L ← L+1, Skip if L=0	24	1	1
	LAM−		A ← M, L ← L−1, Skip if L=F	25	1	1
	LMA+		M ← A, L ← L+1, Skip if L=0	26	1	1
	LMA−		M ← A, L ← L−1, Skip if L=F	27	1	1
	LAMM	n_2	A ← M, H ← H V n_2	39–3B	1	1
	LAMD	mm	A ← Md	10mm	2	2
	LMAD	mm	Md ← A	11mm	2	2
	LMTD	mm	Md(w) ← T (M(w), A), T=ROM table	19mm	2	3
	LMCT		M(w) ← CT	3E59	2	2
	LCTM		CT ← M(w)	3E51	2	2
	LMSR		M(w) ← SR	3E5A	2	2
	LSRM		SR ← M(w)	3E52	2	2
	LTMM		TM ← (M(w), A)	3E50	2	2
	PUSH		ST ← C, A, H, L, SP ← SP−4	1C	1	3
	POP		C, A, H, L, ← ST SP ← SP+4	1D	1	3
Exchange	X		A ⟷ M	28	1	1
	XM	n_2	A ⟷ M, H ← H V n_2	29–2B	1	1
	X+		A ⟷ M, L ← L+1, Skip if L=0	3C	1	1
	X−		A ⟷ M, L ← L−1, Skip if L=F	2C	1	1
Increment/ Decrement	INA		A ← A+1, Skip if A=0	30	1	1
	INM		A ← M+1, Skip if M=0	33	1	1
	INL		L ← L+1, Skip if L=0	31	1	1

INSTRUCTION LIST (Continued)

	Mnemonic		Description	Code	Byte	Cycle
Increment/Decrement	INH		H ← H+1, Skip if M = 0	32	1	1
	INMD	mm	Md ← Md+1, Skip if Md = 0	12mm	2	2
	DCA		A ← A−1, Skip if A = F	34	1	1
	DCM		M ← M−1, Skip if M = F	37	1	1
	DCL		L ← L−1, Skip if L = F	35	1	1
	DCH		H ← H−1, Skip if H = F	36	1	1
	DCMD	mm	Md ← Md−1, Skip if Md = F	13mm	2	2
Arithmetic	ADS		A ← A+M, Skip if Cy = 1	02	1	1
	ADCS		A, C ← A+M+C, Skip if Cy=1	01	1	1
	ADC		A, C ← A+M+C	03	1	1
	AIS	n	A ← A+n, Skip if Cy = 1	3E4n	2	2
	DAA		A ← A+6	06	1	1
	DAS		A ← A+10	0A	1	1
	AND		A ← A ∨ M	0D	1	1
	OR		A ← A ∨ M	05	1	1
	EOR		A ← A ∨ M	04	1	1
	CMA		A ← $\overline{A}$	0B	1	1
	CIA		A ← $\overline{A}$+1	0C	1	1
	RAL		Rotate Left with C	0E	1	1
	RAR		Rotate Right with C	0F	1	1
	TC		Skip if C = 1	09	1	1
	SC		C ← 1	07	1	1
	RC		C ← 0	08	1	1
Compare	CAI	n	Skip if A = n	3E0n	2	2
	CLI	n	Skip if L = n	3E2n	2	2
	CPI	p, n	Skip if P_p = n	17pn	2	2
	CMI	n	Skip if M = n	3E1n	2	2
	CAM		Skip if A = M	16	1	1
Bit operation	TAB	n_2	Skip if Abit (n_2) = 1	54–57	1	1
	RAB	n_2	Abit (n_2) ← 0	64-67	1	1
	SAB	n_2	Abit (n_2) ← 1	74–77	1	1
	TMB	n_2	Skip if Mbit (n_2) = 1	58–5B	1	1
	RMB	n_2	Mbit (n_2) ← 0	68–6B	1	1

INSTRUCTION LIST (Continued)

	Mnemonic		Description	Code	Byte	Cycle
Bit operation	SMB	n	Mbit (n_2) ←1	78–7B	1	1
	TFB	n_2	Skip if Fbit (n_2) =1	5C–5F	1	1
	RFB	n_2	Fbit (n_2) ←0	6C-6F	1	1
	SFB	n_2	Fbit (n_2) ←1	7C–7F	1	1
	TPB	n_2	Skip if Pbit (n_2) =1	50–53	1	1
	RPB	n_2	Pbit (n_2) ←0	60–63	1	1
	SPB	n	Pbit (n_2) ←1	70–73	1	1
	TPBD	p n_2	Skip if $P_p$bit (n_2) =1	3D $p_{0\sim3}$	2	2
	RPBD	p n_2	$P_p$bit (n_2) =0	3D $p_{4\sim7}$	2	2
	SPBD	p n_2	$P_p$bit (n_2) =1	3D $p_{8\sim B}$	2	2
Interrupt	MEI		MEIF ←1	3E60	2	2
	MDI		MEIF ←0	3E61	2	2
	EITB		EITBF ←1	3DC9	2	2
	EITM		EITMF ←1	3DCA	2	2
	EICT		EICTF ←1	3DCB	2	2
	EIEX		EIEXF ←1	3DC8	2	2
	DITB		EITBF ←0	3DC5	2	2
	DITM		EITMF ←0	3DC6	2	2
	DICT		EICTF ←0	3DC7	2	2
	DIEX		EIEXF ←0	3DC4	2	2
	TITB		Skip If EITBF =1	3DC1	2	2
	TITM		Skip If EITMF =1	3DC2	2	2
	TICT		Skip If EICTF =1	3DC3	2	2
	TIEX		Skip If EIEXF =1	3DC0	2	2
	TQEX		Skip If IRQEX =1	3D20	2	2
	TQTB		Skip If IRQTB =1	3DD0	2	2
	TQTM		Skip If IRQTM =1	3DD1	2	2
	TQCT		Skip If IRQCT =1	3DD2	2	2
	TQSR		Skip If IRQSR =1	3DD3	2	2
	RQEX		IRQ EX ←0	3D24	2	2
	RQTB		IRQ TB ←0	3DD4	2	2
	RQTM		IRQ TM ←0	3DD5	2	2
	RQCT		IRQ CT ←0	3DD6	2	2
	RQSR		IRQ SR ←0	3DD7	2	2

6

INSTRUCTION LIST (Continued)

	Mnemonic		Description	Code	Byte	Cycle
Shift resistor / Counter	ECT		CTF ←1 (start)	3DBB	2	2
	ESR		SRF ←1 (start)	3DBA	2	2
	DCT		CTF ←0 (stop)	3DB7	2	2
	DSR		SRF ←0 (stop)	3DB6	2	2
	TCT		Skip If CTF =1	3DB3	2	2
	TSR		Skip If SRF =1	3DB2	2	2
Branch	JCP	a_6	PC ← a_6	C0~FF	1	1
	JP	a_{12}	PC ← a_{12}	$4a_{12}$	2	2
	LJP	a_{13}	PC ← a_{13}	3F	3	4
	CZP	a	ST ←PC+1, PC ←2a, SP ←SP−4	Ba	1	4
	CAL	a_{12}	ST ←PC+2, PC ← a_{12}, SP ←SP−4	Aa_{12}	2	4
	RT		PC ←ST, SP ←SP+4	IE	1	4
	RTS		PC ←ST, SP ←SP+4, Skip unconditional	IF	1	4
	JA		PC ← (PC ←A)+1	IA	1	1
	JM		PC ← (M(w), A)	IB	1	2
Input/Output	IP		A ←P	20	1	1
	IPD	p	A ←Pp	3DpD	2	2
	OP		P ←A	23	1	1
	OPD	p	Pp ←A	3DpC	2	2
CPU control	NOP		No Operation	00	1	1
	HALT		Halt CPU	3DB8	2	2
	STOP		Stop Clock	3DB9	2	2

6

ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	Ta = 25°C	−0.3 to 7	V
Input Voltage	V_I		−0.3 to V_{DD}	V
Output Voltage	V_O		−0.3 to V_{DD}	V
Power Dissipation	P_D	Ta = 25°C per package	200 max.	mW
		Ta = 25°C per output	50 max.	mW
Storage Temperature	Tstg	–	−55 to +150	°C

OPERATING RANGE

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	$f_{(OSC)} \leqq 1$ MHz	3 to 6	V
		$f_{(OSC)} \leqq 4.0$ MHz	4.5 to 5.5	V
Data-Hold Voltage	V_{DDH}	f (OSC) = 0 Hz	2 to 6	V
Operating Temperature	T_{OP}	–	−40 to +85	°C
Fan Out	N	MOS Load	15	–
		TTL Load	1	

DC CHARACTERISTICS

(V_{DD} = 5V±10%, Ta = −40 to +85°C)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
"H" Input Voltage*1,*2	V_{IH}	–	2.4	–	V_{DD}	V
"H" Input Voltage*3,*4	V_{IH}	–	3.6	–	V_{DD}	V
"L" Input Voltage	V_{IL}	–	−0.3	–	0.8	V
"H" Output Voltage*1,*5	V_{OH}	I_O = −15μA	4.2	–	–	V
"L" Output Voltage*1	V_{OL}	I_O = 1.6mA	–	–	0.4	V
"L" Output Voltage*5	V_{OL}	I_O = 15μA	–	–	0.4	V
"L" Output Voltage*6	V_{OL}	I_O = 8mA	–	1	2	V
Input Current*3	I_{IH}/I_{IL}	$V_I = V_{DD}/0V$	–	–	15/−15	μA
Input Current*2,*4	I_{IH}/I_{IL}	$V_I = V_{DD}/0V$	–	–	1/−30	μA
"H" Output Current*1	I_{OH}	V_O = 2.4V	−0.1	–	–	mA
"H" Output Current*1	I_{OH}	V_O = 0.4V	–	–	−1.2	mA
Input Capacity	C_I	f=1MHz Ta=25°C	–	5	–	pF
Output Capacity	C_O		–	7	–	
Current Dissipation (when stop condition)	I_{DDS}	V_{DD}=2V, no load Ta=25°C	–	0.2	5	μA
		No load	–	1	100	μA
Current Dissipation	I_{DD}	Quarts oscillation f=4 MHz, no load	–	6	12	mA

*1 Applied to P0, P1, P3, P4, P5, P6, P7 and P8

*2 Applied to P2 *3 Applied to OSC_0 *4 Applied to $\overline{RESET}$ *5 Applied to $\overline{OSC_1}$

*6 In using LED, total output current should be within the limit of Power dissipitation in "Absolute Maximum Rating."

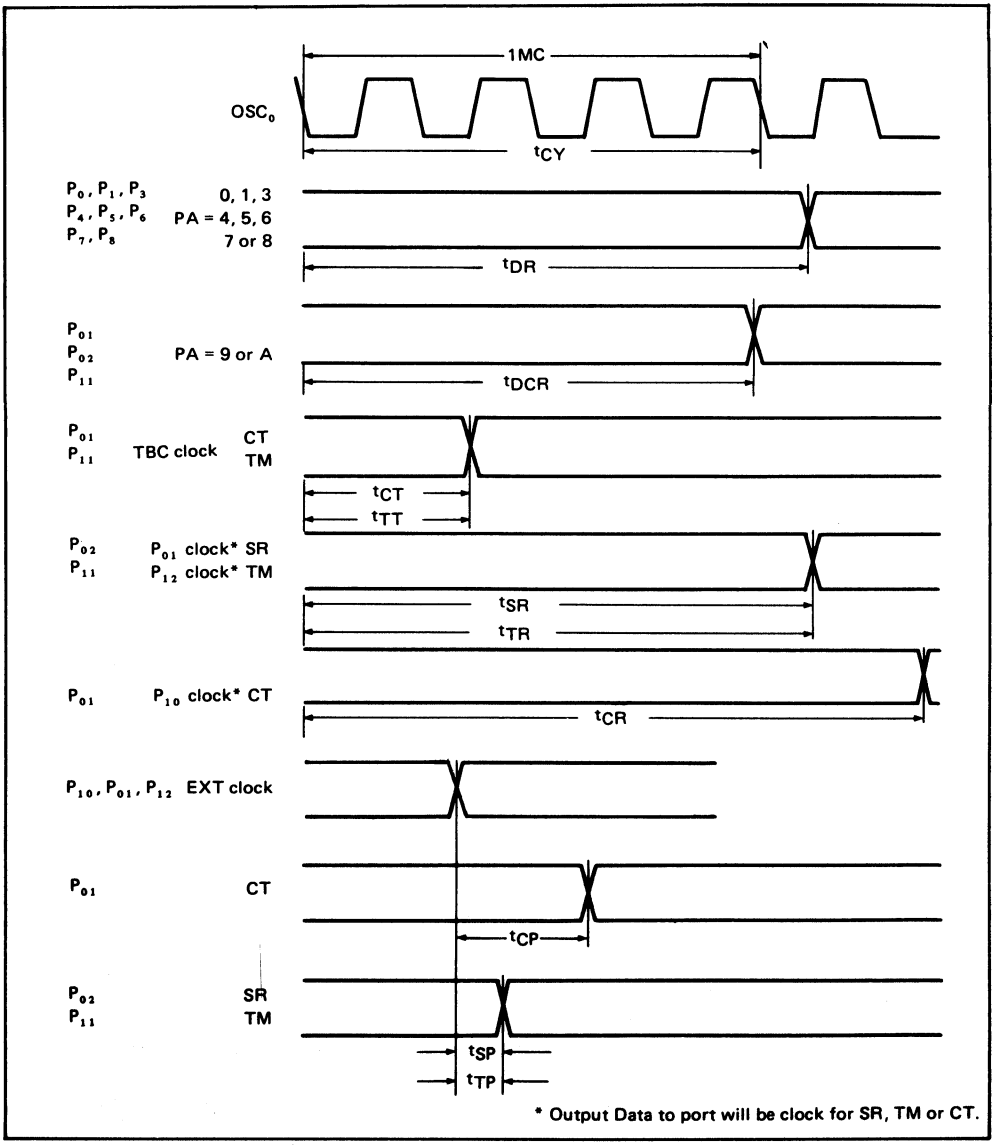
AC CHARACTERISTICS

(V_{DD} = 5V±10%, Ta = −40 to +85°C)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Clock Pulse Width Clock (OSC)	$t_{\phi W}$	–	125	–	–	nS
Cycle Time	t_{CY}	–	1	–	–	μS
Input Data Setup Time	t_{DS}	–	120	–	–	nS
Input Data Hold Time	t_{DH}	–	120	–	–	nS
Input Data, Input Clock Pulse Width	t_{DW}	–	120	–	–	nS
SR Data Setup Time	t_{SS}	–	120	–	–	nS
SR Data Hold Time	t_{SH}	–	120	–	–	nS
Data Delay Time	t_{DR}	C_L=15pF	–	–	t_{CY}+300	nS
Data Delay Time at Mode Switching	t_{DCR}	C_L=15pF	–	–	7/8 t_{CY}+300	nS
CT/TM Data Delay Time using TBC Clock	t_{CT}/t_{TT}	C_L=15pF	–	–	2/8 t_{CY}+360	nS
SR/TM Data Delay Time using PR Clock	t_{SR}/t_{TR}	C_L=15pF	–	–	t_{CY}+480	nS
CT Data Delay Time using PR Clock	t_{CR}	C_L=15pF	–	–	10/8 t_{CY}+480	nS
CT Data Delay Time using External Clock	t_{CP}	C_L=15pF	–	–	2/8 t_{CY}+360	nS
SR/TM Data Delay Time using External Clock	t_{SP}/t_{TP}	C_L=50pF	–	–	360	nS
SR Clock Invalid Time	t_{SINH}	–	2/8 t_{CY}	–	–	nS
INT Invalid Time	t_{IINH}	–	1/8 t_{CY}	–	–	nS

TIMING CHARTS

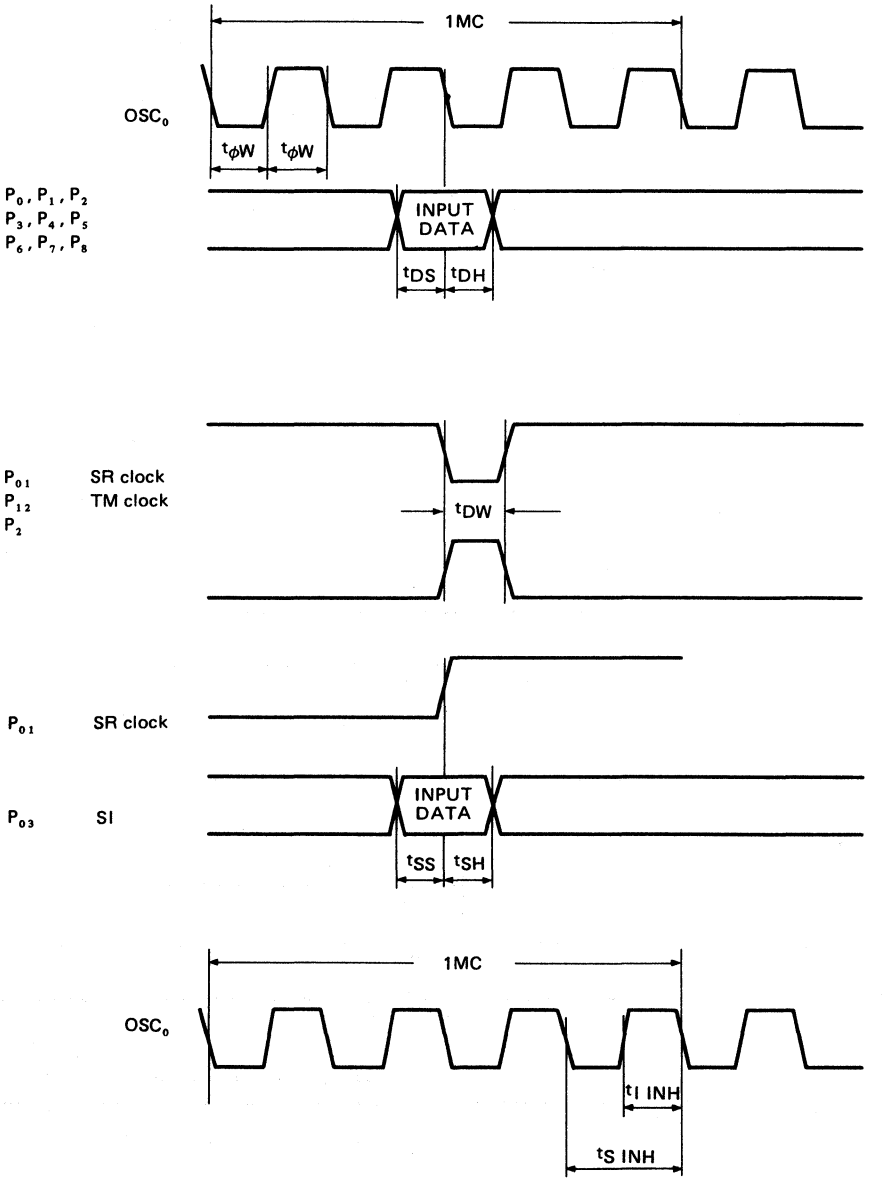
Output Condition



* Output Data to port will be clock for SR, TM or CT.

6

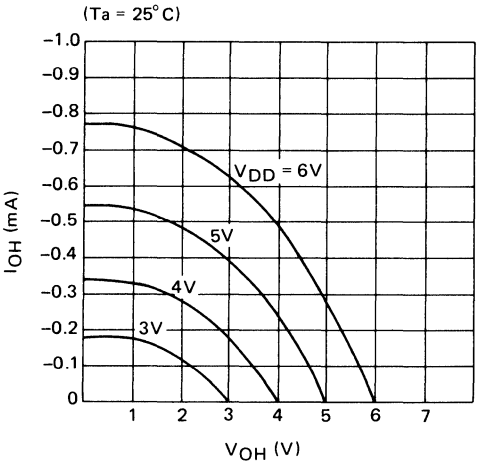
Input Condition



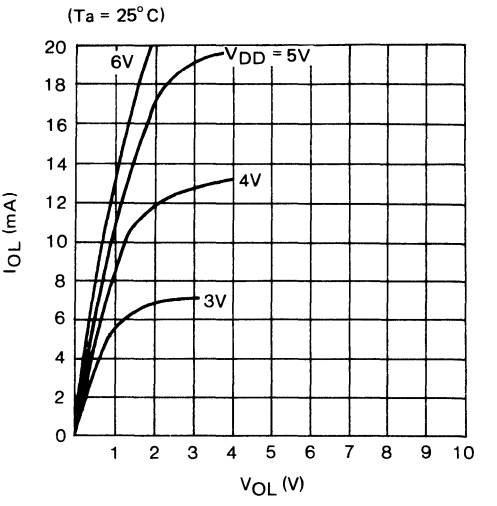
$t_{S\ INH}$: P_{01} (SR clock) INH period during LMSR INST.
(Note: P_{01} is used for clock of SR)

$t_{I\ INH}$: P_{20} (interrupt) INH period during RPB and RPBD INST.

TYP. Current vs Voltage for High State Output (I_{OH}) (V_{OH})

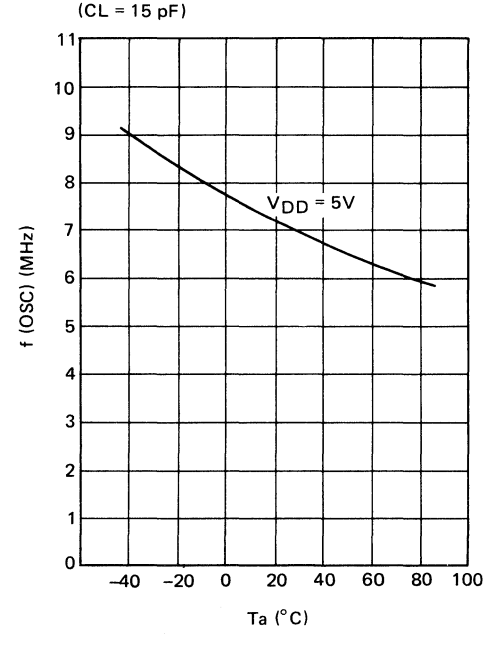


TYP. Current vs. Voltage for Low State Output (I_{OL}) (V_{OL})

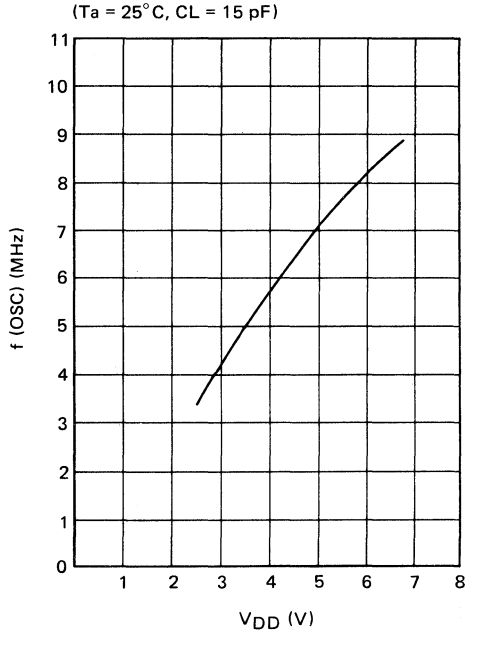


6

TYP. Maximum Oscillator Frequency vs Temperature
f(OSC) (Ta)



TYP. Maximum Oscillator Frequency vs. Supply Voltage
f(OSC) (V_{DD})



MSM6411

CMOS 4-BIT SINGLE CHIP MICROCONTROLLER

GENERAL DESCRIPTION

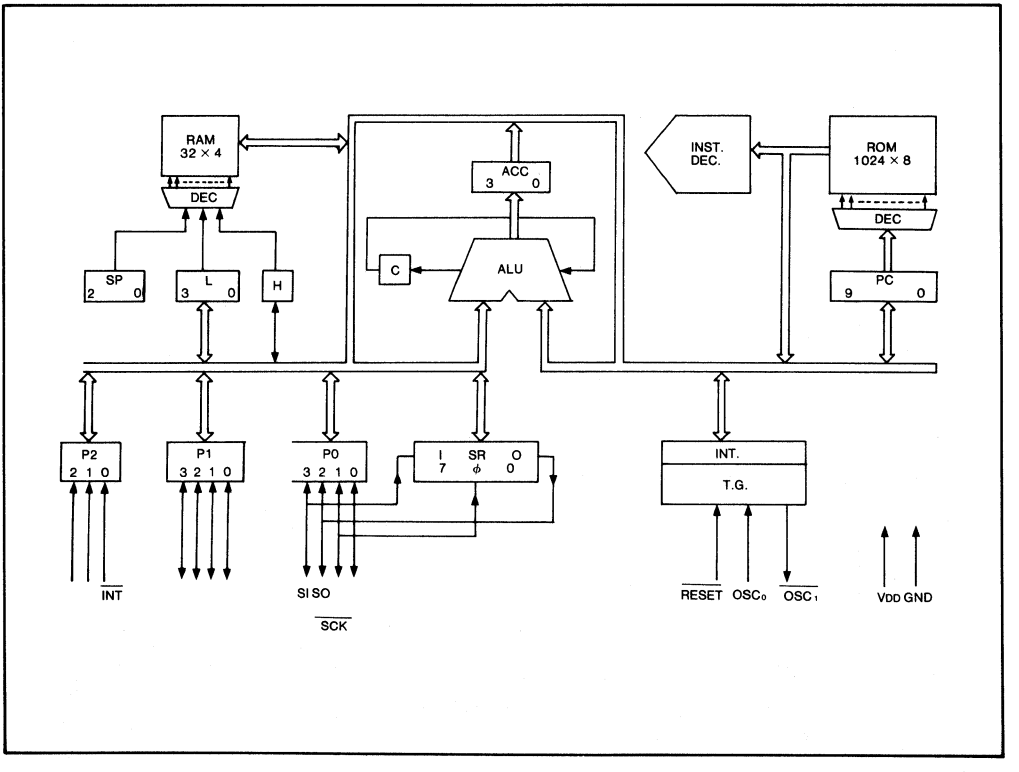
The OKI MSM6411 microcontroller is a low-power, high-performance single-chip device implemented in complementary metal oxide semiconductor technology. 1024 × 8 bits of program ROM, 32 × 4 bits of data RAM, 11 Input/Output lines and oscillator. Program memory is byte wide and data paths are organized as 4-bit wide. 63 instructions include binary, logical operations; bit set, reset, test; multifunctional.

FEATURES

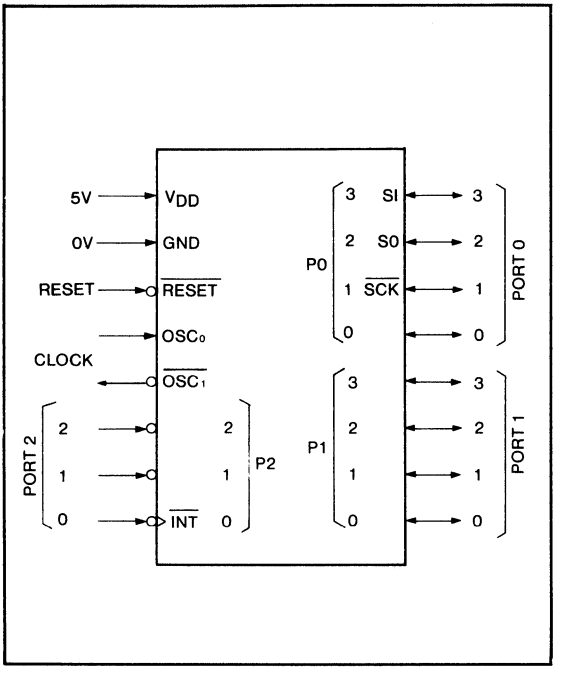
- 1024 × 8 Internal ROM
- 32 × 4 Internal RAM
- 11 I/O Lines (8 I/O Lines, 3 Input Lines)
- 8-bit serial I/O Register
- 2 Interrupt Levels
- 8 Stack Levels
- LED direct drive available (8mA x 5 ports at the same time)
- 952 ns 4.2MHz (V_{DD} 5V±10%)
- 63 Instructions
- Self-Contained Oscillator
- −40 to +85°C Operating Temperature
- 3 to 6V Operating V_{DD}
- Low Power Consumption 5 μW Typical (STOP, V_{DD} = 5V, no load)
- Mask Option Crystal/Ceramic Oscillator

6

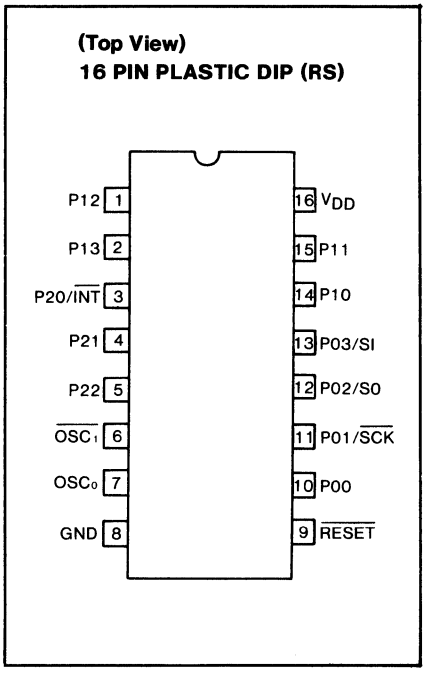
FUNCTIONAL BLOCK DIAGRAM



LOGIC SYMBOL



PIN CONFIGURATION



6

PIN DESCRIPTION

Designation	Input/Output	Pin No.	Function	Reset
P00 P01/$\overline{SCK}$ P02/SO P03/SI	Input/Output	10 11 12 13	4 Bits I/O port. P01 to P03 are used both I/O port and terminal of shift resister	"1"
P10 P11 P12 P13	Input/Output	14 15 1 2	4 Bits I/O port	"1"
P20/$\overline{INT}$ P21 P22	Input	3 4 5	3 Bits input port with latch. P20 is used as both input port and input terminal of INT (input of falling edge trigger).	Latch is reset. ("0")
OSC_0	Input	7	Input terminal of system clock. Oscillation circuit consists of OSC_0 and $\overline{OSC_1}$.	Clock pulse In
$\overline{OSC_1}$	Output	6	Oscillation circuit consists of OSC_0 and $\overline{OSC_1}$.	
$\overline{RESET}$	Input	9	Input terminal of system reset	
V_{DD} GND	Input	16 8	Main power source and circuit GND potential.	

FUNCTIONAL DESCRIPTION

ROM

ROM is organized in 1024 words by 8 bits. It is used to stored developed application programs (instructions). It is addressed by the program counter (PC).

PC

The PC consists of a 10-bit binary counter and is used to address ROM.

Stack and Stack Pointer

An interrupt or CAL instruction causes the contents of the PC to be saved in the stack. The PC is restored from the stack by RT instruction.

All RAM locations (up to 8 levels) are available as the stack. Note that four words of RAM are used for each level.

The stack pointer is a 5-bit up-down counter that points to the address of the next stack to be used. It allows the RAM locations to be used as a push-down stack.

RAM

RAM consists of up to 32 words 4 bits wide. It is addressed by the H- and L-registers or by the contents of the second byte of an instruction.

L-REGISTER

A 4-bit register which specifies RAM locations A3-A0.

H-REGISTER

A 1-bit register which specifies RAM location A4.

ALU

A 4-bit logic circuit that provides arithmetic and logical operations.

ACC

Consisting of a 4-bit register, the accumulator holds the result of operations or the data present on ports.

C FLAG

The flag that holds a carry generated from the result of operations.

INPUT/OUTPUT Ports (2 × 4 bit)

Organized into 4 bits, 2 ports are provided for effecting and controlling data transfer to and from an external source. The ports are selected by codes included in instructions.

Input Ports (1 × 4 bit)

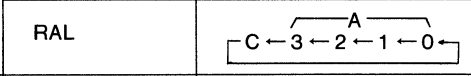
Contained port 2 (P2), which is an input port with latching function. P20 is set at falling edge of the input signal P21 and P22 are set at "0" level inputs. Also, P20 is used as an interrupt request flag. When P20 is set and an interrupt operation occurs, it is automatically reset.

TIMING CONTROL (TC)

A 0 level on the $\overline{\text{RESET}}$ pin for longer than 2 machine cycles initializes the internal circuitry.

Clock pulses are supplied to the XT pin from an external source. Also, by mask-option, it organizes a circuit of oscillation with P20 and produces clock pulses (by connecting externally to crystal, ceramic or CR).

INSTRUCTION LIST

	Mnemonic		Description	Code	Byte	Cycle
Load, Push, Pop	LAI	n	A ← n	90 – 9F	1	1
	LLI	n	L ← n	80 – 8F	1	1
	LHLI	nn	HL ← nn	15nn	2	2
	LAL		A ← L	21	1	1
	LLA		L ← A	2D	1	1
	LAM		A ← M	38	1	1
	LMA		M ← A	2F	1	1
	LAMD	mm	A ← Md	10mm	2	2
	LMAD	mm	Md ← A	11mm	2	2
	LMSR		M(w) ← SR	3E5A	2	2
	LSRM		SR ← M(w)	3E52	2	2
	PUSH		ST ← C, A, H, L, SP ← SP−1	1C	1	3
	POP		C, A, H, L ← ST, SP ← SP+1	1D	1	3
Input Output	IPD	p	A ← Pp	3DpD	2	2
	OPD	p	Pp ← A	3DpC	2	2
Arithmetic	ADS		A ← A+M, SKIP IF Cy = "1"	02	1	1
	ADC		C, A ← C+A+M	03	1	1
	AIS	n	A ← A+n, SKIP IF Cy = "1"	3E4n	2	2
	DAS		A ← A+10	0A	1	1
	AND		A ← A ∧ M	0D	1	1
	EOR		A ← A ∀ M	04	1	1
	CMA		A ← $\overline{A}$	0B	1	1
	CAM		SKIP IF A = M	16	1	1
	SC		C ← "1"	07	1	1
	RC		C ← "0"	08	1	1
	TC		SKIP IF C = "1"	09	1	1
	RAL			0E	1	1
Ex-change	X		A ⟷ M	28	1	1
Increment/ Decrement	INL		L ← L+1, SKIP IF L = "0"	31	1	1
	INH		H ← H+1, SKIP IF H = "0"	32	1	1
	INM		M ← M+1, SKIP IF M = "0"	33	1	1

INSTRUCTION LIST (CONT.)

	Mnemonic		Description	Code	Byte	Cycle
Increment/ Decrement	INMD	mm	Md ← Md+1, SKIP IF Md = "0"	12mm	2	2
	DCL		L ← L−1, SKIP IF L = "F"	35	1	1
	DCH		H ← H−1	36	1	1
	DCM		M ← M−1, SKIP IF M = "F"	37	1	1
Bit operation	TAB	n2	SKIP IF [A bit n2] = "1"	54−57	1	1
	TMB	n2	SKIP IF [M bit n2] = "1"	58-5B	1	1
	RMB	n2	[M bit n2] ← "0"	68−6B	1	1
	SMB	n2	[M bit n2] ← "1"	78−7B	1	1
	TPBD	p, n2	SKIP IF [Pp bit n2] = "1"	3D p0~3	2	2
	RPBD	p, n2	[Pp bit n2] ← "0"	3D p4~7	2	2
	SPBD	p, n2	[Pp bit n2] ← "1"	3D p8~B	2	2
Branch	JCP	a6	PC ← a6	C0~FF	1	1
	JP	a10	PC ← a10	40_43 00~FF	2	2
	CAL	a10	ST ← PC+2, PC ← a10, SP ← SP−1	A0_A3 00~FF	2	4
	RT		PC ← ST, SP ← SP+1	1E	1	4
Interrupt	MEI		MEIF ← "1"	3E60	2	2
	MDI		MEIF ← "0"	3E61	2	2
	EICT		EICTF ← "1"	3DCB	2	2
	EIEX		EIEXF ← "1"	3DC8	2	2
	DICT		EICTF ← "0"	3DC7	2	2
	DIEX		EIEXF ← "0"	3DC4	2	2
	TICT		SKIP IF EICTF = "1"	3DC3	2	2
	TIEX		SKIP IF EIEXF = "1"	3DC0	2	2
	TQEX		SKIP IF IRQEX = "1"	3D20	2	2
	TQSR		SKIP IF IRQSR = "1"	3DD3	2	2
	RQEX		IRQEX ← "0"	3D24	2	2
	RQSR		IRQSR ← "0"	3DD7	2	2
Shift resistor	ESR		SRF ← "1"	3DBA	2	2
	DSR		SRF ← "0"	3DB6	2	2
	TSR		SKIP IF SRF = "1"	3DB2	2	2
CPU control	STOP		STOP CLOCK	3DB9	2	2
	NOP		NO OPERATION	00	1	1

ABSOLUTE MAXIMUM RATING

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	Ta = 25°C	−0.3 ~ 7	V
Input Voltage	V_I		−0.3 ~ V_{DD}	V
Output Voltage	V_O		−0.3 ~ V_{DD}	V
Power Dissipation	P_D	Ta = 25°C per one package	200 max.	mW
		Ta = 25°C per one output	50 max.	mW
Storage Temperature	Tstg		−55 ~ +150	°C

OPERATING CONDITIONS

Parameter	Symbol	Condition	Limits	Unit
Supply Voltage	V_{DD}	f(OSC) ≦ 1MHz	3 ~ 6	V
		f(OSC) ≦ 4.2MHz	4.5 ~ 5.5	V
Data-Hold Voltage	V_{DDH}	f(OSC) = 0Hz	2 ~ 6	V
Operating Temperature	T_{OP}	–	−40 ~ +85	°C
Fan Out	N	MOS Load	15	–
		TTL Load	1	

6

DC CHARACTERISTICS

(V_{DD} = 5V±10%, Ta = −40 ~ +85°C)

Parameter	Symbol	Condition	Min.	Typ	Max.	Unit
"H" Input Voltage *1, *2	V_{IH}	———	2.4	–	V_{DD}	V
"H" Input Voltage *3, *4	V_{IH}	———	3.6	–	V_{DD}	V
"L" Input Voltage	V_{IL}	———	−0.3	–	0.8	V
"H" Output Voltage *1, *5	V_{OH}	I_O = −15μA	4.2	–	–	V
"L" Output Voltage *1	V_{OL}	I_O = 1.6mA	–	–	0.4	V
"L" Output Voltage *5	V_{OL}	I_O = 15μA	–	–	0.4	V
"L" Output Voltage*6	V_{OL}	I_O = 8mA	–	1	2	V
Input Current *3	I_{IH} / I_{IL}	$V_I = V_{DD}/0V$	–	–	15 / −15	μA
Input Current *2, *4	I_{IH} / I_{IL}	$V_I = V_{DD}/0V$	–	–	1 / −30	μA
"H" Output Current *1	I_{OH}	V_O = 2.4V	−0.1	–	–	mA
"H" Output Current *1	I_{OH}	V_O = 0.4V	–	–	−1.2	mA
Input Capacitance	C_I	f = 1MHz, Ta = 25°C	–	5	–	pF
Output Capacitance	C_O		–	7	–	pF
Power Consumption (STOP)	I_{DDS}	V_{DD} = 2V, no load Ta = 25°C	–	0.2	5	μA
		No load	–	1	100	μA
Power Consumption	I_{DD}	Crystal oscillation, No load, 4.2MHz	–	6	12	mA

*1 Applied to P0 and P1.

*2 Applied to P2.

*3 Applied to OSC_0

*4 Applied to $\overline{RESET}$

*5 Applied to $\overline{OSC_1}$

*6 In using LED, total output current should be within the limit of Power dissipitation in "Absolute Maximum Rating."

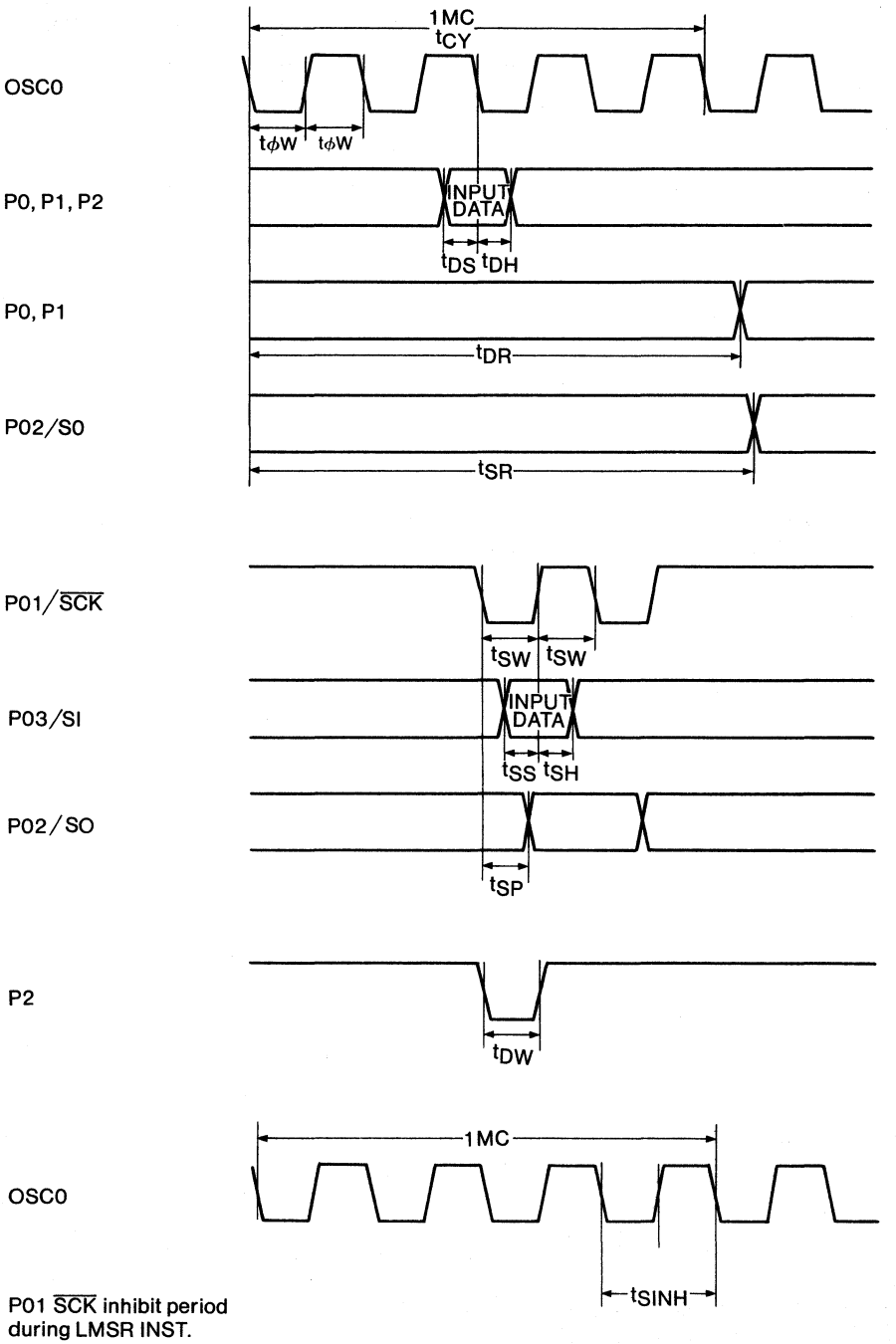
AC CHARACTERISTICS

(V_{DD} = 5V±10%, Ta = −40 ~ +85°C)

Parameter	Symbol	Condition	Min.	Typ	Max.	Unit
Clock (OSC_0) Pulse Width	$t_{\phi W}$	–	119	–	–	ns
Cycle Time	t_{CY}	–	952	–	–	ns
Input Data Setup Time	t_{DS}	–	120	–	–	ns
Input Data Hold Time	t_{DH}	–	120	–	–	ns
Input Data/Input Clock Pulse Width	t_{DW}	–	120	–	–	ns
SR Clock Pulse Width	t_{SW}	–	$t_{\phi W}$	–	–	ns
SR Data Setup Time	t_{SS}	–	120	–	–	ns
SR Data Hold Time	t_{SH}	–	120	–	–	ns
Data Delay Time	t_{DR}	C_L=15pF	–	–	t_{CY} +300	ns
SR Data Delay Time*	t_{SR}	C_L=15pF	–	–	t_{CY} +480	ns
SR Data Delay Time Using External Clock	t_{SP}	C_L=15pF	–	–	360	ns
SR Clock Invalid Time	t_{SINH}	–	2/8 t_{CY}	–	–	ns

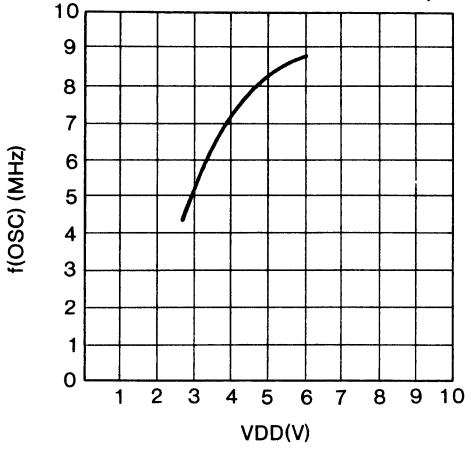
* When SR clock is oscillated by alternate output of "1" or "0" to P01.

TIMING CHART

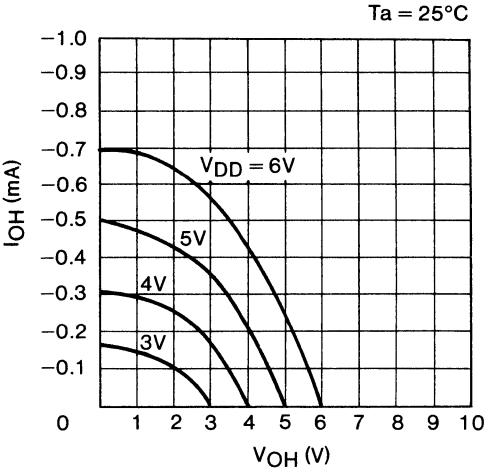


P01 $\overline{SCK}$ inhibit period during LMSR INST.

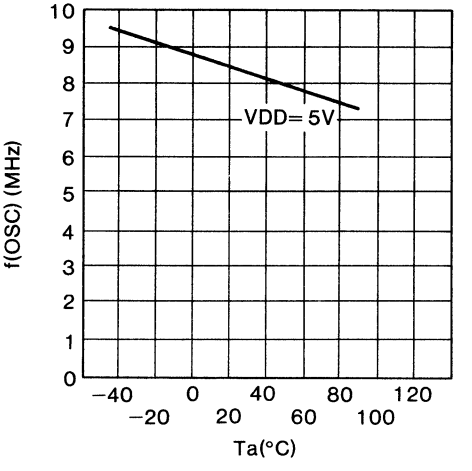
TYP. Maximum Oscillator Frequency f(OSC) vs Supply Voltage (VDD)



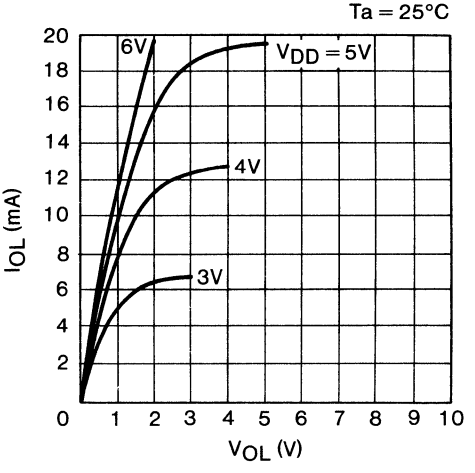
TYP. Current vs Voltage for High State Output (I_{OH}) (V_{OH})



TYP. Maximum Oscillator Frequency f(OSC) vs Temperature (Ta)

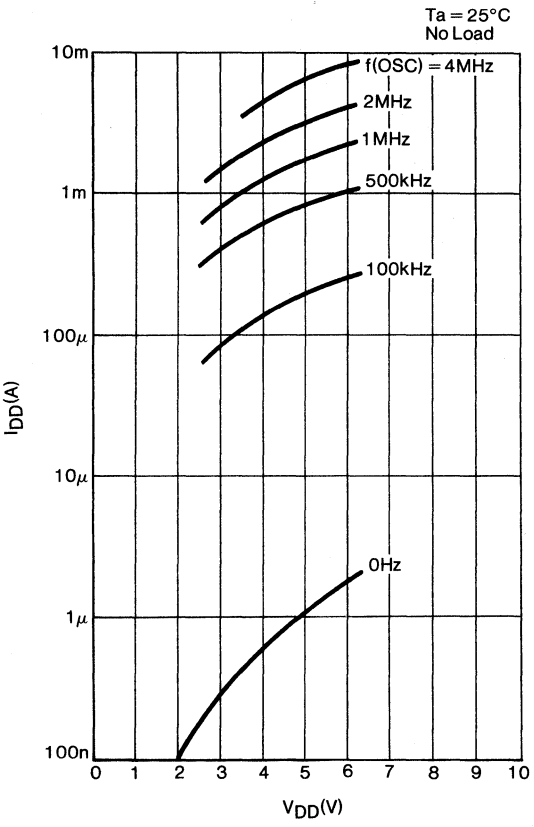


TYP. Current vs Voltage for Low State Output (I_{OL}) (V_{OL})



6

TYP. Supply Current vs Supply Voltage
(I_{DD}) (V_{DD})



OKI semiconductor

MSM6422

CMOS 4-BIT SINGLE CHIP MICROCONTROLLER

GENERAL DESCRIPTION

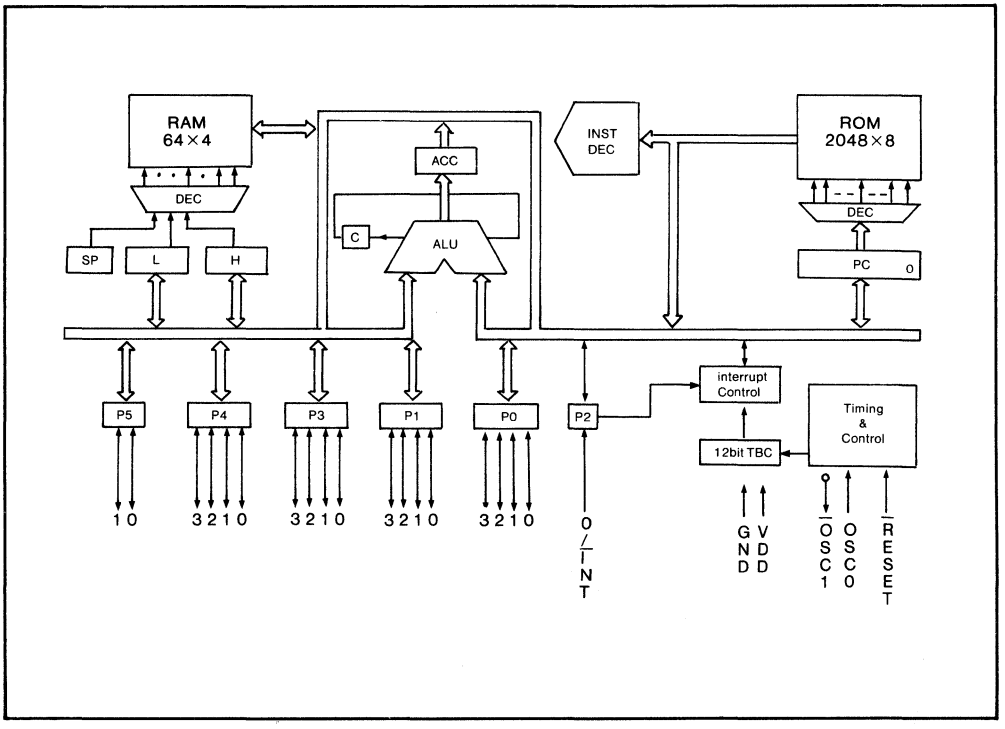
The OKI MSM6422 is a low power, high performance single-chip device implemented in complementary metal oxide semiconductor technology.

Integrated onto a single chip are 16K bits of mask program ROM; 256 bits of data RAM; 18 Input/Output lines and oscillator. Program memory is byte wide and data-paths are organized in 4 bit nibbles. RAM and I/O lines are bit addressable. 63 instructions include Binary, BCD operations; Bit set, Reset, Test; Subroutine call and return.

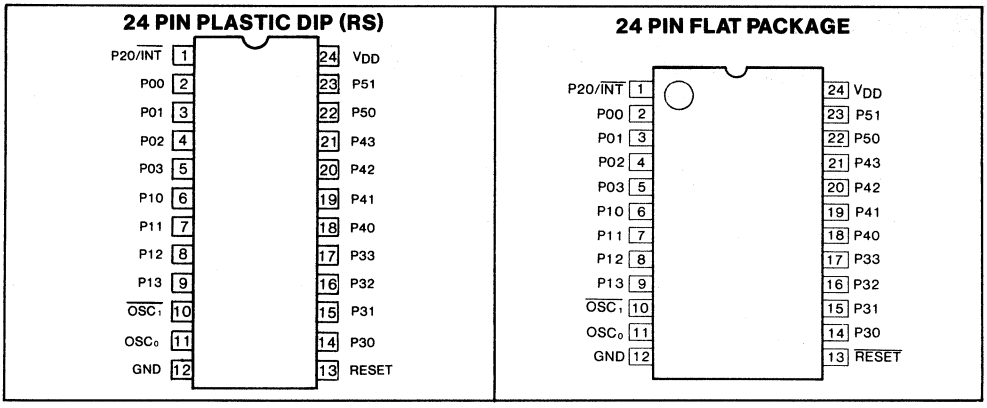
FEATURES

- Low power consumption – 30 mW Typical
- 2048 × 8 Internal ROM
- 64 × 4 Internal RAM
- 18 I/O Lines include 8 Bit Data Bus
- Self-contained Oscillator
- 63 Instructions
- 2 Interrupt Levels
- 16 Stack Levels
- LED direct drive available (8mA x 5 ports at the same time)
- −40 to +85°C Operating Temperature
- 4.5 to 5.5V Operating V_{DD} at 4.2 MHz
  3 to 6V Operating V_{DD} at 1 MHz
- TTL Compatible
- 952ns Cycle Time @ 4.2MHz
  (V_{DD} 5V ±10%)

FUNCTIONAL BLOCK DIAGRAM



LOGIC SYMBOL (Top View)



PIN DESCRIPTION

Terminal symbol	Input/Output	Function	Reset
P00 P01 P02 P03	I/O	4-bit I/O ports (pseudo bidirectional configuration)	"1"
P10 P11 P12 P13	I/O	4-bit I/O ports (pseudo bidirectional configuration)	"1"
P30 P31 P32 P33	I/O	4-bit I/O ports (pseudo bidirectional configuration)	"1"
P40 P41 P42 P43	I/O	4-bit I/O ports (pseudo bidirectional configuration)	"0"
P50 P51	I/O	2-bit I/O ports (pseudo bidirectional configuration)	"0"
P20/$\overline{INT}$	Input	1-bit input port with a latch. Combined use with an interrupt input(falling edge trigger input)	The latch is reset to "0"
OSC_0	Input	System clock (SYSCLK) input terminal. This provides an oscillation circle with $\overline{OSC_1}$ terminal.	–
$\overline{OSC_1}$	Output	System clock output terminal. This provides an oscillation circle with OSC_0 terminal.	–
$\overline{RESET}$	Input	RESET input terminal.	–
V_{DD} GND	Input	Power Supply terminals.	–

FUNCTIONAL DESCRIPTION

Program ROM

Organized into as many as 2,048 words by 8 bits, ROM is used to store developed application programs (instructions). It is addressed by the program counter (PC).

Data RAM

RAM consists of up to 64 words 4 bits wide. It is addressed by the H- and L-registers or by the contents of the second byte of an instruction.

Input/Output Ports

18 input/output port lines are provided for effecting and controlling data transfer to and from an external source. The ports are selected by codes included in instructions.

P20/$\overline{\text{INT}}$ PIN (1 line)

A low on this interrupt input pin sets the interrupt request flag. The flag is automatically reset when an external interrupt occurs.
The line can be used as an input port when interrupt is not used.

12-BIT TIME BASE COUNTER (TBC)

The time base counter consists of a 12-bit binary counter. An interrupt request is generated each time an overflow occurs from the division of OSC_0 input signals by 2^{12}.

PROGRAM COUNTER (PC)

The program counter (PC) consists of a 11-bit binary up counter. It is used to address ROM.

STACK AND STACK POINTER (SP)

An interrupt or subroutine call (CAL) causes the contents of the program counter to be saved in the stack. The program counter is restored from the stack by the RT instruction.

All RAM locations (up to 16 levels) are available as the stack. Note that four words of RAM are used for each level.

The stack pointer is a 4-bit up-down counter that points to the address of the next stack to be used. It allows the RAM locations to be used as a push-down stack.

L-REGISTER

A 4-bit register which specifies RAM locations A3-A0.

H-REGISTER

A 4-bit register whose two low-order bits specify RAM locations A5-A4.

ALU

The 4-bit logic circuit that provides arithmetic and logical operations.

ACCUMULATOR (Acc)

Consisting of a 4-bit register, the accumulator holds the result of operations or the data present on ports.

C-FLAG

The flag that holds a carry generated from the result of operations.

TIMING CONTROL (TC)

A 0 level on the $\overline{\text{RESET}}$ pin for longer than a predetermined period initializes the internal circuitry and ports.

Clock pulses are supplied to the OSC_0 pin from an external source. A crystal or ceramic oscillator may be connected to OSC_0 and $\overline{OSC_1}$ to form an oscillator circuit to produce clock pulses.

ABSOLUTE MAXIMUM RATING

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	Ta = 25°C	−0.3 ~ 7	V
Input Voltage	V_I		−0.3 ~ V_{DD}	V
Output Voltage	V_O		−0.3 ~ V_{DD}	V
Power Dissipation	P_D	Ta = 25°C per one package	200 max.	mW
		Ta = 25°C per one output	50 max.	mW
Storage Temperature	Tstg	—	−55 ~ +150	°C

OPERATING CONDITIONS

Parameter	Symbol	Condition	Limits	Unit
Supply Voltage	V_{DD}	f(OSC) $\leqq$ 1MHz	3 ~ 6	V
		f(OSC) $\leqq$ 4.2MHz	4.5 ~ 5.5	V
Memory-Hold Voltage	V_{DDH}	–	2 ~ 6	V
Operating Temperature	T_{OP}	–	−40 ~ +85	°C
Fan Out	N	MOS Load	15	–
		TTL Load	1	–

DC CHARACTERISTICS

(V_{DD} = 5V±10%, Ta = −40 ~ +85°C)

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
"H" Input Voltage *1, *2	V_{IH}	–	2.4	–	V_{DD}	V
"H" Input Voltage *3, *4	V_{IH}	–	4.2	–	V_{DD}	V
"L" Input Voltage	V_{IL}	–	−0.3	–	0.8	V
"H" Output Voltage *1, *5	V_{OH}	I_O = −15μA	4.2	–	–	V
"L" Output Voltage *1	V_{OL}	I_O = 1.6mA	–	–	0.4	V
"L" Output Voltage *5	V_{OL}	I_O = 15μA	–	–	0.4	V
"L" Output Voltage*6	V_{OL}	I_O = 8mA	–	1	2	V
Input Current *3	I_{IH} / I_{IL}	$V_I = V_{DD}$/0V	–	–	15 / −15	μA
Input Current *2, *4	I_{IH} / I_{IL}	$V_I = V_{DD}$/0V	–	–	1 / −30	μA
"H" Output Current *1	I_{OH}	V_O = 2.4V	−0.1	–	–	mA
"H" Output Current *1	I_{OH}	V_O = 0.4V	–	–	−1.2	mA
Input Capacity	C_I	f = 1MHz, Ta = 25°C	–	5	–	pF
Output Capacity	C_O		–	7	–	pF
Current Consumption (STOP)	I_{DDS}	V_{DD} = 2V, no load Ta = 25°C	–	0.2	5	μA
		No load	–	1	100	μA
Current Consumption	I_{DD}	Crystal oscillation, No load, 4.194304MHz	–	6	12	mA

*1 Applied to P0, P1, P3, P4, and P5

*2 Applied to P2

*3 Applied to OSC_0

*4 Applied to RESET

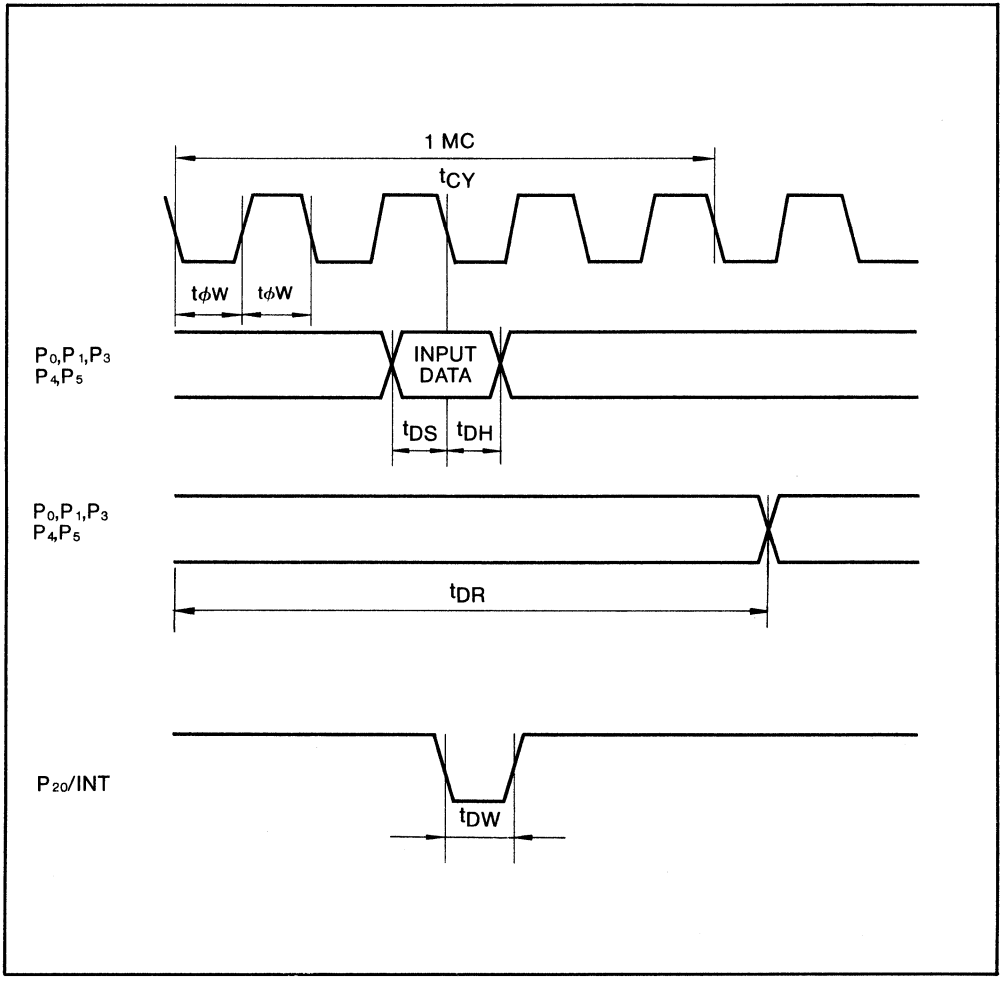
*5 Applied to OSC_1

*6 In using LED, total output current should be within the limit of Power dissipitation in "Absolute Maximum Rating."

AC CHARACTERISTICS

(V_{DD} = 5V±10%, Ta = −40 ~ +85°C)

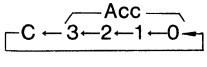
Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
Clock (OSC_0) Pulse Width	$t_{\phi W}$	–	119	–	–	ns
Cycle Time	t_{CY}	–	952	–	–	ns
Input Data Setup Time	t_{DS}	–	120	–	–	ns
Input Data Hold Time	t_{DH}	–	120	–	–	ns
Input Data/Input Clock Pulse Width	t_{DW}	–	120	–	–	ns
Data Delay Time	t_{DR}	C_L=15pF	–	–	t_{CY} +300	ns



INSTRUCTION SET

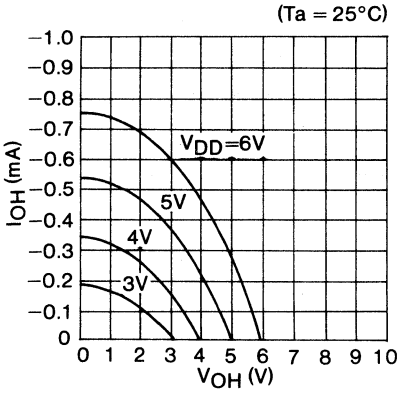
	Mnemonic	Hex op code	Byte	Cycle	Description
Load	LAI n	90 – 9F	1	1	Acc ← n
	LLI n	80 – 8F	1	1	L ← n
	LAL	21	1	1	Acc ← L
	LLA	2D	1	1	L ← Acc
	LAH	22	1	1	Acc ← H
	LHA	2E	1	1	H ← Acc
	LAM	38	1	1	Acc ← M
	LMA	2F	1	1	M ← Acc
	X	28	1	1	Acc ← M
	LMI nn	14 · nn	2	2	M(W) ← nn
	LHLI nn	15 · nn	2	2	HL ← nn
	LAMD mm	10 · mm	2	2	Acc ← Md
	LMAD mm	11 · mm	2	2	Md ← Acc
	IPO p	3D · pD	2	2	Acc ← Pp
	OPD p	3D · pC	2	2	Pp ← Acc
Control	MEI	3E · 60	2	2	MEIF ← "1"
	MDI	3E · 61	2	2	MEIF ← "0"
	EIEX	3D · C8	2	2	EIEXF ← "1"
	EITB	3D · C9	2	2	EITBF ← "1"
	DIEX	3D · C4	2	2	EIEXF ← "0"
	DITB	3D · C5	2	2	EITBF ← "0"
	TIEX	3D · C0	2	2	SKIP IF EIEXF="1"
	TITB	3D · C1	2	2	SKIP IF EITBF="1"
	TQEX	3D · 20	2	2	SKIP IF IRQEX="1"
	TQTB	3D · D0	2	2	SKIP IF IRQTB="1"
	RQEX	3D · 24	2	2	IRQEX ← "0"
	RQTB	3D · D4	2	2	IRQTB ← "0"
Increment/ decrement	INL	31	1	1	L ← L+1, SKIP IF L="0"
	INH	32	1	1	H ← H+1, SKIP IF H="0"
	INM	33	1	1	M ← M+1, SKIP IF M="0"
	DCL	35	1	1	L ← L-1, SKIP LF L="F"
	DCH	36	1	1	H ← H-1, SKIP IF H="F"
	DCM	37	1	1	M ← M-1, SKIP IF M="F"
	INMD mm	12 · mm	2	2	Md ← Md+1, SKIP IF Md="0"

INSTRUCTION SET (CONT.)

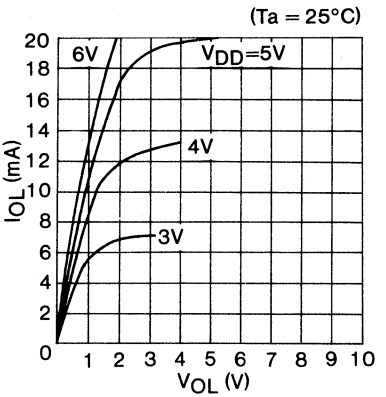
	Mnemonic	Hex op code	Byte	Cycle	Description
Bit handling	TAB n2	54 - 57	1	1	SKIP IF (Acc-Bit n2) = "1"
	TMB n2	58 -5B	1	1	SKIP IF (M-Bit n2) = "1"
	RMB n2	68 - 6B	1	1	(M-Bit n2) ← "0"
	SMB n2	78 - 7B	1	1	(M-Bit n2) ← "1"
	TPBD p n2	3D · p0~3	2	2	SKIP IF (Pp-Bit n2) = "1"
	RPBD p n2	3D · p4~7	2	2	(Pp-Bit n2) ← "0"
	SPBD p n2	3D · p8~B	2	2	(Pp-Bit n2) ← "1"
	TC	09	1	1	SKIP IF C = "1"
	RC	08	1	1	C ← "0"
	SC	07	1	1	C ← "1"
Arithmetic	ADS	02	1	1	Acc ← Acc+M, SKIP IF Cy="1"
	ADC	03	1	1	C, Acc ← C+Acc+M
	AIS n	3E · 4n	2	2	Acc ← Acc+n, SKIP IF Cy="1"
	DAS	0A	1	1	Acc ← Acc+10
	AND	0D	1	1	Acc ← Acc∧M
	OR	05	1	1	Acc ← Acc∨M
	EOR	04	1	1	Acc ← Acc∀M
	CMA	0B	1	1	Acc ← $\overline{\text{Acc}}$
	CAM	16	1	1	SKIP IF Acc=M
	CAI n	3E · 0n	2	2	SKIP IF Acc=n
	RAL	0E	1	1	
Branch	JCP a6	C0 - FF	1	1	PC ← a6
	JP a11	40 - 47 00 - FF	2	2	PC ← a11
	CAL a11	A0 - A7 00 - FF	2	4	STACK ← PC+2, PC←a11, SP←SP-1
	RT	1E	1	4	PC ← STACK, SP ← SP+1
Others	PUSH	1C	1	3	STACK ← PC+2, PC←a11, SP←SP-1
	POP	1D	1	3	C, Acc, H, L ← STACK, SP←SP+1
	STOP	3D · B9	2	2	CLOCK STOP
	NOP	00	1	1	NO OPERATION

6

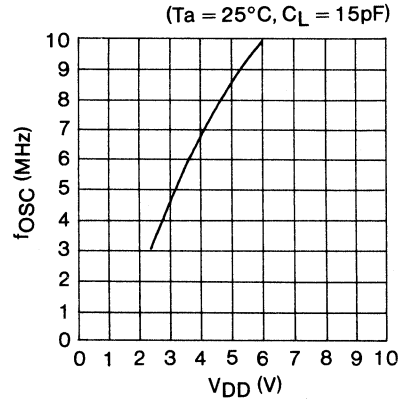
TYP. Current (I_{OH}) vs Voltage (V_{OH}) for High state Output



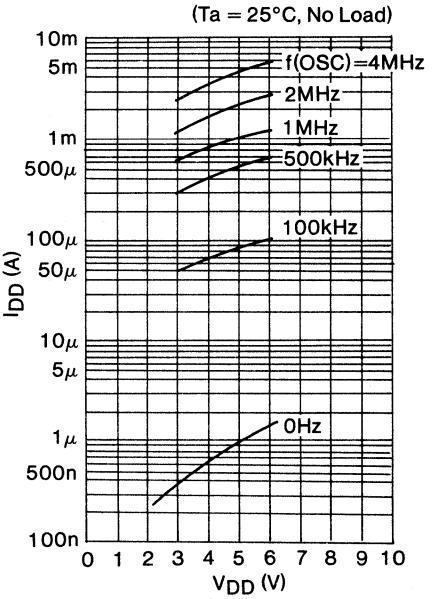
TYP. Current (I_{OL}) vs Voltage (V_{OL}) for Low state Output



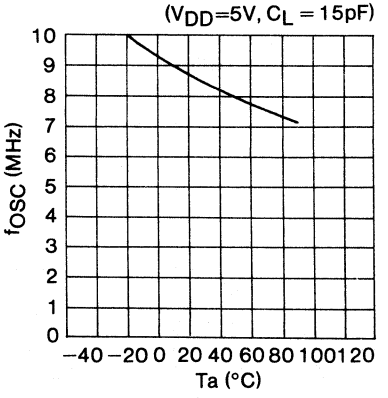
TYP. Maximum Oscillator Frequency f(OSC) vs Supply Voltage (V_D)



TYP. Supply Current (I_{DD}) vs Supply Voltage (V_{DD})



TYP. Maximum Oscillator Frequency f(OSC) vs Temperature (Ta)



MSM6442

CMOS 4-BIT SINGLE CHIP MICROCONTROLLER WITH LCD DRIVER

GENERAL DESCRIPTION

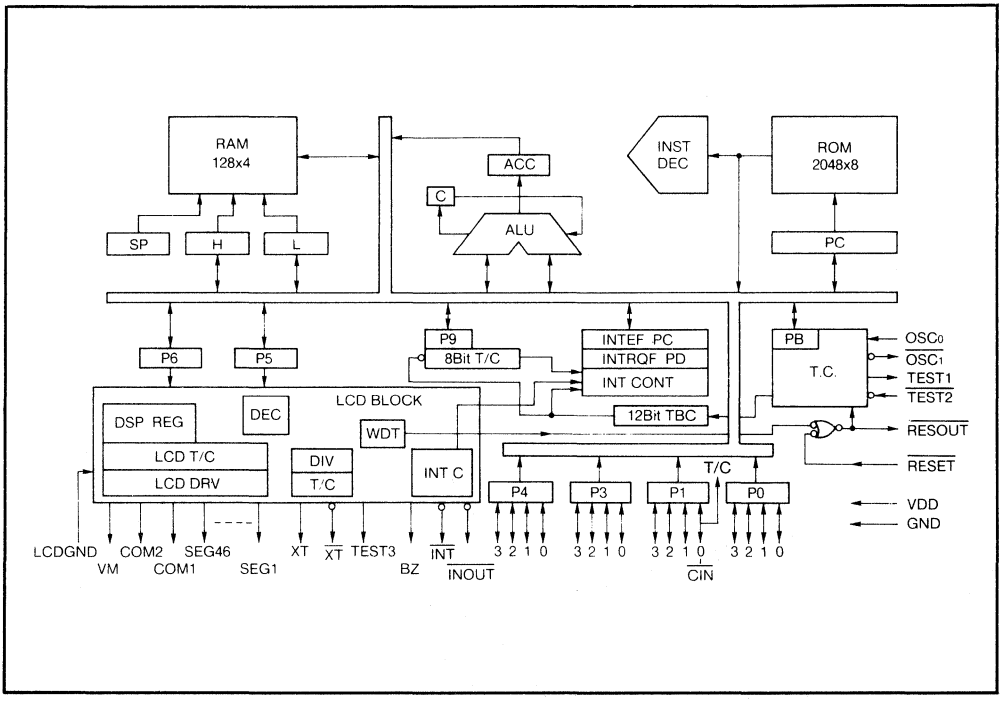
The OKI MSM6442 is a low power, high performance single chip device implemented in complementary metal oxide semiconductor technology with 46 segment outputs and 2 commons. Also integrated onto this chip are 16K bits mask program ROM, 512 bits of data RAM, 28 Input/Output lines and oscillator. 71 instructions include binary, BCD, logical operations; bit set, reset, test; subroutine call and return.

FEATURES

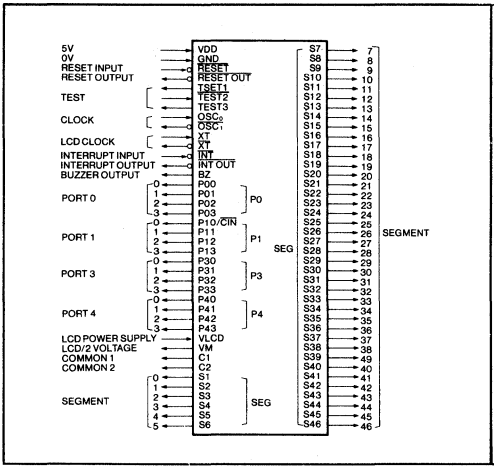
- Low Power Consumption 30mW (typ)
- 2048 × 8 Internal ROM
- 128 × 4 Internal RAM
- Two built-in counters
  12-bit time-base counter
  8-bit programable timer/event counter
- 16 Input/Output Ports and 46 LCD Output Port and 2 Common Output (1/2 Duty, 1/2 Bias)
- LED direct drive available
- Self-contained Oscillator
- 71 Instructions
- 4 Interrupt Levels
- 16 Stack Levels
- −40 to +85°C Operating Temperature
- 4.5 to 5.5V Operating V_{DD} at 4.2 MHz
  3.0 to 6.0V Operating V_{DD} at 1 MHz
- TTL Compatible

6

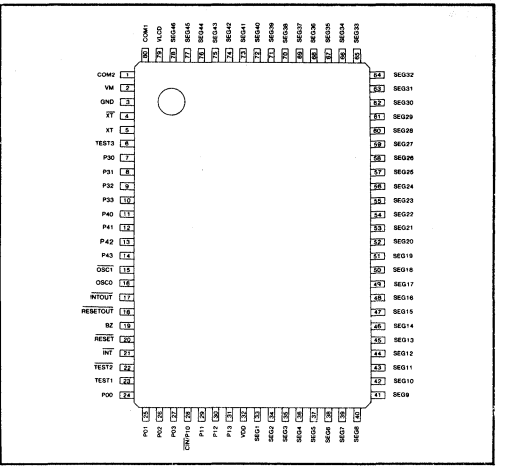
FUNCTIONAL BLOCK DIAGRAM



LOGIC SYMBOL



PIN CONFIGURATION (TOP VIEW)



PIN DESCRIPTION

Terminal	Input/ Output	Function	When reset
P00 ~ P03 P10 ~ P13 P30 ~ P33	Input/ Output	I/O port I/O port (P10 and count input $\overline{CIN}$ are in common) I/O port	"1"
P40 ~ P43	Input/ Output	I/O port	"0"
SEG1 ~ SEG16 SEG17 ~ SEG46	Output*	LCD output port (can be assigned to data output in 4 bit wide) LCD output port	"0"*
COM1 COM2	Output*	LCD common output terminal 1 LCD common output terminal 2	"0"*
$\overline{INT}$	Input	Input port of external interrupt	–
$\overline{INT\ OUT}$	Output	Interrupt output port	"1"
$\overline{RESET}$		Reset input port	–
$\overline{RESET\ OUT}$	Output	Reset output terminal	"1"
BZ	Output	Buzzer pulse output port in 2048 KHz	"0"*
$\overline{OSC_0}$ $\overline{OSC_1}$	Input/ Output	Crystal OSC or ceramic OSC connection Crystal OSC or ceramic OSC connection (System clock)	–
XT $\overline{XT}$	Input/ Output	32.768 kHz crystal oscillator connection (use for LCD control)	–
$\overline{TEST\ 1}$ $\overline{TEST\ 2}$ $\overline{TEST\ 3}$	– –	TEST terminal 1 (open) (Connected to V_{DD}) TEST terminal 2 (open) TEST terminal 3 (open)	
V_{DD}	Input	Power supply (5V)	–
V_{LCD} GND	Input	Power supply for LCD	–
VM	Input/ Output	$(V_{DD}-V_{LCD})/2$ supply voltage output or supply voltage input	"0"*
GND	Input	Power supply (0V)	–

*"0" indicates the LCD GND voltage level

ELECTRIC CHARACTERISTICS

• Absolute Maximum Ratings

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	Ta = 25°C	−0.3 ~ 7	V
Input Voltage	V_I		−0.3 ~ V_{DD}	V
Output Voltage	V_O		−0.3 ~ V_{DD}	V
LCD Voltage	LCDGND		V_{DD} − 9 ~ V_{DD}	V
Storage Temperature	Tstg	—	−55 ~ +150	°C

• Operating Conditions

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	f (OSC) ≦ 1MHz	3 ~ 6	V
		f (OSC) ≦ 4.2MHz	4.5 ~ 5.5	V
LCD Voltage	LCDGND	*1	V_{DD} − 8 ~ 0	V
Memory Retension Voltage	V_{DDH}	Oscillation off	2 ~ 6	V
Operating Temperature	Top	—	−40 ~ +85	°C
LCD Clock Oscillation Frequency	f(XT)	*2	32.768	kHz
Fan Out (I/O Port)	N	MOS Load	15	—
		TTL Load	1	—

*1 Voltage applied to LCD is ($_{DD}$−V_{LCD}).

*2 Oscillation Circuit for LCD Clock (XT, $\overline{XT}$ Port) is for Christal Oscillation only.

DC CHARACTERISTICS

(VDD = 5V ±10%, LDCGND = 0V, Ta = −40 ~ +85°C)

Parameter		Symbol	Condition	MIN	TYP	MAX	Unit
"H" Input Voltage	*1. $\overline{INT}$	VIH	—	2.4	—	VDD	V
	*3.		—	3.6	—	VDD	V
"L" Input Voltage	*1. *4	VIL	—	0	—	0.8	V
"H" Output Voltage	*1. $\overline{OSC1}$	VOH	IO = −15μA	4.2	—	—	V
	*2.		IO = −400μA	2.4	—	—	V
	SEG1~SEG46		IO = −10μA	V_{DD}-0.2	—	—	V
	COM1, COM2		IO = −50μA	V_{DD}-0.2	—	—	V
"L" Output Voltage	*1, *2	VOL	IO = 1.6mA	—	—	0.4	V
	$\overline{OSC1}$		IO = 15mA	—	—	0.4	V
	SEG1~SEG46		IO = 10μA	—	—	0.2	V
	COM1, COM2		IO = 50μA	—	—	0.2	V
"M" Output Voltage	COM1, COM2	VOM	IO ±0.5	VDD/2 0.2	—	VDD/2+0.2	V
"H" Input Current	OSC0	IIH	VI= VDD	—	—	15	μA
	XT			—	—	7	μA
	$\overline{INT}$, $\overline{RESET}$			—	—	1	μA
"L" Input Current	OSC0	IIL	VI = 0V	—	—	−15	μA
	XT			—	—	7	μA
	$\overline{INT}$, $\overline{RESET}$			—	—	−30	μA
"H" Output Current	*1	IOH	VO = 2.4V	−0.1	—	—	mA
			VO = 0.4V	—	—	−1.2	mA
Current Consumption —at stop mode —no oscillation		IDDS	V_{DD} = 2V, TA = 25°C No load Display off XT Port is fixed to "L"	—	0.2	10	μA
			Display off XT Port is fixed to "L"	—	1	100	μA
Current Consumption —at stop mode		IDDL	No load Display off At stop mode f(XT) = 32.768 KHz	—	100	200	μA
Current Consumption		IDD	No load Display off f(osc) = 4.2 MHz f(XT) = 32.768 KHz	—	6	12	mA

*1 Applied to P0, P1, P2, and P4.
*2 Applied to $\overline{INTOUT}$, $\overline{RESET}$, and BZ.
*3 Applied to OSC0, XT, and $\overline{RESET}$.
*4 Applied to XT, $\overline{INT}$, and $\overline{RESET}$.

Note: "M" output voltage is intermediate voltage of the output from common port at dynamic display.

AC CHARACTERISTICS

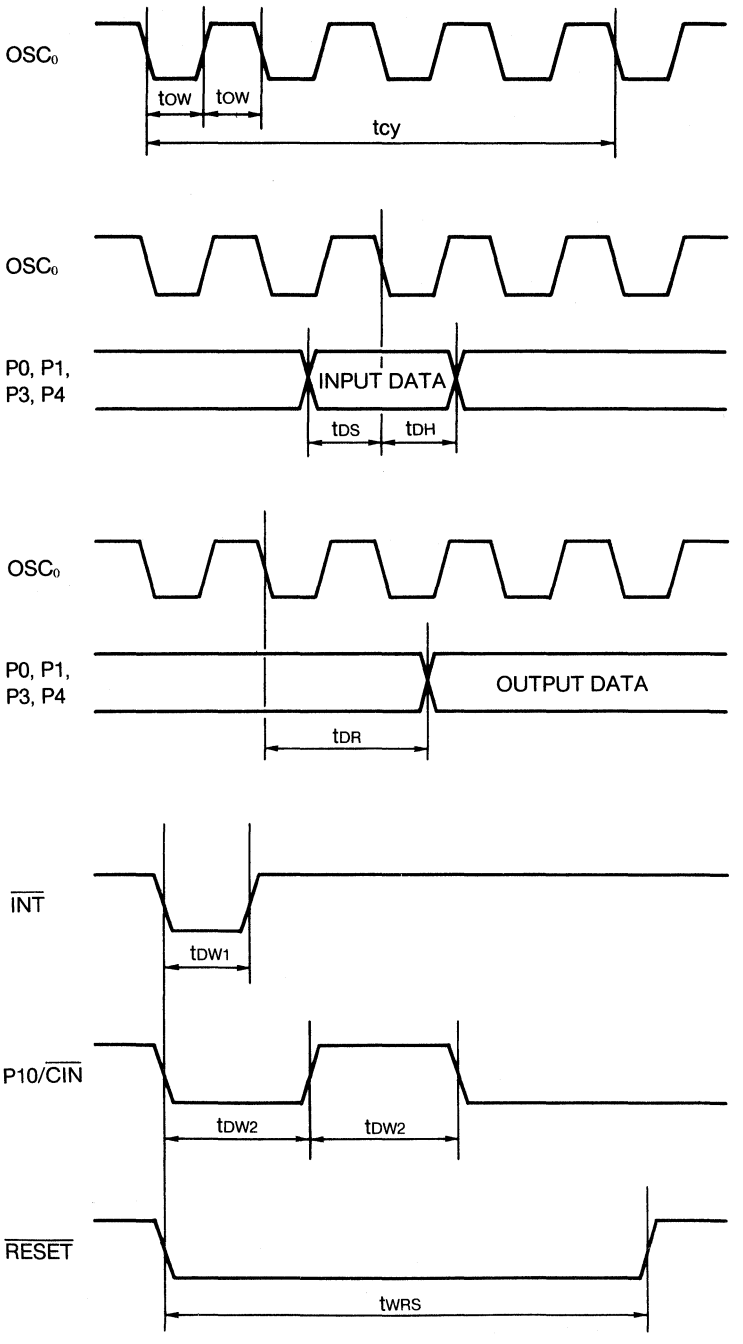
(VDD = 5V ±10%, Ta = −40 ~ +85°)

Parameter	Symbol	Condition	MIN	TYP	MAX	Unit
Clock (OSC_0) Pulse Width	t_{OW}	—	119	—	—	nS
Cycle Time	t_{CY}	—	952	—	—	nS
Input Data Set-up Time	t_{DS}	—	120	—	—	nS
Input Data Hold Time	t_{DH}	Note 1	120	—	—	nS
INT Input Data Pulse Width	t_{DW1}	—	120	—	—	nS
CT Clock Pulse Width	t_{DW2}	—	2/8tcy÷120	—	—	nS
Data Delay Time	t_{DR}	C_L = 15pF	—	—	300	nS
Reset Input Pulse Width	t_{WRS}	Note 2	2tcy	—	—	nS

Note 1* To release powerdown by inputting "L" level into INT Port, pulse width should be longer than the time for the oscillation stabilization at OSC_0.

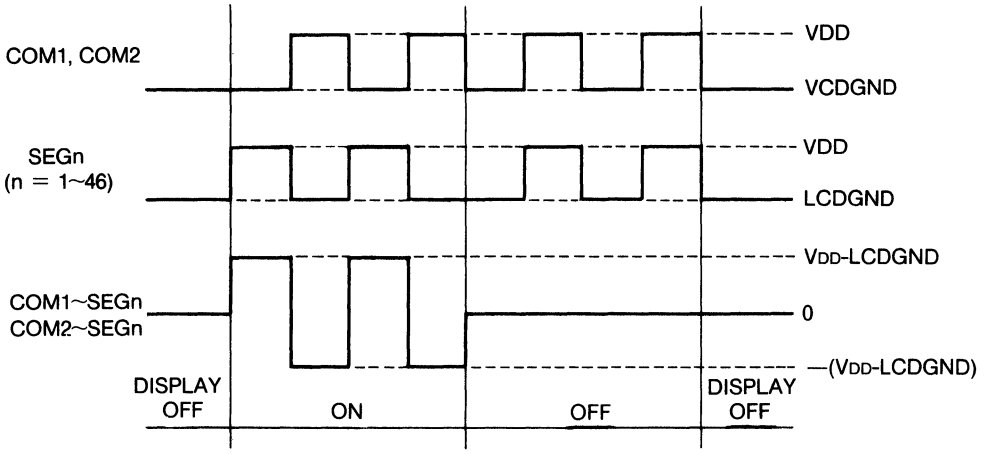
Note 2* The condition of stable oscillation. To release powerdown by reset, pulse width should be longer than the time for oscillation stabilization at OSC_0.

TIMING CHART

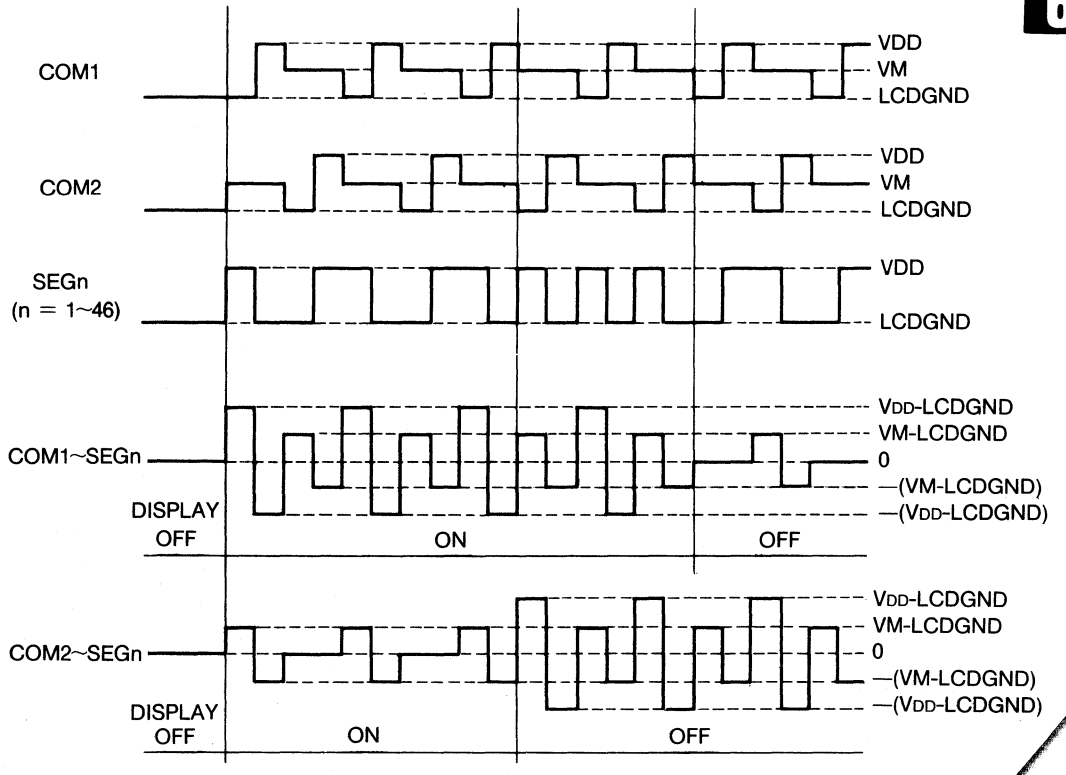


OUTPUT WAVE FORM OF LCD RIVER

STATIC MODE



DYNAMIC MODE



6

FUNCTIONAL DESCRIPTION

ROM

Organized into 2048 words by 8 bits, ROM is used to store developed application programs (instructions). It is addressed by the program counter (PC).

PROGRAM COUNTER (PC)

The program counter consists of a 11-bit binary counter. It is used to address ROM.

STACK

An interrupt or CAL instruction causes the contents of the PC to be saved in the stack. Also, the PUSH instruction causes the contents of accumulator, carry-flag, H- and L-register to be saved in it. These are allowed to be restored by the RT instruction or POP instruction.

RAM

Organized into 128 words of 4 bits, RAM is addressed by the H- and L-register or the contents of the second byte of an instruction.

L-REGISTER

A 4-bit register which specifies the row address of RAM and the port-address in the port operation instructions. It is also used as a working register.

H-REGISTER

A 4-bit register which specifies the column address of RAM and is used as a working register.

ALU

A 4-bit logic circuit which provides arithmetic and logical operations.

ACCUMULATOR (ACL)

Consisting of a 4-bit register, the accumulator holds the result of operations or the date present on ports.

C-FLAG

The flag that holds a carry generated from the result of operations.

INPUT/OUTPUT Ports (16 bits)

16 input/output ports are provided for effecting and controlling data transfer to and from an external source. The ports are selected by codes included in the instructions.

12-bit TIME-BASE COUNTER

The time base counter consists of a 12-bit binary counter. It is used to devide the frequency of the OSC_0 input by 2^{12} and generate the interrupt request at every over-flow signal.

8-bit TIMER EVENT COUNTER

The timer event counter consists of a 8-bit counter (8-bit) register, comparing and controlling circuits. It is used to count pulses of an internal or external source. Coincidently, if value between the counter and the register causes interrupt request occur.

LCD DRIVER

The LCD driver is used to effect LCD display by transferring data in a program to the register assigned as port 5 and 6. It is available to select driving in static or dynamic operation (1/2 duty cycle) and frame frequency (128 Hz/64 Hz) and to drive up to 92 segments at 1/2 duty. Also, 16 outputs(SEG1~SEG16)of the segment terminals can be used as normal data outputs.

A standard LCD clock is produced by the oscillation dividing a crystal oscillator (32.768 kHz) connected to XT and $\overline{XT}$ terminals. This is also used as standard clock of displaying, clock interrupting and watch dog timer. (This clock can be also produced by dividing a frequency of 4.194304 MHz. Note the selection of the frame frequency, when the crystal oscillator is used without a frequency of 4.194304 MHz.)

INTERRUPT

As shown below, 1 ~ 4 is available to interrupt;

1. External interrupt at the falling edge of INT signal input
2. Clock interrupt at every second (32.768 kHz crystal oscillator)
3. Time base counter interrupt at the occurance of an overflow of the timer base counter.
4. Timer event counter interrupt coinciding between the signals of the 8-bit counter and register.

Interrupts 1 and 2 are also used to release the power down mode.

WATCH DOG TIMER (WDT)

A timer for detecting the overrunning of the program. This timer produces the overflow signal by dividing the 64 Hz frequency by 4 generated from the oscillation of a frequency of 32.768 kHz. It can be also halted, when unused.

TIMING CONTROL (T.C)

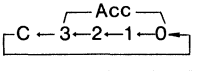
A 0 level on the $\overline{RESET}$ pin for longer than predetermined period initializes the internal circuitry and ports.

Clock pulses are supplied to the OSC_0 pin from an external source. A crystal or ceramic oscillator may be connected to OSC_0 and $\overline{OSC_1}$ to form an oscillator circuit to produce clock pulses.

INSTRUCTION SET

	Mnemonic	Hex op code	Byte	Cycle	Description
Load	LAI n	90 – 9F	1	1	Acc ← n
	LLI n	80 – 8F	1	1	L ← n
	LAL	21	1	1	Acc ← L
	LLA	2D	1	1	L ← Acc
	LAH	22	1	1	Acc ← H
	LHA	2E	1	1	H ← Acc
	LAM	38	1	1	Acc ← M
	LMA	2F	1	1	M ← Acc
	X	28	1	1	Acc ⟷ M
	LMI nn	14 · nn	2	2	M(W) ← nn
	LHLI nn	15 · nn	2	2	HL ← nn
	LAMD mm	10 · mm	2	2	Acc ← Md
	LMAD mm	11 · mm	2	2	Md ← Acc
	LMCT	3E · 59	2	2	M(W) ← CT
	LCTM	3E · 51	2	2	CT ← M(W)
	IPD p	3D · pD	2	2	Acc ← Pp
	OPD p	3D · pC	2	2	Pp ← Acc
Control	MEI	3E · 60	2	2	MEIF ← "1"
	MDI	3E · 61	2	2	MEIF ← "0"
	EIEX	3D · C8	2	2	EIEXF ← "1"
	EICT	3D · CB	2	2	EICTF ← "1"
	DIEX	3D · C4	2	2	EIEXF ← "0"
	DICT	3D · C7	2	2	EICTF ← "0"
	TIEX	3D · C0	2	2	SKIP IF EIEXF="1"
	TICT	3D · C3	2	2	SKIP IF EICTF="1"
	TQEX	3D · 20	2	2	SKIP IF IRQEX="1"
	TQCT	3D · D2	2	2	SKIP IF IRQCT="1"
	RQEX	3D · 24	2	2	IRQEX ← "0"
	RQCT	3D · D6	2	2	IRQCT ← "0"
Increment/ decrement	INL	31	1	1	L ← L+1, SKIP IF L="0"
	INH	32	1	1	H ← H+1, SKIP IF H="0"
	INM	33	1	1	M ← M+1, SKIP IF M="0"
	DCL	35	1	1	L ← L−1, SKIP IF L="F"
	DCH	36	1	1	H ← H−1, SKIP IF H="F"
	DCM	37	1	1	M ← M−1, SKIP IF M="M"
	INMD mm	12 · mm	2	2	Md ← Md+1, SKIP IF Md="0"

INSTRUCTION SET (CONT.)

	Mnemonic	Hex op code	Byte	Cycle	Description
Bit handling	TAB n2	54 - 57	1	1	SKIP IF (ACC−Bit n2) = "1"
	TPB n2	50 - 53	1	1	SKIP IF (P−Bit n2) = "1"
	RPB n2	60 - 63	1	1	(P−Bit n2) ← "0"
	SPB n2	70 - 73	1	1	(P−Bit n2) ← "1"
	TMB n2	58 -5B	1	1	SKIP IF (M−Bit n2) = "1"
	RMB n2	68 - 6B	1	1	(M−Bit n2) ← "0"
	SMB n2	78 - 7B	1	1	(M−Bit n2) ← "1"
	TPBD p n2	3D · p0∼3	2	2	SKIP IF (Pp−Bit n2) = "1"
	RPBD p n2	3D · p4∼7	2	2	(Pp−Bit n2) ← "0"
	SPBD p n2	3D · p8∼B	2	2	(Pp−Bit n2) ← "1"
	TC	09	1	1	SKIP IF C = "1"
	RC	08	1	1	C ← "0"
	SC	07	1	1	C ← "1"
Arithmetic	ADS	02	1	1	Acc ← Acc+M, SKIP IF Cy="1"
	ADC	03	1	1	C, Acc ← C+Acc+M
	AIS n	3E · 4n	2	2	Acc ← Acc+n, SKIP IF Cy="1"
	DAS	0A	1	1	Acc ← Acc+10
	AND	0D	1	1	Acc ← Acc∧M
	OR	05	1	1	Acc ← Acc∨M
	EOR	04	1	1	Acc ← Acc¥M
	CMA	0B	1	1	Acc ← Acc
	CAM	16	1	1	SKIP IF Acc=M
	CAI n	3E · 0n	2	2	SKIP IF Acc=n
	RAL	0E	1	1	
Branch	JCP a6	C0 - FF	1	1	PC ← a6
	JP a11	40 _ 47 00 ¯ FF	2	2	PC ← a11
	CAL a11	A0 _ A7 00 ¯ FF	2	4	STACK ← PC+2, PC←a11, SP←SP−1
	RT	1E	1	4	PC ← STACK, SP ← SP+1
Others	PUSH	1C	1	3	STACK ← C, Acc, H, L, SP←SP−1
	POP	1D	1	3	C, Acc, H, L ← STACK, SP←SP+1
	STOP	3D · B9	2	2	CLOCK STOP
	NOP	00	1	1	NO OPERATION
	ECT	3D·BB	2	2	CTF ← "1" (Counter Start)
	DCT	3D·B7	2	2	CTF ← "0" (Counter Stop)
	TCT	3D·B3	2	2	Skip if CTF = "1"

OKI semiconductor

MSC6458

OKI 4-BIT 1-CHIP MICROCONTROLLER

GENERAL DESCRIPTION

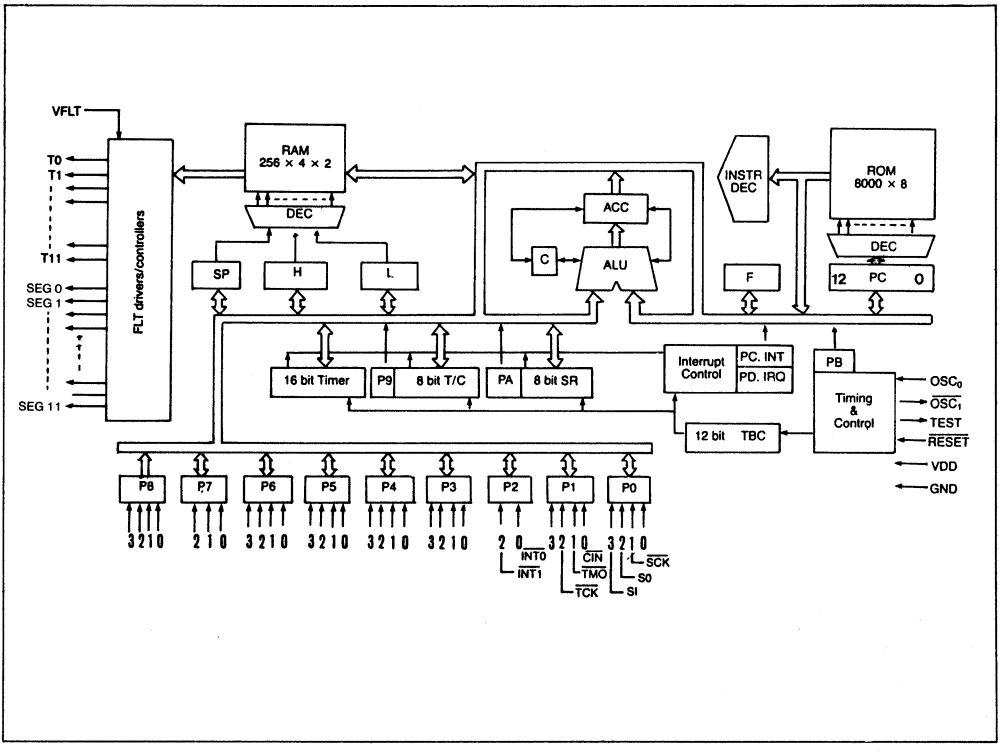
The MSC6458 is a high-speed, 4-bit 1-chip microcontroller with built-in FLT drivers/controllers developed to support relatively large control systems.

FEATURES

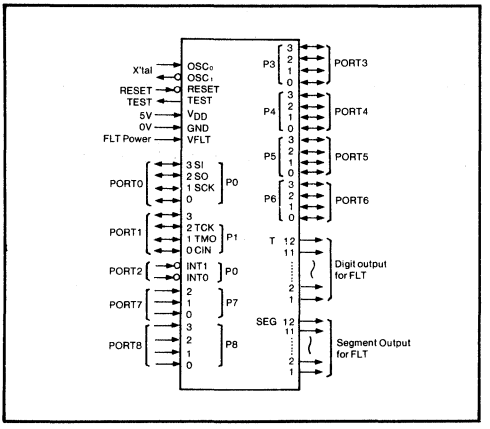
- ROM: 8000 × 8 bits
- RAM: 512 × 4 bits
- Ports: I/O 24 ports (8 having IOL = 20 mA)
  Input 9 (2 also serving as interrupt inputs)
- FLT drivers (Withstand voltage 40V): 12 (IOH = 20mA)
  12 (IOH = 6mA)
- LED direct drive available
- Interrupts: 7 lines (2 external, 5 internal)
- Built-in counters: 12 bits, timebase counter
  16 bits, programmable counter
  8 bits, high-speed programmable timer/event counter
- Serial I/O: Built-in 8-bit SIO register
- Oscillation circuit: Crystal or ceramic oscillation
- Number of instructions: 147
- Cycle time: 930 ns (4.3MHz)
- Operating ranges: 4.5 to 5.5V (4.3MHz)
  Voltage: 3.0 to 6.0V (1MHz)
  Temperature: −40 to +85°C
- Power dissipation (typical)
  (display off): 9mA (5V, 4.3MHz)
  2mA (3V, 1MHz)
- Power down: STOP instruction
- Package: 64-pin shrink DIP/64-pin FLAT

6

BLOCK DIAGRAM

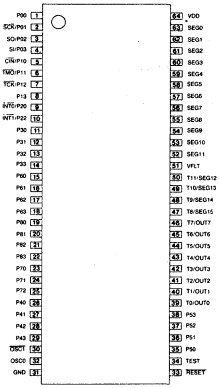


LOGIC SYMBOL



PIN CONFIGURATION (TOP VIEW)

64 PIN PLASTIC SHRINK DIP



PIN DESCRIPTION

Terminal	Input/ Output	Function	When reset
P00 P01/$\overline{SCK}$ P02/SO P03/SI	Input/ Output	I/O port I/O port (also used as serial clock input $\overline{SCK}$) I/O port (also used as serial data output SO) I/O port (also serial data input SI)	"1"
P10/$\overline{CIN}$ P11/$\overline{TMO}$ P12/$\overline{TCK}$ P13	Input/ Output	I/O port (also used as count input $\overline{CIN}$) I/O port (also used as timer output $\overline{TMO}$) I/O port (also used timer clock input $\overline{TCK}$) I/O port	"1"
P20/$\overline{INT0}$ P22/$\overline{INT1}$	Input	Input Port with Latch (falling edge sensitive) also used as interrupt input $\overline{INT0}$ Input Port with Latch ('0' level sensitive) also used as interrupt input $\overline{INT1}$	–
P30 ~ P33	Input/ Output	I/O port	"1"
P60 ~ P63	Input/ Output	I/O port	"0"
P40 ~ P43 P50 ~ P53	Output/ Input	I/O port (I_{OL}=20mA MAX)	"0"
P70 ~ P72 P80 ~ P83	Input	Input port with pull down register Pull down register of P70 ~ P72 can be removed by instruction	
SEG0 ~ SEG11	Output	FLT segment driver (dynamic)	"0"
T11/SEG12 ~ T8/SEG15	Output	FLT segment driver (dynamic)/Timing output	"0"
T7/OUT7 ~ T0/OUT0	Output	FLT segment driver (static)/Timing output	"0"
$\overline{OSC0}$ $\overline{OSC1}$	Input/ Output	Crystal connection terminal for system clock oscillation	–
$\overline{RESET}$	Input	System reset input	–
TEST	Output	Test pin (Open)	–
VFLT	Input	Power supply for FLT driving	–
VDD GND	Input	System Power Supply	–

FUNCTIONAL DESCRIPTION

1. ROM

The ROM, organized in 8 bits, has a maximum capacity of 8000 bytes.

2. RAM

The RAM is organized in 4 bits per word, with a capacity of 512 words.

It is separated into two banks each 256 words long. Bank selection is accomplished via internal ports. The RAM location in the banks is addressed by the H and L registers or by the second byte of each instruction.

3. Ports (24 I/O, 7 input)

The 24 pseudo-bidirectional I/O ports effect or control the exchange of data with external sources. The ports are specified by the L register or by codes contained in instructions. Ports 4 and 5 may draw IOL up to 20mA.

The seven input ports have built-in pulldown resistors. Up to 84 keys can be scanned by assembling them in key matrices with the timing outputs of the FLT drivers (with 12 segments × 12 timings on display; also during automatic display).

4. Interrupt Input Pins (2 terminals)

The $\overline{INT0}$/P20 and $\overline{INT1}$/P22 pins are interrupt input pins. External interrupt request flags of $\overline{INT0}$/P20 pin and $\overline{INT1}$/P22 pin can be set by using interrupt input pins:

$\overline{INT0}$/P20 pin ... positive edge or negative edge input.

$\overline{INT1}$/P22 pin ... "0" level input.

These flags are automatically reset when the appropriate external interrupts occur. These pins are available for use as input ports when not used as interrupt input pins.

5. FLT Drivers/Controllers (Automatic Display)

The FLT drivers have a withstand voltage of 40V in the positive direction from the GND level. They comprise 12 ports that can draw 20mA as IOH (Timing outputs) and 12 ports that can draw 6mA as such (Segment outputs).

A choice of four display modes is supported as listed below. A display RAM area is allocated as part of the RAM space. Data is automatically displayed when transferred to the display RAM. (Two different display frequencies are selectable.) Static output data can be displayed by controlling the FLT drivers by programming.

Display modes (@4.194304 MHz)

(1) 12 Segments × 12 Timings
1/12 duty (85.3/341.3 Hz)

(2) 16 Segments × 8 Timings
1/8 duty (128/512 Hz)

(3) 16 Segments × 4 Timings +4 output*
1/4 duty (256/1024Hz)

(4) 16 Segments+8 output*
Program controlled

*output: static outputs

6. Stack (STACK) and Stack Pointer (SP)

The PC is saved in the stack when an interrupt occurs or a CAL instruction is executed. It is recovered by the execution of an RT instruction.

One fourth of the RAM space (128 words maximum, 32 levels) is available as a stack area. A 4-word RAM area is used for "one" level in the stack.

The stack pointer is an 8-bit up-down counter (the MSB and 2 bits from LSB being fixed at '1') indicating the next stack address to use. It enables the RAM space to be used as a pushdown stack. Data can also be transferred between stack pointer and the H/L registers.

7. Interrupts

Seven interrupt lines are provided for eight sources and eight levels of interrupts as follows (two external inputs):

(1) Display interrupt
Update to timing signals (positive edge)

(2) External interrupt1
Negative edge on the $\overline{INT0}$/P20 pin

(3) External interrupt2
Positive edge on the $\overline{INT0}$/P20 pin

(4) External interrupt3
'0' input on the $\overline{INT1}$/P22 pin

(5) Timebase interrupt
12-Bit timebase counter overflow

(6) Timer interrupt
16-Bit timer and timer register matched signal

(7) Counter interrupt
8-Bit counter and counter register matched signal

(8) Serial/O interrupt
8-Bit shift register shift end signal

8. 12-Bit Timebase Counter

The timebase counter is made up of a 12-bit binary counter. It generates an interrupt request every time it overflows as a result of dividing the OSC0 input 2^{12}.

9. 16-Bit Programmable Timer/Event Counter

Comprising a 16-bit register, a 16-bit binary counter, a comparator circiut, and a control circuit, the programmable timer generates an interrupt request when the register and counter values are matched.

10. 8-Bit High-Speed Programmable Timmer/Event Counter

The high-speed programmable timer/event counter comprises an 8-bit register, an 8-bit binary counter, a comparator circuit, and a control circuit. Starting and stopping the counter can be controlled by instructions. It generates an interrupt request when the register and counter values are matched.

11. 8-Bit Serial I/O

Serial I/O consists of an 8-bit shift register, a 3-bit shift counter, and a control circuit. It is used for serial data input and output. Serial data input and output takes place synchronized with a shift clock, which is selectable between internal and external clocks. The shift counter automatically terminates a data transfer on counting eight shift clock pulses and generates an interrupt request.

12. Registers (Acc, H, L, F)

The accumulator (Acc) is a 4-bit register used to perform data transfers or calculations with the RAM, other registers, ports and so on.

The H and L registers are each a 4-bit register. They transfer data to and from Acc and SP (stack pointer) and address the RAM. The L register is also used to specify ports to use.

The F register is made up of four independent flip-flops. It can be used as a program "flag" or general-purpose register because each of these flip-flops permits set/reset testing and transferring 4-bit parallel data to and from Acc by instructions.

13. Timing Control (TC)

A '0' input on the $\overline{\text{RESET}}$ pin for a certain period initializes internal circuitry and ports.

As the input side of clock pulses, the OSC0 pin accepts clock pulses from an external source. Clock pulses may also be obtained by configuring an oscillation circuit with a crystal oscillator or ceramic resonator connected to OSC0 and $\overline{\text{OSC1}}$.

Load Instructions, etc.

Mnemonic		Code	Bytes	Cycles	Description
LAI	n	90–9F	1	1	A ← n
LLI	n	80–8F	1	1	L ← n
LHI	n	3E · 7n	2	2	H ← n
LHLI	nn	15 · nn	2	2	HL ← nn
LMI	nn	14 · nn	2	2	M (w) ← nn
LAL		21	1	1	A ← L
LLA		2D	1	1	L ← A
LAH		22	1	1	A ← H
LHA		2E	1	1	H ← A
LAM		38	1	1	A ← M
LMA		2F	1	1	M ← A
LAM+		24	1	1	A ← M, L ← L+1, Skip if L = "0"
LAM−		25	1	1	A ← M, L ← L−1, Skip if L = "F"
LMA+		26	1	1	M ← A, L ← L+1, Skip if L = "0"
LMA−		27	1	1	M ← A, L ← L−1, Skip if L = "F"
LAMM	n2	39–3B	1	1	A ← M, H ← H ⩛ n2
LAMD	mm	10 · mm	2	2	A ← Md
LMAD	mm	11 · mm	2	2	Md ← A
X		28	1	1	A ⟷ M
X+		3C	1	1	A ⟷ M, L ← L+1, Skip if L = "0"
X−		2C	1	1	A ⟷ M, L ← L−1, Skip if L = "F"
XM	n2	29–2B	1	1	A ⟷ M, H ← H ⩛ n2
LMT	mm	19 · mm	2	4	M (w) ← T (Md (w), A)
LAF		3E · 54	2	2	A ← F
LFA		3E · 5C	2	2	F ← A
LHLS		3E · 53	2	2	HL ← SP
LSHL		3E · 5B	2	2	SP ← HL
IP		20	1	1	A ← P
OP		23	1	1	P ← A
IPD	p	3D · pD	2	2	A ← Pp
OPD	p	3D · pC	2	2	Pp ← A
OPT		18	1	3	P4, P5 ← T (M (w), A)

Interrupt Control Instructions

Mnemonic	Code	Bytes	Cycles	Description
MEI	3E · 60	2	2	MEIF ← "1"
MDI	3E · 61	2	2	MEIF ← "0"
EIXD	3D · E8	2	2	EIXDF ← "1"
EIXU	3D · E9	2	2	EIXUF ← "1"
EIXL	3D · EA	2	2	EIXLF ← "1"
EIDP	3D · EB	2	2	EIDPF ← "1"
EITB	3D · D8	2	2	EITBF ← "1"
EITM	3D · D9	2	2	EITMF ← "1"
EICT	3D · DA	2	2	EICTF ← "1"
EISR	3D · DB	2	2	EISRF ← "1"
DIXD	3D · E4	2	2	EIXDF ← "0"
DIXU	3D · E5	2	2	EIXUF ← "0"
DIXL	3D · E6	2	2	EIXLF ← "0"
DIDP	3D · E7	2	2	EIDPF ← "0"
DITB	3D · D4	2	2	EITBF ← "0"
DITM	3D · D5	2	2	EITMF ← "0"
DICT	3D · D6	2	2	EICTF ← "0"
DISR	3D · D7	2	2	EISRF ← "0"
TIXD	3D · E0	2	2	Skip if EIXDF = "1"
TIXU	3D · E1	2	2	Skip if EIXUF = "1"
TIXL	3D · E2	2	2	Skip if EIXLF = "1"
TIDP	3D · E3	2	2	Skip if EIDPF = "1"
TITB	3D · D0	2	2	Skip if EITBF = "1"
TITM	3D · D1	2	2	Skip if EITMF = "1"
TICT	3D · D2	2	2	Skip if EICTF = "1"
TISR	3D · D3	2	2	Skip if EISRF = "1"
TQXD	3D · 20	2	2	Skip if IRQXDF = "1"
TQXU	3D · 21	2	2	Skip if IRQXUF = "1"
TQXL	3D · 22	2	2	Skip if IRQXLF = "1"
TQDP	3D · 23	2	2	Skip if IRQDPF = "1"
TQTB	3D · C0	2	2	Skip if IRQTBF = "1"
TQTM	3D · C1	2	2	Skip if IRQTMF = "1"
TQCT	3D · C2	2	2	Skip if IRQCTF = "1"
TQSR	3D · C3	2	2	Skip if IRQSRF = "1"
RQXD	3D · 24	2	2	IRQXDF ← "0"
RQXU	3D · 25	2	2	IRQXUF ← "0"
RQXL	3D · 26	2	2	IRQXLF ← "0"
RQDP	3D · 27	2	2	IRQDPF ← "0"
RQTB	3D · C4	2	2	IRQTBF ← "0"
RQTM	3D · C5	2	2	IRQTMF ← "0"
RQCT	3D · C6	2	2	IRQCTF ← "0"
RQSR	3D · C7	2	2	IRQSRF ← "0"

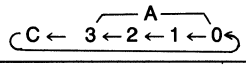
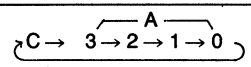
Increment/Decrement Instructions

Mnemonic		Code	Bytes	Cycles	Description
INA		30	1	1	A ← A+1, Skip if A = "0"
INL		31	1	1	L ← L+1, Skip if L = "0"
INH		32	1	1	H ← H+1, Skip if H = "0"
INM		33	1	1	M ← M+1, Skip if M = "0"
DCA		34	1	1	A ← A−1, Skip if A = "F"
DCL		35	1	1	L ← L−1, Skip if L = "F"
DCH		36	1	1	H ← H−1, Skip if H = "F"
DCM		37	1	1	M ← M−1, Skip if M = "F"
INMD	mm	12 · mm	2	2	Md ← Md+1, Skip if Md = "0"
DCMD	mm	13 · mm	2	2	Md ← Md−1, Skip if Md = "F"

Bit Handling Instructions, etc.

Mnemonic		Code	Bytes	Cycles	Description
TAB	n2	54–57	1	1	Skip if A (n2) = "1"
RAB	n2	64–67	1	1	A (n2) ← "0"
SAB	n2	74–77	1	1	A (n2) ← "1"
TPB	n2	50–53	1	1	Skip if P (n2) = "1"
RPB	n2	60–63	1	1	P (n2) ← "0"
SPB	n2	70–73	1	1	P (n2) ← "1"
TMB	n2	58–5B	1	1	Skip if M (n2) = "1"
RMB	n2	68–6B	1	1	M (n2) ← "0"
SMB	n2	78–7B	1	1	M (n2) ← "1"
TFB	n2	5C–5F	1	1	Skip if F (n2) = "1"
RFB	n2	6C–6F	1	1	F (n2) ← "0"
SFB	n2	7C–7F	1	1	F (n2) ← "1"
TPBD	p, n2	3D · p0~3	2	2	Skip if Pp (n2) = "1"
RPBD	p, n2	3D · p4~7	2	2	Pp (n2) ← "0"
SPBD	p, n2	3D · p8~B	2	2	Pp (n2) ← "1"
TC		09	1	1	Skip if C = "1"
RC		08	1	1	C ← "0"
SC		07	1	1	C ← "1"

Arithmetic Instructions

Mnemonic		Code	Bytes	Cycles	Description
ADCS		01	1	1	C, A ← C+A+M, Skip if C = "1"
ADS		02	1	1	A ← A+M, Skip if Cy = "1"
ADC		03	1	1	C, A ← C+A+M
AIS	n	3E · 4n	2	2	A ← A+n, Skip if Cy = "1"
DAA		06	1	1	A ← A+6
DAS		0A	1	1	A ← A+10
AND		0D	1	1	A ← A ∧ M
OR		05	1	1	A ← A ∨ M
EOR		04	1	1	A ← A ⊻ M
CMA		0B	1	1	A ← $\overline{A}$
CIA		0C	1	1	A ← $\overline{A}$+1
RAL		0E	1	1	
RAR		0F	1	1	
CAM		16	1	1	Skip if A = M
CAI	n	3E · 0n	2	2	Skip if A = n
CMI	n	3E · 1n	2	2	Skip if M = n
CLI	n	3E · 2n	2	2	Skip if L = n
CPI	p, n	17 · pn	2	2	Skip if Pp = n

Branch Instructions, etc.

Mnemonic		Code	Bytes	Cycles	Description
JCP	a6	C0–FF	1	1	PC ← a6
JA		1A	1	2	PC ← (PC ← A) +1
JM		1B	1	2	PC ← (M (w), A)
JP	a12	40–4F 00–FF	2	2	PC ← a12
CAL	a12	A0–AF 00–FF	2	4	ST ← PC+2, PC ← a12, SP ← SP–4
CZP	a	Ba	1	4	ST ← PC+1, PC ← 2a, SP ← SP–4
LJP	a13	3F 3F 00–1F 00 FF	3	4	PC ← a13
LCAL	a13	3F 3F 80–9F 00 FF	3	4	ST ← PC+3, PC ← a13, SP ← SP–4
RT		1E	1	4	PC ← ST, SP ← SP+4
RTS		1F	1	4	PC ← ST, SP ← SP+4, then Skip

Counter Control Instructions, etc.

Mnemonic	Code	Bytes	Cycles	Description
LCTM	3E · 51	2	2	CTR ← M (w)
LMCT	3E · 59	2	2	M (w) ← CT
ECT	3D · BB	2	2	CTF ← "1" (Counter Start)
DCT	3D · B7	2	2	CTF ← "0" (Counter Stop)
TCT	3D · B3	2	2	Skip if CTF = "1"
LTMM	3E · 50	2	3	TMR ← M (2w)
LMTM	3E · 58	2	3	M (2w) ← TM
LSRM	3E · 52	2	2	SR ← M (w), SC ← "0" SC: Shift Counter
LMSR	3E · 5A	2	2	M (w) ← SR
ESR	3D · BA	2	2	SRF ← "1" (Shift Register Start)
DSR	3D · B6	2	2	SRF ← "0" (Shift Register Stop)
TSR	3D · B2	2	2	Skip if SRF = "1"

CPU Control Instructions, etc.

Mnemonic	Code	Bytes	Cycles	Description
PUSH	1C	1	3	ST ← C, A, H, L, SP ← SP−4
POP	1D	1	3	C, A, H, L ← ST, SP ← SP+4
HALT	3D · B8	2	2	Halt CPU
STOP	3D · B9	2	2	Stop CPU
NOP	00	1	1	No Operation

Explanations of Instruction Symbols

A	: Accumulator (4-bit)
H	: H register (4-bit)
L	: L register (4-bit)
F	: F register (4-bit)
M	: RAM word addressed by the H and L registers
Md	: RAM word addressed by second byte of an instruction code
M (w)	: Two RAM words addressed by the H and L register/H3-0 and L3-1 (8-bit)
Md (w)	: Two RAM words addressed by second byte of an instruction code (8-bit)
M (2w)	: Four RAM words addressed by the H and L register/H3-0 and L3-2 (16-bit)
ST	: Four RAM words (16-bit) allocated as a stack area
SP	: Stack pointer (8-bit)
PC	: Program counter
P	: Port specified by the L register (4-bit)
Pp	: Port specified by 4 high-order bits of second byte of an instruction code (4-bit)
CTR	: 8-Bit counter/register
CT	: 8-Bit programmable counter
CTF	: Programmable counter start flag
TMR	: 16-Bit timer/register
TM	: 16-Bit programmable timer
SR	: 8-Bit shift register
SRF	: Shift register start flag
(X, Y)	: ROM address data specified by a11-4 as X and a3-0 as Y (12-bit)
T (X, Y)	: ROM table data specified by a11-4 as X and a3-0 as Y (8-bit)
n	: Immediate data (4-bit)
nn	: Immediate data (8-bit)
n2	: Two low-order bits of an instruction code
(n2)	: Bit specified by the two low-order bits of an instruction code
a	: ROM address data
aX	: ROM address data (X-bit)
mm	: RAM address data (8-bit)
C	: Carry flag
Cy	: Flag indicating a carry in a calculation result

ELECTRIC CHARACTERISTICS

• Absolute Maximum Ratings

Parameter	Symbol	Conditions		Limits	Unit
Supply Voltage	V_{DD}			−0.3 ~ 7	V
Indicated Supply Voltage	V_{FLT}	Ta = 25°C		V_{DD} ~ 45	V
Input Voltage	V_I			−0.3 ~ V_{DD}	V
Input Voltage	V_O	Ta = 25°C	Input/output	−0.3 ~ V_{DD}	V
			Indicated output	−0.3 ~ V_{FLT}	V
"H" Output Current (Indicated Output)	I_{OH}	Per pin	SEG0 ~ SEG1	10	mA
			T0 ~ T11	40	mA
			OUT0 ~ OUT7	* 30	mA
		Output terminal total	SEG0 ~ SEG11	72	mA
			T0 ~ T11	72	mA
"L" Output Current (P4, P5)	I_{OL}	Per terminal		20	mA
		P4 total		40	mA
		P5 total		40	mA
Power Dissipation	P_D	Per package		600	mW
		Per input/output terminal		50	mW
Storage Temperature	Tstg	—		−55 ~ +150	°C

* When timing output is used as static output

• Operating Conditions

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	f (osc) ≦ 4.3MHz	4.5 ~ 5.5	V
		f (osc) ≦ 1MHz	3 ~ 6	V
Indicated Supply Voltage	V_{FLT}	—	10 ~ 40	V
Memory Retension Voltage	V_{DDH}	Oscillation off	2 ~ 6	V
Operating Temperature	Topr	—	−40 ~ +85	°C
(Fan Out (Input/Output Port)	N	MOS Load	15	—
		TTL Load	1	—

• DC Characteristics

(V_{DD} = 5V ±10%, Ta = −40 ~ +85°C)

Parameter	Terminal applied	Symbol	Conditions	Min.	Typ.	Max.	Unit
"H" Input Voltage	*1	V_{IH}	—	2.4	—	V_{DD}	V
	OSC0, RESET		—	3.8	—	V_{DD}	V
	P7, P8		—	3.4	—	V_{DD}	V
"L" Input Voltage	*2	V_{IL}	—	0	—	0.8	V
	P7, P8		—	0	—	1.6	V
"H" Output Voltage	*3	V_{OH}	IO = −15μA	4.2	—	—	V
	SEG0 ~ SEG11		IO = −6mA	V_{FLT}-2.5	—	—	V
	T0 ~ T11		IO = −20mA	V_{FLT}-3.5	—	—	V
"L" Output Voltage	P0, P1, P3, P6	V_{OL}	IO = 1.6mA		—	0.4	V
	P4, P5		IO = 10mA		—	0.8	V
	OSC1		IO = 15μA		—	0.4	V
	SEG0 ~ SEG11		IO = 1mA		—	1.6	V
	T0 ~ T11		IO = 1mA		—	1.4	V
"H" Input Current	OSC0	I_{IH}	VI = VDD		—	15	μA
	P2, RESET				—	1	μA
	P7(P73=0), P8				—	60	μA
	P7(P73=1)				—	1	μA

Parameter	Terminal applied	Symbol	Conditions	Min.	Typ.	Max.	Unit
"L" Input Current	OSC0	I_{IL}	VI = 0V	–	–	–15	μA
	P2, $\overline{RESET}$			–	–	–30	μA
	P7, P8			–	–	–1	μA
"H" Output Current	P0, P1, P3, P4, P5, P6	I_{OH}	VO = 2.4V	–0.1	–	–	mA
			VO = 0.4V	–	–	–1.2	mA
Current Consumption		I_{DD}	No load f (osc) = 4.3MHz	–	12	20	mA
Current Consumption (When stop mode condition)		I_{DDS}	No load	–	1	100	μA
			No load V_{DD} = 2V Ta = 25°C	–	0.5	10	μA
Current Consumption (FLT driver section)		I_{FLT}	No load All FLT driver, "L" level	–	2	100	μA

*1. Applied to P0, P1, P2, P3, P4, P5, P6
*2. Applied to P0, P1, P2, P3, P4, P5, P6, OSC0, RESET
*3. Applied to P0, P1, P3, P4, P5, P6, OSC1

• AC Characteristics

(V_{DD} = 5V ±10%, Ta = 40 ~ +85°C)

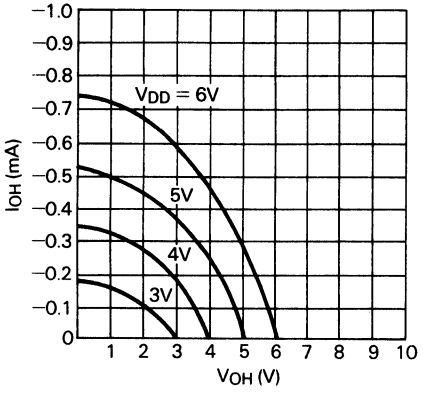
Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Clock (O.S.C $_0$) Pulse Width	$t_{\phi W}$	–	116	–	–	nS
Cycle Time	t_{CY}	–	928	–	–	nS
Input Data Setup Time	t_{DS}	–	120	–	–	nS
Input Data Hold Time	t_{DH}	–	120	–	–	nS
P2 Input Data Pulse Width	t_{DWP2}	Note 1	120	–	–	nS
SR Clock. Pulse Width	t_{DW1}	–	120	–	-	nS
CT Clock. Pulse Width	t_{DW2}	–	$2/8t_{CY}+120$	–	–	nS
TM Clock. Pulse Width	t_{DW3}	–	$t_{CY}+120$	–	–	nS
SR Data Setup Time	t_{SS}	–	120	–	–	nS
SR Data Hold Time	t_{SH}	–	120	–	–	nS
SR Clock Invalid Time *	t_{SINH}	–	2/8tcy	–	–	nS
Data Delay Time	t_{DR}	C_L = 15pF	–	–	300	nS
SR Clock Delay Time	t_{SP}	C_L = 15pF	–	–	360	nS
Reset Input. Rise Time	t_{WRS}	Note 2	2tcy	–	–	nS
Segment Output. Rise Time	t_{TLHS}	VFLT = 40V	–	–	3	μSS
Segment Output. Rise Time	t_{THLS}	C_{LD} = 15pF	–	–	1	μS
Timing Output. Rise Time	t_{TLHT}	VFLT = 40V	–	–	3	μS
Timing Output. Rise Time	t_{THLT}	C_{LD} = 15pF	–	–	1	μS

*1. When stop mode is to be released by "L" level input from P20/INT0, it is necessary to keep the pulse width of more than oscillation stability time for OSC_0.
*2. This indicates when OSC_0 oscillation is stabilized. However, when stop mode is released by reset input, the pulse width of more than OSC_0 oscillation stability time as requested.
*3. t_{SINH}: When shift register commands LMSR during shift in operation, its inner part will not change if clock, which inputs P01/SCK during t_{SINH} period, changes.

STANDARD CHARACTERISTICS

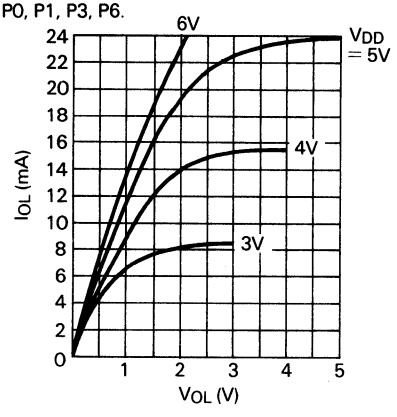
- **"H" Output Current I_{OH} – Output Voltage V_{OH} Characteristics (Ta = 25°C)**

P0, P1, P3, P4, P5, P6

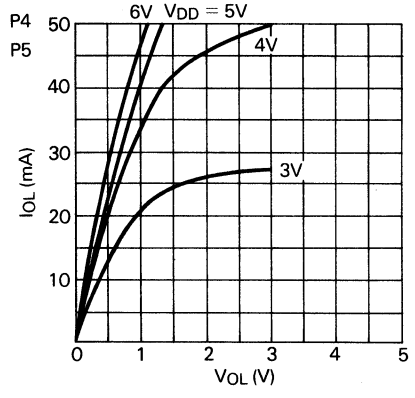


- **"L" Output Current I_{OL}– Output Voltage V_{OL} Characteristics (Ta = 25°C)**

P0, P1, P3, P6.

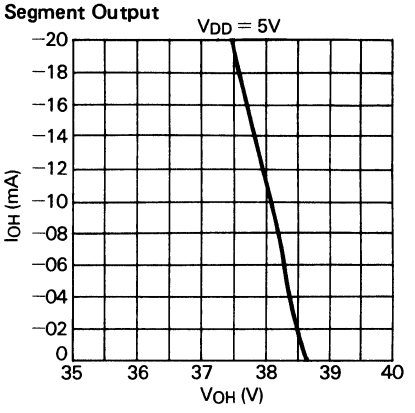


- **"L" Output Current I_{OL}– Output Voltage V_{OL} Characteristics (Ta = 25°)**



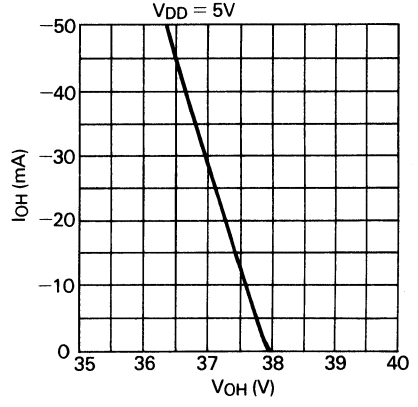
- **"H" Output Current I_{OH}– Output Voltage V_{OH} Characteristics (Ta = 25°C, VFLT = 40V)**

Segment Output



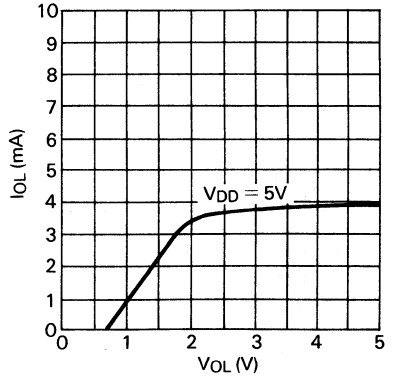
- **"L" Output Current I_{OH}– Output Voltage V_{OL} Characteristics (Ta = 25°C, VFLT = 40V)**

Timing Output



- **"L" Output Current I_{OL}– Output Current V_{OL} Characteristics (Ta = 25°C, VFLT = 40V)**

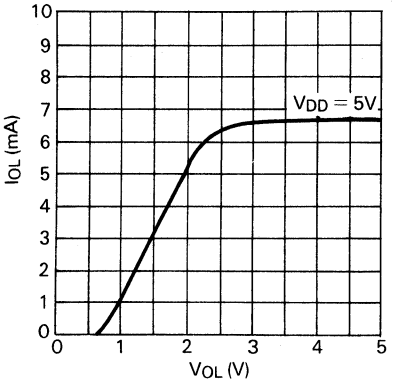
Segment Output



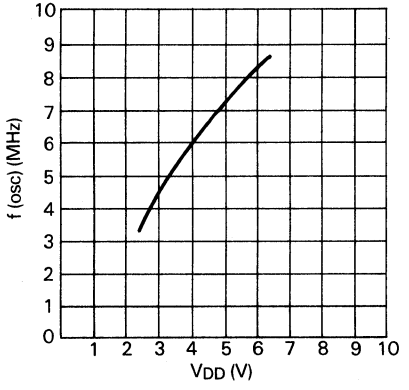
6

• **"L" Output Current I_{OL} – Output Current V_{OL} Characteristics (Ta = 25°C, VFLT = 40V)**

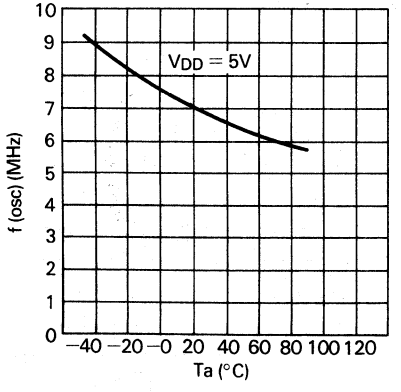
Timing Output



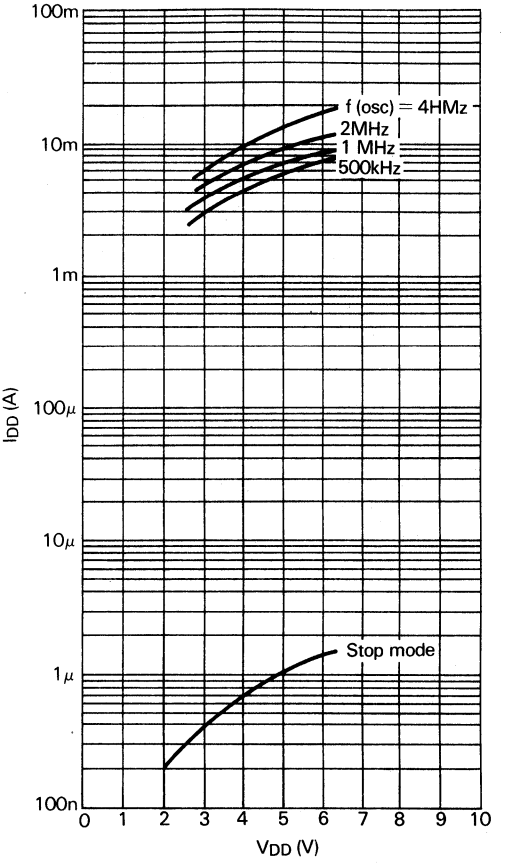
• **Maximum Clock Freqency f(osc) – Supply Voltage V_{DD} Characteristics ($Ta = 25°C$, C_L = 15pF)**



• **Maximum Clock Frequency f(osc) – Ambient Temperature Ta (V_{DD} = 5V, C_L = 15pF)**

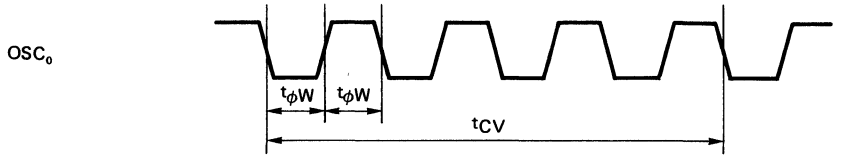


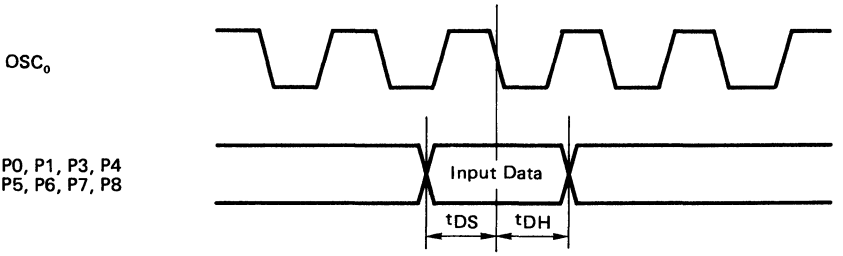
• **Current Consumption I_{DD} – Supply Voltage V_{DD} (Ta = 25°C, No load)**

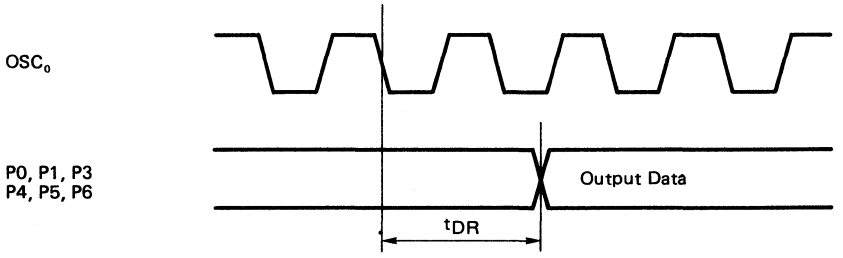


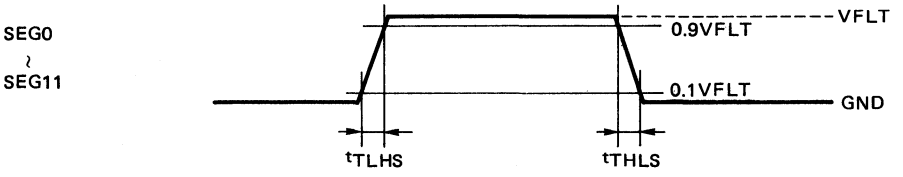
6

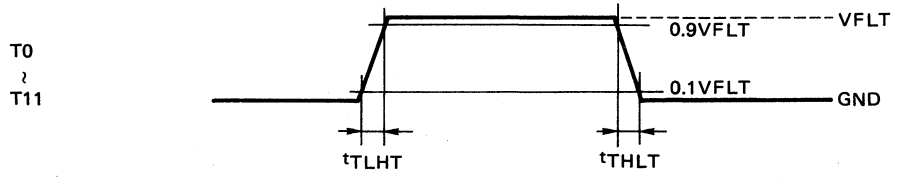
TIMING CHART

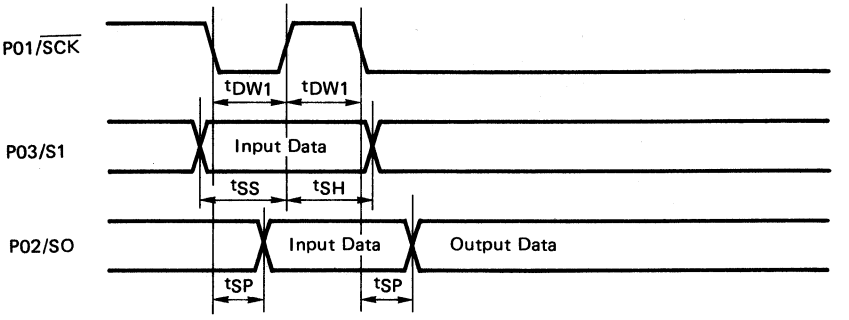


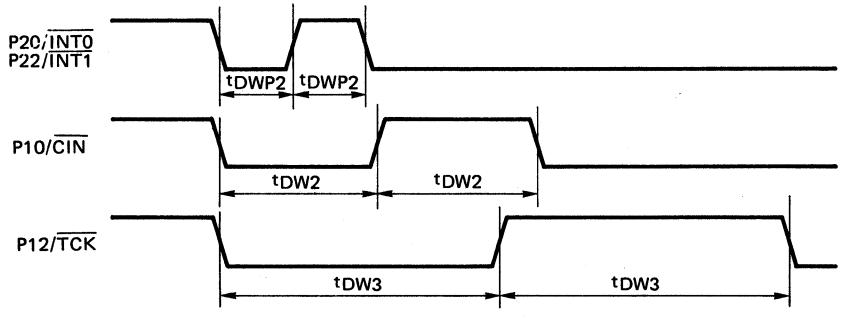


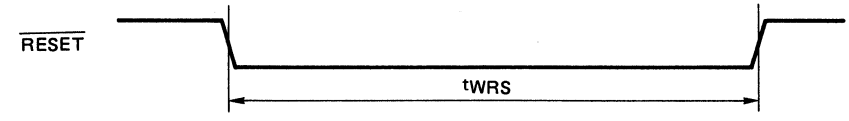


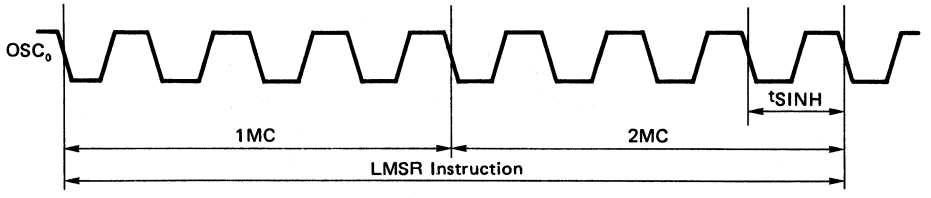





P01/SCK
tDW1
tDW1
P03/S1
Input Data
tSS
tSH
P02/SO
Input Data
Output Data
tSP
tSP


P20/INT0
P22/INT1
tDWP2
tDWP2
P10/CIN
tDW2
tDW2
P12/TCK
tDW3
tDW3


RESET
tWRS


OSC0
tSINH
1MC
2MC
LMSR Instruction

6

OKI semiconductor

MSC6458VS

MSC6458 PIGGY BACK

GENERAL DESCRIPTION

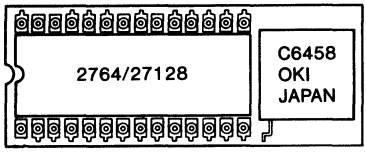
The MSC6458VS is a device whose built-in ROM is replaced by external EPROM using the piggyback method.

FEATURES

- Supply Voltage: 5V±5%
- Frequency: DC ~ 4.3MHz
- Operating Temperature: 0 ~ 70°C

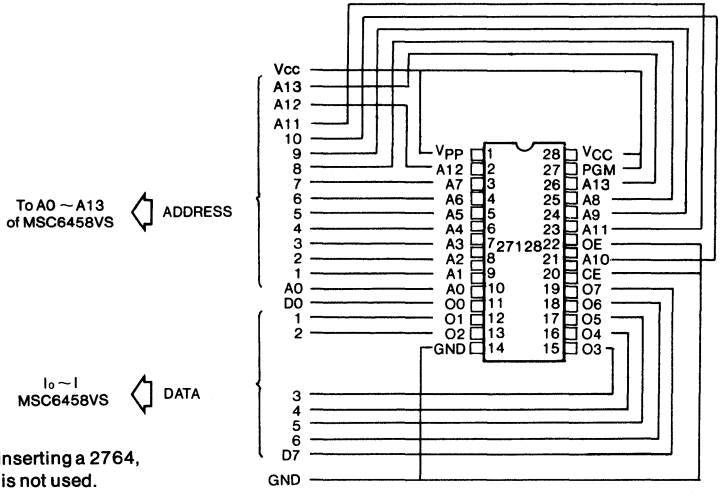
ROM INSERTION

Please refer to drawing below.



PIN CONFIGURATION

Pin Connection between MSC6458 VS and EPROM



Note: When inserting a 2764, pin 26 is not used.

6

OLMS-65 SERIES

6

MSM6502/6512

CMOS 4 BIT SINGLE CHIP MICROCONTROLLER WITH LCD DRIVER

GENERAL DESCRIPTION

The OKI MSM6502/6512 is a low-power, high-performance 4 bit single-chip microcontroller implemented in complementary metal oxide semiconductor technology.

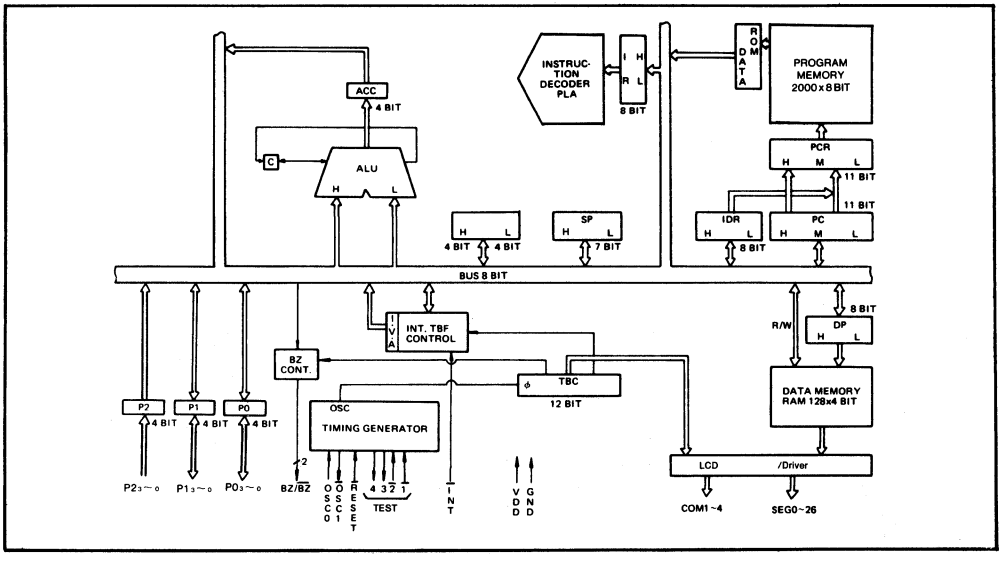
Integrated within this one chip is a 108 (4×27) dot LCD Driver. Also integrated in this chip are 16K bits of mask program ROM, 512 bits of data RAM, Input/Output lines, a programmable timer/counter, and oscillator.

The advantages of the MSM6502/6512 in comparison with the OLMS-40 Series include, among other features, a lower drive voltage, multiplexed interrupts, a larger number of drivable liquid crystal elements, a buzzer output, and larger memories.

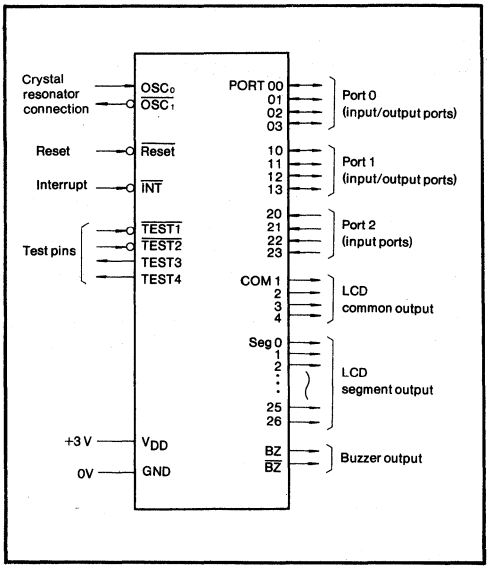
FEATURES

- ROM : 2000 × 8 bit
- RAM : 128 × 4 bit
- Number of Instructions : 68
- Clock Oscillation : Crystal 32.768 kHz
- Cycle Time : 91.5 μs
- Timer Interrupt : Dual (16 & 128 Hz)
- I/O Ports : 4 bit × 2 Port
- Input ports : 4 bit × 1 Port
- LCD Drive : 108 (4×27) picture elements
- Buzzer : 2K/1K/512 Hz/Soft
- Interrupt : Three types (external, two timer types)
- Stack : Nesting RAM stack pointer = 7 bits
- Power down : Halt mode available
- Operating power supply voltage : 2.4V – 3.6V
- Consumption current (V_{DD}=3V, OSC = 32.768 kHz) : MSM6502 : 45μA (Typical) 30μA (Halt mode) MSM6512 : 30μA (Typical) 12μA (Halt mode)
- Package : 56 pin FLAT (Small type)/ CHIP FORM

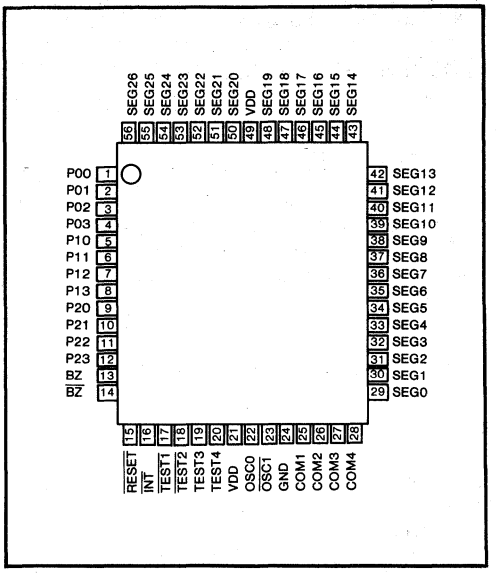
FUNCTIONAL BLOCK DIAGRAM



LOGIC SYMBOL



PIN CONFIGURATION (TOP VIEW)



6

PIN DESCRIPTION

Designation	Pin No.	Function
GND	24	Circuit GND potential
V_{DD}	21, 49	Main power source
OSC_0	22	Crystal OSC input, internal clock input
$\overline{OSC_1}$	23	Crystal OSC input, internal clock output
P0, P1	1 to 4 5 to 8	Pseudo-bidirectional ports for 4-bits parallel I/O. To input data from these ports, it is necessary to write "1" beforehand. The port to be selected is specified by the L register. The register contents and the corresponding specified ports are listed below. Content of L register / Port Specified 0.8 / P0 1.9 / P1 2.0 AH / P2 3.0 BH / P3 4.0 CH / P4 5.0 DH / P5 6.7.0EH. OFH / No designation Note: P3, P4, P5 are internal ports.
P2	9 to 12	Input ports for 4-bit parallel input with no latching function.
$\overline{INT}$	16	Input pin to request an interrupt from the external circuit. The input flag is set by the fall of the input signal.

Designation	Pin No.	Function
$\overline{\text{RESET}}$	15	The reset mode starts after "0" is input to the $\overline{\text{RESET}}$ pin for more than 2 machine cycles. The reset signal has priority over all of other signals and performs the following operations automatically: (1) Resets all bits of the PC (Program counter) to "0". (2) Sets all bits of the parallel I/O ports (P00 to P13) to "1". (3) Resets the internal resister (H, L, ACC, C, P3, P4, P5). (4) Resets the skip flag. (5) Resets all bits of the time base counter (TBC). (6) Resets the interrupt request flag (IRQF). (7) Resets the interrupt enable flag (EIF). (8) Resets the master interrupt enable flag (MEIF). (9) Sets all bits of the stack pointer (SP) to "1". (10) Initializes the segment and common outputs. (11) Sets all bits of the index register (IDR) to "1". Since the $\overline{\text{RESET}}$ pin is pulled up to V_{DD} by an internal resister (approx. 800 kΩ), it is possible to achieve power ON/reset by connecting it with an external capacitor.
LCD Drive Pins SEG 0 ~ 26 COM 1 ~ 4	29 to 48 50 to 56 25 to 28	A special AC waveform designed to comply with liquid crystal properties is required for liquid crystal drive purpose. The MSM6502B is equipped with a 1/4 duty 1/3 bias liquid crystal drive circuit with four common output ports and 27 segments to enable displays of 108 picture elements. On/off selection of picture element displays involves writing "0" or "1" in the corresponding bits in the RAM 00H to thru 1AH display area, and subsequent automatic hardware controlled display. The frame frequency is 64 Hz.
BZ/$\overline{\text{BZ}}$	13, 14	BZ and $\overline{\text{BZ}}$ are used in the generation of alarms and other sounds. The selectable frequencies include three hardware frequencies (TBC output) 512, 1024, and 2048 Hz, and a software type based on P50 data. These frequencies are selected at P3. When one of the hardware frequencies is selected by P3, output of that frequency is continuous. But selection of the software type results in output of P50 contents to generate a melody by program.

INSTRUCTION LIST

Grouping	Mnemonic	Code	Byte	Cycle	Function
Load	LAI n	8n	1	1	Acc ← n
	LLI n	7n	1	1	L ← n
	LHLI n8	6A n8	2	2	HL ← n8
	LXI n8	69 n8	2	2	X ← n8
	LAM	B0	1	1	Acc ← M<HL>
	LAL	B1	1	1	Acc ← L
	LAH	B2	1	1	Acc ← H
	LMA	B4	1	1	M<HL> ← Acc
	LLA	B5	1	1	L ← Acc
	LHA	B6	1	1	H ← Acc
	LMAD m7	1B m7	2	2	M<m7> ← Acc

INSTRUCTION LIST (CONT.)

Grouping	Mnemonic	Code	Byte	Cycle	Function
Load	LAMD m7	1A m7	2	2	Acc ← M<m7>
	LMT	67	1	2	M<HL+1><HL> ← ROM<X>
	PUSH HL	BC	1	2	STACK ← HL, SP ← SP−2
	PUSH CA	BD	1	2	STACK ← C, Acc, SP ← SP−2
	POP HL	BE	1	2	HL ← STACK, SP ← SP+2
	POP CA	BF	1	2	C, Acc ← STACK, SP ← SP+2
Exchange	XAM	B8	1	1	Acc ⟷ M<HL>
	XAMD m7	1C m7	2	2	Acc ⟷ M<m7>
	XHS	3F	1	1	HL ⟷ SP
Increment and decrement	INA	10	1	1	Acc ← Acc + 1, Skip if Acc=0
	INL	11	1	1	L ← L+1, Skip if L=0
	INM	12	1	1	M<HL> ← M<HL> +1, Skip if M<HL> = 0
	INX	5C	1	1	X ← X + 1
	DCA	14	1	1	Acc ← Acc−1, Skip if Acc= F
	DCL	15	1	1	L ← L − 1 Skip if L= F
	DCM	16	1	1	M<HL> ← M<HL> −1, Skip if M<HL> = F
	DCX	5D	1	1	X ← X − 1
Operation	DSC	60	1	1	C,Acc ← C+Acc+$\overline{M}$<HL>, Adjust if C=0
	DAC	61	1	1	C,Acc ← C+Acc+(M<HL>+6), Adjust if C=0
	ADS	62	1	1	Acc ← Acc+M<HL>, Skip if Cy=1
	ADCS	63	1	1	C,Acc ← C+Acc+M<HL>, Skip if C=1
	AIS n	0n	1	1	Acc ← Acc+n, Skip if Cy=1
	CMA	65	1	1	Acc ← $\overline{Acc}$
	EOR	66	1	1	Acc ← Acc ∀ M<HL>
	AND	4C	1	1	Acc ← Acc ∧ M<HL>
	OR	4D	1	1	Acc ← Acc ∨ M<HL>
	CAM	6B	1	1	Skip if Acc = M<HL>
	CPAL	6C	1	1	Skip if Acc = L
	CAXL	6D	1	1	Skip if Acc = XL
	CAXH	6E	1	1	Skip if Acc = XH
	SC	1F	1	1	C ← "1"
	RC	1E	1	1	C ← "0"
	TC	4E	1	1	Skip if C= "1"

INSTRUCTION LIST (CONT.)

Grouping	Mnemonic	Code	Byte	Cycle	Function
Operation	RAL	18	1	1	(*) ← C ← Acc: 3←2←1←0 ← (*)
	RAR	19	1	1	(*) → C → Acc: 3→2→1→0 → (*)
Bit Operation	SMB n2	30 ~ 33	1	1	M<HL> bit n2 ← "1"
	SMBD m7, n2	50 (m7) ~ 53 (m7)	2	2	M<m7> bit n2 ← "1"
	SPB n2	20 ~ 23	1	1	P bit n2 ← "1"
	RMB n2	34 ~ 37	1	1	M<HL> bit n2 ← "0"
	RMBD m7, n2	54 (m7) ~ 57 (m7)	2	2	M<m7> bit n2 ← "0"
	RPB n2	24 ~ 27	1	1	P bit n2 ← "0"
	TMB n2	38 ~ 3B	1	1	Skip if M<HL> bit n2 = "1"
	TMBD m7, n2	58 (m7) ~ 5B (m7)	2	2	Skip if M<m7> bit n2 = "1"
	TPB n2	28 ~ 2B	1	1	Skip if P bit n2 = "1"
	TAB n2	2C ~ 2F	1	1	Skip if Acc bit n2 = "1"
	TIRB n2	49 ~ 4B	1	1	if IRQF bit n2="1" Skip & IRQF bit n2 ← "0"
Interrupt	EI	5E	1	1	MEIF ← "1"
	DI	5F	1	1	MEIF ← "0"
Branch	J a11	9000~97CF	2	2	$PC_{10} \sim _0$ ← a11
	CAL a11	A000~A7CF	2	3	STACK←PC+2, SP←SP-3, $PC_{10} \sim _0$ ←a11
	JCP a6	C0~FF	1	1	$PC_5 \sim _0$ ←a6
	RT	3C	1	2	PC ← STACK, SP←SP+3
	RTS	3D	1	1	PC ← STACK, SP←SP+3, then Skip
Input/ Output	IP	B3	1	1	Acc ← P
	OP	B7	1	1	P ← Acc
CPU Control	HALT	4F	1	1	HLF ← "1"
	NOP	00	1	1	No Operation

ELECTRICAL CHARACTERISTIC

ABSOLUTE MAXIMUM RATING

Parameter	Symbol	Conditions	Limits	Unit
Power Supply Voltage	V_{DD}	Ta = 25°C	−0.3 ~ 7	V
Input Voltage	V_I		−0.3 ~ V_{DD}	V
Output Voltage	Vo		−0.3 ~ V_{DD}	V
Power Dissipation	P_D	Ta = 25°C per package	200	mV
StorageTemperature	Tstg	—	−55 ~ +150	°C

OPERATING RANGE

Parameter	Symbol	Conditions	Limits	Unit
Power Supply Voltage	V_{DD}	f(osc)=32.768 kHz	2.4 ~ 3.6	V
Operating Temperature	Top	—	−20 ~ +70	°C

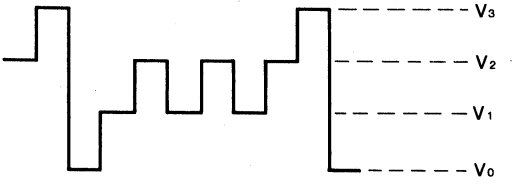
D.C. CHARACTERISTICS

(V_{DD} = 3V, Ta = −20 ~ +70°C)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
"H" Input Voltage	V_{IH}	−		2.6	−	−	V
"L" Input Voltage	V_{IL}	−		−	−	0.4	V
"H" Output Voltage(1)	V_{OH}	I_O = −1.0 mA		2.0	−	−	V
"L" Output Voltage(2)	V_{OL}	I_O = 1.0 mA		−	−	1.0	V
LCD Drive Output Voltage(3)	V_3	MSM6502	I_O = −5 μA	2.8	−	3.0	V
		MSM6512	I_O = −2 μA				
	V_2	MSM6502	I_O = ±2 μA	1.8	−	2.2	V
		MSM6512	I_O = ±0.5 μA				
	V_1	MSM6502	I_O = ±2 μA	0.8	−	1.2	V
		MSM6512	I_O = ±0.5 μA				
	V_0	MSM6502	I_O = 5μA	0.0	−	0.2	V
		MSM6512	I_O = ±2 μA				
OSC_0 Input Current	I_{IH}/I_{IL}	$V_I = V_{DD}/V_I$ = 0V		−	−	2/−2	μA
Input Current(4)	I_{IH}/I_{IL}	$V_I = V_{DD}/V_I$ = 0V		−	−	1/−10	μA
Input Current(5)	I_{IH}/I_{IL}	$V_I = V_{DD}/V_I$ = 0V		−	−	1000/−1	μA
Input Current(6)	I_{IH}/I_{IL}	$V_I = V_{DD}/V_I$ = 0V		−	−	1/−40	μA
P0, P1 "H" Output Current	I_{OH}	V_0 = 0 V		−	−	−50	μA
Current Consumption	I_{DD}	MSM6502	f(osc) = 32.768 kHz at no load	−	45	70	μA
		MSM6512		−	30	55	
	I_{DDHLT}	MSM6502	f(osc) = 32.768 kHz at HLT execution	−	30	40	μA
		MSM6512		−	12	25	
	I_{DDS}	MSM6502	Statis	−	15	25	μA
		MSM6512		−	5	15	
Oscillation Start Time	T_{OSC}	−		−	−	10	SEC

6

(1) Applied to BZ, $\overline{BZ}$
(2) Applied to BZ, $\overline{BZ}$, P0, P1
(3) Applied to COMMON, SEGMENT



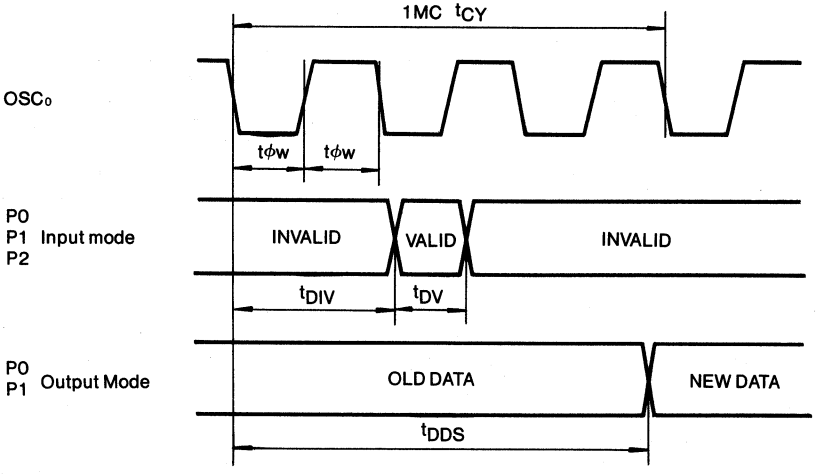
(4) Applied to $\overline{RESET}$, $\overline{INT}$
(5) Applied to P2 (When input is unable)
(6) Applied to P2 (When input is able)

SWITCHING CHARACTERISTIC

(V_{DD} = 3V, Ta = −20 ~ +70°C)

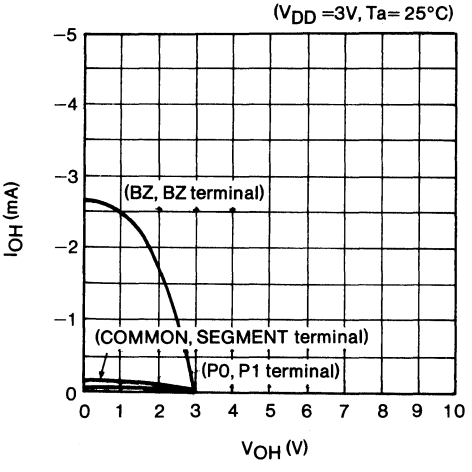
Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Clock (OSC_0) Pulse Width	$t_{\phi W}$	–	15	–	–	μS
Cycle Time	t_{CY}	–	(1)	–	–	μS
P0 P1 P2 Data Valid Time	t_{DV}	–	(2)	–	–	μS
P0 P1 P2 Data Invalid Time	t_{DIV}	–	–	–	(3)	μS
P0 P1 Data Delay Time	t_{DDS}	C_L = 50 pF	–	–	(4)	μS

(1) $t_{CY} = 3 \times 1/f(osc)$
(2) $t_{DV} = 1/2 \times 1/f(osc)$
(3) $t_{DIV} = 1 \times 1/f(osc) + 10\mu S$
(4) $t_{DDS} = 5/2 \times 1/f(osc) + 15\mu S$

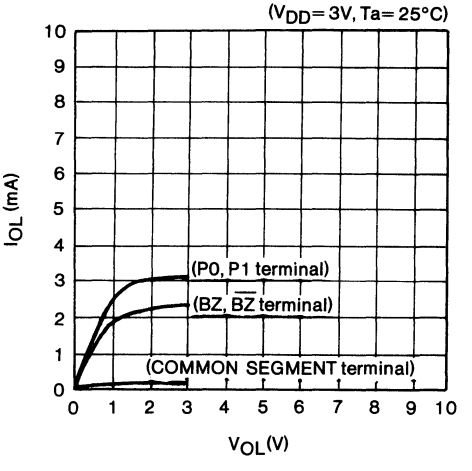


TYPICAL PERFORMANCE CURVES for MSM6502

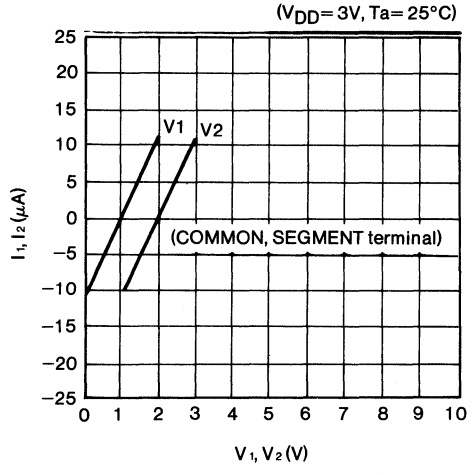
High-level Output Current (I_{OH}) – Output Voltage (V_{OH})



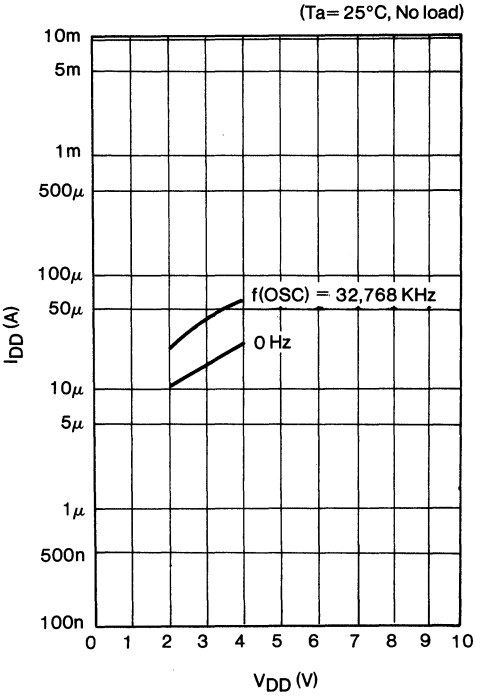
Low-level Output Current (I_{OL}) – Output Voltage (V_{OL})



Middle-level Output Current (I1, I2) – Output Voltage (V1, V2)



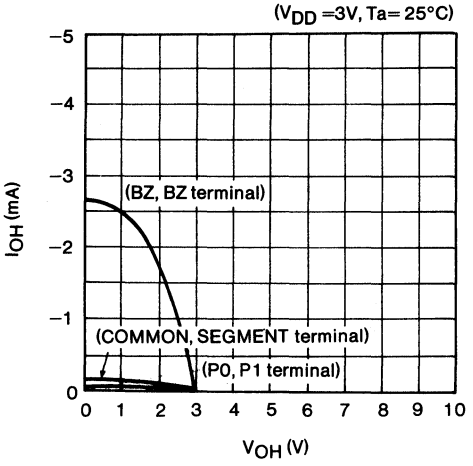
Current Consumption (I_{DD}) – Power Supply Voltage (V_{DD})



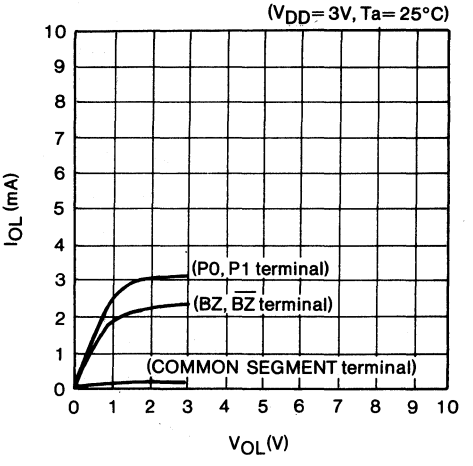
6

TYPICAL PERFORMANCE CURVES for MSM6512

High-level Output Current (I_{OH}) – Output Voltage (V_{OH})

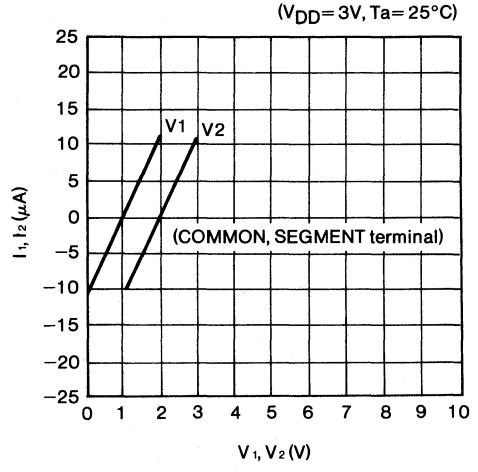


Low-level Output Current (I_{OL}) – Output Voltage (V_{OL})

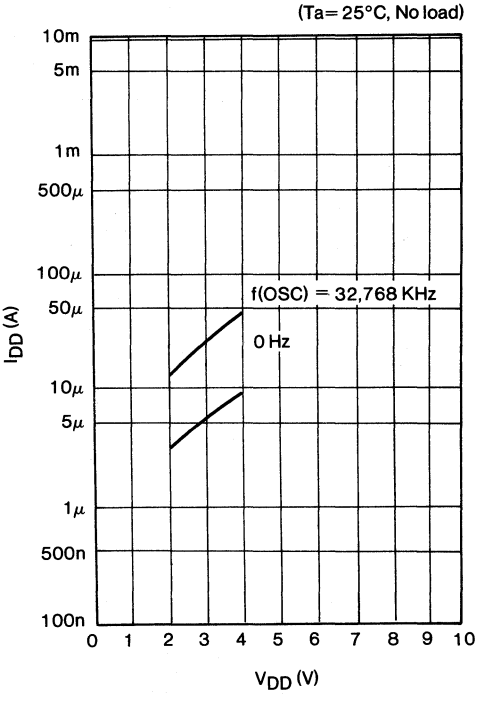


6

Middle-level Output Current (I1, I2) – Output Voltage (V1, V2)



Current Consumption (I_{DD}) – Power Supply Voltage (V_{DD})



8 BIT SERIES (OKI ORIGINAL)

OKI semiconductor

MSM62580

CMOS 8-BIT SINGLE CHIP MICROCONTROLLER WITH 16K BIT E²PROM

GENERAL DESCRIPTION

The MSM62580 is a CMOS single-chip microcontroller with on-board 16K bit E²PROM for applications such as IC-cards, etc.

The powerful instruction set consists of 95 instructions including special instructions for IC cards, executed by the 8-bit CPU in 800 ns at 5.0 MHz clock frequency.

The MSM62580 has improved hardware and software for security. Consequently, this chip suits application such as IC cards of low cost, high security and high reliability.

APPLICATION EXAMPLES

- IC Cards
- Mechanical Controls
- Automobile Controls
- Industrial Controls
- Compact Disc Players
- Audio/Video Equipment
- Household Appliances
- Musical Instruments

FEATURES

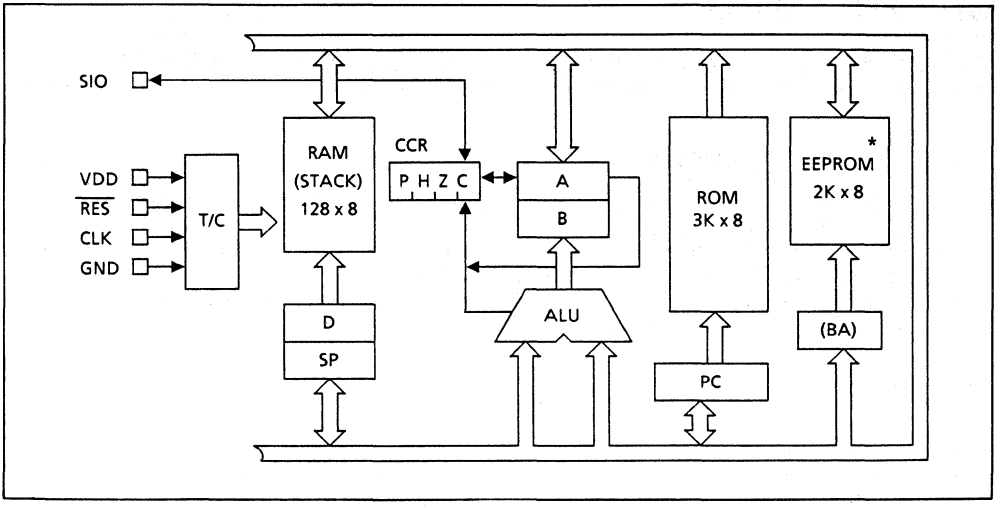
High speed instruction cycle
High number of instructions, and an efficient instruction set
(For example)

- Instructions for serial interface : SIN, SOUT, DLY
- Index addressing
- 1 byte call addresing : CZP
- Instructions for auto increment and auto decrement : INC, DEC
- Small Die size
- Simplified E²PROM write/erase operation by using control ROM.
- From D.C to 5 MHz clock frequency
- 9600 baud-rate serial interface using "DLY" instruction.

SPECIFICATIONS

- Single chip, low power CMOS
- 8-Bit Microcontroller
- 3K Bytes program ROM
- 512 Bytes control ROM
- 2K Bytes E²PROM
- 128 Bytes data RAM
- Clock frequency : 0 ~ 5.0 MHz
- Instrucion cycle : 800ns @ 5 MHz
- Number of instructions: 95
- Operation current : 4 mA typ.
- Ambient range : 0 to 70°C
- Number of pads : 5
- Supply voltage : +5 V ± 10%
- Die size : 5.0 × 4.5 mm

BLOCK DIAGRAM



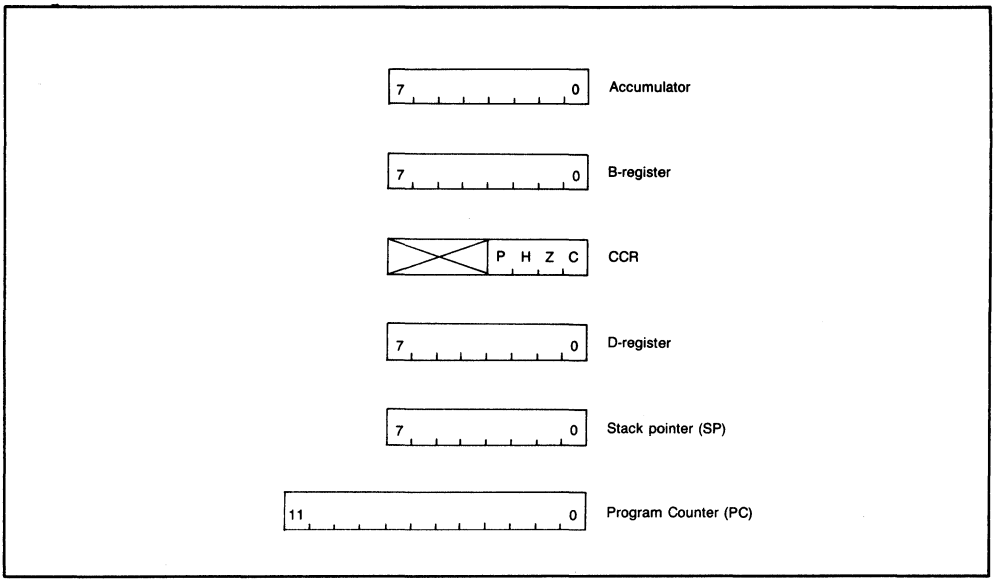
A	:	Accumulator	H	:	Half carry flag (for decimal operations)
ALU	:	Arithmetic circuit	P	:	Parity flag
B	:	B register (auxiliary register)	PC	:	Program counter
BA	:	B register paired with accumulator (B register higher rank)	RAM	:	Data memory
C	:	Carry flag	$\overline{RES}$	:	Reset input pin
CCR	:	Condition code register	SIO	:	Serial input/output pin
CLK	:	Clock input pin	SP	:	Stack pointer
D	:	D register (data pointer)	T/C	:	Timing and control circuit
EEPROM	:	Rewritable read-only memory	VDD	:	Power supply pin (5V)
GND	:	Power supply pin (0V)	Z	:	Zero flag

* EEPROM is not used as instruction area.

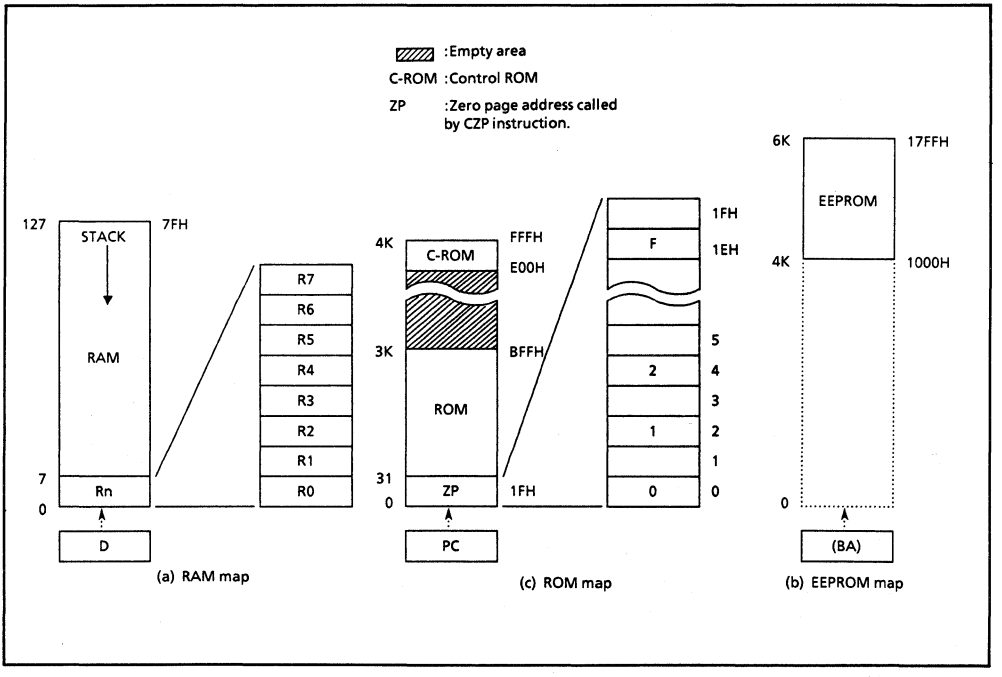
PIN DESCRIPTION

Description	Input/Output	Function
S-I/O	Input/Output	Serial data input/output port. Quasi bidirectional I/O port. Set "1" level after "Reset".
V_{DD}	—	Main power source
GND	—	Circuit GND potential
$\overline{RES}$	Input	"Reset" has priority over every other signal. $\overline{RES}$ input initialize the processor. Active "0" level.
CLK	Input	External clock input

REGISTER DIAGRAM



MEMORY MAP



ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{DD}	Ta = 25°C	−0.5 to 7	V
Input Voltage	V_I	Ta = 25°C	−0.3 to V_{DD} +0.5	V
Output Voltage	V_O	Ta = 25°C	−0.3 to V_{DD} +0.5	V
Storage Temperature	T_{stg}		0 to +70	°C

OPERATING CONDITIONS

Parameter	Symbol	Limits	Unit
Supply Voltage	V_{DD}	4.5 to 5.5	V
Operating Temperature	T_{OP}	0 to +70	°C

D.C. CHARACTERISTICS

6

(V_{DD} = 5V± 10%, Ta = 0 to +70°C)

Parameter		Symbol	Conditions	Min	Typ	Max	Unit
Operating Current		I_{DD}	f = 5 MHz	—	4	10	mA
Low Input Voltage	CLK	V_{IL}	—	−0.3	—	0.5	V
	RES			−0.3	—	0.5	
	SIO			−0.3	—	0.8	
High Input Voltage	CLK	V_{IH}	—	2.4	—	V_{DD}	V
	RES			4	—	V_{DD}	
	SIO			2.0	—	V_{DD}	
Low Output Voltage		V_{OL}	I_{OL} MAX = 1.6mA	0	—	0.4	V
High Output Voltage		V_{OH}	I_{OH} MAX = −100≥A	2.4	—	V_{DD}	V
Input Current (CLK, $\overline{RES}$)		I_{IH1}/I_{IL1}	V_I = O/V_{DD}	—	—	20	μA
Input Current (SIO)		I_{IH2}/I_{IL2}		—	—	−1	mA
Input Capacitance		C_I	f = 1MHz Ta = 25°C	—	15	—	pF
Output Capacitance		C_O		—	20	—	pF

Notes: = CLK, $\overline{RES}$ has poll down resistance SIO has pull up resistance.

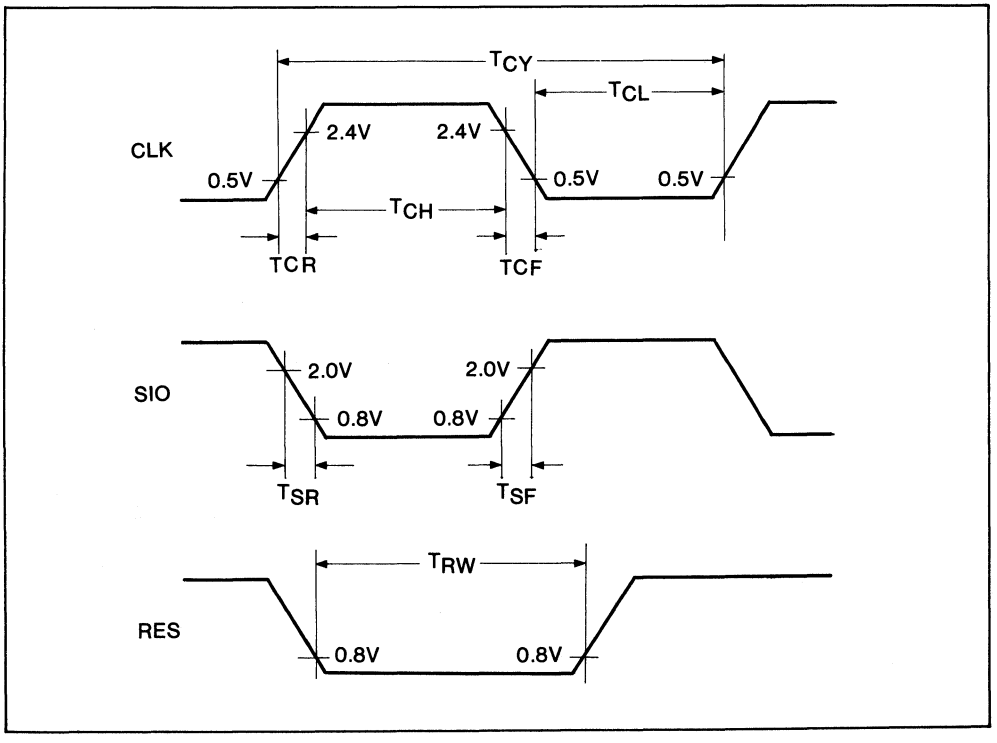
A.C. CHARACTERISTICS

(V_{DD} = 5V± 10%, Ta = 0 to +70°C)

Parameter	Symbol	Min	Typ	Max	Unit
CLK Cycle Time	T_{CY}	200	—	—	ns
CLK Cycle Low Width	T_{CL}	0.4*T_{CY}	—	0.6*T_{CY}	ns
CLK Cycle High Width	T_{CH}	0.4*T_{CY}	—	0.6*T_{CY}	ns
CLK Cycle Rise Time	T_{CR}	—	—	5.0	μs
CLK Cycle Fall Time	T_{CR}	—	—	5.0	μs
$\overline{RES}$ Pulse Width	T_{RW}	8*T_{CY}	—	—	μs
SIO INPUT Rise Time	T_{SR}	—	—	5.0	μs
SIO INPUT Fall Time	T_{SF}	—	—	5.0	μs

Note: at output load capacitance C_O = 30pF

TIMING CHARTS



INSTRUCTION LIST

MNEMONIC	opr	OPERATION	BYTE	CYCLE	FLAGS			
					C	P	H	Z
MOV A, opr	B	A ← B	1	1				*
	D	A ← D	1	1				*
	@D	A ← (D)	1	1				*
	@D+	A←(D), D←D+1	1	2				*
	@D−	A←(D), D←D−1	1	2				*
	N	A ← (N)	2	2				*
	N + @D	A ← (N+D)	2	3				*
	#N	A ← #N	2	2				*
MOV opr, A	B	B ← A	1	1				
	D	D ← A	1	1				
	@D	(D) ← (A)	1	1				
	@D+	(D)←A, D←D+1	1	2				
	@D−	(D)←A, D←D−1	1	2				
	N	(N) ← A	2	2				
	N+@D	(N+D) ← A	2	3				
MOV D, opr	Rn	D ← Rn	1	2				
	#N	D ← #N	2	2				
MOV Rn, opr	D	Rn ← D	1	2				
	#N	Rn ← #N	2	3				
MOV @BA, opr	@D	(BA) ← (D)	1	4				
MOV @D, opr	@BA	(D) ← (BA)	1	4				
MOV @D+, opr	#N	(D)← #N,D←D+1	2	2				
MOV @D+, opr	BA	(D)←A, (D+1)←B	1	3				
MOVW BA, opr	@D	A←(D), B←(D+1)	1	3				*
MOVW BA, opr	#N	A← #N1, B← #N2	3	3				*

MNEMONIC	opr	OPERATION	BYTE	CYCLE	FLAGS			
					C	P	H	Z
XCH A, opr	B	A ↔ B	1	2				*
	D	A ↔ D	1	2				*
	@D	A↔(D)	1	2				*
	N	A ↔ (N)	2	2				*
XCH D, opr	B	D ↔ B	1	2				
	SP	D ↔ SP	1	2				
XCH C, opr	P	C ↔ P	1	1	*	*		
ADD A, opr	@D	A ← A+(D)	1	1	*		*	*
	N	A ← A+(N)	2	2	*		*	*
	#N	A← A+#N	2	2	*		*	*
ADC A, opr	@D	A←A+(D)+C	1	1	*		*	*
	N	A←A+(N)+C	2	2	*		*	*
	#N	A←A+#N+C	2	2	*		*	*
DAA		Decimal adjust	1	1	*			*
CMP A, opr	@D	A − (D)	1	1	*			*
	N	A − (N)	2	2	*			*
	#N	A − #N	2	2	*			*
CMP @D, opr	#BA	(D) − (BA)	1	4	*			*
EOR A, opr	@D	A ← A∀(D)	1	1				*
	N	A ← A∀(N)	2	2				*
	#N	A ← A∀#N	2	2				*
OR A, opr	@D	A ← AV(D)	1	1				*
	N	A ← AV(N)	2	2				*
	#N	A ← AV#N	2	2				*
AND A, opr	@D	A ← A∧(D)	1	1				*
	N	A ← A∧#(N)	2	2				*
	#N	A ← A∧#N	2	2				*

6

MNEMONIC	opr	OPERATION	BYTE	CYCLE	FLAGS			
					C	P	H	Z
INC opr	A	A ← A+1	1	1				*
	D	D ← D+1	1	1				
	@D	(D) ← (D)+1	1	1				*
	N	(N) ← (N)+1	2	2				*
DEC opr	A	A ← A−1	1	1				*
	D	D ← D−1	1	1				
	@D	(D) ← (D)−1	1	1				*
	N	(N) ← (N)−1	2	2				*
RRC opr	A	↳C → A7 ~0┘	1	1	*			*
	@D	↳C → (D)7~0┘	1	1	*			*
	N	↳C → (N)7~0┘	2	2	*			*
RLC opr	A	└C ← A7 ~0↲	1	1	*			*
	@D	└C ← (D)7~0↲	1	1	*			*
	N	└C ← (N)7~0↲	2	2	*			*
PUSH opr	PSW	(SP)←A,(SP−1)←CCR, SP←SP−2	1	3				
	D	(SP)←D, SP←SP−1	1	2				
POP opr	PSW	CCR←(SP−1), A←(SP−2), SP←SP+2	1	3	*	*	*	*
	D	D ← (SP + 1), SP ← SP + 1	1	2				
JZ opr	addr	if Z=1, PC←PC+2+addr	2	2/3				
JNZ opr	addr	if Z ≠ 1, PC←PC+2+addr	2	2/3				
JC opr	addr	if C=1, PC←PC+2+addr	2	2/3				
JNC opr	addr	if C ≠ 1, PC←PC+2+addr	2	2/3				
JB opr	baddr,addr	if(baddr)=1,PC←PC+3+addr	3	3/4				
JNB opr	baddr,addr	if(baddr) ≠ 1, PC←PC+3+addr	3	3/4				
DJNZ opr	Rn,addr	Rn←Rn-1, if Rn ≠ 0, PC←PC+2+addr (n=4~7)	2	3/4				
JMNE opr	#N,addr	if (D) ≠ #N, PC←PC+3+addr	3	3/4				
JDNE opr	#N,addr	if D ≠ #N, PC←PC+3+addr	3	3/4				

MNEMNIC	opr	OPERATION	BYTE	CYCLE	FLAGS			
					C	P	H	Z
JMP opr	addr	PC←addr(0~4K)	2	2				
CAL opr	addr	(SP)←PC+2, PC←addr(0~4K) SP←SP-2	2	4				
CZP opr	addr	(SP)←PC+2,PC←ZP,SP←SP-2	1	4				
RT		PC←(SP), SP←SP+2	1	3				
NOP		No Operation	1	1				
CLR opr	A	A ← 0	1	1				1
RC		C ← 0	1	1	0			
SC		C ← 1	1	1	1			
RB	baddr	(baddr) ← 0	2	2				
SB	baddr	(baddr) ← 1	2	2				
CPL opr	A	A ← $\overline{A}$	1	1				*
	C	C ← $\overline{C}$	1	1	*			
CHK opr	P	P←C, if A=odd, C←1 ELSE C←0	1	1	*	*		
SIN		C ← SIO	1	1	*			
SOUT		SI/O ← C	1	1				
DLY opr	#N	DELAY N+3 CYCLES	2	3~258				

MSM66301

OKI ORIGINAL HIGH PERFORMANCE CMOS SINGLE-COMPONENT 8/16-BIT MICROCONTROLLER

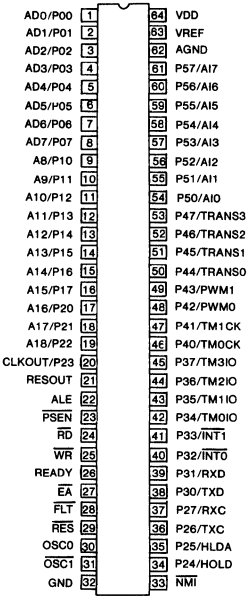
GENERAL DESCRIPTION

The OKI MSM66301 is a an new generation, high performance single component microcontroller inplemented in sillicon gate complementary metal oxide semiconductor technology (CMOS). Integrated within this chip are 16-bit ALU, 16K bytes of mask program ROM, 512 bytes of data RAM, 48 I/O lines, built-in 16-bit timers, 10-bit A/D converter, serial I/O port, pulse width modulator (PWM), and oscillator.

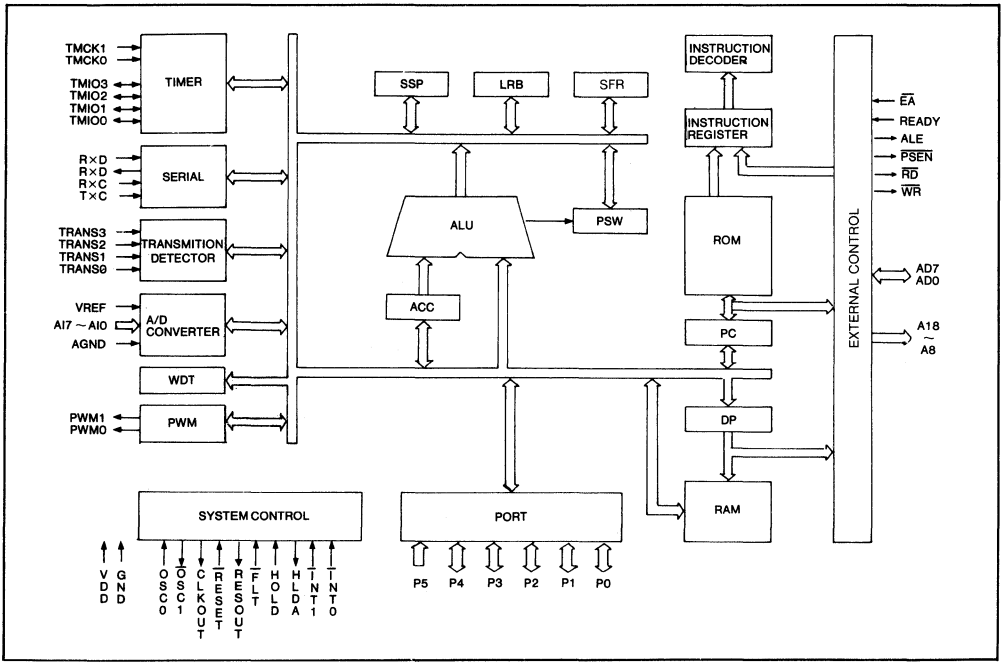
FEATURES

- 8-Bit External Data Bus Interface
- 16-Bit Internal Architecture
- 64K address space for program memory (including 16K bytes onchip ROM)
- 512K address space for data memory (including 512 bytes onchip RAM)
- High speed execution
  Minimum Cycle for Instruction: 400ns (10MHz)
- The Abundance of Powerful Instructions
  8/16 data transfer operation
  8/16 bit arithmetic operation
  16(8) bit × 16(8) bit → 32(16) bit
  32(16) bit/16 (8) bit → 32(16) bit
  16(8) bit ± 16(8) bit → 16(8) bit
  8/16 logic operation
  Bit operation
  String operation
  User stack operation
  ROM table access operation
- The same instruction allows both byte and word width operation according to Data Descriptor.
  That is to say, the same algorithm and the same source program lines are applicable to byte and word width data manipulation with only changing Data Descriptor.
- Many Addressing Modes
- 8 Input lines, $\overline{40}$ Input/Output lines
- Built-in 16 bit timer × 4
  Each timer has the following 4 modes.
  Auto reload timer mode
  Clock out mode
  Capture register mode
  Real time output mode
- Serial Port × 1ch.
  (variable bit length, baud rate generators for transmitter & receiver)
  Asyncronous normal mode
  Asyncronous multi processor communication mode
  Syncronous normal mode
  Syncronous multi processor communication mode
- 16 bit Pulse Width Modulator × 2
- Transition Detector × 4
- 10 bit A/D converter (8 channel)
- 1 non-maskable interrupt, 16 maskable interrupts
- Stand-by Function
  software Clock stop
  software CPU stop
  hardware CPU stop
- 64 pin Shrink DIP/64 pin plastic flat package/68 pin PLCC

PIN CONFIGURATION



FUNCTIONAL BLOCK DIAGRAM



6

PIN DESCRIPTION

Designation	Input/Output	Function
$P_{00} - P_{07}$ / $AD_0 - AD_7$	I/O	P0: 8-bit I/O port. Each bit can be assigned to input or output. AD: Outputs the lower 8 bits of program counter during external program memory fetch, and receives the addressed instruction under the control of $\overline{PSEN}$. Also outputs the address, Outputs or inputs data during an external data memory access instruction, under control of ALE, $\overline{RD}$, and $\overline{WR}$.
$P_{10} - P_{17}$ / $A_8 - A_{15}$	I/O	P1: 8-bit I/O port. Each bit can be assigned to input or output. A: Outputs the middle 8 bits of program counter (PC_{8-15}) during external program memory fetch. Also outputs the middle 8 bits of address during an external data memory access instructions.
$P_{20} - P_{22}$ / $A_{16} - A_{18}$	I/O	P2: 8-bit I/O port. Each bit can be assigned to input or output. A: Outputs the upper 3 bits of address during external data memory access instructions.
P_{23}/CLKOUT		CLKOUT: clock output pin. Output frequency range is equal to or twice the system clock.
P_{24}/HOLD		HOLD: Input pin to request the CPU to enter the hardware power-down state.
P_{25}/HLDA		HLDA: HOLD ACKNOWLEDGE: the HLDA signal appears in reponse to the HOLD signal and indicates that the CPU has entered the power-down state.
P_{26}/TxC		TxC: Transmitter clock input pin.
P_{27}/RxC		RxC: Receiver clock input pin.
P_{30}/TxD	I/O	P3: 8-bit I/O port. Each bit can be assigned to input or output. TxD: Transmitter data output pin.
P_{31}/RxD		RxD: Receiver data input pin.
P_{32}/$\overline{INT0}$		$\overline{INT}$: Interrupt Request Input pin. Falling edge trigger or level trigger is selectable.
P_{33}/$\overline{INT1}$		
P_{34}/TM0IO		TM0IO ~ TM3IO: One of the following signals is output or input: • clock twice the frequency range of the 16 bit timer overflow • load trigger signal to the capture register input • setting value output The signal that is input or output depends on the mode.
P_{35}/TM1IO		
P_{36}/TM2IO		
P_{37}/TM3IO		

PIN DESCRIPTION (Continued)

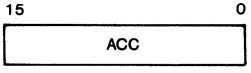
Designation	Input/Output	Function
P_{40}/TM0CK P_{41}/TM1CK P_{42}/PWM0 P_{43}/PWM1 $P_{44} - P_{47}$ / TRANS0 – TRNS3	I/O	P4: 8 bit I/O port. Each bit can be assigned to input or output. TM0CK, TM1CK: clock input pins of timer 0, timer 1. TRNS: Transition Detector. The input pins which sense the falling edge and set the flag. PWM: Pulse Wide Modulator output pin.
$P_{50} - P_{57}$ / AI0 – AI7	INPUT	P5: 8 bit input port. AI: analog sygnal input pin to A/D converter.
RESOUT	OUTPUT	Output 'H' level when the CPU is in RESET cycle. Reset to 'L' level by program.
ALE	OUTPUT	Address Latch Enable: The timing pulse to latch the lower 8 bit of the address output from port 0 when the CPU accesses the external data memory.
$\overline{PSEN}$	OUTPUT	Program Store Enable: The strobe pulse to fetch to external program memory.
$\overline{RD}$	OUTPUT	Output strobe activated during a BUS read. Can be used to enable data on to the bus from the external data memory.
$\overline{WR}$	OUTPUT	Output strobe during a bus write. Used as write strobe to external data memory.
READY	INPUT	Used when the CPU accesses low speed peripherals.
$\overline{EA}$	INPUT	Normaly set to 'H' level. If set to 'L' level, the CPU fetches the code to external program memory.
$\overline{FLT}$	INPUT	If $\overline{FLT}$ is 'H' level, ALE, $\overline{WR}$, $\overline{RD}$, $\overline{PSEN}$ are set 'H' level when reset. If $\overline{FLT}$ is set to 'L', ALE, $\overline{WR}$, $\overline{RD}$ $\overline{PSEN}$ are set to floating level when reset.
$\overline{RESET}$	INPUT	RESET input pin

PIN DESCRIPTION (Continued)

Designation	Input/Output	Function
OSC_0, $\overline{OSC_1}$		Oscillation circuit consists of OSC_0, $\overline{OSC_1}$.
$\overline{NMI}$	INPUT	non maskable interrupt input pin (falling edge)
VREF		reference voltage input pin for A/D converter
AGND		ground for A/D converter
VDD		system power supply
GND		ground

REGISTERS

■ **ACCUMULATOR**



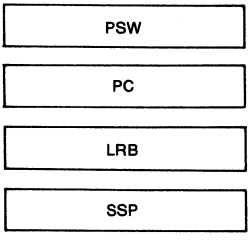
■ **CONTROL REGISTERS (CR)**

PROGRAM STATUS WORD

PROGRAM COUNTER

LOCAL REGISTER BASE

SYSTEM STACK POINTER



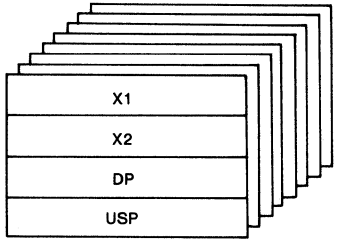
■ **POINTING REGISTERS (PR)**

INDEX REGISTER1

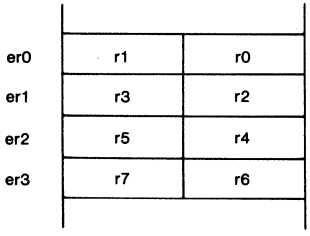
INDEX REGISTER2

DATA POINTER

USER STACK POINTER



■ **LOCAL REGISTERS**



■ **SPECIAL FUNCTION REGISTERS (SFR)**

All of the I/O functions are controlled by SFRs.Also, some of the internal functions (Timer, WDT, etc,...) are controlled by SFRs. SFRs are located in the top of RAM space (000H ~ 007FH).

MSM66301 Special Function Registers [1]

Address	Name	Abbrebiation	R/W	8/16 Operation	When reset
0000	System Stack Pointer	SSPL	R/W	8/16	FF
0001		SSPH			FF
0002	Local Register Base	LRBL	R/W	8/16	unknown
0003		LRBH			
0004	Program Status Word	PSWL	R/W	8/16	00
0005		PSWH			00
0006	Accumulator	ACCL	R/W	8/16	00
0007		ACCH			00
0008	Source Index Register	SI	R/W	16	unknown
0009	for Block Transfer				
000A	Destination Index Register	DI	R/W	16	unknown
000B	for Block Transfer				
000C	Counter Register	CX	R/W	16	unknown
000D	for Block Transfer				
000E	Source/Destination Bank Register	BSDI	R/W	8	unknown
0010	Stand-By Control Register	SBYCON	R/W	8	00
0011	Watch Dog Timer	WDT	W	8	
0012	Peripheral Control Register	PRPHF	R/W	8	01
0016	Work Area for Emulator		R/W	16	unknown
0017					
0018	Interrupt Request Flag	IRQL	R/W	8/16	00
0019		IRQH			00
001A	Interrupt Enable Flag	IEL	R/W	8/16	00
001B		IEH			00
001C	External Interrupt Control Register	EXICON	R/W	8	00

MSM66301 Special Function Registers [2]

Address	Name	Abbrebiation	R/W	8/16 Operation	When reset
0020	Port 0 Data Register	P0	R/W	8	Unknown
0021	Port 0 Mode Register	P0IO	R/W	8	00
0022	Port 1 Data Register	P1	R/W	8	unknown
0023	Port 1 Mode Register	P1IO	R/W	8	00
0024	Port 2 Data Register	P2	R/W	8	unknown
0025	Port 2 Mode Register	P2IO	R/W	8	00
0026	Port 2 Special Function Control Register	P2SF	R/W	8	00
0028	Port 3 Data Register	P3	R/W	8	unknown
0029	Port 3 Mode Register	P3IO	R/W	8	00
002A	Port 3 Special Function Control Register	P3SF	R/W	8	00
002C	Port 4 Data Register	P4	R/W	8	unknown
002D	Port 4 Mode Register	P4IO	R/W	8	00
002E	Port 4 Special Function Control Register	P4SF	R/W	8	00
002F	Port 5	P5	R	8	
0030	Timer 0 Counter	TM0	R/W	16	00
0031					00
0032	Timer 0 Register	TMR0	R/W	16	00
0033					00
0034	Timer 1 Counter	TM1	R/W	16	00
0035					00
0036	Timer 1 Register	TMR1	R/W	16	00
0037					00
0038	Timer 2 Counter	TM2	R/W	16	00
0039					00
003A	Timer 2 Register	TMR2	R/W	16	00
003B					00
003C	Timer 3 Counter	TM3	R/W	16	00
003D					00
003E	Timer 3 Register	TMR3	R/W	16	00
003F					00

MSM66301 Special Function Registers [3]

Address	Name	Abbrebiation	R/W	8/16 Operation	When reset
0040	Timer 0 Control Register	TCON0	R/W	8	00
0041	Timer 1 Control Register	TCON1	R/W	8	00
0042	Timer 2 Control Register	TCON2	R/W	8	00
0043	Timer 3 Control Register	TCON3	R/W	8	00
0046	Transition Detector Reg.	TRNS	R/W	8	00
0013	Stop Mode Buffer Flag		W	8	

MSM66301 Special Function Registers [4]

Address	Name	Abbrebiation	R/W	8/16 Operation	When reset
0046	Transition Detector	TRNS	R/W	8	00
0048	Transmitter Clock Generator	STTM	R/W	8	00
0049	Transmitter Clock Generator Control Register	STTMR	R/W	8	00
004A	Transmitter Control Register	STTMC	R/W	8	00
004C	Receiver Clock Generator	SRTM	R/W	8	00
004D	Receiver Clock Generator Control Register	SRTMR	R/W	8	00
004E	Receiver Control Register	SRTMC	R/W	8	00
0050	Transmitter Mode Control Register	STCON	R/W	8	00
0051	Transmitter Data Buffer	STBUF	W	8	unknown
0054	Receiver Mode Control Register	SRCON	R/W	8	00
0055	Receiver Data Buffer	SRBUF	R	8	unknown
0056	Receiver Eroor Status Register	SRSTAT	R	8	00
0058	A/D Scanning Mode Register	ADSCAN	R/W	8	00
0059	A/D Select Mode Register	ADSEL	R/W	8	00
0060	A/D Convert Result Register 0	ADCR0L	R	8/16	unknown
0061		ADCR0H			
0062	A/D Convert Result Register 1	ADCR1L	R	8/16	unknown
0063		ADCR1H			
0064	A/D Convert Result Register 2	ADCR2L	R	8/16	unknown
0065		ADCR2H			
0066	A/D Convert Result Register 3	ADCR3L	R	8/16	unknown
0067		ADCR3H			
0068	A/D Convert Result Register 4	ADCR4L	R	8/16	unknown
0069		ADCR4H			
006A	A/D Convert Result Register 5	ADCR5L	R	8/16	unknown
006B		ADCR5H			
006C	A/D Convert Result Register 6	ADCR6L	R	8/16	unknown
006D		ADCR6H			
006E	A/D Convert Result Register 7	ADCR7L	R	8/16	unknown
006F		ADCR7H			

MSM66301 Special Function Registers [5]

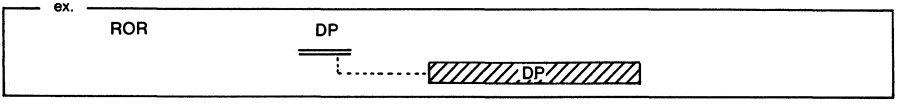
Address	Name	Abbrebiation	R/W	8/16 Operation	When reset
0070	PWM 0 Counter	PWMC0L	R/W	8/16	00
0071		PWMC0H			00
0072	PWM 0 Register	PWMR0L	R/W	8/16	00
0073		PWMR0H			00
0074	PWM 1 Counter	PWMC1L	R/W	8/16	00
0075		PWMC1H			00
0076	PWM 1 Register	PWMR1L	R/W	8/16	00
0077		PWMR1H			00
0078	PWM 0 Control Register	PWCON0	R/W	8	00
007A	PWM 1 Control Register	PWCON1	R/W	8	00

ADDRESSING MODE

The MSM66301 supports 512KB (64KB×8BANKs) 05 data space and 64KB of program space with verious types of addressing methods. These methods divide into the following types.

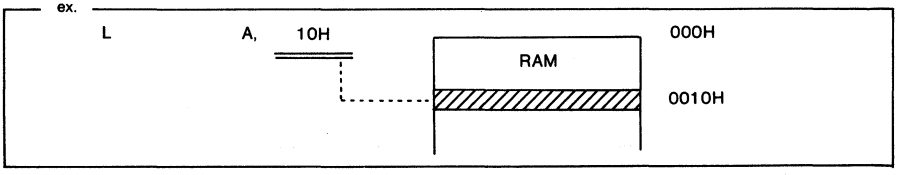
1. RAM ADDRESSING (FOR DATA SPACE)

1.1 Register Direct Addressing

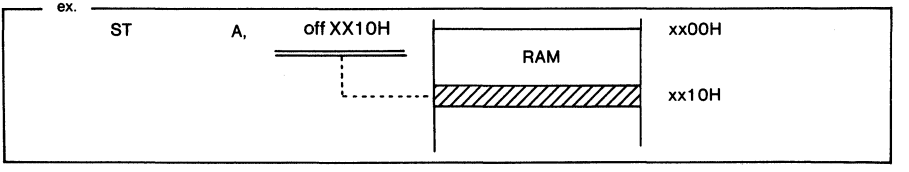


1.2 Displacement Addressing

a) Zero Page

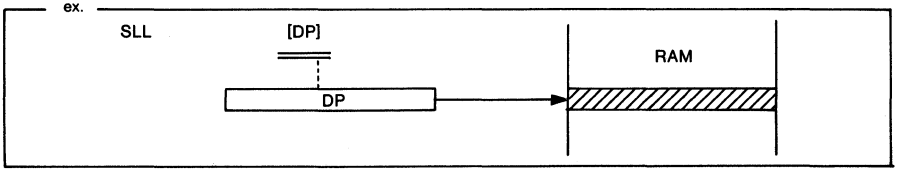


b) Direct Page

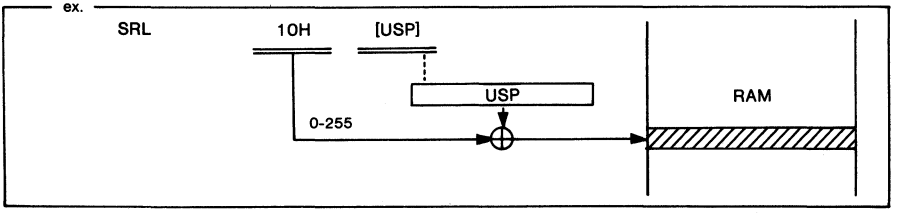


1.3 Pointing Register Indirect Addressing

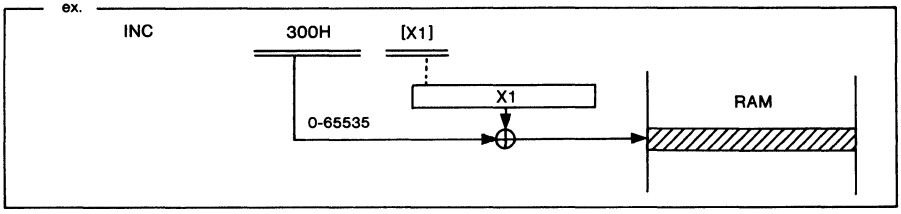
a) Data Pointer (DP) Indirect



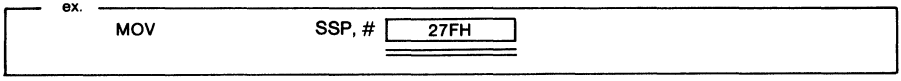
b) User Stack Pointer (USP) Indirect



c) Index register (X1, X2) Indirect

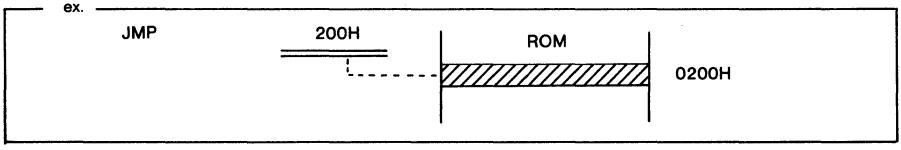


1.4 Immediate Addressing



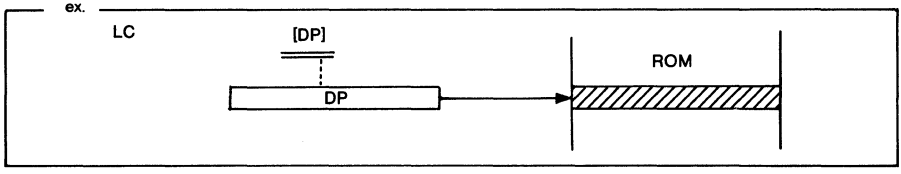
2. ROM ADDRESSING (FOR PROGRAM SPACE)

2.1 Direct Addressing

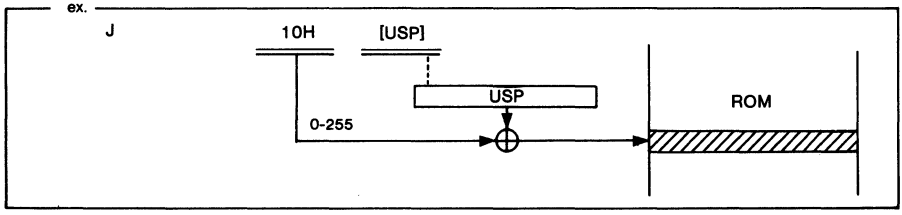


2.2 Pointing Register Indirect Addressing

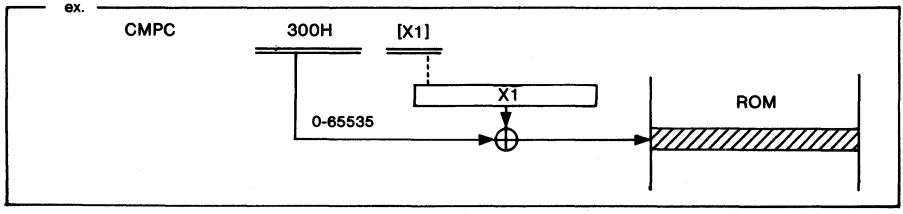
a) Data Pointer (DP) Indirect



b) User Stack Pointer (USP) Indirect



c) Pointing Register (X1, X2, DP, USP) Indirect with 16bit displacement



INSTRUCTIONS

Data Transfer

Instruction	Function
L A,*	A ← *
ST A,*	A → *
MOV *, A	* ← A
MOV *, #	* ← Imme.
MOV er.*	er ← *
MOV d,*	(d) ← *
MOV DP,*	DP ← *
MOV X1,*	X1 ← *
MOV X2,*	X2 ← *
MOV USP,*	USP ← *
MOV PSW,*	PSW ← *
MOV SSP,*	SSP ← *
MOV LRB,*	LRB ← *
CLR *	* ← 0
SWAP	$A_{15-8} \longleftrightarrow A_{7-0}$
XCHG A,*	A ⟷ *
XNBL A,*	$A_{3-0} \longleftrightarrow *_{3-0}$
TRNS USP, LRB	$USP_{15-3} \leftarrow LRB_{12-0}$ $USP_{2-0} \leftarrow 0$
TRNS LRB, USP	$LRB_{12-0} \leftarrow USP_{15-3}$ $LRB_{15-13} \leftarrow 0$

Instruction	Function
LB A,*	$A_L \leftarrow *$
STB A,*	$A_L \rightarrow *$
MOVB A,*	$A_L \leftarrow *$
MOVB *, A	* ← AL
MOVB *, #	* ← Imme.
MOVB r,*	r ← *
MOVB d,*	(d) ← *
MOVB PSW0,*	PSW0 ← *
MOVB SCB,*	SCB ← *
CLRB *	* ← 0
SWAPB	$A_{7-4} \longleftrightarrow A_{3-0}$
XCHGB	$A_L \longleftrightarrow *$

Push & Pop

Instruction	Function
PUSHU *	* → USER STACK
POPU A	A ← USER STACK
PUSHS *	* → SYSTEM STACK
POPS *	* ← SYSTEM STACK

Instruction	Function
PUSHUB *	* → USER STACK
POPUB A	A_L ← USER STACK

Rotate & Shift

Instruction	Function
ROL *	Rotate ←
ROR *	Rotate →
SLL *	Shift ← Logical
SRL *	Shift → Logical
SRA *	Shift → Arithmetic

Instruction	Function
ROLB *	Rotate ←
RORB *	Rotate →
SLLB *	Shift ← Logical
SRLB *	Shift → Logical
SRAB *	Shift → Arithmetic

Increment & Decrement

Instruction	Function
INC *	* ← *+1
DEC *	* ← *−1

Instruction	Function
INCB *	* ← * + 1
DECB *	* ← * − 1

ROM Table Reference

Instruction	Function
LC A,*	A ← * (ROM)
CMPC A,*	A − * (ROM)

Instruction	Function
LCB A, *	A_L ← * (ROM)
CMPCB A, *	A_L − * (ROM)

Arithmetic Operation

Instruction		Function
MUL		er1: A ← A×er0
DIV		er0: A ← er0: A/er2
ADD	A, *	A ← A+*
ADD	*, A	* ← *+A
ADD	*, d	* ← *+(d)
ADD	*, #	* ← *+Imme.
ADC	A, *	A ← A+*+C
ADC	*, A	* ← *+A+C
ADC	*, d	* ← *+(d)+C
ADC	*, #	* ← *+Imme.+C
SUB	A, *	A ← A−*
SUB	*, A	* ← *−A
SUB	*, d	* ← *−(d)
SUB	*, #	* ← *−Imme.
SBC	A, *	A ← A−*−C
SBC	*, A	* ← *−A−C
SBC	*, d	* ← *−(d)−C
SBC	*, #	* ← *−Imme.−C

Instruction		Function
MULB		A ← A_L×r0
DIVB		A ← A/r0
ADDB	A, *	A_L ← A_L+*
ADDB	*, A	* ← *+A_L
ADDB	*, d	* ← *+(d)
ADDB	*, #	* ← *+Imme.
ADCB	A, *	A_L ← A_L+*+C
ADCB	*, A	* ← *+A_L+C
ADCB	*, d	* ← *+(d)+C
ADCB	*, #	* ← *+Imme.+C
SUBB	A, *	A_L ← A_L−*
SUBB	*, A	* ← *−A_L
SUBB	*, d	* ← *−(d)
SUBB	*, #	* ← *−Imme.
SBCB	A, *	A_L ← A_L−*−C
SBCB	*, A	* ← *−A_L−C
SBCB	*, d	* ← *−(d)−C
SBCB	*, #	* ← *−Imme.−C

Logical Operation

Instruction		Function
AND	A,*	A ← A and *
AND	*, A	* ← * and A
AND	*, d	* ← * and (d)
AND	*, #	* ← * and Imme.
OR	A, *	A ← A or *
OR	*, A	* ← * or A
OR	*, d	* ← * or (d)
OR	*, #	* ← * or Imme.
XOR	A, *	A ← A xor *
XOR	*, A	* ← * xor A
XOR	*, d	* ← * xor (d)
XOR	*, #	* ← * xor Imme.

Instruction		Function
ANDB	A, *	A_L ← A_L and *
ANDB	*, A	* ← * and A_L
ANDB	*, d	* ← * and (d)
ANDB	*, #	* ← * and Imme.
ORB	A, *	A_L ← A_L or *
ORB	*, A	* ← * or A_L
ORB	*, d	* ← * or (d)
ORB	*, #	* ← * or Imme.
XORB	A, *	A_L ← A_L xor *
XORB	*, A	* ← * xor A_L
XORB	*, d	* ← * xor (d)
XORB	*, #	* ← * xor Imme.

Comparison

Instruction		Function
CMP	A, *	A − *
CMP	*, A	* − A
CMP	*, d	* − (d)
CMP	*, #	* − Imme.

Instruction		Function
CMPB	A, *	A_L − *
CMPB	*, A	* − A_L
CMPB	*, d	* − (d)
CMPB	*, #	* − Imme.

Stack Operation

Instruction	Function
ADD A. [+USP]	USP ← USP + 2, A ← A + [USP]
ADC A. [+USP]	USP ← USP + 2, A ← A + [USP] + C
SUB A. [+USP]	USP ← USP + 2, A ← A − [USP]
SBC A. [+USP]	USP ← USP + 2, A ← A − [USP] − C
AND A. [+USP]	USP ← USP + 2, A ← A and [USP]
OR A. [+USP]	USP ← USP + 2, A ← A_{or} [USP]
XOR A. [+USP]	USP ← USP + 2, A ← A_{xor} [USP]
CMP A. [+USP]	USP ← USP + 2, A − [USP]

Decimal Adjust

Instruction	Function
DAA	Decimal Adjust for Addition
DAS	Decimal Adjust for Substruct

6

Bit Operation

Instruction	Function
SBR *	bit ← 1
RBR *	bit ← 0
MBR C, *	C ← bit
MBR *, C	bit ← C
TRB *	Z ← bit

Instruction	Function
SB *	bit ← 1
RB *	bit ← 0
MB C, *	C ← bit
MB *, C	bit ← C

Execute

Instruction	Function
EX *	Execute Specified Data as the Instruction

Jump & Call

Instruction	Function
SJ adrs	Short Jump
J *	Jump
JC EQ, adrs	Jump if '='
JC LE, adrs	Jump if '< ='
JC GE, adrs	Jump if '> ='
JBS bit, adrs	Jump if bit-on
JRNZ, DP, adrs	Loop Function

Instruction	Function
SCAL adrs	Short Call
CAL *	Call Subroutine
JC NE, adrs	Jump if '≠'
JC LT, adrs	Jump if '<'
JC GT, adrs	Jump if '<'
JBR bit, adrs	Jump if bit-off
VCAL adrs	Vector Call

Return

Instruction	Function
RT	Return from Subroutine
RTI	Return from Interrupt

String Operation

Instruction	Function
SMOVI	String Transfer with Increasing Pointers
SMOVD	String Transfer with Decreasing Pointers
SCMP	String Compare

Others

Instruction	Function
SC	C ← 1
SS	STACK FLAG ← 1
NOP	No Operation

Instruction	Function
RC	C ← 0
BRK	Software Reset

(Note)

'*'	Addressing expression (See Addressing mode)
'#' and 'imme.'	Immediate value
'←' and '→'	Bit-shift direction

ABSOLUTE MAXIMUM RATING (TARGET SPECIFICATION)

Item	Symbol	Condition	Value	Unit
Power Supply Voltage	V_{DD}	Ta = 25°C	−0.3 ~ 7.0	V
Input Voltage	V_I		−0.3 ~ V_{DD}	V
Output Voltage	V_O		−0.3 ~ V_{DD}	V
Analog Reference Voltage	V_R		−0.3 ~ V_{DD}	V
Analog Input Voltage	V_{AI}		−0.3 ~ V_R	V
Power Desipation	P_D	Ta = 25°C, per Package	400 Max.	mW
		Ta = 25°C, per Output	50 Max.	mW
Storage Temperature	T_{STG}	–	−55 ~ 150	°C

OPERATING RANGE

6

Item	Symbol	Condition		Value	Unit
Power Supply Voltage	V_{DD}	f (OSC) ≦ 10 MHz		4.5 ~ 5.5	V
Data Retention Voltage	V_{DDH}	f (OSC) = 0 Hz		2 ~ 6	V
Analog Reference Voltage	V_R	–		4.5 ~ V_{DD}	V
Analog Ground Voltage	V_{AG}			0 ~ 0.2	V
Analog Input Voltage	V_{AI}			V_{AG} ~ V_R	V
Fan Out	N	MOS Load		30	
		TTL Load	P_{00} ~ P_{07}	2	
			P_{10} ~ P_{47}	1	

DC CHARACTERISTICS (TARGET SPECIFICATION)

(V_{DD} = 5V ±10%, Ta = −40 ~ +85°C)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
"H" Input Voltage *1,*2,*3,*7	V_{IH}	—	2.4		V_{DD}	V
"H" Input Voltage *6,*7,*9	V_{IH}	—	4.0		V_{DD}	V
"L" Input Voltage *1,*2,*3,*7	V_{IL}	—	−3.0		0.8	V
"L" Input Voltage *6,*8,*9	V_{IL}	—	−0.3		0.8	V
"H" Output Voltage *1,*2,*4,*5	V_{OH}	I_O = −30 μA	4.2			V
"H" Output Voltage *10	V_{OH}	I_O = −15 μA	4.2			V
"L" Output Voltage *1,*4	V_{OL}	I_O = 3.2 mA			0.45	V
"L" Output Voltage *2,*5	V_{OL}	I_O = 1.6 mA			0.45	V
"L" Output Voltage *10	V_{OL}	I_O = −15 μA			0.4	V
Input Current *1,*2,*3,*7,*8	I_{IH}/I_{ILI}	V_I = VDD/0V			1/−1	μA
Input Current *6	I_{IH}/I_{IL}	V_I = VDD/0V			1/−20	μA
Input Current *9	I_{IH}/I_{IL}	V_I = VDD/0V			10/−10	μA
"H" Output Current *1,*2,*4,*5	I_{OH}	V_O = 2.4V	−0.2			mA
Analog Reference Voltage Supply Resistance	R_R	V_{AG} = 0V		16		KΩ
Analog Input Leak Current	R_{LAI}	V_{AG} = 0V			±1	μA
Analog Input Impedance	R_{AI}	A/D Converter Active and In Sampling Condition		1		MΩ
	C_{AI}			10		pF
Input Capacity	C_1	f = 1MHz, Ta = 25°C		5		pF
Output Capacity	C_0	f = 1MHz, Ta = 25°C		7		pF
Current Consumption (Stop)	I_{DDS}	V_{DD} = 2V, No Load, Ta = 25°C		0.2		μA
		No Load		1		μA
Current Consumption	I_{DD}	f (OSC) = 10MHz, No Load		20		mA

*1 Specification applied to P0
*2 Specification applied to P1, P2, P3, P4
*3 Specification applied to P5
*4 Specification applied to ALE, PSEN, RD, WR
*5 Specification applied to RESOUT
*6 Specification applied to RES, NMI
*7 Specification applied to READY, EA
*8 Specification applied to FLT
*9 Specification applied to OSC_0
*10 Specification applied to OSC_1

AC CHARACTERISTICS (TARGET SPECIFICATION)

External Program Memory Control

(V_{DD} = 5V ±10%, Ta = –40 ~ +85°C)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
OSC Clock Pulse Width	$t_{\phi W}$	–	50			nS
ALE Pulse Width	t_{AW}	–	100			nS
PSEN Pulse Width	t_{PW}	–	250			nS
PSEN Pulse Delay Time	t_{PAD}	C_L = 50pF			$t_{\phi W}$+20	nS
"L" Address Set Up Time	t_{AAS}		$t_{\phi W}$-30		$t_{\phi W}$+30	nS
"L" Address Hold Time	t_{AAH}		$t_{\phi W}$-30		$t_{\phi W}$+30	nS
"H" Address Delay Time	t_{AAD}				$t_{\phi W}$+30	nS
"H" Address Hold Time	t_{APH}		$t_{\phi W}$-10		$t_{\phi W}$+40	nS
Instruction Set Up Time	t_{IS}		80			nS
Instruction Hold Time	t_{IH}		50			nS

External Data Memory Control

(V_{DD} = 5V ±10%, Ta = –40 ~ +85°C)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
OSC Clock Pulse Width	$t_{\phi W}$	–	50			nS
ALE Pulse Width	t_{AW}	–	100			nS
RD Pulse Width	t_{RW}	–	250			nS
WR Pulse Width	t_{WW}	–	200			nS
RD Pulse Delay Time	t_{RAD}	C_L = 50pF			$t_{\phi W}$+20	nS
WR Pulse Delay Time	t_{WAD}				$t_{\phi W}$+20	nS
"L" Address Set Up Time	t_{AAS}		$t_{\phi W}$-30		$t_{\phi W}$+30	nS
"L" Address Hold Time	t_{AAH}		$t_{\phi W}$-30		$t_{\phi W}$+30	nS
"H" Address Delay Time	t_{AAD}				$t_{\phi W}$+30	nS
"H" Address Hold Time (RD)	t_{ARH}		$t_{\phi W}$-10		$t_{\phi W}$+40	nS
"H" Address Hold Time (WR)	t_{AWH}		$2t_{\phi W}$-10		$2t_{\phi W}$+40	nS
Memory Data Set Up Time	t_{MS}		80			nS
Memory Data Hold Time	t_{MH}		50			nS
Data Delay Time	t_{DD}				$t_{\phi W}$+30	nS
Data Hold Time	t_{DH}		$2t_{\phi W}$-10		$2t_{\phi W}$+40	nS

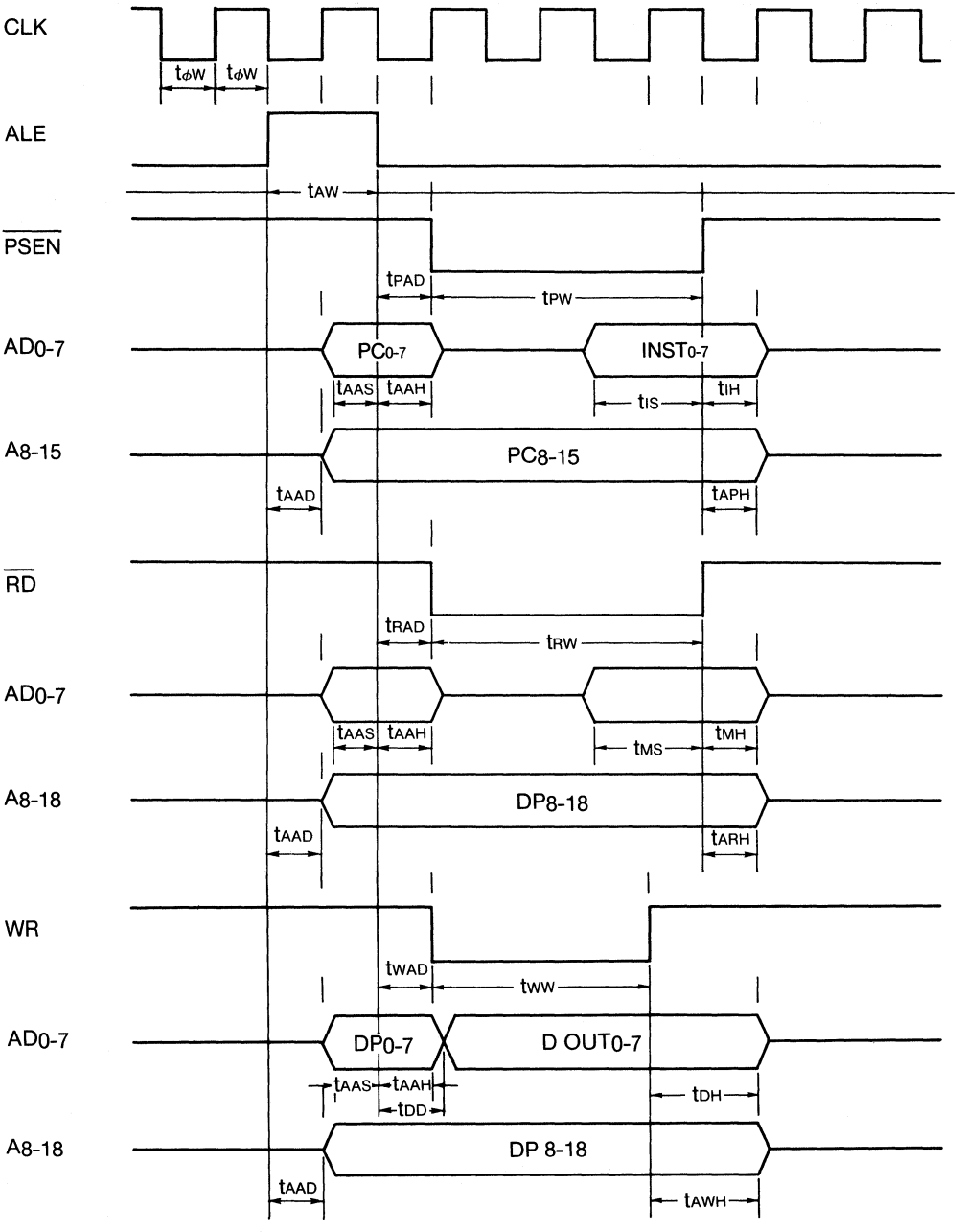
A/D CONVERSION CHARACTERISTICS (TARGET SPECIFICATION)

(V_{DD} = 5V ±10%,)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Resolution	n	$V_R = V_{DD}$ V_{AG} = GND = 0V Analog Input Impedance ≦ 5KΩ Ta = 25°C		10		Bits
Zero Point Error	E_{ZS}			±2		LSB
Full Scale Error	E_{FS}			±2		LSB
Linearity Error	E_L			±3		LSB
Input Crosstalk				±2		LSB
Channel Conversion Speed	t_C		64		320	μs/CH

6

TIMING CHART



8 BIT SERIES (INTEL COMPATIBLE)

MSM80C35/48
MSM80C39/49
MSM80C40/50

CMOS 8-BIT SINGLE CHIP MICROCONTROLLER

GENERAL DESCRIPTION

The OKI MSM80C48/MSM80C49/MSM80C50 microcontroller is a low-power, high-performance 8-bit single chip device implemented in silicon gate complementary metal oxide semiconductor technology. Integrated within these chips are 8K/16K/32K bits of mask program ROM, 512/1024/2048 bits of data RAM, 27 I/O lines, built-in 8 bit timer/counter, and oscillator. Program memory and data paths are byte wide. Eleven new instructions have been added to the NMOS version's instruction set, thereby optimizing power down, port data transfer, decrement and port float functions.

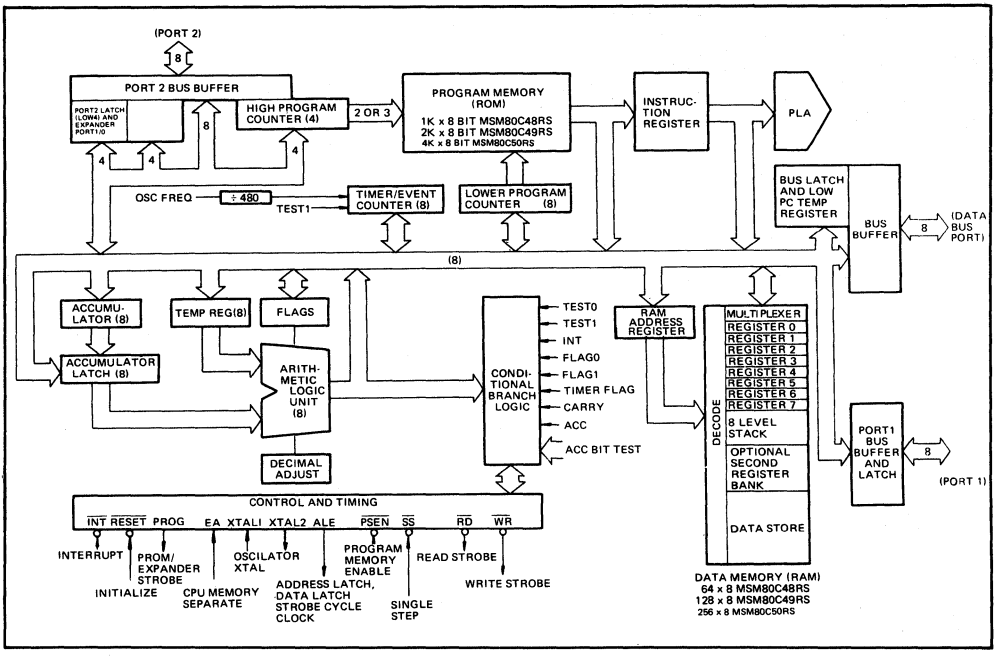
Available in 40-pin plastic DIP (RS) or 44-pin plastic flat packages (GSK).

FEATURES

- Lower power consumption enabled by CMOS silicon gate process
- Completely static operation
- Improved power-down feature
- Minimum instruction cycle 1.36 μs (11MHz) @ V_{CC} = +5V ±10%
  11 MHz version of MSM80C40/50 (6 MHz < XTAL1.2 < 11 MHz) is under development.
- Every signal input terminal is provided with a Schmitt circuit, except XTAL1 Pin.
- Every signal output terminal is capable of driving a standard TTL, except XTAL2 Pin.
- 111 instructions
- All instructions are usable even during execution of external ROM instructions.
- Operation facility
  Addition, logical operations, and decimal adjust
- Program memory (ROM) : 1K × 8 bits (MSM80C48)
  : 2K × 8 bits (MSM80C49)
  : 4K × 8 bits (MSM80C50)
- Data memory (RAM) : 64 × 8 bits (MSM80C48)
  : 128 × 8 bits (MSM80C49)
  : 256 × 8 bits (MSM80C50)
- Two sets of working registers
- External and timer interrupts
- Two test inputs
- Built-in 8-bit timer counter
- Extendable external memory and I/O ports
- Input/output ports : Input/output ports – 8 bits × 2
  : Data bus input/output – 8 bits × 1
- Single-step execution function
- Every signal input terminal is provided with a Schmitt circuit, except XTAL 2 Pin
- Every signal output terminal is capable of driving a standard TTL, except X'tal 2 Pin.
- Wide range of operating voltage, from +2.5V to +6V of V_{CC}.
- High noise margin action
- Two kinds of package; 40-pin plastic DIP and 44-pin plastic flat package
- Compatible with Intel's 8048, 8049 and 8050

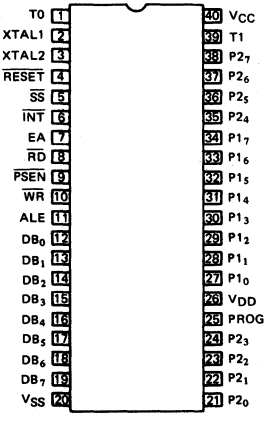
6

FUNCTIONAL BLOCK DIAGRAM

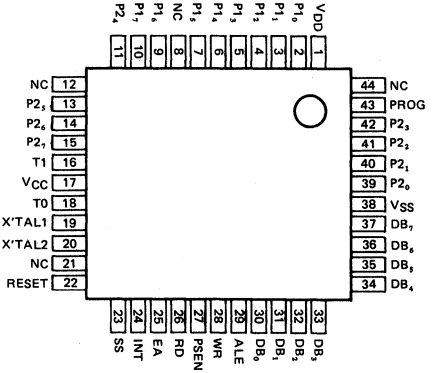


PIN CONFIGURATION

(Top View) 40 Lead Plastic DIP



(Top View) 44 Lead Plastic FLAT



Pin Name

$P1_0 \sim P1_7$	: Input/output port (PORT1)	ALE	: Address Latch Enable
$P2_0 \sim P2_7$	: Input/output port (PORT2)	$\overline{PSEN}$	: Program Store Enable
$DB_0 \sim DB_7$	: Data bus port (BUS PORT)	$\overline{RESET}$	: Reset
T0, T1	: Test	$\overline{SS}$	: Single Step
$\overline{INT}$	: Interrupt	EA	: ROM Mode
$\overline{RD}$	: Read	XTAL 1, 2	: Crystal Controlled Oscillator
$\overline{WR}$	: Write		

PIN DESCRIPTION

Designation	Input/Output	Function
$P1_0 \sim P1_7$ (PORT 1)	Input/Output	8-bit quasi-bidirectional port
$P2_0 \sim P2_7$ (PORT 2)	Input/Output	8-bit quasi-bidirectional port The high-order four bits of external program memory addresses can be output from P20-P23, to which the I/O expander MSM82C43RS may also be connected.
$DB_0 \sim DB_7$ (BUS)	Input/Output	Bidirectional port The low-order eight bits of external program memory address can be output from this port, and the addressed instruction is fetched under the control of $\overline{PSEN}$ signal. Also, the external data memory address is output, and data is read and written synchronously using $\overline{RD}$ and $\overline{WR}$ signals. The port can also serve as either a statically latched output port or a non-latching input port.
T0 (Test 0)	Input/Output	The input can be tested with the conditional jump instructions JT0 and JNT0. The execution of the ENT0 CLK instruction causes a clock output to be generated.
T1	Input	The input can be tested with the conditional jump instructions JT1 and JNT1. The execution of a STRT CNT instruction causes an internal counter input to be activated.
$\overline{INT}$ (Interrupt)	Input	Interrupt input. If interrupt is enabled, $\overline{INT}$ input initiates an interrupt. Interrupt is disabled after a reset. Also testable with a JNI instruction. Can be used to terminate the power-down mode. (Active "0" level)
$\overline{RD}$ (Read)	Output	A signal to read data from external data memory. (Active "0" level)
$\overline{WR}$ (Write)		A signal to write data to external data memory. (Active "0" level)
ALE Address & Data Latch Clock		This signal is generated in each cycle. It may be used as a clock output. External data memory or external program memory is addressed upon the falling edge. For the external ROM, this signal is used to latch the bus port data upon the ALE signal rise-up after the execution of the OUTL BUS, A instruction.
$\overline{PSEN}$ Program Store Enable	Output	A signal to fetch an instruction from external program memory (Active "0" level)
$\overline{RESET}$	Output	($\overline{RESET}$) input initialize the processor. (Active "0" level) Used to terminate the power-down mode.
$\overline{SS}$ (Single Step)	Output	A program is executed step by step. This pin can also be used to control internal oscillation when the power-down mode is reset. (Active "0" level)
EA (External Access)	Input	When held at high level, all instructions are fetched from external memory. (Active "1" level)
$\overline{PROG}$ (Expander Strobe)	Output	This output strobes the MSM82C43RS I/O expander.

PIN DESCRIPTION (CONT.)

Designation	Input/Output	Function
XTAL 1 (Crystal 1)	Input	One side of the crystal input for the internal oscillator. An external source can also be input.
XTAL 2 (Crystal 2)	Output	Other side of Crystal input for internal oscillator.
V_{CC}	–	Power supply terminal
V_{DD}	–	Standby control input. Normally, "1" level. When set to "0" level, oscillation is stopped and processor goes into standby mode.
V_{SS}	–	GND

Note: The required $\overline{\text{RESET}}$ pulse duration is at least two machine cycles under the condition that the power supply and the oscillator have been stabilized.

ADDED FUNCTIONS OF MSM80C48, MSM80C49 AND MSM80C50

The MSM80C48, MSM80C49 and MSM80C50 basically incorporate the capabilities of Intel's 8048, 8049, and 8050 plus the following new functions:

1. Power-Down Mode Enhancements

1.1 Power-down by software

(1) Clock (See item 4, "Power-down mode", for details.)
   a. Crystal-controlled oscillator halt (HLTS instruction)
      Power requirements can be minimized.
   b. Clock supply halt (HALT instruction)
      Restart is accomplished without oscillator wait.

(2) I/O ports (See Table 4-1 and 4-2 for details.)
   I/O port floating instructions
   Power consumption resulting from inputs/outputs can be minimized with FLT and FLTT instructions.
   Port floating is cancelled by executing FRES instruction, "0" level at $\overline{INT}$ pin or "0" level at $\overline{RESET}$ pin.

(3) Six types of power-down can be done by a combination of HLTS/HALT and FLT/FLTT instructions.

1.2 Power-down by hardware (See 4.3, Power-down mode by V_{DD} pin utilization for details.)

Crystal-controlled oscillators can be halted by controlling the V_{DD} terminal, thereby floating all I/O ports for minimum power consumption.

2. Additional Instructions (11)

HLTS	MOV A, P2
HALT	MOV P1, @ R3
FLT	MOVP1 P, @ R3
FLTT	DEC @ Rr
FRES	DJNZ @ R, addr
MOV A, P1	

3. Improved Uses of BUS $P_{0\sim7}$, $P1_{0\sim7}$, $P2_{0\sim7}$, and $\overline{SS}$ terminals

3.1 BUS $P_{0\sim7}$

The MSM80C48, MSM80C49, and MSM80C50 remove the limitation on the use of OUTL BUS, A instructions during the external ROM access mode by having an independent data latch and external ROM mode address latch in BUS $P_{0\sim7}$.

Consequently, there is no need to relocate bus port instructions when in the external ROM access mode.

3.2 $P1_{0\sim7}$ and $P2_{0\sim7}$

The MSM80C48, MSM80C49 and MSM80C50 are designed to minimize power consumption when $P1_{0\sim7}$ and $P2_{0\sim7}$ are used as input/output ports, to maximize the performance of CMOS.

When these ports are used as output ports, the acceleration circuit is actuated only when output data changes from "0" to "1", thus speeding up the rise time of the output signals.

When these ports are used as input ports, the internal pullup resistance becomes approximately 9 kΩ when input data is "1".

The internal pullup resistance rises to approximately 100 kΩ when input data is "0". Thus, a high noise margin can be obtained by selecting the impedance and thus the outflow of current is minimized whenever these ports are used as output or input ports.

3.3 Clock generation control via the $\overline{SS}$ terminal

When the crystal-controlled oscillator is halted in the HLTS or hardware power-down mode, the $\overline{SS}$ terminal is pulled down by a resistor of 20 – 50 kΩ, while its internal pullup resistor of 200 – 500 kΩ is isolated from V_{CC}. When the power-down mode is cancelled, the internal resistor of the $\overline{SS}$ terminal is changed from pull-down to pullup. Consequently, the CPU can be halted for any period of time until the crystal-controlled oscillator resumes normal oscillation when a capacitor is connected to the SS terminal.

4. Power-Down Mode

The MSM80C48, MSM80C49, and MSM80C50 power-down mode can be enabled in 2 different ways-through software by a combination of clock control and port floating instructions, and through hardware by control of the V_{DD} pin.

4.1 Software power-down mode

Power-down mode can be done by a combination of the following instructions.

(1) HALT (clock supply halt to control circuit)

Instruction code:

0	0	0	0	0	0	0	1

Description: Although crystal-controlled oscillator operation is continued, the clock supply to the CPU control circuit is halted and CPU operations suspended. When cancelling this software mode, restart is accomplished without oscillator wait. Timing charts are outlined in Figs. 4-1 and 4-2.

(2) HLTS (oscillation stop)

Instruction code:

1	0	0	0	0	0	1	0

Description: The oscillator operation is halted and CPU operations suspended. In cancelling this power down mode, connecting a capacitor to the $\overline{SS}$ pin enables a reasonable

wait period to be accomplished before normal operation is resumed. [Except in the case of using the $\overline{\text{RESET}}$ pin]
Timing charts are outlined in Figs. 4-3 and 4-4.

(3) FLT (floating $P1_0 \sim 7$, $P2_0 \sim 7$, and $BP_0 \sim 7$)

Instruction code: 1 0 1 0 0 0 1 0

Description:

	Internal ROM mode	External ROM mode
P1	Floating	Floating
P2	Floating	$P2_0 \sim 3$ operation $P2_4 \sim 7$ floating
BP	Floating	Operation

Details of IC pin status as a result of executing the FLT instruction are shown in Table 4-1.

(4) FLTT (floating of all output pins)

Instruction code: 1 1 0 0 0 0 1 0

Description:

	Internal ROM mode	External ROM mode
ALE	Floating	Operation
$\overline{\text{PSEN}}$	Floating	Operation
PROG	Floating	Floating
$\overline{\text{WR}}$	Floating	Floating
$\overline{\text{RD}}$	Floating	Floating
TO OUT	Floating	Floating
P1	Floating	Floating
P2	Floating	$P2_0 \sim 3$ operation $P2_4 \sim 7$ floating
BP	Floating	Operation
XTAL	Operation	Operation

Details of IC pin status as a result of executing the FLTT instruction are shown in Table 4-2.

Example 1: Power-down mode accomplished by stopping oscillation.
- ○ Setting by execution of HLTS [82H] instruction.

Example 2: Power-down mode accomplished by stopping the clock supply to the CPU control circuit.
- ○ Setting by execution of HALT [01H] instruction.

Example 3: Power-down mode by floating of $P1_0 \sim 7$, $P2_0 \sim 7$ and $BP_0 \sim 7$, and subsequent stopping of CPU oscillation.
- ○ Setting by first executing the FLT[A2H] instruction and then the HLTS[82H] instruction.

Example 4: Power-down mode by floating $P1_0 \sim 7$, $P2_0 \sim 7$ and $BP_0 \sim 7$, and then stopping the clock supply to the CPU control circuit.
- ○ Setting by first executing the FLT[A2H] instruction, and then the HALT[01H] instruction.

Example 5: Power-down mode by floating all output pins, followed by stopping oscillation.
- ○ Setting by first executing the FLTT[C2H] instruction followed by execution of the HLTS[82H] instruction.

Example 6: Power-down mode by floating all output pins, followed by stopping of the clock supply to the CPU control circuit.
- ○ Setting by first executing the FLTT[C2H] instruction, followed by execution of the HALT[01H] instruction.

4.2 Cancellation of software power-down mode

The power-down mode status outlined above in examples 1 to 6 can be cancelled by using either the interrupt pin or the $\overline{\text{RESET}}$ pin.

(1) Use of the $\overline{\text{INT}}$ pin during external interrupt enabled mode (i.e. following execution of EN I instruction).
- ○ The clock generator is activated and the CPU started up when a "0" level is applied to the $\overline{\text{INT}}$ pin. If this "0" level is maintained until at least 2 ALE output signals occur, an external interrupt is generated, and execution proceeds from address 3. If, however, the power-down mode has been done during the interrupt processing routine, execution is resumed just after the power-down instruction.

(2) Use of the $\overline{\text{INT}}$ pin during external interrupt disabled mode (i.e. following execution of DIS I instruction or hardware reset)
- ○ The clock generator is activated and the CPU started up when a "0" level is applied to the $\overline{\text{INT}}$ pin. If this "0" level is maintained until at least 2 ALE output signals occur, execution is resumed just after the power-down instruction.

(3) Use of the $\overline{\text{RESET}}$ pin
- ○ The clock generator is activated and the CPU started up when a "0" level is applied to the $\overline{\text{RESET}}$ pin. If this "0" level is maintained until at least 2 ALE output signals occur, the CPU is reset and execution proceeds from address 0. In case cancellation is done in oscillation stop mode, the "0" level must be input to the $\overline{\text{RESET}}$ PIN until oscillation is stabilized.

Table 4-1 Details of Pin Status Following Execution of FLT Instruction

Pin No.	Pin Name	Internal ROM	External ROM
1P	T0	Active	Active
2P	XTAL1	Active	Active
3P	XTAL2	Active	Active
4P	$\overline{\text{RESET}}$	Active	Active
5P	$\overline{\text{SS}}$	200 ~ 500 kΩ pullup	200 ~ 500 kΩ pullup
6P	$\overline{\text{INT}}$	Active	Active
7P	EA	Active	Active
8P	$\overline{\text{RD}}$	Active	Active
9P	$\overline{\text{PSEN}}$	Active	Active
10P	$\overline{\text{WR}}$	Active	Active
11P	ALE	Active	Active
12P	DB0	Floating	Active
13P	DB1	Floating	Active
14P	DB2	Floating	Active
15P	DB3	Floating	Active
16P	DB4	Floating	Active
17P	DB5	Floating	Active
18P	DB6	Floating	Active
19P	DB7	Floating	Active
20P	V_{SS}	0 [V]	0 [V]
21P	P20	Floating	Active
22P	P21	Floating	Active
23P	P22	Floating	Active
24P	P23	Floating	Active
25P	PROG	Active	Active
26P	V_{DD}	"1" level	"1" level
27P	P10	Floating	Floating
28P	P11	Floating	Floating
29P	P12	Floating	Floating
30P	P13	Floating	Floating
31P	P14	Floating	Floating
32P	P15	Floating	Floating
33P	P16	Floating	Floating
34P	P17	Floating	Floating
35P	P24	Floating	Floating
36P	P25	Floating	Floating
37P	P26	Floating	Floating
38P	P27	Floating	Floating
39P	T1	Active	Active
40P	V_{CC}	+2 to +6 [V]	+2 to +6 [V]

Note: The FLT mode itself is reset by executing the FRES instruction, or supplying "0" level to $\overline{\text{INT}}$ or $\overline{\text{RESET}}$ pin.

Table 4-2 Details of Pin Status Following Execution of FLTT Instruction

Pin No.	Pin Name	Internal ROM	External ROM
1P	T0	Floating if output enabled	Floating if output enabled
2P	XTAL1	Active	Active
3P	XTAL2	Active	Active
4P	$\overline{\text{RESET}}$	Active	Active
5P	SS	200 to 500 kΩ pullup	200 to 500 kΩ pullup
6P	$\overline{\text{INT}}$	Active	Active
7P	EA	Active	Active
8P	$\overline{\text{RD}}$	Floating	Floating
9P	$\overline{\text{PSEN}}$	Floating	Active
10P	$\overline{\text{WR}}$	Floating	Floating
11P	ALE	Floating	Active
12P	DB0	Floating	Active
13P	DB1	Floating	Active
14P	DB2	Floating	Active
15P	DB3	Floating	Active
16P	DB4	Floating	Active
17P	DB5	Floating	Active
18P	DB6	Floating	Active
19P	DB7	Floating	Active
20P	V_{SS}	0 [V]	0 [V]
21P	P20	Floating	Active
22P	P21	Floating	Active
23P	P22	Floating	Active
24P	P23	Floating	Active
25P	PROG	Floating	Floating
26P	V_{DD}	"1" level	"1" level
27P	P10	Floating	Floating
28P	P11	Floating	Floating
29P	P12	Floating	Floating
30P	P13	Floating	Floating
31P	P14	Floating	Floating
32P	P15	Floating	Floating
33P	P16	Floating	Floating
34P	P17	Floating	Floating
35P	P24	Floating	Floating
36P	P25	Floating	Floating
37P	P26	Floating	Floating
38P	P27	Floating	Floating
39P	T1	Active	Active
40P	V_{CC}	+2.5 to +6 [V]	+2.5 to +6 [V]

Note: The FLTT mode itself is reset by executing the FRES instruction, or supplying "0" level to $\overline{\text{INT}}$ or $\overline{\text{RESET}}$ pin.

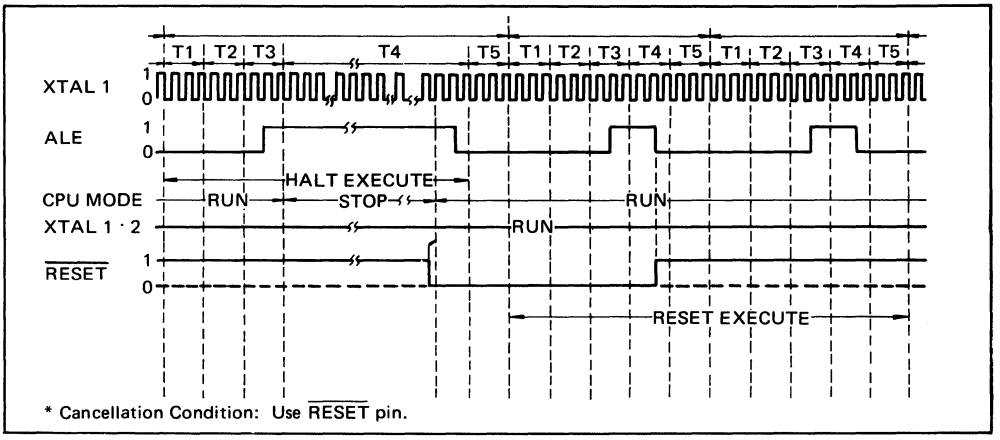


Fig. 4-1 HALT [01H] Instruction Execution Timing Chart

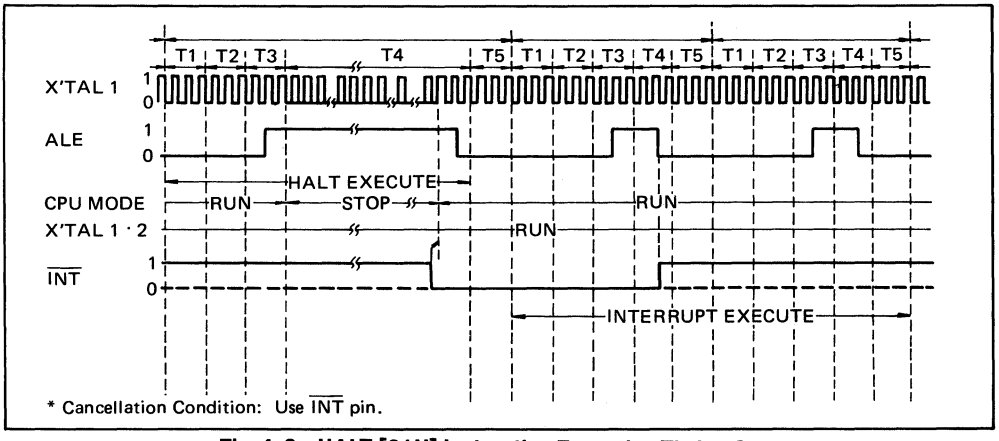


Fig. 4-2 HALT [01H] Instruction Execution Timing Chart

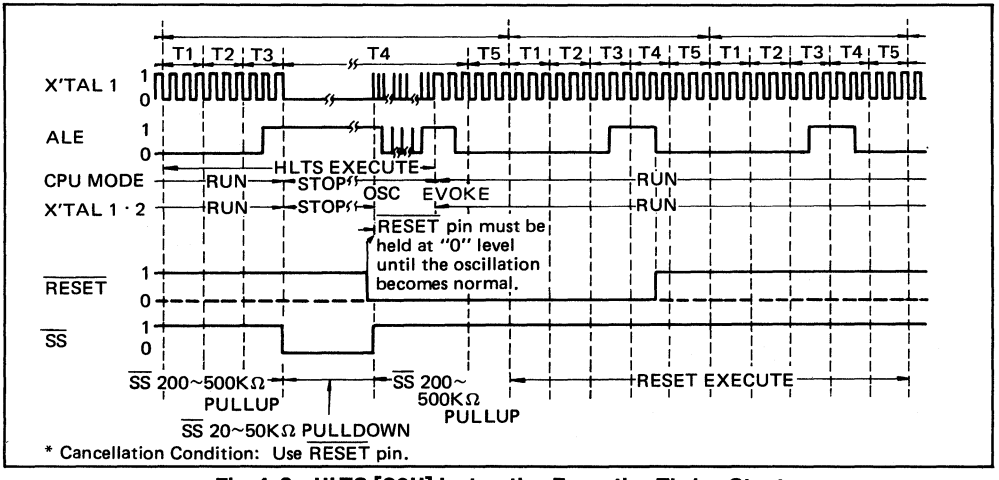
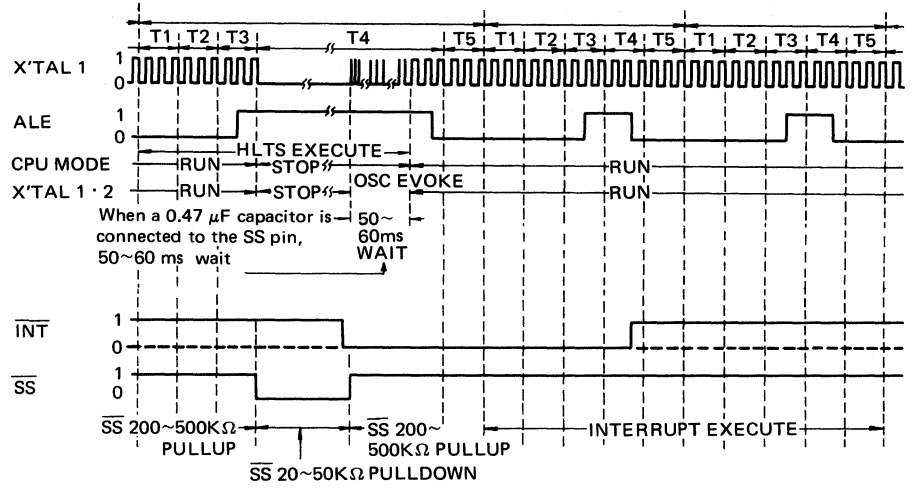


Fig. 4-3 HLTS [82H] Instruction Execution Timing Chart



* Cancellation Condition: Use $\overline{INT}$ pin.

Fig. 4-4 HLTS [82H] Instruction Execution Timing Chart

4.3 Hardware power-down mode

In the MSM80C48, MSM80C49 and MSM80C50, forcing the level at the V_{DD} pin [pin 26] to a "0" during either external ROM or internal ROM mode results in suspension of the oscillator function and subsequent floating (high impedance) of all the I/O pins except the $\overline{RESET}$, $\overline{SS}$ and XTAL 1/2 pins. The CPU is thereby stopped while maintaining internal status. Details of the IC pin status at this time are outlined in Table 4-3.

4.4 Cancellation of hardware power-down mode

(1) Use of $\overline{RESET}$ pin

○ The clock generator is activated and the CPU started up when a "1" level is applied to the V_{DD} pin while a "0" level is input to the $\overline{RESET}$ pin. If this "0" level is kept applied to the $\overline{RESET}$ pin until oscillation become stable, the CPU will be reset and will start executing from address 0. The timing chart is outlined in Fig. 4-5.

(2) Use of the $\overline{INT}$ pin during external interrupt enabled status (i.e. following execution of EN I instruction)

○ The clock generator is activated and the CPU started up when a "1" level is applied to the V_{DD} pin while a "0" level is applied to the $\overline{INT}$ pin.

If this "0" level is maintained until at least 2 ALE output signals occur, an external interrupt is generated, and execution starts from address 3.

However, if the power-down mode is started during an interrupt processing routine, execution will be continued on the next instruction after the present instruction. The timing chart is outlined in Fig. 4-6.

(3) Use of the $\overline{INT}$ pin during external interrupt disabled mode (i.e. following execution of DIS I instruction or hardware reset)

○ The clock generator is activated and the CPU started up when a "1" level is applied to the V_{DD} pin while a "0" level is applied to the $\overline{INT}$ pin. If this "0" level is maintained until at least 2 ALE output signals occur, execution is continued on the next instruction after the present instruction. The timing chart is outlined in Fig. 4-6.

(4) Use of V_{DD} pin only

○ The clock generator is activated and the CPU started up when a "1" level is applied to the V_{DD} pin while a "1" level is also applied to both the $\overline{RESET}$ and $\overline{INT}$ pins. In this case, execution is resumed from the stopped position. The timing chart is outlined in Fig. 4-7.

Table 4-3 Details of Pin Status during Hardware Power-Down Mode

Pin No.	Pin Name	Normal Operation (V_{DD} = "1" level)	Power Down Mode (V_{DD} = "0" level)
1P	T0	Active	Floating if output enabled
2P	XTAL1	Active	Active
3P	XTAL2	Active	Active
4P	$\overline{RESET}$	Active	Active
5P	$\overline{SS}$	200 to 500 kΩ pullup	20 to 50 kΩ pulldown
6P	$\overline{INT}$	Active	Active
7P	EA	Active	Active
8P	$\overline{RD}$	Active	Floating
9P	$\overline{PSEN}$	Active	Floating
10P	$\overline{WR}$	Active	Floating
11P	ALE	Active	Floating
12P	DB0	Active	Floating
13P	DB1	Active	Floating
14P	DB2	Active	Floating
15P	DB3	Active	Floating
16P	DB4	Active	Floating
17P	DB5	Active	Floating
18P	DB6	Active	Floating
19P	DB7	Active	Floating
20P	V_{SS}	0 [V]	0 [V]
21P	P20	Active	Floating
22P	P21	Active	Floating
23P	P22	Active	Floating
24P	P23	Active	Floating
25P	PROG	Active	Floating
26P	V_{DD}	"1" level	"0" level
27P	P10	Active	Floating
28P	P11	Active	Floating
29P	P12	Active	Floating
30P	P13	Active	Floating
31P	P14	Active	Floating
32P	P15	Active	Floating
33P	P16	Active	Floating
34P	P17	Active	Floating
35P	P24	Active	Floating
36P	P25	Active	Floating
37P	P26	Active	Floating
38P	P27	Active	Floating
39P	T1	Active	Active
40P	V_{CC}	+2 to +6 [V]	+2 to +6 [V]

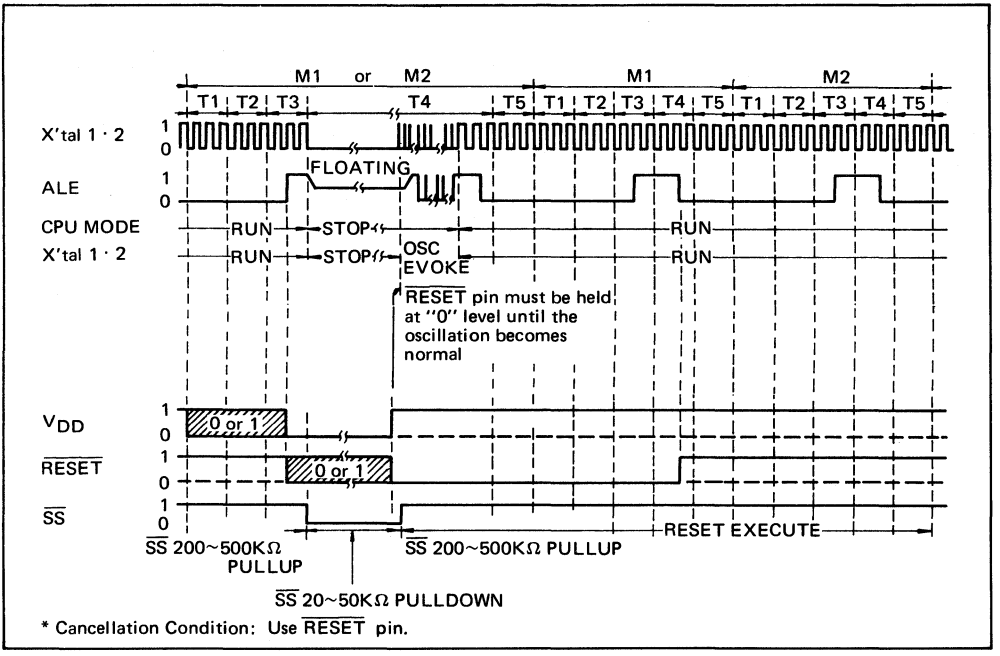


Fig. 4-5 Hardware Power-Down Mode Timing Chart

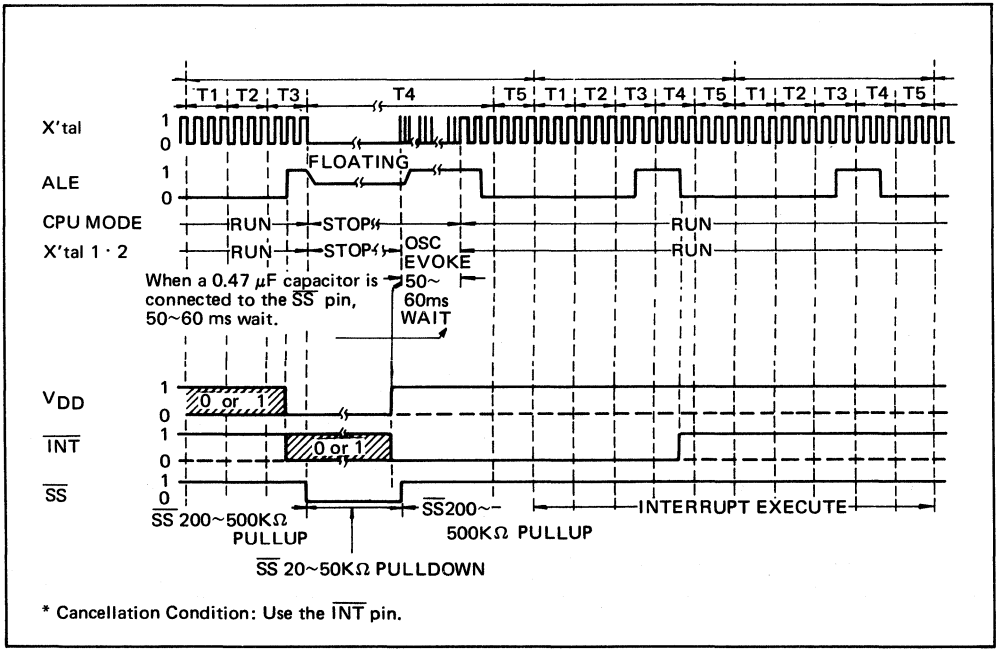
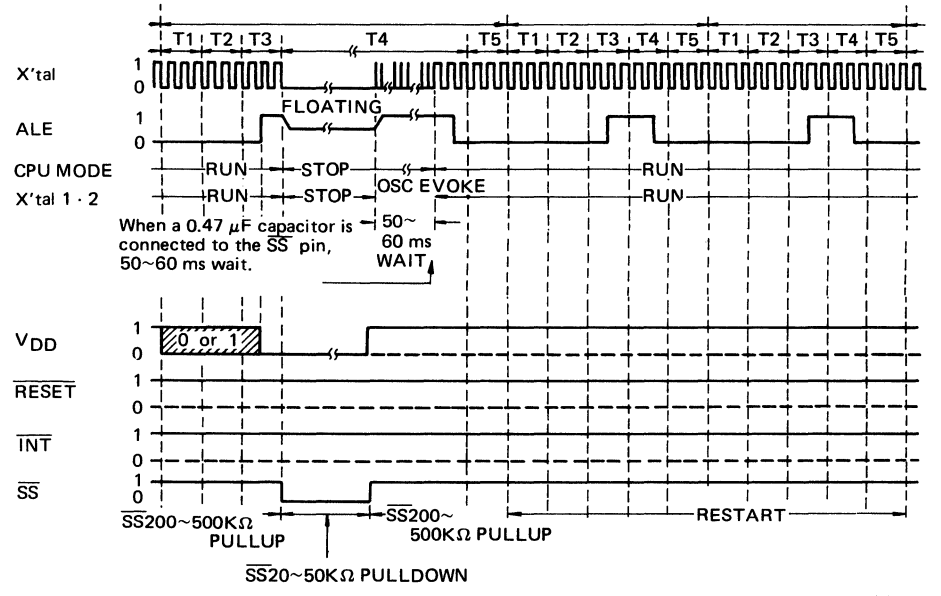


Fig. 4-6 Hardware Power-Down Mode Timing Chart



* Cancellation Condition: Use the V_{DD} pin on condition RESET and INT pins are in "1" level.

Fig. 4-7 Hardware Power-Down Mode Timing Chart

6

MSM80C48/MSM80C49/MSM80C50 INSTRUCTION TABLE

H \ L	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111
0 0000	NOP	HALT Added	OUTL BUS A	ADD A, # data	JMP	EN I		DEC A	INS A, BUS	IN A, P1	INA, P2		MOVD A, Pp			
1 0001	INC @, R0	INC @, R1	JB0 addr	ADDC A, # data	CALL	DIS I	JTF addr	INC A	INC Rr							
2 0010	XCHA @, R0	XCHA @, R1		MOV A, # data	JMP	EN TCNTI	JNT0 addr	CLR A	XCH A, Rr							
3 0011	XCHD A, @R0	XCHD A, @R1	JB1 addr		CALL	DIS TCNTI	JT0 addr	CPL A		OUTL P1, A	OUTL P2, A		MOVD Pp, A			
4 0100	ORL A, @R0	ORL A, @R1	MOV A, T	ORL A, # data	JMP	STRT CNT	JNT1 addr	SWAP A	ORLA, Rr							
5 0101	ANL A, @R0	ANL A, @R1	JB2 addr	ANL A, # data	CALL	STRT T	JT1 addr	DA A	ANL A, Rr							
6 0110	ADD A, @ R0	ADD A, @ R1	MOV T, A	MOV A, P1 Added	JMP	STOP TCNT		RRC A	ADD A, Rr							
7 0111	ADDC A, @ R0	ADDC A, @ R1	JB3 addr	MOV A, P2 Added	CALL	ENT0 CLK	JF1 addr	RR A	ADDC A, Rr							
8 1000	MOVX A, @ R0	MOVX A, @ R1	HLTS Added	RET	JMP	CLR F0	JN1 addr		ORL BUS, # data	ORL P1, # data	ORL P2, # data		ORLD Pp, A			
9 1001	MOVX @ R0, A	MOVX @ R1, A	JB4 addr	RETR	CALL	CLR F0	JNZ addr	CLR C	ANL BUS, # data	ANL P1, # data	ANL P2, # data		ANL D Pp, A			
A 1010	MOV @ R0, A	MOV @ R1, A	FLT Added	MOVP A, @A	JMP	CLR F1		CPL C	MOV Rr, A							
B 1011	MOV @ R0, #data	MOV @R1, #data	JB5 addr	JMPP @A	CALL	CPL F1	JF0 addr		MOV R, #data							
C 1100	DEC @R0 Added	DEC @R1 Added	FLTT Added	MOVP1 P,@ R3 Added	JMP	SEL RB0	JZ addr	MOV A, PSW	DEC Rr							
D 1101	XRL A, @R0	XRL A, @R1	JB6 addr	XRLA, #data	CALL	SEL RB1		MOV PSW, A	XRLA, Rr							
E 1110	DJNZ @R0 Added	DJNZ @R1 Added	FRES Added	MOVP3 A, @A	JMP	SEL MB0	JNC addr	RL A	DJNZ Rr							
F 1111	MOV A, @R0	MOV A, @R1	JB7 addr	MOVP1, @ R3 Added	CALL	SEL MB1	JC addr	RLC A	MOV A, Rr							

EXPLANATION OF INSTRUCTION SYMBOLS

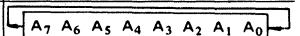
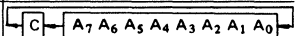
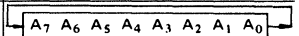
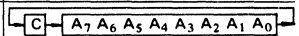
Symbols are listed below.

A	: Accumulator	PC	: Program counter
AC	: Auxiliary carry	Pp	: Port indicator (p = 4 ~ 7)
addr	: 12-bit program memory address or its part	PSW	: Program status word
Bb	: Bit indicator (b = 0 ~ 7)	Rr	: Resister indicator (r = 0 ~ 7)
BS	: Bank switch	SP	: Stack pointer
BUS	: BUS PORT	T	: Timer
C	: Carry	TF	: Timer flag
CLK	: Clock	T0, T1	: Test pins T0 and T1
CNT	: Counter	X	: External RAM
D	: 4-bit data	#	: Symbol denoting immediate data
data	: 8-bit numerical value	@	: Symbol denoting indirect address
DBF	: Memory data bank flip-flop	(X)	: Denotes contents of X
F0, F1	: F0 flag and F1 flag	((X))	: Denotes contents addressed by X
I	: Interrupt	←	: Transference

LIST OF INSTRUCTIONS

Classification	Mnemonic	Instruction Code D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal	Byte	Cycle	Description
Accumulator operation instructions	ADD A, Rr	0	1	1	0	1	r_2	r_1	r_0	68 ~ 6F	1	1	(AC), (C), (A) ← (A) + (Rr)
	ADD A, @Rr	0	1	1	0	0	0	0	r_0	60~61	1	1	(AC), (C), (A) ← (A) + ((Rr))
	ADD A, #data	0 d_7	0 d_6	0 d_5	0 d_4	0 d_3	0 d_2	1 d_1	1 d_0	03 Byte 2	2	2	(AC), (C), (A) ← (A) + data
	ADDC A, Rr	0	1	1	1	1	r_2	r_1	r_0	78~7F	1	1	(AC), (C), (A) ← (A) + (Rr) + (C)
	ADDC A, @Rr	0	1	1	1	0	0	0	r_0	70~71	1	1	(AC), (C), (A) ← (A) + ((Rr)) + (C)
	ADDC A, #data	0 d_7	0 d_6	0 d_5	1 d_4	0 d_3	0 d_2	1 d_1	1 d_0	13 Byte 2	2	2	(AC), (C), (A) ← (A) + data + (C)
	ANL A, Rr	0	1	0	1	1	r_2	r_1	r_0	58~5F	1	1	(A) ← (A) AND (Rr)
	ANL A, @Rr	0	1	0	1	0	0	0	r_0	50~51	1	1	(A) ← (A) AND ((Rr))
	ANL A, #data	0 d_7	1 d_6	0 d_5	1 d_4	0 d_3	0 d_2	1 d_1	1 d_0	53 Byte 2	2	2	(A) ← (A) AND data
	ORL A, Rr	0	1	0	0	1	r_2	r_1	r_0	48~4F	1	1	(A) ← (A) OR (Rr)
	ORL A, @Rr	0	1	0	0	0	0	0	r_0	40~41	1	1	(A) ← (A) OR ((Rr))
	ORL A, #data	0 d_7	1 d_6	0 d_5	0 d_4	0 d_3	0 d_2	1 d_1	1 d_0	43 Byte 2	2	2	(A) ← (A) OR data
	XRLA, Rr	1	1	0	1	1	r_2	r_1	r_0	D8~DF	1	1	(A) ← (A) XOR (Rr)
	XRLA, @Rr	1	1	0	1	0	0	0	r_0	D0~D1	1	1	(A) ← (A) XOR ((Rr))
	XRL A, #data	1 d_7	1 d_6	0 d_5	1 d_4	0 d_3	0 d_2	1 d_1	1 d_0	D3 Byte 2	2	2	(A) ← (A) XOR data
	INC A	0	0	0	1	0	1	1	1	17	1	1	(A) ← (A) + 1
	DEC A	0	0	0	0	0	1	1	1	07	1	1	(A) ← (A) − 1
	CLR A	0	0	1	0	0	1	1	1	27	1	1	(A) ← 0
	CPL A	0	0	1	1	0	1	1	1	37	1	1	(A) ← $(\bar{A})$
	DA A	0	1	0	1	0	1	1	1	57	1	1	Add 6 to bits 0 ~ 3 when contents of accumulator bits 0 ~ 3 exceed 9 or when auxiliary carry (AC) is 1. Then add 6 to bits 4 ~ 7 when the result of adding the carry from the lower 0 ~ 3 exceeds 9, or when carry (C) is 1. Set 1 in the carry flag if an overflow is generated in the end result, or when the carry prior to adjustment is 1.

LIST OF INSTRUCTIONS (CONT.)

Classification	Mnemonic	Instruction Code D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal	Byte	Cycle	Description
Accumulator operation instructions	SWAP A	0	1	0	0	0	1	1	1	47	1	1	$(A_{4\sim7}) \rightleftarrows (A_{0\sim3})$
	RL A	1	1	1	0	0	1	1	1	E7	1	1	 Rotate accumulator contents to the left by 1 bit.
	RLC A	1	1	1	1	0	1	1	1	F7	1	1	 Rotate accumulator contents with carry to the left by 1 bit.
	RR A	0	1	1	1	0	1	1	1	77	1	1	 Rotate accumulator contents to the right by 1 bit.
	RRC A	0	1	1	0	0	1	1	1	67	1	1	 Rotate accumulator contents with carry to the right by 1 bit.
Input/output instructions	IN A, P1	0	0	0	0	1	0	0	1	09	1	2	(A) ← (P1)
	IN A, P2	0	0	0	0	1	0	1	0	0A	1	2	(A) ← (P2)
	OUTL P1, A	0	0	1	1	1	0	0	1	39	1	2	(P1) ← (A)
	OUTL P2, A	0	0	1	1	1	0	1	0	3A	1	2	(P2) ← (A)
	ANL P1, #data	1 d_7	0 d_6	0 d_5	1 d_4	1 d_3	0 d_2	0 d_1	1 d_0	99 Byte 2	2	2	(P1) ← (P1) AND data
	ANL P2, #data	1 d_7	0 d_6	0 d_5	1 d_4	1 d_3	0 d_2	1 d_1	0 d_0	9A Byte 2	2	2	(P2) ← (P2) AND data
	ORL P1, #data	1 d_7	0 d_6	0 d_5	0 d_4	1 d_3	0 d_2	0 d_1	1 d_0	89 Byte 2	2	2	(P1) ← (P1) OR data
	ORL P2, #data	1 d_7	0 d_6	0 d_5	0 d_4	1 d_3	0 d_2	1 d_1	0 d_0	8A Byte 2	2	2	(P2) ← (P2) OR data
	INS A, BUS	0	0	0	0	1	0	0	0	08	1	2	(A) ← (BUS)
	OUTL BUS, A	0	0	0	0	0	0	1	0	02	1	2	(BUS) ← (A)
	ANL BUS, #data	1 d_7	0 d_6	0 d_5	1 d_4	1 d_3	0 d_2	0 d_1	0 d_0	98 Byte 2	2	2	(BUS) ← (BUS) AND data
	ORL BUS, #data	1 d_7	0 d_6	0 d_5	0 d_4	1 d_3	0 d_2	0 d_1	0 d_0	88 Byte 2	2	2	(BUS) ← (BUS) OR data
	MOVD A, Pp	0	0	0	0	1	1	P_1	P_0	0C~0F	1	2	$(A_{0\sim3}) \leftarrow$ (Pp) p=4~7 $(A_{4\sim7}) \leftarrow 0$
	MOVD Pp, A	0	0	1	1	1	1	P_1	P_0	3C~3F	1	2	(Pp) $\leftarrow (A_{0\sim3})$ p=4~7
	ANLD Pp, A	1	0	0	1	1	1	P_1	P_0	9C~9F	1	2	(Pp) ← (Pp) AND $(A_{0\sim3})$ p=4~7
	ORLD Pp, A	1	0	0	0	1	1	P_1	P_0	8C~8F	1	2	(Pp) ← (Pp) OR $(A_{0\sim3})$ p=4~7
Register operation instructions	INC Rr	0	0	0	1	1	r_2	r_1	r_0	18~1F	1	1	(Rr) ← (Rr) + 1
	INC @Rr	0	0	0	1	0	0	0	r_0	10~11	1	1	((Rr)) ← ((Rr)) + 1
	DEC Rr	1	1	0	0	1	r_2	r_1	r_0	C8~CF	1	1	(Rr) ← (Rr) − 1
	DEC @Rr	1	1	0	0	0	0	0	r_0	C0~C1	1	1	((Rr)) ← ((Rr)) − 1
Branching instructions	JMP addr	a_{10} a_7	a_9 a_6	a_8 a_5	0 a_4	0 a_3	1 a_2	0 a_1	0 a_0	ϕ4 ~ E4 Byte 2	2	2	$(PC_{8\sim10}) \leftarrow$ addr 8~10 $(PC_{0\sim7}) \leftarrow$ addr 0~7 $(PC_{11}) \leftarrow$ DBF
	JMPP @A	1	0	1	1	0	0	1	1	B3	1	2	$(PC_{0\sim7}) \leftarrow$ ((A))

LIST OF INSTRUCTIONS (CONT.)

Classification	Mnemonic	Instruction Code D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal	Byte	Cycle	Description
Branching instructions	DJNZ Rr, addr	1	1	1	0	1	r_2	r_1	r_0	E8~EF	2	2	(Rr) ←(Rr) − 1
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			($PC_{0\sim7}$) ←addr if (Rr) = 0
													(PC) ←(PC) + 2 if (Rr) = 0
	DJNZ @Rr, addr	1	1	1	0	0	0	0	r_0	E0~E1	2	2	((Rr)) ←((Rr)) − 1
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			($PC_{0\sim7}$) ←addr if ((Rr)) = 0
													(PC) ←(PC) + 2 if ((Rr)) = 0
	JC addr	1	1	1	1	0	1	1	0	F6	2	2	($PC_{0\sim7}$) ←addr if C = 1
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			(PC) ←(PC) + 2 if C = 0
	JNC addr	1	1	1	0	0	1	1	0	E6	2	2	($PC_{0\sim7}$) ←addr if C = 0
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			(PC) ←(PC) + 2 if C = 1
	JZ addr	1	1	0	0	0	1	1	0	C6	2	2	($PC_{0\sim7}$) ←addr if A = 0
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			(PC) ←(PC) + 2 if A ≠ 0
	JNZ addr	1	0	0	1	0	1	1	0	96	2	2	($PC_{0\sim7}$) ←addr if A ≠ 0
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			(PC) ←(PC) + 2 if A = 0
	JT0 addr	0	0	1	1	0	1	1	0	36	2	2	($PC_{0\sim7}$) ←addr if T0 = 1
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			(PC) ←(PC) + 2 if T0 = 0
	JNT0 addr	0	0	1	0	0	1	1	0	26	2	2	($PC_{0\sim7}$) ←addr if T0 = 0
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			(PC) ←(PC) + 2 if T0 = 1
	JT1 addr	0	1	0	1	0	1	1	0	56	2	2	($PC_{0\sim7}$) ←addr if T1 = 1
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			(PC) ←(PC) + 2 if T1 = 0
	JNT1 addr	0	1	0	0	0	1	1	0	46	2	2	($PC_{0\sim7}$) ←addr if T1 = 0
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			(PC) ←(PC) + 2 if T1 = 1
	JF0 addr	1	0	1	1	0	1	1	0	B6	2	2	($PC_{0\sim7}$) ←addr if F0 = 1
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			(PC) ←(PC) + 2 if F0 = 0
	JF1 addr	0	1	1	1	0	1	1	0	76	2	2	($PC_{0\sim7}$) ←addr if F1 = 1
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			(PC) ←(PC) + 2 if F1 = 0
	JTF addr	0	0	0	1	0	1	1	0	16	2	2	($PC_{0\sim7}$) ←addr TF ←0 } if TF = 1
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			(PC) ←(PC) + 2 if TF = 0
	JNI addr	1	0	0	0	0	1	1	0	86	2	2	($PC_{0\sim7}$) ←addr if $\overline{INT}$ = 0
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			(PC) ←(PC)+2 if $\overline{INT}$ = 1
	JBb addr	b_2	b_1	b_0	1	0	0	1	0	12 ~ F2	2	2	($PC_{0\sim7}$) ←addr if Bb = 1
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			(PC) ←(PC)+2 if Bb = 0
Sub-routine instructions	CALL addr	a_{10}	a_9	a_8	1	0	1	0	0	14 ~ F4	2	2	((SP)) ←(PC) + 2, ($PSW_{4\sim7}$)
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			($PC_{8\sim10}$) ←addr 8~10 ($PC_{0\sim7}$) ←addr 0~7 (PC_{11}) ←DBF (SP) ←(SP) + 1
	RET	1	0	0	0	0	0	1	1	83	1	2	(SP) ←(SP) − 1 (PC) ←((SP))
	RETR	1	0	0	1	0	0	1	1	93	1	2	(SP) ←(SP) − 1 (PC) ←((SP)) ($PSW_{4\sim7}$) ←((SP)) INT END
Flag operation instructions	CLR C	1	0	0	1	0	1	1	1	97	1	1	(C) ←0
	CPL C	1	0	1	0	0	1	1	1	A7	1	1	(C) ←$\overline{(C)}$
	CLR F0	1	0	0	0	0	1	0	1	85	1	1	(F0) ←0
	CPL F0	1	0	0	1	0	1	0	1	95	1	1	(F0) ←$\overline{(F0)}$
	CLR F1	1	0	1	0	0	1	0	1	A5	1	1	(F1) ←0
	CPL F1	1	0	1	1	0	1	0	1	B5	1	1	(F1) ←$\overline{(F1)}$
Data transfer instructions	MOV A, Rr	1	1	1	1	1	r_2	r_1	r_0	F8~FF	1	1	(A) ←(Rr)
	MOV A, @Rr	1	1	1	1	0	0	0	r_0	F0~F1	1	1	(A) ←((Rr))
	MOV A, #data	0	0	1	0	0	0	1	1	23	2	2	(A) ←data
		d_7	d_6	d_5	d_4	d_3	d_2	d_1	d_0	Byte 2			

LIST OF INSTRUCTIONS (CONT.)

Classification	Mnemonic	Instruction Code D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal	Byte	Cycle	Description
Data transfer instructions	MOV Rr, A	1	0	1	0	1	r_2	r_1	r_0	A8~AF	1	1	(Rr) ←(A)
	MOV @Rr, A	1	0	1	0	0	0	0	r_0	A0~A1	1	1	((Rr)) ←(A)
	MOV Rr, #data	1 d_7	0 d_6	1 d_5	1 d_4	1 d_3	r_2 d_2	r_1 d_1	r_0 d_0	B8~BF Byte 2	2	2	(Rr) ←data
	MOV @Rr, #data	1 d_7	0 d_6	1 d_5	1 d_4	0 d_3	0 d_2	0 d_1	r_0 d_0	B0~B1 Byte 2	2	2	((Rr)) ←data
	MOV A, PSW	1	1	0	0	0	1	1	1	C7	1	1	(A) ←(PSW)
	MOV PSW, A	1	1	0	1	0	1	1	1	D7	1	1	(PSW) ←(A)
	XCH A, Rr	0	0	1	0	1	r_2	r_1	r_0	28~2F	1	1	(A) ⇄(Rr)
	XCH A, @Rr	0	0	1	0	0	0	0	r_0	20~21	1	1	(A) ⇄((Rr))
	XCHD A, @Rr	0	0	1	1	0	0	0	r_0	30~31	1	1	($A_{0\sim3}$)⇄(($Rr_{0\sim3}$))
	MOVX A, @Rr	1	0	0	0	0	0	0	r_0	80~81	1	2	(A) ←((Rr)) External RAM
	MOVX @Rr, A	1	0	0	1	0	0	0	r_0	90~91	1	2	((Rr)) ←(A) External RAM
	MOVP A, @A	1	0	1	0	0	0	1	1	A3	1	2	(A) ←($PC_{8\sim10}$, A))
	MOVP3 A, @A	1	1	1	0	0	0	1	1	E3	1	2	(A) ←(PC_{011}, A))
	MOVP1 P, @R3	1	1	0	0	0	0	1	1	C3	1	2	(P1) ←((($PC_{0\sim7}$)←((R3))))
	MOV P1, @R3	1	1	1	1	0	0	1	1	F3	1	2	(P1) ←((R3))
	MOV A, P1	0	1	1	0	0	0	1	1	63	1	1	(A) ←(P1) Latch data
	MOV A, P2	0	1	1	1	0	0	1	1	73	1	1	(A) ←(P2) Latch data
Control instructions	EN TCNTI	0	0	1	0	0	1	0	1	25	1	1	TINT Enable F/F ←1
	DIS TCNTI	0	0	1	1	0	1	0	1	35	1	1	TINT Enable F/F ←0
	EN I	0	0	0	0	0	1	0	1	05	1	1	EXINT Enable F/F ←1
	DIS I	0	0	0	1	0	1	0	1	15	1	1	EXINT Enable F/F ←0
	SEL RB0	1	1	0	0	0	1	0	1	C5	1	1	(BS) ←0
	SEL RB1	1	1	0	1	0	1	0	1	D5	1	1	(BS) ←1
	SEL MB0	1	1	1	0	0	1	0	1	E5	1	1	(DBF) ←0
	SEL MB1	1	1	1	1	0	1	0	1	F5	1	1	(DBF) ←1
	$\overline{\text{ENT0 CLK}}$	0	1	1	1	0	1	0	1	75	1	1	T0 ←1/3 XTAL 1
	FLT	1	0	1	0	0	0	1	0	A2	1	1	P1, P2, BUS Floating
	FLTT	1	1	0	0	0	0	1	0	C2	1	1	CPU Output Signal Floating
	FRES	1	1	1	0	0	0	1	0	E2	1	1	FLT, FLTT RESET
	HLT	0	0	0	0	0	0	0	1	01	1	1	CPU Control Clock Stop
	HALTS	1	0	0	0	0	0	1	0	82	1	1	XTAL 1·2 Stop
Timer/counter instructions	MOV A, T	0	1	0	0	0	0	1	0	42	1	1	(A) ←(T)
	MOV T, A	0	1	1	0	0	0	1	0	62	1	1	(T) ←(A)
	STRT T	0	1	0	1	0	1	0	1	55	1	1	(T) ←÷32←÷15←XTAL
	STRT CNT	0	1	0	0	0	1	0	1	45	1	1	(T) ←T1 Clock
	STOP TCNT	0	1	1	0	0	1	0	1	65	1	1	(T) Count Stop
Other instruction	NOP	0	0	0	0	0	0	0	0	00	1	1	(PC) ←(PC) + 1

ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{CC}	Ta = 25°C	−0.3 to 7	V
Input Voltage	V_I	Ta = 25°C	−0.3 to V_{CC}	V
Storage Temperature	T_{stg}		−55 to +150	°C

OPERATING RANGE

- MSM80C35/48, 80C39/49 ... DC to 11 MHz, VCC = 5V ±20%
- MSM80C40/50 ... DC to 6 MHz, VCC = 50 ±20%

Parameter	Symbol	Conditions	Limits	Unit
Supply Voltage	V_{CC}	fosc = DC~11 MHz*	+2.5 to +6	V
RAM Retention Voltage	V_{CC}		+2 to +6	V
Ambient Temperature	T_A		−40 to +85	°C
Fan Out	N	MOS load	10	
		TTL load	1	

* 11 MHz version of MSM80C40/50 (6 MHz < XTAL1.2 < 11 MHz) is under development.

6

DC ELECTRICAL CHARACTERISTICS

(V_{CC} = 5V±10%, Ta = −40 to +85°C)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit	Measuring Circuit
"L" Input Voltage	V_{IL}		−0.3		0.8	V	1
"H" Input Voltage (1)	V_{IH}		2.2		V_{CC}	V	
"H" Input Voltage (2)	V_{IH}		3.8		V_{CC}	V	
"L" Output Voltage (3)	V_{OL}	I_{OL} = 2 mA			0.45	V	
"L" Output Voltage (4)	V_{OL}	I_{OL} = 1.6 mA			0.45	V	
"H" Output Voltage (3)	V_{OH}	I_{OH} = 400 μA	2.4			V	
"H" Output Voltage (4)	V_{OH}	I_{OH} = 50 μA	2.4			V	
"H" Output Voltage (3)	V_{OH}	I_{OH} = 20 μA	4.2			V	
"H" Output Voltage (4)	V_{OH}	I_{OH} = 10 μA	4.2			V	
Input Leak Current	I_{IL}	$V_{SS} \leqq V_{IN} \leqq V_{CC}$			±10	μA	2
Output Leak Current (5)	I_{OL}	$V_{SS} \leqq V_O \leqq V_{CC}$			±10	μA	3
$\overline{RESET}$ Pull up Resistance	R_R	$V_{IN} \geqq V_{IH} \leqq$ / $V_{IN} \leqq V_{IL}$	20/500		50/750	kΩ	2
$\overline{SS}$ Pull up Resistance (6)	R_{SS}	Oscillation stop/oscillation	20/200		50/500	kΩ	
P1, P2 Pull up Resistance	$R_{P1, P2}$	$V_{IN} \geqq V_{IH}$/ $V_{IN} \leqq V_{IL}$	5/75		15/150	kΩ	3
Power Supply Current	I_{CC}	At hardware power down V_{CC}=2V (TA = +25°C) (7)		1	10	μA	4
		At HLTS execution *V_{CC}=2V (TA = +25°C) (7)		1	10	μA	
		At HALT (6 MHz)		1.5	3	mA	
		At HALT (11 MHz) (8)		2.5	5	mA	
		Atexecution (6 MHz)		5	10	mA	
		At execution (11 MHz) (8)		10	20	mA	

Notes: (1) This does not apply to $\overline{RESET}$, XTAL1, XTAL2, and V_{DD}.
(2) $\overline{RESET}$, XTAL1, XTAL2, V_{DD}
(3) BUS, $\overline{RD}$, $\overline{WR}$, $\overline{PSEN}$, ALE
(4) Other outputs
(5) High-impedance state
(6) This operates as a pull-down resistor when the oscillation is stopped in the HLTS or hardware power-down mode and as a pull-up resistor in other states.
(7) This does not contain flow out current from I/O Ports and Signal pins.
(8) MSM80C35/48, 80C39/49

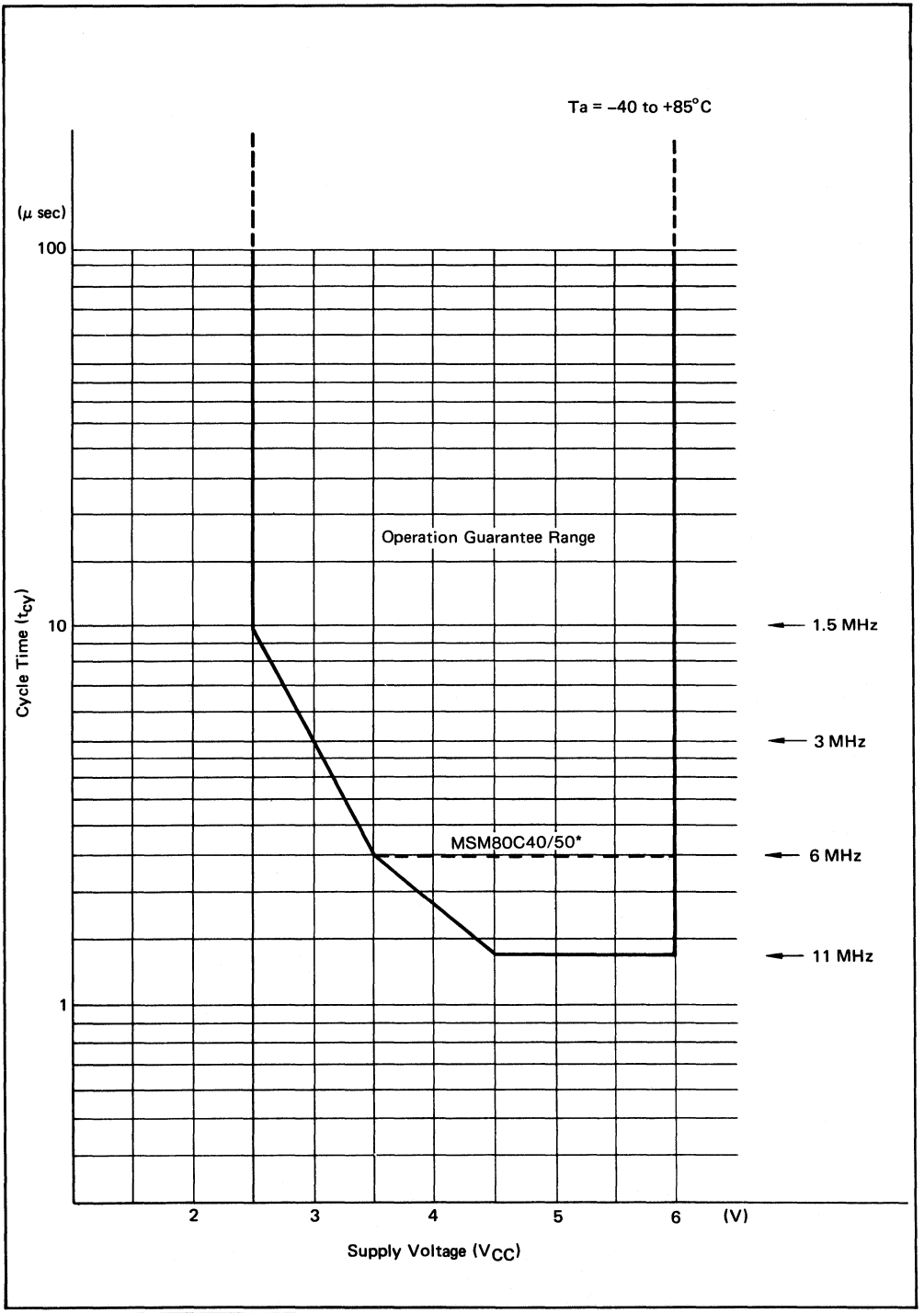
AC CHARACTERISTICS

(V_{CC} = 5V±10%, TA = −40°C to +85°C)

Parameter	Symbol	Limits				Unit
		11 MHz Clock		Variable Clock (0 – 11 MHz)		
		Min.	Max.	Min.	Max.	
Cycle Time	t_{CY}	1.36		1.36		μS
ALE Pulse Width	t_{LL}	150		$7/30t_{CY}-165$		ns
Address Set up ALE	t_{AL}	70		$2/15t_{CY}-110$		ns
Address Hold from ALE	t_{LA}	50		$1/15t_{CY}-40$		ns
Bus Port Latch Data Setup to ALE	t_{BL}	110		$5/30t_{CY}-115$		ns
Bus Port Latch Data Hold from ALE	t_{LB}	90		$3/30t_{CY}-45$		ns
Control Pulse Width ($\overline{PSEN}$, $\overline{RD}$, and $\overline{WR}$)	t_{CC}	300		$6/15t_{CY}-245$		ns
Data Setup before $\overline{WR}$	t_{DW}	250		$6/15t_{CY}-295$		ns
Data Hold after $\overline{WR}$	t_{WD}	40		$2/15t_{CY}-140$		ns
Data Hold after $\overline{RD}$	t_{DR}	0	100	0	100	ns
$\overline{PSEN}$, $\overline{RD}$ to Data-in	t_{RD}		200		$5/15t_{CY}-250$	ns
Address Setup to $\overline{WR}$	t_{AW}	200		$6/15t_{CY}-345$		ns
Address Setup to Data-in	t_{AD}		400		$8/15t_{CY}-325$	ns
Address Float to $\overline{RD}$, $\overline{PSEN}$	t_{AFC}	0		0		ns
Port Control Setup to $\overline{PROG}$	t_{CP}	100		$2/15t_{CY}-80$		ns
Port Control Hold from $\overline{PROG}$	t_{PC}	60		$4/15t_{CY}-300$		ns
$\overline{PROG}$ to P2 Input Valid	t_{PR}	–	650		$9/15t_{CY}-165$	ns
Output Data Setup	t_{DP}	200		$6/15t_{CY}-345$		ns
Output Data Hold	t_{PD}	20		$3/15t_{CY}-250$		ns
Input Data Hold from $\overline{PROG}$	t_{PF}	0	150	0	150	ns
$\overline{PROG}$ Pulse Width	t_{PP}	700		$10/15t_{CY}-205$		ns
Port 2 I/O Setup to ALE	t_{PL}	150		$9/30t_{CY}-255$		ns
Port 2 I/O Hold from ALE	t_{LP}	20		$3/30t_{CY}-115$		ns

Note: Control output: C_L = 80 pF
Bus output: C_L = 150 pF [for 20 pF (t_{WD})]

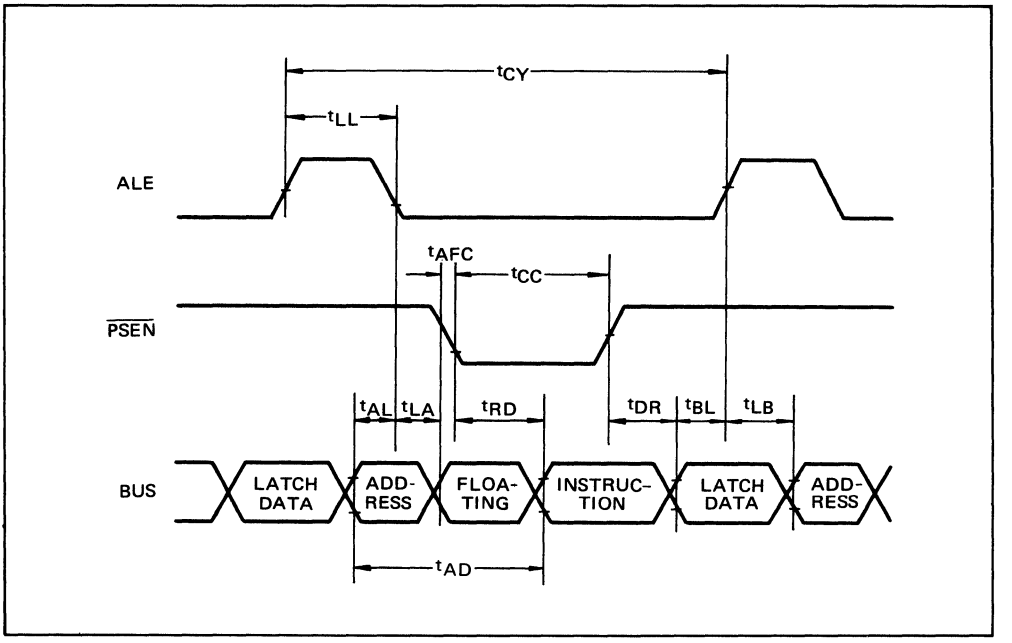
MSM80C49 OPERATION GUARANTEE RANGE



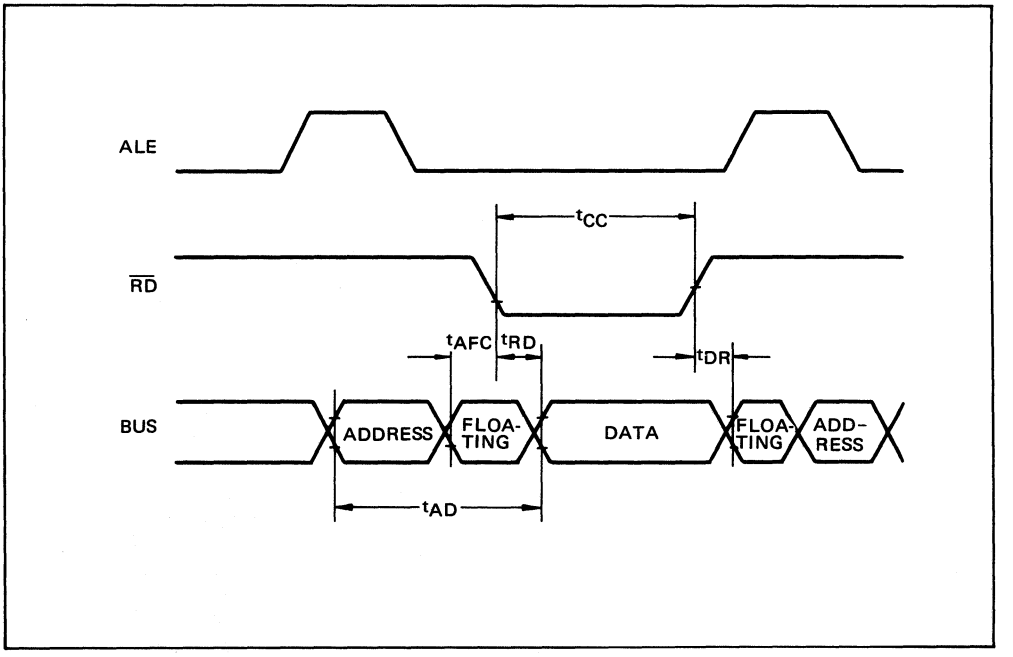
* 11 MHz version of MSM80C40/50 is under development

TIMING CHART

Instruction Fetch (from external program memory)

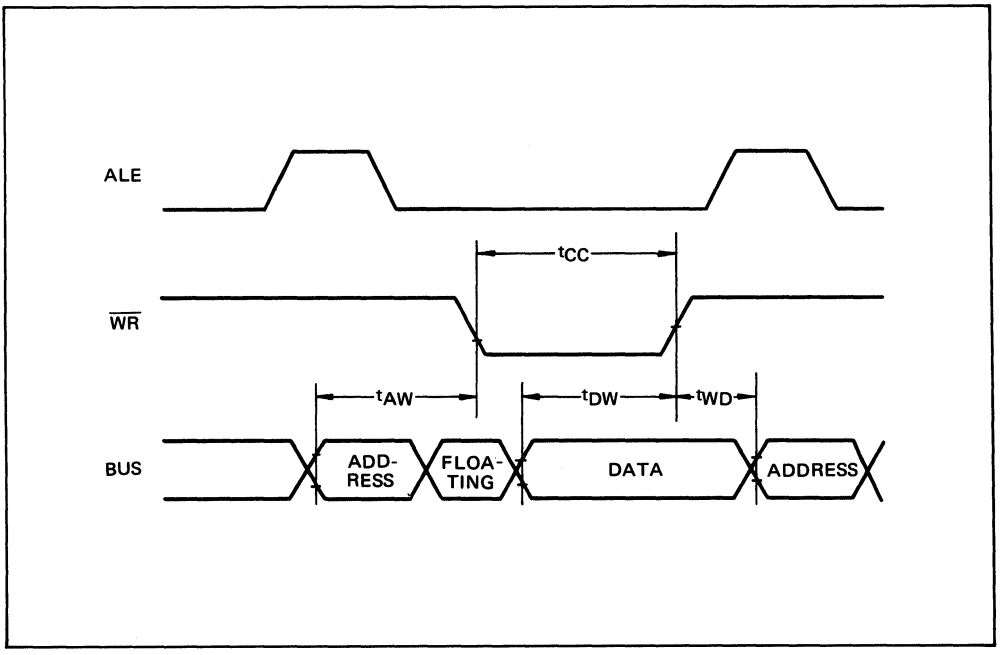


Read (from external data memory)



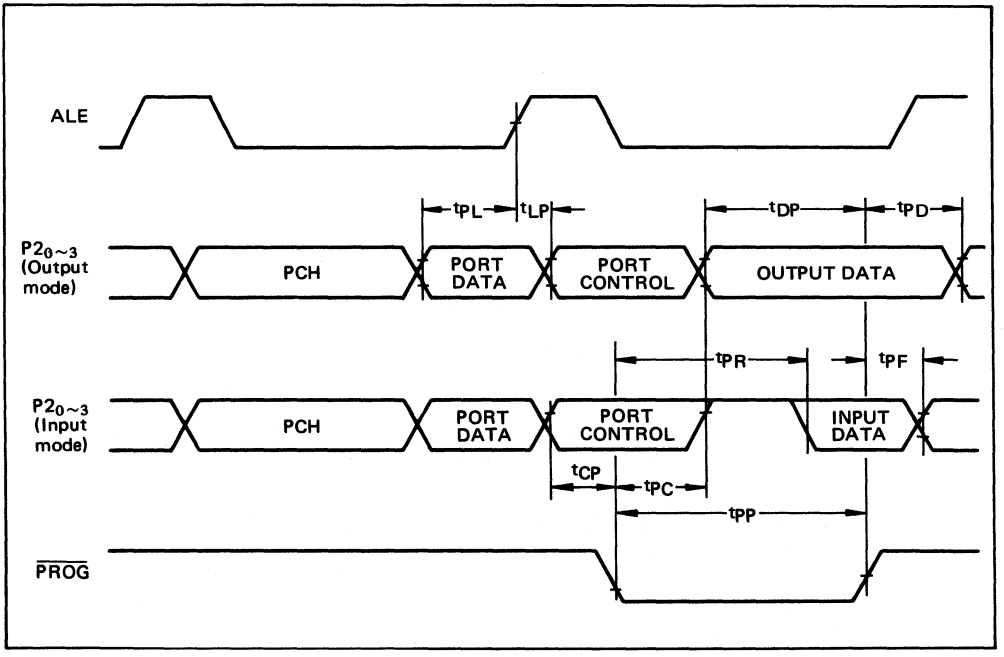
6

Write (to external memory)

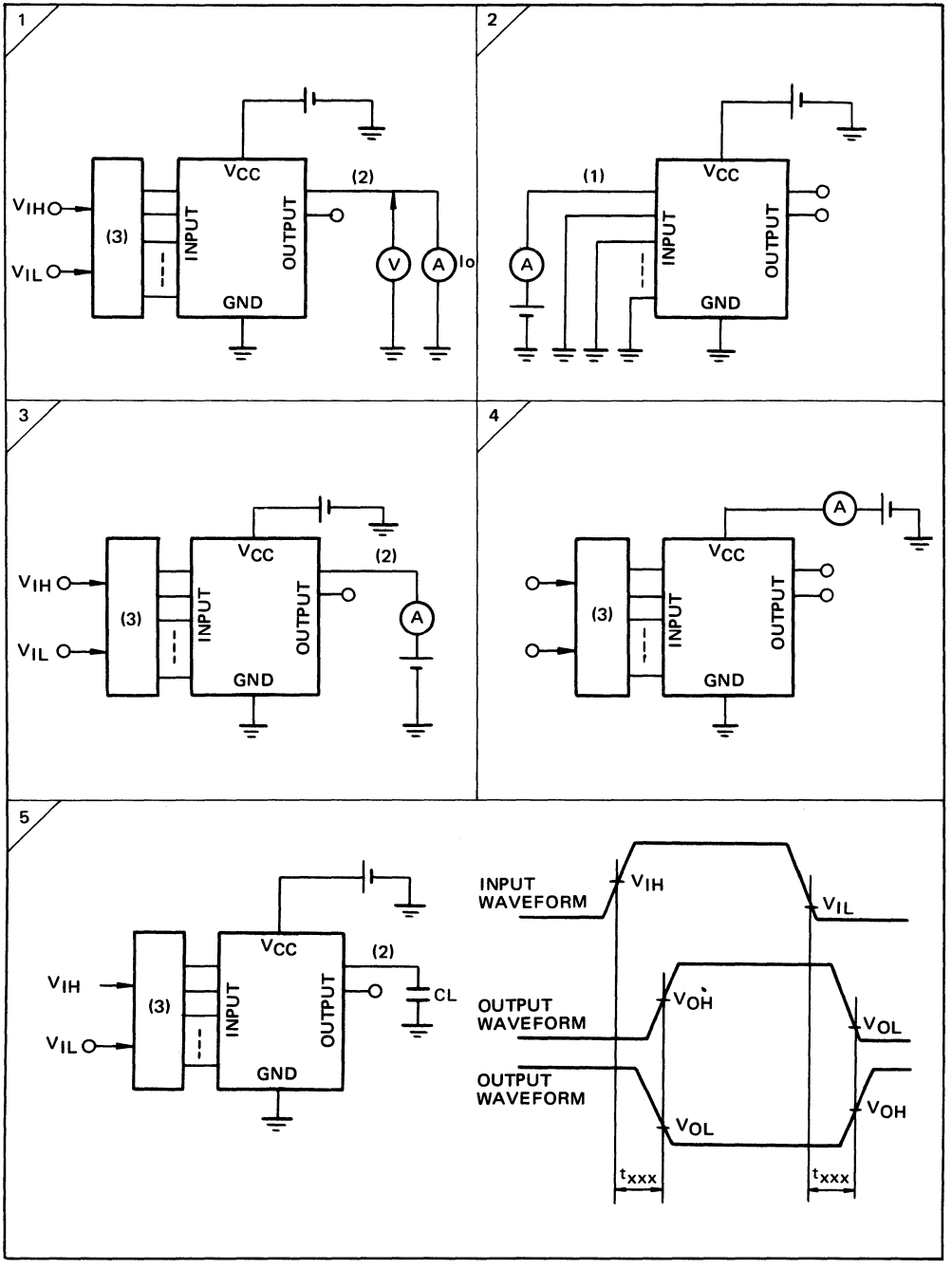


6

4 low-order bits input/output of port 2 when expanded I/O is used
(in external program memory access mode)



MEASUREMENT CIRCUIT



Notes: (1) This is repeated for each specified input pin.
(2) This is repeated for each specified output pin.
(3) Input logic for setting the specified state.

6

MSM80C31F/MSM80C51F

CMOS SINGLE-COMPONENT 8-BIT MICROCONTROLLER

GENERAL DESCRIPTION

The OKI MSM80C31F/MSM80C51F microcontroller is a low power, high performance 8-bit single component device implemented in OKI's silicon gate complementary metal oxide semiconductor process technology. Integrated within the device is 4K bytes of mask programmable ROM (MSM80C51F only), 128 bytes of data RAM, 32 I/O lines, two 16-bit timer/counters, a five-source two-level interrupt structure, a full duplex serial port, and an on chip oscillator and clock circuitry. In addition, the device has two software selectable modes for further power reduction – Idle and Power Down. Idle mode freezes the CPU's instruction execution while maintaining RAM and allowing the timers, serial port and interrupt system to continue functioning. Power Down mode saves the RAM contents but freezes the oscillator causing all other device functions to be inoperative.

FEATURES

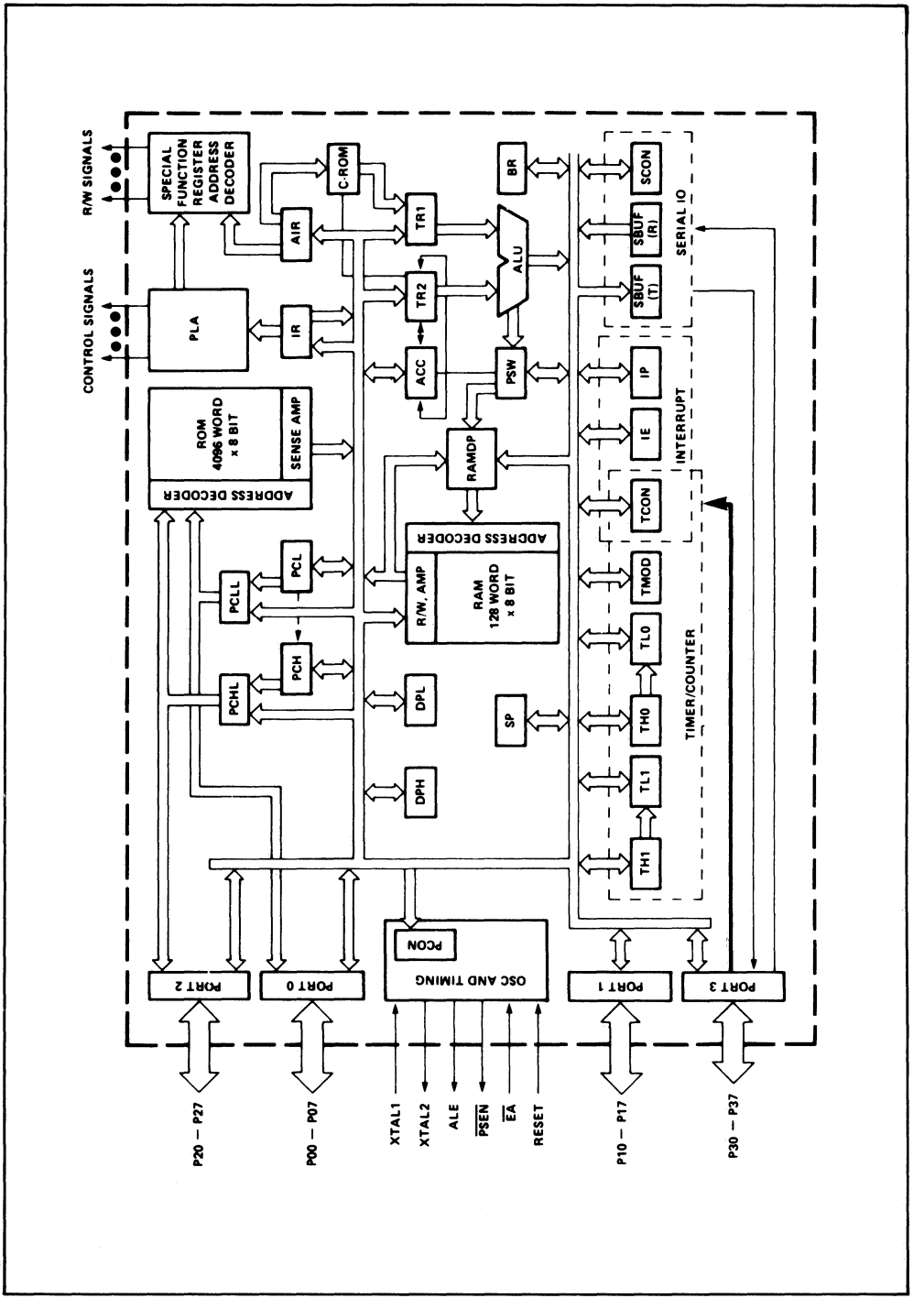
- Operating temperature: −40 ~ +85°C
- Operating frequency: 0.5 ~ 16 MHz
- CMOS technology, 2μm Silicon gate
- Minimum instruction cycle:
  1.0μs (@ 12MHz, Vcc = 5V±20%)
  0.75μs (@ 16MHz, Vcc = 5V±10%)
- Low power consumption:
  Normal Operation 16 mA @ 5V, 12MHz
  Idle Mode 3.7 mA @ 5V, 12MHz
  Power Down Mode 50μA @ 2V
- Instruction set includes 111 instructions
- 8-bit CPU
- On chip oscillator and clock circuitry
- 32 Input/Output lines
- 4069 × 8 bits on chip ROM (MSM80C51F)
- 128 × 8 bits on chip RAM
- 64K address space for program memory
- 64K address space for external data memory
- Two 16-bit timer/counters
- Five source two-priority level interrupt structure
- Full duplex serial port
- Boolean processor
- CMOS and TTL compatible
- CMOS ROM LESS development device (MSM80C31F)

DIFFERENCES BETWEEN MSM80C31F/MSM80C51F AND MSM80C31/MSM80C51

- Operating frequency
  0.5 ~ 16 MHz MSM80C31F-1/MSM80C51F-1
  0.5 ~ 12 MHz MSM80C31/MSM80C51/MSM80C31F/MSM80C51F
- External clock input terminal
  XTAL1 ..... MSM80C31F(-1)/MSM80C51F(-1)
  XTAL2 ..... MSM80C31/MSM80C51
- Emulation mode
  Output impedance of ALE and $\overline{PSEN}$ pins becomes about 20kΩ while CPU is being reset in MSM80C31F/MSM80C51F.

Any other functions and electrical characteristics of MSM80C31F/MSM80C51F except for the above three differences are the same as those of MSM80C31/MSM80C51.

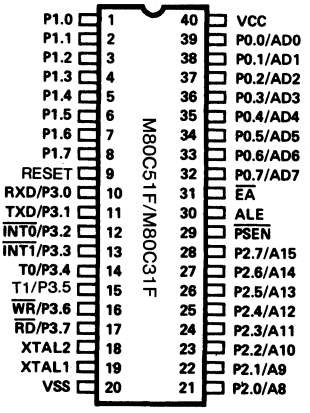
FUNCTIONAL BLOCK DIAGRAM



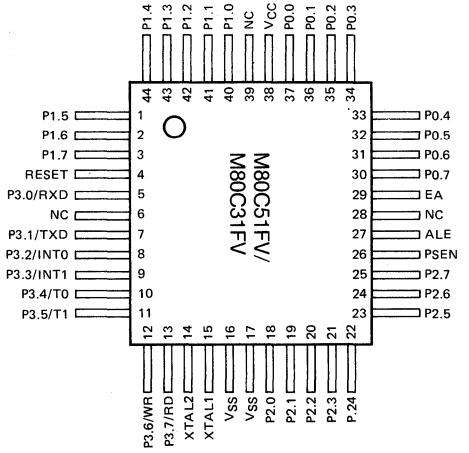
6

PIN CONFIGURATION

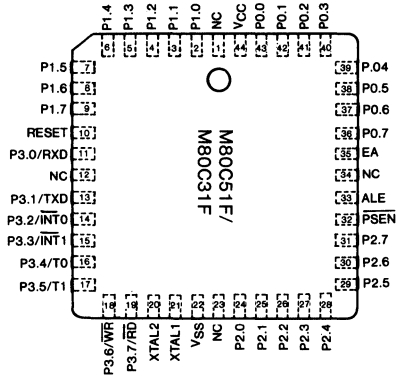
MSM80C51F-XXRS/MSM80C31FRS
(Top View) 40-Lead Plastic DIP



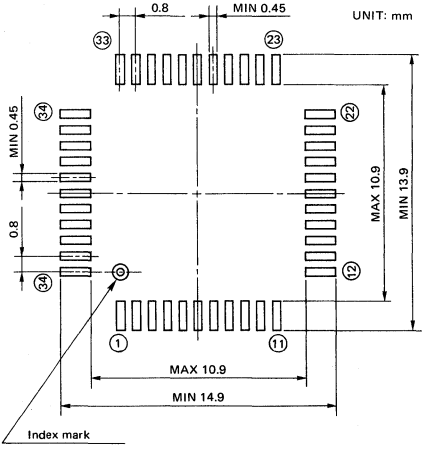
MSM80C51FV-XXGSK/MSM80C31FVGSK
(Top View) 44-Lead Plastic Flat Package



MSM80C51F-XXJS/MSM80C31FJS
(Top View) 44 Lead Plastic Leaded Chip Carrier



RECOMMENDED FOOTPRINT
FOR QFD PACKAGE



6

PIN DESCRIPTIONS

Designation	Description
V_{SS}	Ground potential
V_{CC}	Supply voltage during Normal, Idle and Power Down operation
Port 0	Port 0 is an 8-bit open drain bidirectional I/O port. It is also the multiplexed low-order address and data bus during accesses to external memory.
Port 1	Port 1 is an 8-bit bidirectional I/O port with internal pullups. It can drive CMOS inputs without external pullups.
Port 2	Port 2 is an 8-bit bidirectional I/O port with internal pullups. It receives the high-order address byte during accesses to external memory. It can drive CMOS inputs without external pullups.
Port 3	Port 3 is an 8-bit bidirectional I/O port with internal pullups. It also serves the functions of various special features, as shown below: **Port Pin** **Alternate Fucntion** P3.0 RXD (serial input port) P3.1 TXD (serial output port) P3.2 $\overline{INT0}$ (external interrupt) P3.3 $\overline{INT1}$ (external interrupt) P3.4 T0 (Timer 0 external input) P3.5 T1 (Timer 1 external input) P3.6 $\overline{WR}$ (external data memory write strobe) P3.7 $\overline{RD}$ (external data memory read strobe) It can drive CMOS inputs without external pullups.
RESET	Reset input pin. A reset is accomplished by holding the RESET pin high for at least 1 μ second, even if the oscillator has been stopped. The CPU responds by executing an internal reset. An internal pulldown resistor permits Power-On reset using only a capacitor connected to V_{CC}. This pin does not receive the power down voltage since the function has been transferred to the V_{CC} pin.
ALE	Address Latch Enable output for latching the low byte of the address during accesses to external memory. For this purpose, ALE is activated twice every machine cycle or at a constant rate of 1/6 the oscillator frequency, except during an external memory access at which time one ALE pulse is skipped. It can drive CMOS inputs without an external pullup.
$\overline{PSEN}$	Program Store Enable output is the read strobe to external Program Memory. $\overline{PSEN}$ is activated twice each machine cycle during fetches from external Program Memory. (However, when executing out of external Program Memory, two activations of $\overline{PSEN}$ are skipped during each access to external Data Memory.) $\overline{PSEN}$ is not activated during fetches from internal Program Memory. It can drive CMOS inputs without an external pullup.
$\overline{EA}$	External Access input pin. When $\overline{EA}$ is held high, the CPU executes out of internal Program Memory (unless the Program Counter exceeds 0FFFH). When $\overline{EA}$ is held low, the CPU executes only out of external Program Memory. $\overline{EA}$ must not be floated.
XTAL1	Crystal 1 pin. It is an input to the inverting amplifier which forms the internal oscillator. It also receives the external clock signal when an external oscillator is used. (External clock signal should be at 50% duty and C-MOS level).
XTAL2	Crystal 2 pin. It is an output of the inverting amplifier that forms the internal oscillator.

ELECTRICAL CHARACTERISTICS

ABSOLUTE MAXIMUM RATINGS*

Item	Symbol	Conditions	Rating	Units
Supply Voltage	V_{CC}		−0.5 to +7.0	V
Voltage from any Pin to V_{SS}	VAP		−0.5 to V_{CC} +0.5	V
Storage Temperature	TSTG		−55 to +150	°C
Power Dissipation	PD		1.0	W

OPERATING RANGE

- MSM80C31F-1/80C51F-1 ... 0.5 to 16 MHz, VCC = ±10%
- MSM80C31F/80C51F, MSM80C31/80C51 ... 0.5 to 12 MHz, VCC = ±20%

Item	Symbol	Conditions	Rating	Units
Supply Voltage	V_{CC}	** fosc = 0.5 ~ 16 MHz	2.5 to +6.0	V
RAM Retention Voltage	VPD	Power Down	2.0 to +6.0	V
Operating Temperature	TOP		−40 to +85	°C

6

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

** DC and AC characteristics in the range of 12 MHz $< f \leq$ 16 MHz and 2.5 V $\leq$ Vcc $<$ 4 V will be specified elsewhere.

DC CHARACTERISTICS

(TA = −40°C to +85°; V_{CC} = 4 to 6V: V_{SS} = 0V)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Input Low Voltage (Except EA)	V_{IL}	–	−0.5	–	$0.2V_{CC}-0.1$	V
Input Low Voltage To EA	V_{IL1}	–	−0.5	–	$0.2V_{CC}-0.3$	V
Input High Voltage (Except XTAL1, RESET)	V_{IH}	–	$0.2V_{CC}+0.9$	–	$V_{CC}+0.5$	V
Input High Voltage (XTAL1 and RESET)	V_{IH}	–	$0.7V_{CC}$	–	$V_{CC}+0.5$	V
Power Down Voltage to V_{CC} in PD mode	V_{PD}	–	2.0	–	6.0	V
Output Low Voltage Ports 1, 2, 3 (Note 1)	V_{OL}	I_{OL} = 1.6mA	–	–	0.45	V
Output Low Voltage Port 0, ALE, $\overline{PSEN}$ (Note 1)	V_{OL1}	I_{OL} = 3.2mA	–	–	0.45	V
Output High Voltage Ports 1, 2, 3	V_{OH}	$I_{OH} = -10\mu A$	$0.9V_{CC}$	–	–	V
		$I_{OH} = -30\mu A$	$0.75V_{CC}$	–	–	V
		$I_{OH} = -60\mu A$	2.4	–	–	V

NOTE 1: V_{OL} is degraded when the 80C31F/80C51F rapidly discharges external capacitance. This AC noise is most pronounced during the emission of address data. When using external memory, locate the latch or buffer as close to the 80C31F/80C51F as possible.

DC CHARACTERISTICS (CONT.)

($T_A = -40°C$ to $+85°$; $V_{CC} = 4$ to 6V: $V_{SS} = 0V$)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Output High Voltage Port 0(in External Bus Mode), ALE, PSEN	V_{OH1}	$I_{OH} = -40\mu A$ (Note 2)	$0.9V_{CC}$	–	–	V
		$I_{OH} = -150\mu A$	$0.75V_{CC}$	–	–	V
		$I_{OH} = -400\mu A$	2.4	–	–	V
Logical 0 Input Current Ports 1, 2, 3	I_{IL}	$V_{in} = 0.45V$	–	–	−200	μA
Logical 1 To 0 Transition Current-Ports 1, 2, 3	I_{TL}	$V_{in} = 2.0V$	–	–	−500	μA
Input Leakage Current Port 0, EA	I_{LI}	$V_{SS} < V_{in} < V_{CC}$	–	–	±10	μA
Maximum Power Supply Current Normal Operation (Note 3)	I_{CC}	V_{CC} =	4V	5V	6V	
		f_{osc} = 12 MHz	12	16	20	mA
		f_{osc} = 8 MHz	8.3	11	14	mA
		f_{osc} = 3.5 MHz	4.3	5.7	7.5	mA
		f_{osc} = 0.5 MHz	1.6	2.2	3	mA
Maximum Power Supply Current Idle Mode (Note 4)	I_{CC1}	f_{osc} = 12 MHz	2.5	3.7	5	mA
		f_{osc} = 8 MHz	1.8	2.7	3.7	mA
		f_{osc} = 3.5 MHz	1.1	1.6	2.2	mA
		f_{osc} = 0.5 MHz	0.6	0.9	1.2	mA
Maximum Power Supply Current Normal Operation (Note 3)	I_{CC}	V_{CC} =	4.5V	5V	5.5V	
		f_{osc} = 16 MHz	18	20	23	mA
		f_{osc} = 12 MHz	14	16	18	mA
		f_{osc} = 8 MHz	10	11	12.5	mA
		f_{osc} = 1.2 MHz	2.0	2.3	2.6	mA
Maximum Power Supply Current Idle Mode (Note 4)	I_{CC1}	f_{osc} = 16 MHz	4.0	5.0	6.0	mA
		f_{osc} = 12 MHz	3.0	3.7	5.0	mA
		f_{osc} = 8 MHz	2.3	2.7	3.2	mA
		f_{osc} = 1.2 MHz	1.4	1.5	1.6	mA
Power Supply Current (Power Down Mode)	I_{PD}	V_{CC} = 2.0V (Note 5)	–	–	50	μA
RESET Pulldown Resistor	R_{RST}		20	40	125	kΩ
Capacitance of I/O Buffer	C_{IO}	f_C = 1 MHz $T_A = 25°C$	–	–	10	pF

6

NOTE 2: Capacitive loading on Ports 0 and 2 may cause the V_{OH} on ALE and PSEN to momentarily fall below the 0.9 V_{CC} specification when the address bits are stabilizing.

NOTE 3: I_{CC} is measured with all output pins disconnected; XTAL1 driven with TCLCH, TCHCL = 10 ns, $V_{IL} = V_{SS} + 0.5V$, $V_{IH} = V_{CC} - 0.5V$; $\overline{EA}$ = RESET = PORT 0 = V_{CC}. I_{CC} may be higher if a crystal oscillator is used.

NOTE 4: Idle I_{CC} is measured with all output pins disconnected; XTAL1 driven with TCLCH, TCHCL = 10 ns, $V_{IL} = V_{SS} + 0.5V$, $V_{IH} = V_{CC} - 0.5V$; $\overline{EA}$ = PORT 0 = V_{CC}. I_{CC} may be higher if a crystal oscillator is used.

NOTE 5: Power Down I_{PD} is measured with all output pins disconnected; $\overline{EA}$ = PORT 0 = V_{CC}; XTAL2 N.C.; RESET = V_{SS}.

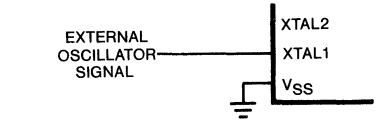
Datum	Emitting Ports	Degraded I/O Lines	V_{OL} (peak/max)
Address	P2, P0	P1, P3	0.8V
Write Data	P0	P1, P2, P3	0.8V

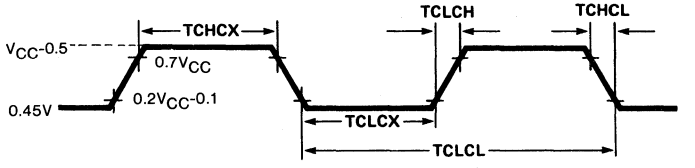
CLOCK PARAMETERS

Item	Symbol	Variable Clock freq = 0.5 MHz to 16 MHz		
		Min.	Max.	Unit
Oscillator Period	T_{CLCL}	62.5	–	ns
High Time	T_{CHCX}	20	–	ns
Low Time	T_{CLCX}	20	–	ns
Rise Time	T_{CLCH}		20	ns
Fall Time	T_{CHCL}	–	20	ns

6

EXTERNAL CLOCK DRIVE





*XTAL1
Input to the inverting oscillator amplifier and input to the internal clock generator circuits.

*XTAL2
Output from the inverting oscillator amplifier.

AC CHARACTERISTICS

(T_{OP} = −40°C to 85°C; V_{CC} = 4V to 6V; V_{SS} =0V; 0.5 to 12 MHz; Load Capacitance for Port 0, ALE and $\overline{PSEN}$ = 100pF; Load Capacitance for all other outputs = 80 pF)

EXTERNAL PROGRAM MEMORY CHARACTERISTICS

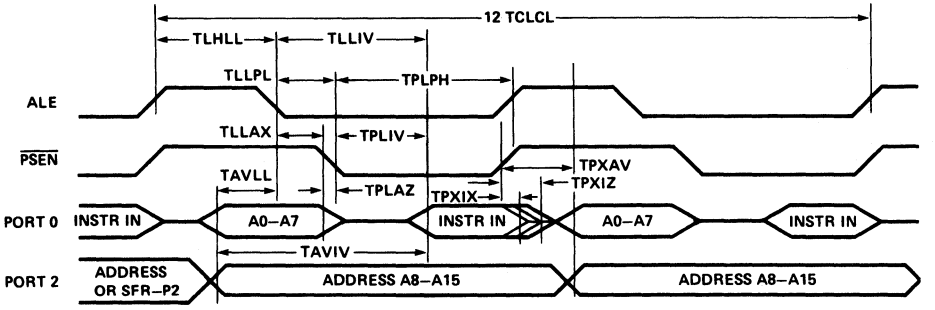
Item	Symbol	Min.	Typ.	Max.	Unit
ALE Pulse Width	TLHLL	2TCLCL-40	–	–	ns
Address Valid to ALE Low	TAVLL	TCLCL-40	–	–	ns
Address Hold after ALE Low	TLLAX	TCLCL-35	–	–	ns
ALE Low to Valid Instr. In	TLLIV	–	–	4TCLCL-100	ns
ALE Low to $\overline{PSEN}$ Low	TLLPL	TCLCL-25	–	–	ns
$\overline{PSEN}$ Pulse Width	TPLPH	3TCLCL-35	–	–	ns
$\overline{PSEN}$ Low to Valid Instr. In	TPLIV	–	–	3TCLCL-105	ns
Input Instr. Hold after $\overline{PSEN}$	TPXIX	0	–	–	ns
Input Instr. Float after $\overline{PSEN}$	TPXIZ	–	–	TCLCL-20	ns
$\overline{PSEN}$ to Address Valid	TPXAV	TCLCL-8	–	–	ns
Address to Valid Instr. In	TAVIV	–	–	5TCLCL-105	ns
$\overline{PSEN}$ Low to Address Float	TPLAZ	–	–	0	ns

EXTERNAL DATA MEMORY CHARACTERISTICS

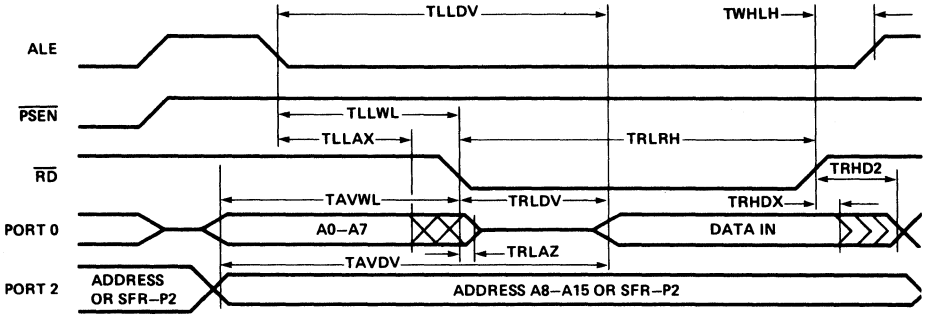
Item	Symbol	Min.	Typ.	Max.	Unit
$\overline{RD}$ Pulse Width	TRLRH	6TCLCL-100	–	–	ns
$\overline{WR}$ Pulse Width	TWLWH	6TCLCL-100	–	–	ns
Data Address Hold after ALE Low	TLLAX	TCLCL-35	–	–	ns
$\overline{RD}$ Low to Valid Data In	TRLDV	–	–	5TCLCL-165	ns
Data Hold after $\overline{RD}$	TRHDX	0	–	–	ns
Data Float after $\overline{RD}$	TRHDZ	–	–	2TCLCL-70	ns
ALE Low to Valid Data In	TLLDV	–	–	8TCLCL-150	ns
Address to Valid Data In	TAVDV	–	–	9TCLCL-165	ns
ALE Low to $\overline{WR}$ or $\overline{RD}$ Low	TLLWL	3TCLCL-50	–	3TCLCL+50	ns
Address to WR or RD Low	TAVWL	4TCLCL-130	–	–	ns
Data Valid to $\overline{WR}$ Transition	TQVWX	TCLCL-60	–	–	ns
Data Valid to $\overline{WR}$ High	TQVWH	7TCLCL-150	–	–	ns
Data Hold after $\overline{WR}$	TWHQX	TCLCL-50	–	–	ns
$\overline{RD}$ Low to Address Float	TRLAZ	–	–	0	ns
$\overline{RD}$ or $\overline{WR}$ High to ALE High	TWHLH	TCLCL-40	–	TCLCL+40	ns

AC TIMING DIAGRAMS

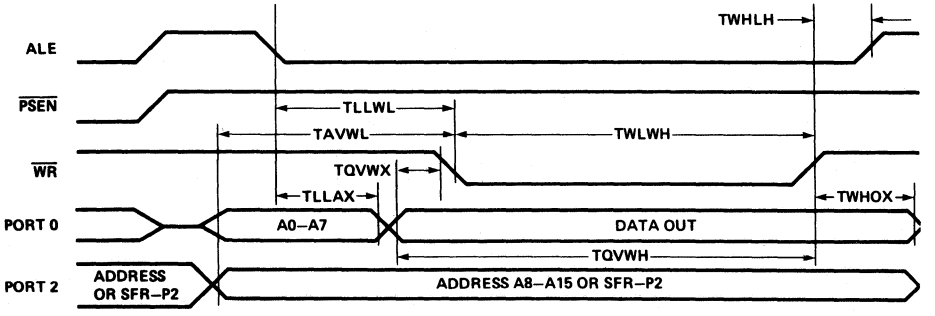
EXTERNAL PROGRAM MEMORY READ CYCLE



EXTERNAL DATA MEMORY READ CYCLE

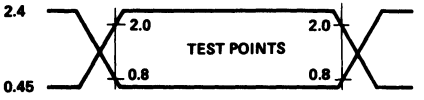


EXTERNAL DATA MEMORY WRITE CYCCLE

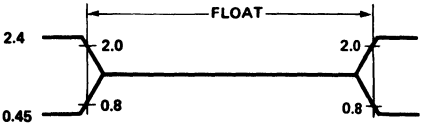


AC TESTING INPUT/OUTPUT, FLOAT WAVEFORMS

INPUT/OUTPUT



FLOAT



AC inputs during testing are driven at 2.4V for a logic "1" and 0.45V for a logic "0". Timing measurements are made at 2.0V for a logic "1" and 0.8V for a logic "0". For timing purposes, the float state is defined as the point at which a P0 pin sinks 3.2 mA or sources 400μA at the voltage test levels.

SERIAL PORT TIMING

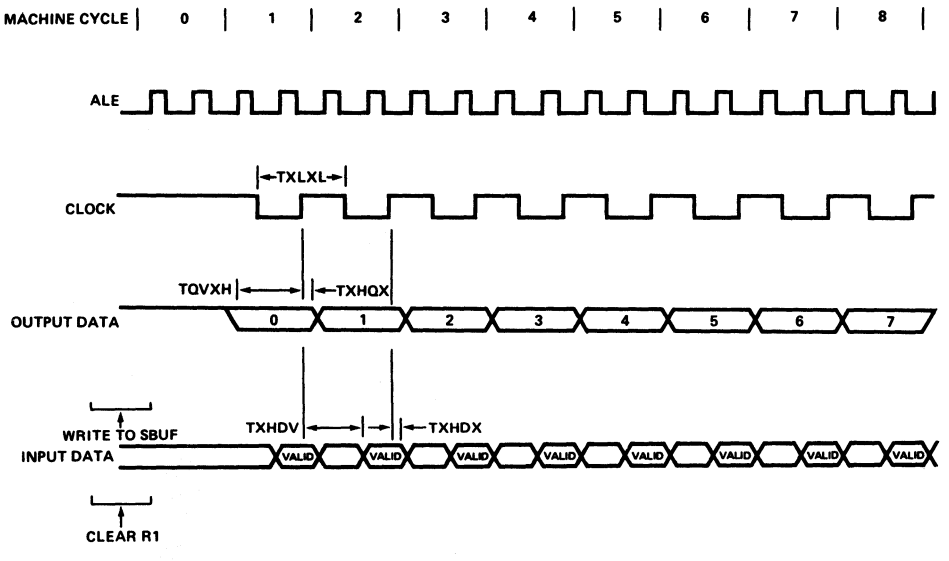
I/O EXPANSION MODE

(TA = −40°C to +85°C; V_{CC} = 4 to 6V; V_{SS} = 0V, 0.5 to 12 MHz; Load capacitance = 80 pF)

Symbol	Parameter	Min.	Max.	Units
TXLXL	Serial port clock cycle time	12TCLCL	–	μs
TQVXH	Output data setup to clock rising edge	10TCLCL-133	–	ns
TXHQX	Output data hold after clock rising edge	2TCLCL-117	–	ns
TXHDX	Input data hold after clock rising edge	0	–	ns
TXHDV	Clock rising edge to input data valid	–	10TCLCL-133	ns

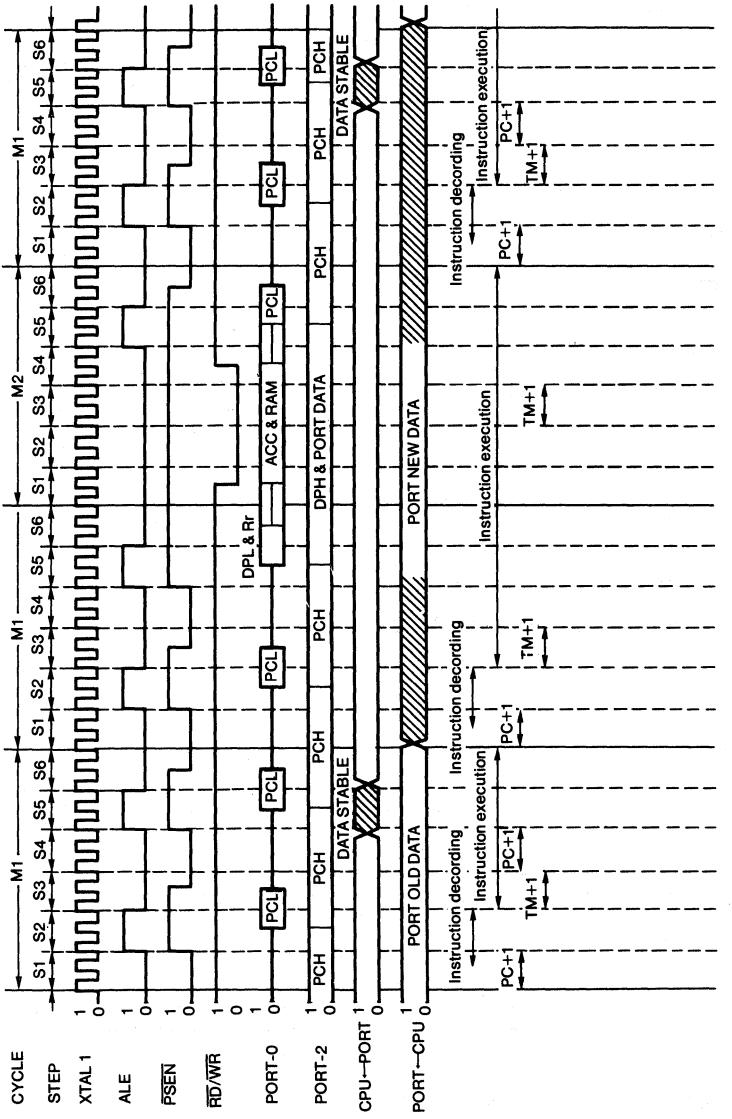
6

SERIAL PORT (I/O EXPANSION MODE)

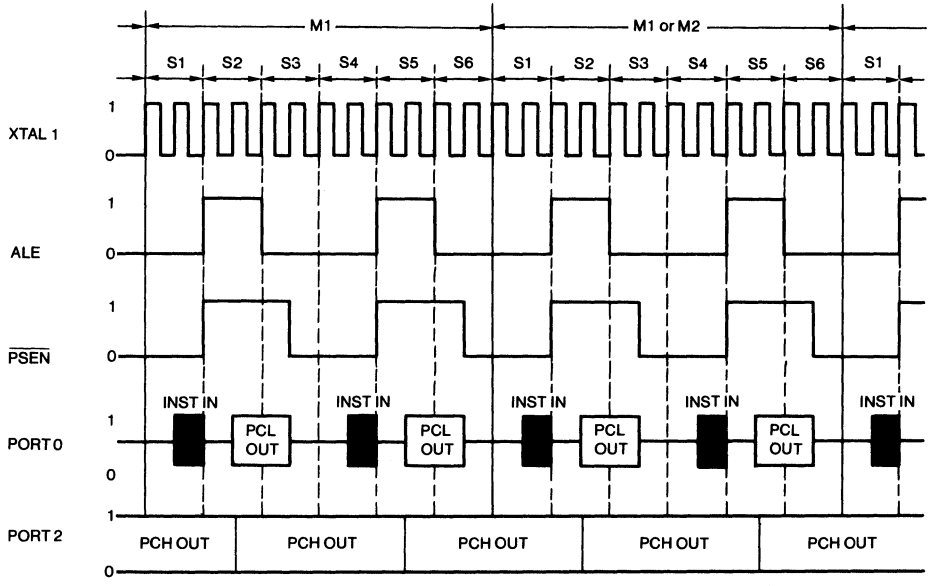


CLOCK WAVEFORMS

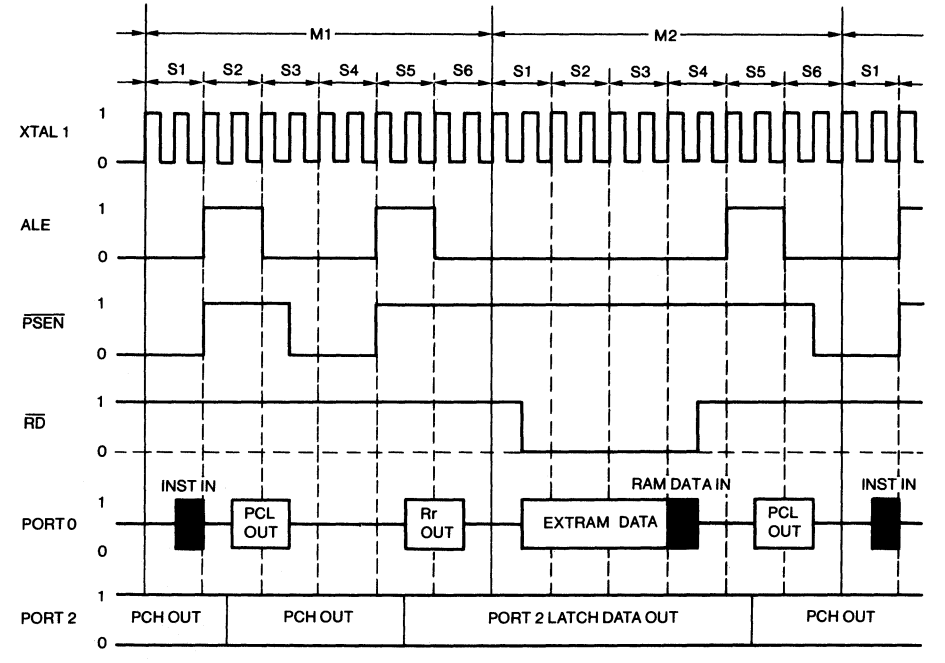
BASIC TIMING CHART



EXTERNAL PROGRAM MEMORY FETCH

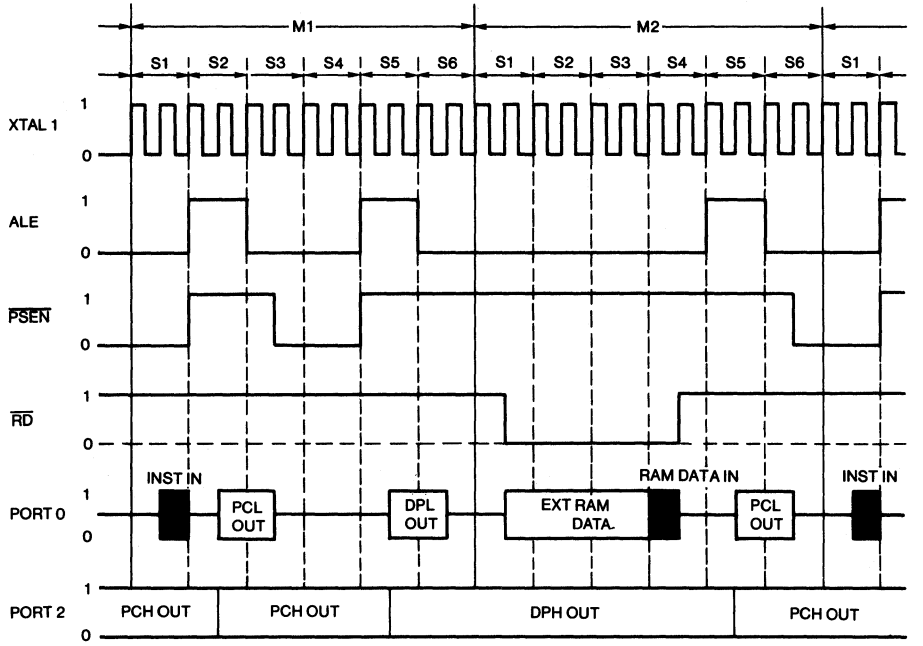


READ CYCLE 1 (MOVX A, @Rr)

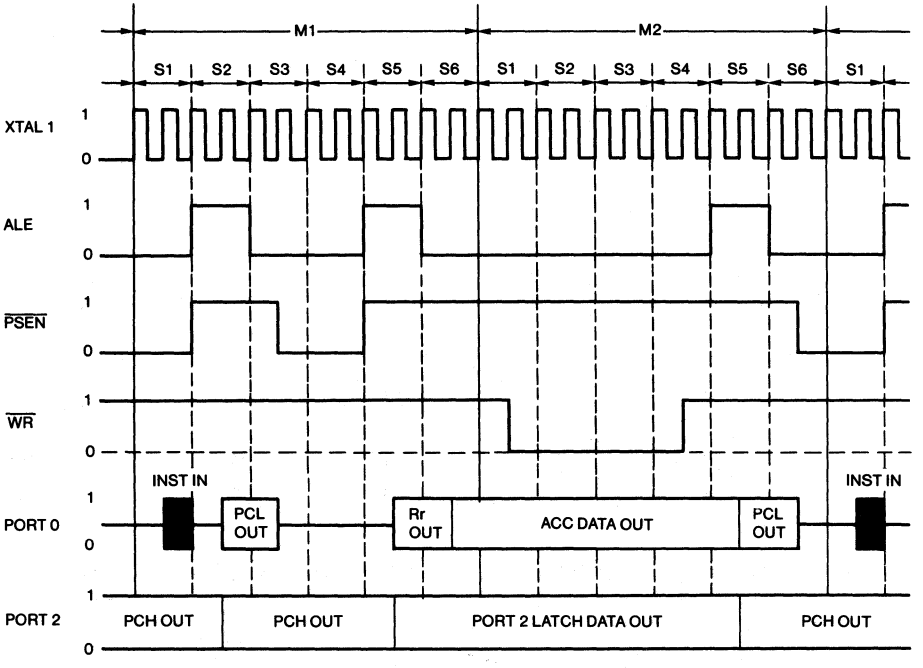


6

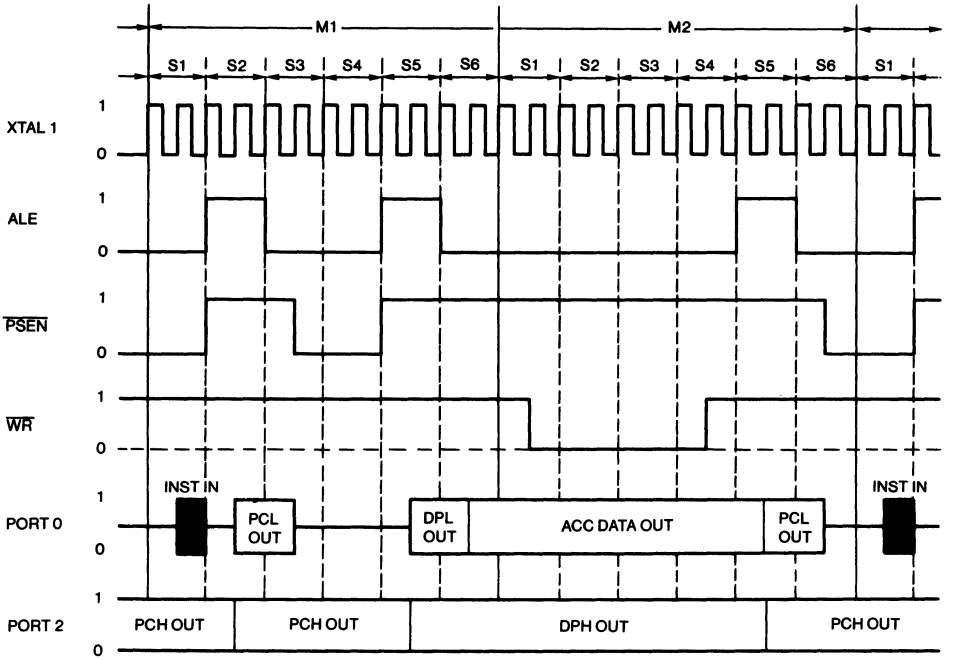
READ CYCLE 2 (MOVX A, @DPTR)



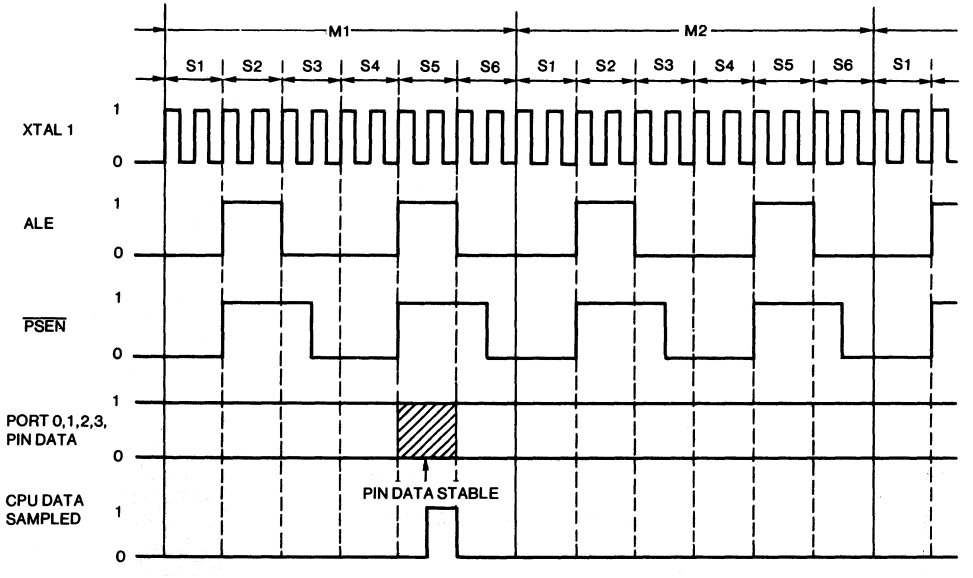
WRITE CYCLE 1 (MOVX, @Rr, A)



WRITE CYCLE 2 (MOVX A, @DPTR, A)



PORT OPERATION

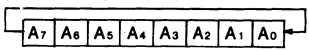
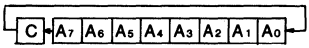
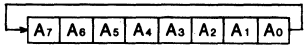
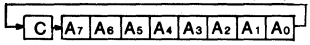


6

INSTRUCTION SET DETAILS

	Mnemonic	Instruction Code D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexa-decimal	Byte	Cycle	Explanation
Arithmetic operations	ADD A, Rn	0	0	1	0	1	n_2	n_1	n_0	28 ~ 2F	1	1	(A) ← (A) + (Rn)
	ADD A, direct	0	0	1	0	0	1	0	1	25	2	1	(A) ← (A) + (direct)
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			
	ADD A, @Ri	0	0	1	0	0	1	1	1	26 ~ 27	1	1	(A) ← (A) + ((Ri))
	ADD A, #data	0	0	1	0	0	1	0	0	24	2	1	(A) ← (A) + #data
		d_7	d_6	d_5	d_4	d_3	d_2	d_1	d_0	Byte 2			
	ADDC A, Rn	0	0	1	1	1	n_2	n_1	n_0	38 ~ 3F	1	1	(A) ← (A) + (C) + (Rn)
	ADDC A, direct	0	0	1	1	0	1	0	1	35	2	1	(A) ← (A) + (C) + (direct)
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			
	ADDC A, @Ri	0	0	1	1	0	1	1	1	36 ~ 37	1	1	(A) ← (A) + (C) + ((Ri))
	ADDC A, #data	0	0	1	1	0	1	0	0	34	2	1	(A) ← (A) + (C) + #data
		d_7	d_6	d_5	d_4	d_3	d_2	d_1	d_0	Byte 2			
	SUBB A, Rn	1	0	0	1	1	n_2	n_1	n_0	98 ~ 9F	1	1	(A) ← (A) − ((C) + (Rn))
	SUBB A, direct	1	0	0	1	0	1	0	1	95	2	1	(A) ← (A) − ((C) + (direct))
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			
	SUBB A, @Ri	1	0	0	1	0	1	1	1	96 ~ 97	1	1	(A) ← (A) − ((C) + ((Ri)))
	SUBB A, #data	1	0	0	1	0	1	0	0	94	2	1	(A) ← (A) − ((C) + #data)
		d_7	d_6	d_5	d_4	d_3	d_2	d_1	d_0	Byte 2			
	INC A	0	0	0	0	0	1	0	0	04	1	1	(A) ← (A) + 1
	INC Rn	0	0	0	0	1	n_2	n_1	n_0	08 ~ 0F	1	1	(Rn) ← (Rn) + 1
	INC direct	0	0	0	0	0	1	0	1	05	2	1	(direct) ← (direct) + 1
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			
	INC @Ri	0	0	0	0	0	1	1	1	06 ~ 07	1	1	((Ri)) ← ((Ri)) + 1
	INC DPTR	1	0	1	0	0	0	1	1	A3	1	2	(DPTR) ← (DPTR) + 1
	DEC A	0	0	0	1	0	1	0	0	14	1	1	(A) ← (A) − 1
	DEC Rn	0	0	0	1	1	n_2	n_1	n_0	18 ~ 1F	1	1	(Rn) ← (Rn) − 1
	DEC direct	0	0	0	1	0	1	0	1	15	2	1	(direct) ← (direct) − 1
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0				
	DEC @Ri	0	0	0	1	0	1	1	1	16 ~ 17	1	1	((Ri) ← ((Ri)) − 1
	MUL AB	1	0	1	0	0	1	0	0	A4	1	4	(B_{15} ~ $_8$), (A_7 ~ $_0$) ← (A) × (B)
	DIV AB	1	0	0	0	0	1	0	0	84	1	4	(A_{15} ~ $_8$), (B_7 ~ $_0$) ← (A)/(B)
	DA A	1	1	0	1	0	1	0	0	D4	1	1	Contents of Accumulator are BCD, IF [[(A_3 ~ $_0$) > 9] or [(AC) = 1]] THEN (A_3 ~ $_0$) ← (A_3 ~ $_0$) + 6 AND IF [[(A_7 ~ $_4$) > 9] OR [(C) = 1]] THEN (A_7 ~ $_4$) ← (A_7 ~ $_4$) + 6
Logical operations	ANL A, Rn	0	1	0	1	1	n_2	n_1	n_0	58 ~ 5F	1	1	(A) ← (A) AND (Rn)
	ANL A, direct	0	1	0	1	0	1	0	1	55	2	1	(A) ← (A) AND (direct)
		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	Byte 2			
	ANL A, @Ri	0	1	0	1	0	1	1	1	56 ~ 57	1	1	(A) ~ (A) AND ((Ri))
	ANL A, #data	0	1	0	1	0	1	0	0	54	2	1	(A) ← (A) AND #data
		d_7	d_6	d_5	d_4	d_3	d_2	d_1	d_0	Byte 2			

INSTRUCTION SET DETAILS (CONT.)

	Mnemonic	Instruction Code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Hexa-decimal	Byte	Cycle	Explanation
Logical operations	ANL direct, A	0 1 0 1 0 0 1 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	52 Byte 2	2	1	(direct) ← (direct) AND (A)
	ANL direct, #data	0 1 0 1 0 0 1 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	53 Byte 2 Byte 3	3	2	(direct) ← (direct) AND #data
	ORL A, Rn	0 1 0 0 1 n_2 n_1 n_0	48 ~ 4F	1	1	(A) ← (A) OR (Rn)
	ORL A, direct	0 1 0 0 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	45 Byte 2	2	1	(A) ← (A) OR (direct)
	ORL A, @Ri	0 1 0 0 0 1 1 1	46 ~ 47	1	1	(A) ← (A) OR ((Ri))
	ORL A, #data	0 1 0 0 0 1 0 0 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	44 Byte 2	2	1	(A) ← (A) OR #data
	ORL direct, A	0 1 0 0 0 0 1 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	42 Byte 2	2	1	(direct) ← (direct) OR (A)
	ORL direct, #data	0 1 0 0 0 0 1 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	43 Byte 2 Byte 3	3	2	(direct) ← (direct) OR #data
	XRL A, Rn	0 1 1 0 1 n_2 n_1 n_0	68 ~ 6F	1	1	(A) ← (A) XOR (Rn)
	XRL A, direct	0 1 1 0 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	65 Byte 2	2	1	(A) ← (A) XOR (direct)
	XRL A, @Ri	0 1 1 0 0 1 1 1	66 ~ 67	1	1	(A) ← (A) XOR ((Ri))
	XRL A, #data	0 1 1 0 0 1 0 0 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	64 Byte 2	2	1	(A) ← (A) XOR #data
	XRL direct, A	0 1 1 0 0 0 1 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	62 Byte 2	2	1	(direct) ← (direct) XOR (A)
	XRL direct, #data	0 1 1 0 0 0 1 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	63 Byte 2 Byte 3	3	2	(direct) ← (direct) XOR #data
	CLR A	1 1 1 0 0 1 0 0	E4	1	1	(A) ← 0
	CPL A	1 1 1 1 0 1 0 0	F4	1	1	(A) ← $(\overline{A})$
	RL A	0 0 1 0 0 0 1 1	23	1	1	 The contents of the accumulator are rotated left by one bit.
	RLC A	0 0 1 1 0 0 1 1	33	1	1	 The contents of the accumulator and carry are rotated left by one bit.
	RR A	0 0 0 0 0 0 1 1	03	1	1	 The contents of the accumulator are rotated right by one bit.
	RRC A	0 0 0 1 0 0 1 1	13	1	1	 The contents of the accumulator and carry are rotated right by one bit.
	SWAP A	1 1 0 0 0 1 0 0	C4	1	1	$(A_{3 \sim 0}) \rightleftarrows (A_{7 \sim 4})$

INSTRUCTION SET DETAILS (CONT.)

	Mnemonic	Instruction Code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Hexa-decimal	Byte	Cycle	Explanation
Data transfer	MOV A, Rn	1 1 1 0 1 n_2 n_1 n_0	E8 ~ EF	1	1	(A) ← (Rn)
	MOV A, direct	1 1 1 0 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	E5 Byte 2	2	1	(A) ← (direct)
	MOV A, @Ri	1 1 1 0 0 1 1 1	E6 ~ E7	1	1	(A) ← ((Ri))
	MOV A, #data	0 1 1 1 0 1 0 0 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	74 Byte 2	2	1	(A) ← #data
	MOV Rn, A	1 1 1 1 1 n_2 n_1 n_0	F8 ~ FF	1	1	(Rn) ← (A)
	MOV Rn, direct	1 0 1 0 1 n_2 n_1 n_0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	A8 ~ AF Byte 2	2	2	(Rn) ← (direct)
	MOV Rn, #data	0 1 1 1 1 n_2 n_1 n_0 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	78 ~ 7F Byte 2	2	1	(Rn) ← #data
	MOV direct, A	1 1 1 1 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	F5 Byte 2	2	1	(direct) ← (A)
	MOV direct, Rn	1 0 0 0 1 n_2 n_1 n_0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	88 ~ 8F Byte 2	2	2	(direct) ← (Rn)
	MOV direct 1, direct 2	1 0 0 0 0 1 0 1 a_7^2 a_6^2 a_5^2 a_4^2 a_3^2 a_2^2 a_1^2 a_0^2 a_7^1 a_6^1 a_5^1 a_4^1 a_3^1 a_2^1 a_1^1 a_0^1	85 Byte 2 Byte 3	3	2	(direct 1) ← (direct 2)
	MOV direct, @Ri	1 0 0 0 0 1 1 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	86 ~ 87 Byte 2	2	2	(direct) ← ((Ri))
	MOV direct, #data	0 1 1 1 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	75 Byte 2 Byte 3	3	2	(direct) ← #data
	MOV @Ri, A	1 1 1 1 0 1 1 1	F6 ~ F7	1	1	((Ri)) ← A
	MOV @Ri, direct	1 0 1 0 0 1 1 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	A6 ~ A7 Byte 2	2	2	((Ri)) ← (direct)
	MOV @Ri, #data	0 1 1 1 0 1 1 1 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	76 ~ 77 Byte 2	2	1	((Ri)) ← #data
	MOV DPTR, #data 16	1 0 0 1 0 0 0 0 d_{15} d_{14} d_{13} d_{12} d_{11} d_{10} d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	90 Byte 2 Byte 3	3	2	(DPTR) ← #data$_{16}$
	MOVC A, @A+ DPTR	1 0 0 1 0 0 1 1	93	1	2	(A) ← ((A) + (DPTR))
	MOVC A, @A+PC	1 0 0 0 0 0 1 1	83	1	2	(PC) ← (PC) + 1, (A) ← ((A) + (PC))
	MOVX A, @Ri	1 1 1 0 0 0 1 1	E2 ~ E3	1	2	(A) ← ((Ri)) External RAM
	MOVX A, @DPTR	1 1 1 0 0 0 0 0	E0	1	2	(A) ← ((DPTR)) External RAM
	MOVX @Ri, A	1 1 1 1 0 0 1 1	F2 ~ F3	1	2	((Ri)) ← (A) External RAM
	MOVX @DPTR, A	1 1 1 1 0 0 0 0	F0	1	2	((DPTR)) ← (A) External RAM
	PUSH direct	1 1 0 0 0 0 0 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	C0 Byte 2	2	2	(SP) ← (SP) + 1 ((SP)) ← (direct)
	POP direct	1 1 0 1 0 0 0 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	D0 Byte 2	2	2	(direct) ← ((SP)) (SP) ← (SP) − 1
	XCH A, Rn	1 1 0 0 1 n_2 n_1 n_0	C8 ~ CF	1	1	(A) ⇄ (Rn)
	XCH A, direct	1 1 0 0 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	C5 Byte 2	2	1	(A) ⇄ (direct)

INSTRUCTION SET DETAILS (CONT.)

	Mnemonic	Instruction Code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Hexa-decimal	Byte	Cycle	Explanation
Data transfer	XCH A, @Ri	1 1 0 0 0 1 1 1	C6 ~ C7	1	1	(A) ⇄ ((Ri))
	XCHD A, @Ri	1 1 0 1 0 1 1 1	D6 ~ D7	1	1	($A_{3\sim0}$) ⇄ (($Ri_{3\sim0}$))
Boolean variable manipulation	CLR C	1 1 0 0 0 0 1 1	C3	1	1	(C) ← 0
	CLR bit	1 1 0 0 0 0 1 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	C2 Byte 2	2	1	(bit) ← 0
	SETB C	1 1 0 1 0 0 1 1	D3	1	1	(C) ← 1
	SETB bit	1 1 0 1 0 0 1 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	D2 Byte 2	2	1	(bit) ← 1
	CPL C	1 0 1 1 0 0 1 1	B3	1	1	(C) ← ($\overline{C}$)
	CPL bit	1 0 1 1 0 0 1 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	B2 Byte 2	2	1	(bit) ← ($\overline{bit}$)
	ANL C, bit	1 0 0 0 0 0 1 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	82 Byte 2	2	2	(C) ← (C) AND (bit)
	ANL C,/bit	1 0 1 1 0 0 0 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	B0 Byte 2	2	2	(C) ← (C) AND ($\overline{bit}$)
	ORL C, bit	0 1 1 1 0 0 1 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	72 Byte 2	2	2	(C) ← (C) OR (bit)
	ORL C,/bit	1 0 1 0 0 0 0 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	A0 Byte 2	2	2	(C) ← (C) OR ($\overline{bit}$)
	MOV C, bit	1 0 1 0 0 0 1 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	A2 Byte 2	2	1	(C) ← (bit)
	MOV bit, C	1 0 0 1 0 0 1 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	92 Byte 2	2	2	(bit) ← (C)
Program branching	ACALL addr 11	a_{10} a_9 a_8 1 0 0 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	Byte 1 Byte 2	2	2	(PC) ← (PC) + 2 (SP) ← (SP) + 1 ((SP)) ← ($PC_{7\sim0}$) (SP) ← (SP) + 1 ((SP)) ← ($PC_{15\sim8}$) (PC) ← page address
	LCALL addr 16	0 0 0 1 0 0 1 0 a_{15} a_{14} a_{13} a_{12} a_{11} a_{10} a_9 a_8 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	12 Byte 2 Byte 3	3	2	(PC) ← (PC) + 3 (SP) ← (SP) + 1 ((SP)) ← ($PC_{7\sim0}$) (SP) ← (SP) + 1 ((SP)) ← ($PC_{15\sim8}$) (PC) ← $addr_{15\sim0}$
	RET	0 0 1 0 0 0 1 0	22	1	2	($PC_{15\sim8}$) ← ((SP)) (SP) ← (SP) − 1 ($PC_{7\sim0}$) ← ((SP)) (SP) ← (SP) − 1
	RETI	0 0 1 1 0 0 1 0	32	1	2	($PC_{15\sim8}$) ← ((SP)) (SP) ← (SP) − 1 ($PC_{7\sim0}$) ← ((SP)) (SP) ← (SP) − 1
	AJMP addr 11	a_{10} a_9 a_8 0 0 0 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	Byte 1 Byte 2	2	2	(PC) ← (PC) + 2 ($PC_{10\sim0}$) ← page address
	LJMP addr 16	0 0 0 0 0 0 1 0 a_{15} a_{14} a_{13} a_{12} a_{11} a_{10} a_9 a_8 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	02 Byte 2 Byte 3	3	2	(PC) ← $addr_{15\sim0}$

6

INSTRUCTION SET DETAILS (CONT.)

	Mnemonic	Instruction Code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Hexa-decimal	Byte	Cycle	Explanation
Program branching	SJMP rel	1 0 0 0 0 0 0 0 r_7 r_6 r_5 r_4 r_3 r_2 r_1 r_0	80 Byte 2	2	2	(PC) ← (PC) + 2 (PC) ← (PC) + rel
	JMP @A+DPTR	0 1 1 1 0 0 1 1	73	1	2	(PC) ← (A) + (DPTR)
	JZ rel	0 1 1 0 0 0 0 0 r_7 r_6 r_5 r_4 r_3 r_2 r_1 r_0	60 Byte	2	2	(PC) ← (PC) + 2 IF (A) = 0 THEN (PC) ← (PC) + rel
	JNZ rel	0 1 1 1 0 0 0 0 r_7 r_6 r_5 r_4 r_3 r_2 r_1 r_0	70 Byte 2	2	2	(PC) ← (PC) + 2 IF (A) ≠ 0 THEN (PC) ← (PC) + rel
	JC rel	0 1 0 0 0 0 0 0 r_7 r_6 r_5 r_4 r_3 r_2 r_1 r_0	40 Byte 2	2	2	(PC) ← (PC) + 2 IF (C) ≠ 1 THEN (PC) ← (PC) + rel
	JNC rel	0 1 0 1 0 0 0 0 r_7 r_6 r_5 r_4 r_3 r_2 r_1 r_0	50 Byte 2	2	2	(PC) ← (PC) + 2 IF (C) ≠ 0 THEN (PC) ← (PC) + rel
	JB bit, rel	0 0 1 0 0 0 0 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0 r_7 r_6 r_5 r_4 r_3 r_2 r_1 r_0	20 Byte 2 Byte 3	3	2	(PC) ← (PC) + 3 IF (bit) = 1 THEN (PC) ← (PC) + rel
	JNB bit, rel	0 0 1 1 0 0 0 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0 r_7 r_6 r_5 r_4 r_3 r_2 r_1 r_0	30 Byte 2 Byte 3	3	2	(PC) ← (PC) + 3 IF (bit) = 0 THEN (PC) ← (PC) + rel
	JBC bit, rel	0 0 0 1 0 0 0 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0 r_7 r_6 r_5 r_4 r_3 r_2 r_1 r_0	10 Byte 2 Byte 3	3	2	(PC) ← (PC) + 3 IF (bit) = 1 THEN (bit) ← 0 (PC) ← (PC) + rel

	CJNE A, direct, rel	1 0 1 1 0 1 0 1
		a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0
		r_7 r_6 r_5 r_4 r_3 r_2 r_1 r_0
			B5
			Byte 2
			Byte 3
				3	2	(PC) ← (PC) + 3
						IF (direct) < (A) THEN (PC) ← (PC) + rel and (C) ← 0
						IF (direct) > (A) THEN (PC) ← (PC) + rel and (C) ← 1

	CJNE A, #data, rel	1 0 1 1 0 1 0 0 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0 r_7 r_6 r_5 r_4 r_3 r_2 r_1 r_0	B4 Byte 2 Byte 3	3	2	(PC) ← (PC) + 3 IF #data < (A) THEN (PC) ← (PC) + rel and (C) ← 0 IF #data > (A) THEN (PC) ← (PC) + rel and (C) ← 1
	CJNE Rn, #data, rel	1 0 1 1 1 n_2 n_1 n_0 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0 r_7 r_6 r_5 r_4 r_3 r_2 r_1 r_0	B8 ~ BF Byte 2 Byte 3	3	2	(PC) ← (PC) + 3 IF #data < (Rn) THEN (PC) ← (PC) + rel and (C) ← 0 IF #data > (Rn) THEN (PC) ← (PC) + rel and (C) ← 1
	CJNE @Ri, #data, rel	1 0 1 1 0 1 1 1 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0 r_7 r_6 r_5 r_4 r_3 r_2 r_1 r_0	B6 ~ B7 Byte 2 Byte 3	3	2	(PC) ← (PC) + 3 IF #data < ((Ri)) THEN (PC) ← (PC) + rel and (C) ← 0 IF #data > ((Ri)) THEN (PC) ← (PC) + rel and (C) ← 1
	DJNZ Rn, rel	1 1 0 1 1 n_2 n_1 n_0 r_7 r_6 r_5 r_4 r_3 r_2 r_1 r_0	D8 ~ DF Byte 2	2	2	(PC) ← (PC) + 2 (Rn) ← (Rn) − 1 IF (Rn) ≠ 0 THEN (PC) ← (PC) + rel
	DJNZ direct, rel	1 1 0 1 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0 r_7 r_6 r_5 r_4 r_3 r_2 r_1 r_0	D5 Byte 2 Byte 3	3	2	(PC) ← (PC) + 3 (direct) ← (direct) − 1 IF (direct) ≠ 0 THEN (PC) ← (PC) + rel
	NOP	0 0 0 0 0 0 0 0	00	1	1	(PC) ← (PC) + 1

NOTES ON THE INSTRUCTION SET AND THE ADDRESSING MODES

Rn	– Register R7-R0 of the currently selected Register Bank.
direct	– 8-bit internal data location's address. This could be an Internal Data RAM location (0 – 127) or a SFR [i.e., I/O port, control register, status register, etc. (128 – 255)].
@Ri	– 8-bit internal data RAM location (0 – 255) addressed indirectly through register R1 or R0.
#data	– 8-bit constant included in instruction.
#data 16	– 16-bit constant included in instruction.
addr 16	– 16-bit destination address. Used by LCALL & LJMP. A branch can be anywhere within the 64K-byte Program Memory address space.
addr 11	– 11-bit destination address. Used by ACALL & AJMP. The branch will be within the same 2K-byte page of program memory as the first byte of the following instruction.
rel	– Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is −128 to +127 bytes relative to first byte of the following instruction.
bit	– Direct Addressed bit in Internal Data RAM or Special Function Register.

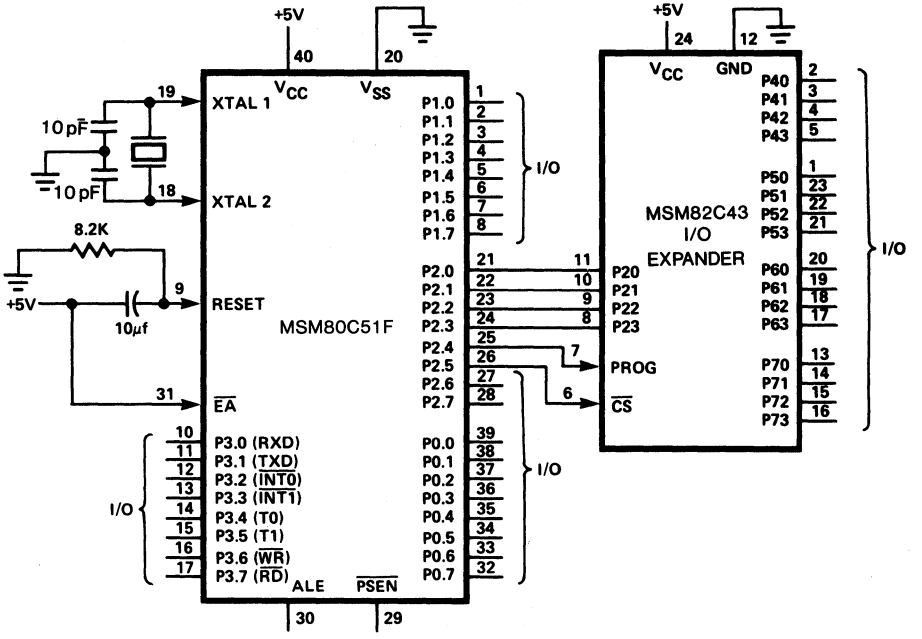
INSTRUCTIONS THAT AFFECT FLAG SETTINGS[1]

INSTRUCTION	FLAG		
	C	OV	AC
ADD	X	X	X
ADDC	X	X	X
SUBB	X	X	X
MUL	O	X	
DIV	O	X	
DA	X		
RRC	X		
RLC	X		
SETB C	I		

INSTRUCTION	FLAG		
	C	OV	AC
CLR C	O		
CPL C	X		
ANL C, bit	X		
ANL C,/bit	X		
ORL C, bit	X		
ORL C, /bit	X		
MOV C, bit	X		
CJNE	X		

[1] Note that operations on SFR byte address 208 or bit addresses 208-215 (i.e., the PSW or bits in the PSW) will also affect flag settings.

APPLICATION EXAMPLES



The following software driver is required to interface to the 82C43
Mixing Parallel Output, Input, and
Control Strobes on Port 2.

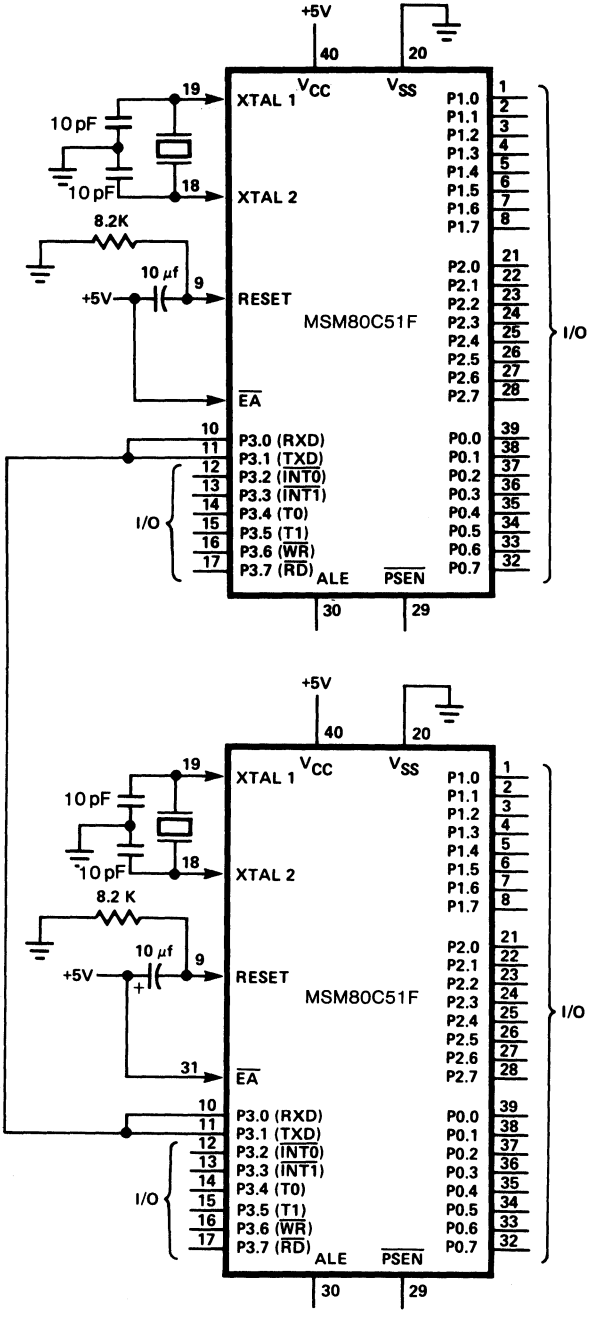
```
IN82C43   INPUT DATA FROM AN 82C43 I/O EXPANDER
          CONNECT TO P23-P20
          P25 & P24 MIMIC CS/ & PROG
          P27-P26 USED AS INPUTS
          PORT TO BE READ IN ACC

IN82C43   MOV     A, #11010000B
          MOV     P2,A          ;OUTPUT INSTRUCTION CODE
          CLR     P2.4          ;FALLING EDGE OF PROG
          ORL     P2,#00001111B ;SET FOR INPUT
          MOV     A,P2          ;READ INPUT DATA
          SETB    P2.4          ;RETURN PROG HIGH
```

P21	P20	Address Code	P23	P22	Instruction Code
0	0	Port 4	0	0	Read
0	1	Port 5	0	1	Write
1	0	Port 6	1	0	ORLD
1	1	Port 7	1	1	ANLD

I/O Expansion Using an 82C43

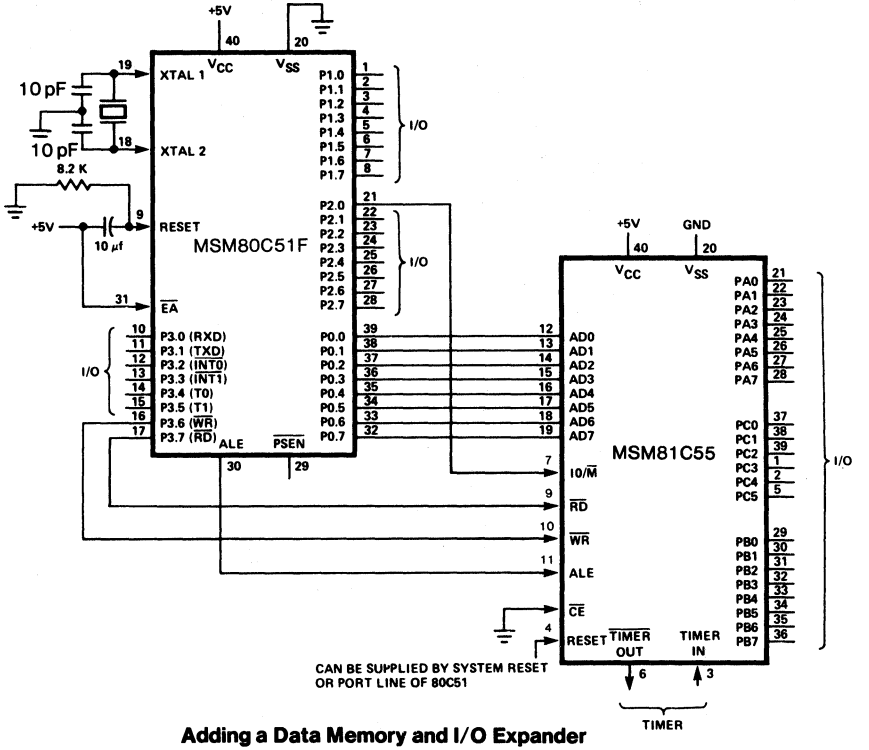
APPLICATION EXAMPLES (CONT.)



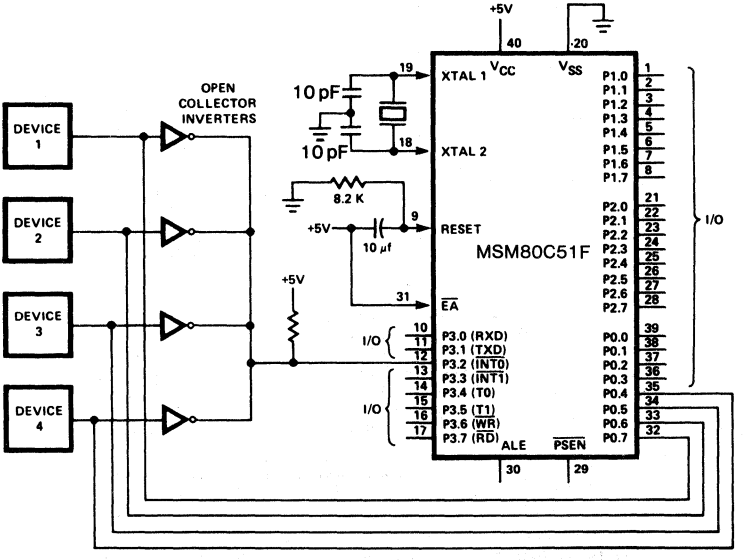
Multiple 80C51F's Using Half-Duplex Serial Communication

6

APPLICATION EXAMPLES (CONT.)



Adding a Data Memory and I/O Expander



Multiple Interrupt Sources

MSM80C31/MSM80C51 INSTRUCTION CODES

H \ L	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111
0 0000	NOP	AJMP address 11 (Page 0)	LJMP address 16	RR A	INC A	INC direct	INC @R0	INC @R1
1 0001	JBC bit, rel	ACALL address 11 (Page 0)	LCALL address 16	RRC A	DEC A	DEC direct	DEC @R0	DEC @R1
2 0010	JB bit, rel	AJMP address 11 (Page 1)	RET	RL A	ADD A, #data	ADD A, direct	ADD A, @R0	ADD A, @R1
3 0011	JNB bit, rel	ACALL address 11 (Page 1)	RETI	RLC A	ADDC A, #data	ADDC A, direct	ADDC A, @R0	ADDC A, @R1
4 0100	JC bit, rel	AJMP address 11 (Page 2)	ORL direct, A	ORL direct, #data	ORL A, #data	ORL A, direct	ORL A, @R0	ORL A, @R1
5 0101	JNC rel	ACALL address 11 (Page 2)	ANL direct, A	ANL direct, #data	ANL A, #data	ANL A, direct	ANL A, @R0	ANL A, @R1
6 0110	JZ rel	AJMP address 11 (Page 3)	XRL direct, A	XRL direct, #data	XRL A, #data	XRL A, direct	XRL A, @R0	XRL A, @R1
7 0111	JNZ rel	ACALL address 11 (Page 3)	ORL C, bit	JMP @A+DPTR	MOV A, #data	MOV direct, #data	MOV @R0, #data	MOV @R1, #data
8 1000	SJMP rel	AJMP address 11 (Page 4)	ANL C, bit	MOVC A, @A+PC	DIV AB	MOV direct 1, direct 2	MOV direct, @R0	MOV direct, @R1
9 1001	MOV DPTR, #data 16	ACALL address 11 (Page 4)	MOV bit, C	MOVC A, @A+DPTR	SUBB A, #data	SUBB A, direct	SUBB A, @R0	SUBB A, @R1
A 1010	ORL C/bit	AJMP address 11 (Page 5)	MOV C, bit	INC DPTR	MUL AB		MOV @R0, direct	MOV @R1, direct
B 1011	ANL C/bit	ACALL address 11 (Page 5)	CPL bit	CPL C	CJNE A, #data, rel	CJNE A, direct, rel	CJNE @R0, #data, rel	CJNE @R1, #data, rel
C 1100	PUSH direct	AJMP address 11 (Page 6)	CLR bit	CLR C	SWAP A	XCH A, direct	XCH A, @R0	XCH A, @R1
D 1101	POP direct	ACALL address 11 (Page 6)	SETB bit	SETB C	DA A	DJNZ direct, rel	XCHD A, @R0	XCHD A, @R1
E 1110	MOVX A, @DPTR	AJMP address 11 (page 7)	MOVX A, @R0	MOVX A, @R1	CLR A	MOV A, direct	MOV A, @R0	MOV A, @R1
F 1111	MOVX @DPTR, A	ACALL address 11 (page 7)	MOVX @R0, A	MOVX @R1, A	CPL A	MOV direct, A	MOV @R0, A	MOV @R1, A

2 BYTE	3 BYTE
MNEMONIC	
2 CYCLE	4 CYCLE

8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111
INC R0	INC R1	INC R2	INC R3	INC R4	INC R5	INC R6	INC R7
DEC R0	DEC R1	DEC R2	DEC R3	DEC R4	DEC R5	DEC R6	DEC R7
ADD A, R0	ADD A, R1	ADD A, R2	ADD A, R3	ADD A, R4	ADD A, R5	ADD A, R6	ADD A, R7
ADDC A, R0	ADDC A, R1	ADDC A, R2	ADDC A, R3	ADDC A, R4	ADDC A, R5	ADDC A, R6	ADDC A, R7
ORL A, R0	ORL A, R1	ORL A, R2	ORL A, R3	ORL A, R4	ORL A, R5	ORL A, R6	ORL A, R7
ANL A, R0	ANL A, R1	ANL A, R2	ANL A, R3	ANL A, R4	ANL A, R5	ANL A, R6	ANL A, R7
XRL A, R0	XRL A, R1	XRL A, R2	XRL A, R3	XRL A, R4	XRL A, R5	XRL A, R6	XRL A, R7
MOV R0, #data	MOV R1, #data	MOV R2, #data	MOV R3, #data	MOV R4, #data	MOV R5, #data	MOV R6, #data	MOV R7, #data
MOV direct, R0	MOV direct, R1	MOV direct, R2	MOV direct, R3	MOV direct, R4	MOV direct, R5	MOV direct, R6	MOV direct, R7
SUBB A, R0	SUBB A, R1	SUBB A, R2	SUBB A, R3	SUBB A, R4	SUBB A, R5	SUBB A, R6	SUBB A, R7
MOV R0, direct	MOV R1, direct	MOV R2, direct	MOV R3, direct	MOV R4, direct	MOV R5, direct	MOV R6, direct	MOV R7, direct
CJNE R0, #data, rel	CJNE R1, #data, rel	CJNE R2, #data, rel	CJNE R3, #data, rel	CJNE R4, #data, rel	CJNE R5, #data, rel	CJNE R6, #data, rel	CJNE R7, #data, rel
XCH A, R0	XCH A, R1	XCH A, R2	XCH A, R3	XCH A, R4	XCH A, R5	XCH A, R6	XCH A, R7
DJNZ R0, rel	DJNZ R1, rel	DJNZ R2, rel	DJNZ R3, rel	DJNZ R4, rel	DJNE R5, rel	DJNE R6, rel	DJNE R7, rel
MOV A, R0	MOV A, R1	MOV A, R2	MOV A, R3	MOV A, R4	MOV A, R5	MOV A, R6	MOV A, R7
MOV R0, A	MOV R1, A	MOV R2, A	MOV R3, A	MOV R4, A	MOV R5, A	MOV R6, A	MOV R7, A

OKI semiconductor

MSM80C154/ MSM83C154

CMOS 8-bit One-Chip Microcontroller

GENERAL DESCRIPTION

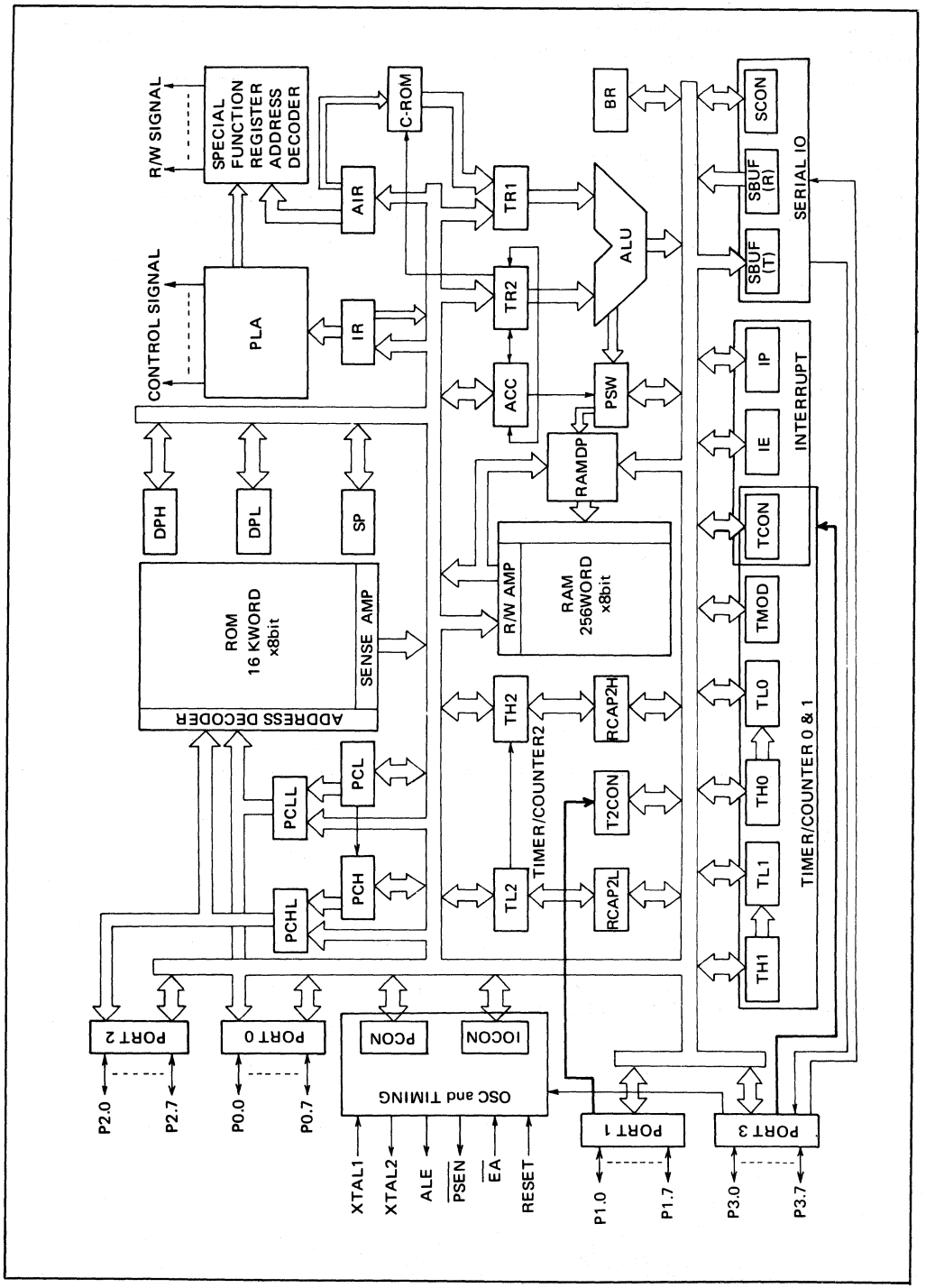
The MSM83C154/MSM80C154 is a high performance 8-bit one-chip microcontroller implementing large integration, high speed and low power consumption by 2 μm silicon gate CMOS process technology.
The MSM83C154 features 16K byte ROM, 256 byte RAM, 32 I/O ports, three 16-bit timer/counters, multifunctional serial port and clock generator. In addition, the MSM83C154 has three standby modes enabling further power reduction.
The MSM80C154 is identical to the MSM83C154 except the omission of 16K byte ROM.

FEATURES

- Fully static circuit
- On-chip program memory : 16K x 8 bit ROM (MSM83C154 only)
- On-chip data memory : 256 x 8 bit RAM
- External program memory address space : 64K bytes
- External data memory address space : 64K bytes
- I/O ports
  (Port 1, 2, 3, impedance programmable) : 32
- 16-bit timer/counters
  (includes watch dog timer & 32 bit timer) : 3
- Multifunctional serial port : I/O Expansion mode
  : UART mode (featuring error detection)
- 6-source 2-priority level
  interrupt and multi-level
  interrupt available by programming IP and IE registers
- Memory-mapped special function registers
- Bit addressable data memory and SFRs
- Minimum instruction cycle : 1.0 μs @ 12 MHz operation
  : 0.75 μs @ 16 MHz operation (MSM80C154-1)
  16 MHz version of MSM83C154 (12 MHz < XTAL1 • 2 ⩽ 16 MHz) is now under development.
- "Multiply"/"divide" instruction cycle : 3 μs @16 MHz operation
- Standby functions : Idle mode (CPU halt)
  : Power down mode (Oscillator stop)
  Activated by Software or Hardware; Providing ports with floating or active status
  The software power down mode is terminated by interrupt signal enabling excution from the interrupted address.
- Lower power consumption achieved by 2 μm silicon gate CMOS process
- Upward compatible with MSM80C51/80C31
- Packages : 40-pin DIP, 44-pin flat package and 44-pin PLCC

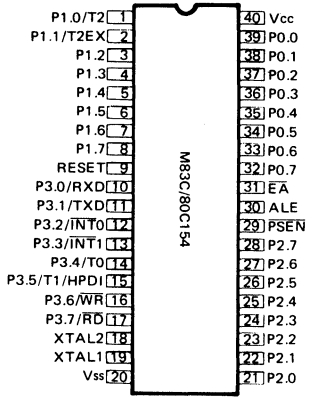
6

CIRCUIT BLOCK DIAGRAM

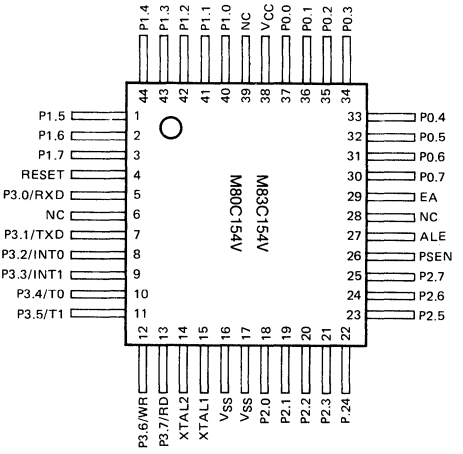


PIN CONFIGURATION

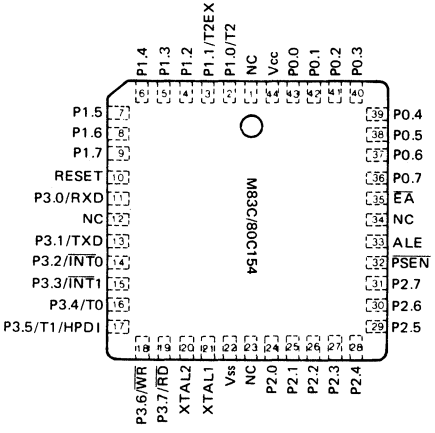
MSM83C154-XXRS/MSM80C154RS
(Top-View) 40 Lead Plastic DIP



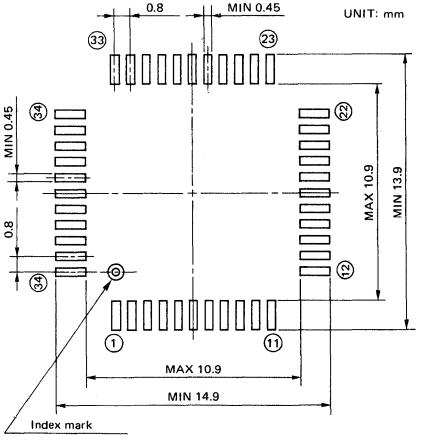
MSM83C154V-XXGS/MSM80C154VGS
(Top-View) 44 Lead Plastic Package



MSM83C154-XXJS/MSM80C154JS *1
(Top-View) 44 Lead Plastic Leaded Chip Carrier



RECOMMENDED FOOTPRINT
FOR QPD PACKAGE



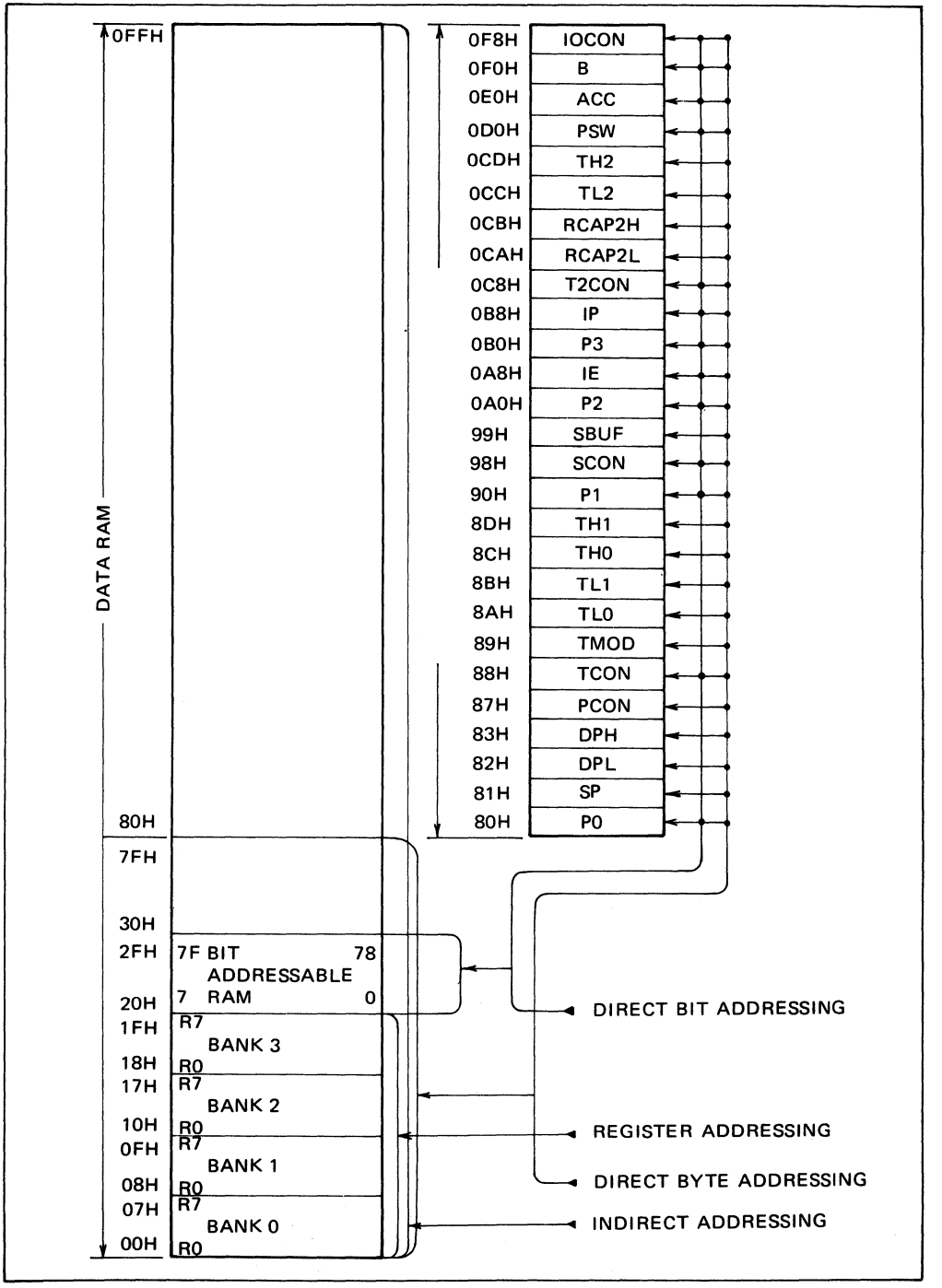
PIN FUNCTIONS

Pin Name	Description
P0.0 ~ P0.7	Bidirectional I/O ports. They are also the data/address bus (input/output of data and output of lower 8-bit address when external memory is accessed. They are open drain output when used as I/O ports, but tri-state output when used as data/address bus.
P1.0 ~ P1.7	P1.0 to P1.7 are quasi-bidirectional I/O ports. They are pulled up internally when used as input ports. Two of them have the following secondary functions: • P1.0 (T2) : Used as external clock input pin for the timer/counter 2. • P1.1 (T2EX) : Used as trigger input for the timer/counter 2 to be reloaded or captured; causing the timer/counter 2 interrput.

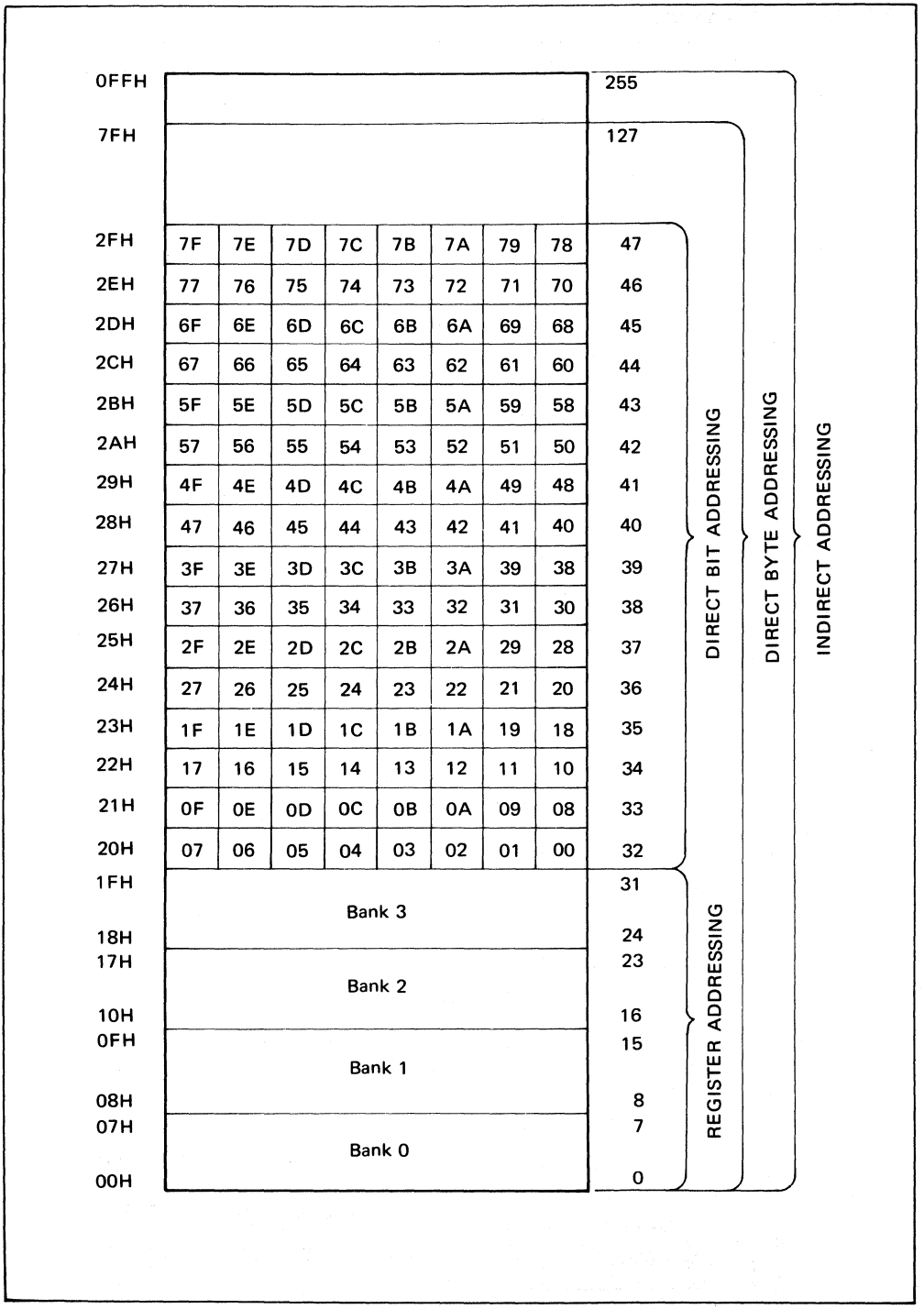
PIN FUNCTIONS (CONT.)

Pin Name	Description
P2.0 ~ P2.7	P2.0 to P2.7 are quasi-bidirectional I/O ports. They also output the higher 8-bit address when an external memory is accessed. They are pulled up internally when used as input ports.
P3.0 ~ P3.7	P3.0 to P3.7 are quasi-bidirectional I/O ports. They are pulled up internally when used as input ports. They also have the following secondary functions: • P3.0 (RXD) Serial data input/output in the I/O expansion mode and serial data input in the UART mode when the serial port is used. • P3.1 (TXD) Synchronous clock output in the I/O expansion mode and serial data output in the UART mode when the serial port is used. • P3.2 ($\overline{INT0}$) Used as input pin for the external interrupt 0, and as count-up control pin for the timer/counter 0. • P3.3 ($\overline{INT1}$) Used as input pin for the external interrupt 1, and as count-up control pin for the timer/counter 1. • P3.4 (T0) Used as external clock input pin for the timer/counter 0. • P3.5 (T1) Used as external clock input pin for the timer/counter 1 and power down mode control input pin. • P3.6 ($\overline{WR}$) Output of the write strobe signal when data is written into external data memory. • P3.7 ($\overline{RD}$) Output of the read strobe signal when data is read from external data memory.
ALE	Address latch enable output for latching the lower 8-bit address during external memory access. Two ALE pulses are activated per machine cycle except during external data memory access at which time one ALE pulse is skipped.
$\overline{PSEN}$	Program store enable output which enable the external memory output to the bus during external program memory access. Two $\overline{PSEN}$ pulses are activated per machine cycle except during external data memory access at which two $\overline{PSEN}$ pulses are skipped.
$\overline{EA}$	When $\overline{EA}$ is held at "H" level, the MSM83C154 executes instructions from internal program memory at address 0000H to 3FFFH, and executes instructions from external program memory above address 3FFFH. When $\overline{EA}$ is held at "L" level, the MSM80C154/MSM83C154 executes instructions from external program memory for all addresses.
RESET	If this pin remains "H" for at least 1 μ second, the MSM80C154/MSM83C154 is reset. Since this pin is pulled down internally, a power-on reset is achieved by simply connecting a capacitor between Vcc and this pin.
XTAL1	Oscillator inverter input pin. External clock is input through XTAL1 pin.
XTAL2	Oscillator inverter output pin.
V_{CC}	Power supply pin during both normal operation and standby operations.
V_{SS}	GND pin.

DATA MEMORY AND SPECIAL FUNCTION REGISTER LAYOUT DIAGRAM



DETAILED DIAGRAM OF DATA MEMORY (RAM)



DETAILED DIAGRAM OF SPECIAL FUNCTION REGISTERS

Direct Byte Address	Bit Address (MSB)							(LSB)	Special Function Register Symbol
	WDT	T32	SERR	IZC	P3HZ	P2HZ	P1HZ	ALF	
0F8H	FF	FE	FD	FC	FB	FA	F9	F8	IOCON
0F0H	F7	F6	F5	F4	F3	F2	F1	F0	B
0E0H	E7	E6	E5	E4	E3	E2	E1	E0	ACC
	CY	AC	F0	RS1	RS0	0V	F1	P	
0D0H	D7	D6	D5	D4	D3	D2	D1	D0	PSW
0CDH	Not Bit Addressable								TH2
0CCH	Not Bit Addressable								TL2
0CBH	Not Bit Addressable								RCAP2H
0CAH	Not Bit Addressable								RCAP2L
	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$	
0C8H	CF	CE	CD	CC	CB	CA	C9	C8	T2CON
	PCT		PT2	PS	PT1	PX1	PT0	PX0	
0B8H	BF	–	BD	BC	BB	BA	B9	B8	IP
0B0H	B7	B6	B5	B4	B3	B2	B1	B0	P3
	EA		ET2	ES	ET1	EX1	ET0	EX0	
0A8H	AF	–	AD	AC	AB	AA	A9	A8	IE
0A0H	A7	A6	A5	A4	A3	A2	A1	A0	P2
99H	Not Bit Addressable								SBUF
	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	
98H	9F	9E	9D	9C	9B	9A	99	98	SCON
90H	97	96	95	94	93	92	91	90	P1

Direct Byte Address	Bit Address (MSB)							(LSB)	Special Function Register Symbol
8DH	Not Bit Addressable								TH1
8CH	Not Bit Addressable								TH0
8BH	Not Bit Addressable								TL1
8AH	Not Bit Addressable								TL0
89H	Not Bit Addressable								TMOD
	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	
88H	8F	8E	8D	8C	8B	8A	89	88	TCON
87H	Not Bit Addressable								PCON
83H	Not Bit Addressable								DPH
82H	Not Bit Addressable								DPL
81H	Not Bit Addressable								SP
80H	87	86	85	84	83	82	81	80	P0

SPECIAL FUNCTION REGISTERS

Timer mode register (TMOD)

NAME	ADDRESS	MSB 7	6	5	4	3	2	1	LSB 0
TMOD	89H	GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0

BIT LOCATION	FLAG	FUNCTION		
TMOD.0	M0	M1	M0	Timer/counter 0 mode setting
		0	0	8-bit timer/counter with 5-bit prescalar.
		0	1	16-bit timer/counter.
		1	0	8-bit timer/counter with 8-bit auto reloading.
TMOD.1	M1	1	1	Timer/counter 0 separated into TL0 (8-bit) timer/counter and TH0 (8-bit) timer/counter. TF0 is set by TL0 carry, and TF1 is set by TH0 carry.
TMOD.2	C/$\overline{T}$	Timer/counter 0 count clock designation control bit. XTAL1 • 2 divided by 12 clocks is the input applied to timer/counter 0 when C/$\overline{T}$ = "0". The external clock applied to the T0 pin is the input applied to timer/counter 0 when C/$\overline{T}$ = "1".		
TMOD.3	GATE	When this bit is "0", the TR0 bit of TCON (timer control register) is used to control the start and stop of timer/counter 0 counting. If this bit is "1", timer/counter 0 starts counting when both the TR0 bit of TCON and $\overline{INT0}$ pin input signal are "1", and stops counting when either is changed to "0".		
TMOD.4	M0	M1	M0	Timer/counter 1 mode setting.
		0	0	8-bit timer/counter with 5-bit prescalar.
		0	1	16-bit timer/counter.
		1	0	8-bit timer/counter with 8-bit auto reloading.
TMOD.5	M1	1	1	Timer/counter 1 operation stopped.
TMOD.6	C/$\overline{T}$	Timer/counter 1 count clock designation control bit. XTAL1 • 2 divided by 12 clocks is the input applied to timer/counter 1 when C/$\overline{T}$ = "0". The external clock applied to the T1 pin is the input applied to timer/counter 1 when C/$\overline{T}$ = "1".		
TMOD.7	GATE	When this bit is "0", the TR1 bit of TCON is used to control the start and stop of timer/counter 1 counting. If this bit is "1", timer/counter 1 starts counting when both the TR1 bit of TCON and $\overline{INT1}$ pin input signal are "1", and stops counting when either is changed to "0".		

Power control register (PCON)

NAME	ADDRESS	MSB 7	6	5	4	3	2	1	LSB 0
PCON	87H	SMOD	HPD	RPD	–	GF1	GF0	PD	IDL

BIT LOCATION	FLAG	FUNCTION
PCON.0	IDL	IDLE mode set when this bit is set to "1". CPU operations are stopped when IDLE mode is set, but XTAL1•2, timer/counters 0, 1, and 2, the interrupt circuits, and serial port remain active. IDLE mode is cancelled when the CPU is reset or when an interrupt is generated.
PCON.1	PD	PD mode set when this bit is set to "1". CPU operations and XTAL1•2 are stopped when PD mode is set. PD mode is cancelled when the CPU is reset or when an interrupt is generated.
PCON.2	GF0	General purpose bit. Testing this flag when IDLE mode is cancelled by an interrupt shows whether the interrupt is a normal interrupt or an IDLE mode release interrupt.
PCON.3	GF1	General purpose bit. Testing this flag when PD mode is cancelled by an interrupt shows whether the interrupt is a normal interrupt or a PD mode release interrupt.
PCON.4	–	Reserved bit. The output data is "1" if the bit is read.
PCON.5	RPD	Bit used to specify cancellation of CPU power down mode (IDLE or PD) by interrupt signal. Power down mode cannot be cancelled by interrupt signal if interrupt is not enabled by IE (interrupt enable register) when this bit is "0". If the interrupt flag is set to "1" by an interrupt request signal when this bit is "1" (even if interrupt is disabled), the program is executed from the next address of the power down mode setting instruction. The flag is reset to "0" by software.
PCON.6	HPD	The hard power down setting mode is enabled when this bit is set to "1". If the level of the power failure detect signal applied to the HPDI pin (pin 3.5) is changed from "1" to "0" when this bit is "1", XTAL1•2 oscillation is stopped and the system is put into hard power down mode. HPD mode is cancelled when the CPU is reset.
PCON.7	SMOD	When the timer/counter 1 carry signal is used as a clock in mode 1, 2 or 3 of the serial port, this bit has the following functions. The serial port operation clock is reduced by 1/2 when the bit is "0" for delayed processing. And when the bit is "1", the serial port operation clock is normal for faster processing.

Timer control register (TCON)

NAME	ADDRESS	MSB 7	6	5	4	3	2	1	LSB 0
TCON	88H	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

BIT LOCATION	FLAG	FUNCTION
TCON.0	IT0	External interrupt 0 signal used in level detect mode when this bit is "0", and in trigger detect mode when "1".
TCON.1	IE0	Interrupt request flag for external interrupt 0. Bit is reset automatically when interrupt is serviced. Bit can be set and reset by software when IT0 = "1".
TCON.2	IT1	External interrupt 1 signal used in level detect mode when this bit is "0", and in trigger detect mode when "1".
TCON.3	IE1	Interrupt request flag for external interrupt 1. Bit is reset automatically when interrupt is serviced. Bit can be set and reset by software when IT1 = "1".
TCON.4	TR0	Counting start and stop control bit for timer/counter 0. Timer/counter 0 starts counting when this bit is "1", and stops counting when "0".
TCON.5	TF0	Interrupt request flag for timer interrupt 0. Bit is reset automatically when interrupt is serviced. Bit is set to "1" when carry signal is generated from timer/counter 0.
TCON.6	TR1	Counting start and stop control bit for timer/counter 1. Timer/counter 1 starts counting when this bit is "1", and stops counting when "0".
TCON.7	TF1	Interrupt request flag for timer interrupt 1. Bit is reset automatically when interrupt is serviced. Bit is set to "1" when carry signal is generated from timer/counter 1.

Serial port control register (SCON)

NAME	ADDRESS	MSB 7	6	5	4	3	2	1	LSB 0
SCON	98H	SM0	SM1	SM2	REN	TB8	RB8	TI	RI

BIT LOCATION	FLAG	FUNCTION
SCON.0	RI	"End of serial port reception" interrupt request flag. This flag must be reset by software during interrupt service routine. This flag is set after the eighth bit of data has been received when in mode 0, or by the STOP bit when in any other mode. In mode 2 or 3, however, RI is not set if the RB8 data is "0" with SM2 = "1". RI is set in mode 1 if STOP bit is received when SM2 = "1".
SCON.1	TI	"End of serial port transmission" interrupt request flag. This flag must be reset by software during interrupt service routine. This flag is set after the eighth bit of data has been sent when in mode 0, or after the last bit of data has been sent when in any other mode.
SCON.2	RB8	The ninth bit of data received in mode 2 or 3 is passed to RB8. The STOP bit is applied to RB8 if SM2 = "0" when in mode 1. RB8 can not be used in mode 0.
SCON.3	TB8	The TB8 data is sent as the ninth data bit when in mode 2 or 3. Any desired data can be set in TB8 by software.
SCON.4	REN	Reception enable control bit No reception when REN = "0". Reception enabled when REN = "1".
SCON.5	SM2	If the ninth bit of received data is "0" with SM2 = "1" in mode 2 or 3, the "end of reception" signal is not set in the RI flag. Nor is the "end of reception" signal set in the RI flag if the STOP bit is not "1" when SM2 = "1" in mode 1.
SCON.6	SM1	SM0 / SM1 / MODE /
		0 / 0 / 0 / 8-bit shift register I/O
		0 / 1 / 1 / 8-bit UART variable baud rate
SCON.7	SM0	1 / 0 / 2 / 9-bit UART 1/32 XTAL1, 1/64 XTAL1 baud rate
		1 / 1 / 3 / 9-bit UART variable baud rate

Interrupt enable register (IE)

NAME	ADDRESS	MSB 7	6	5	4	3	2	1	LSB 0
IE	0A8H	EA	–	ET2	ES	ET1	EX1	ET0	EX0

BIT LOCATION	FLAG	FUNCTION
IE.0	EX0	Interrupt control bit for external interrupt 0. Interrupt disabled when bit is "0". Interrupt enabled when bit is "1".
IE.1	ET0	Interrupt control bit for timer interrupt 0. Interrupt disabled when bit is "0". Interrupt enabled when bit is "1".
IE.2	EX1	Interrupt control bit for external interrupt 1. Interrupt disabled when bit is "0". Interrupt enabled when bit is "1".
IE.3	ET1	Interrupt control bit for timer interrupt 1. Interrupt disabled when bit is "0". Interrupt enabled when bit is "1".
IE.4	ES	Interrupt control bit for serial port. Interrupt disabled when bit is "0". Interrupt enabled when bit is "1".
IE.5	ET2	Interrupt control bit for timer interrupt 2. Interrupt disabled when bit is "0". Interrupt enabled when bit is "1".
IE.6	–	Reserved bit. The output data is "1" if the bit is read.
IE.7	EA	Overall interrupt control bit. All interrupts are disabled when bit is "0". All interrupts are controlled by IE.0 thru IE.5 when bit is "1".

Interrupt priority register (IP)

NAME	ADDRESS	MSB 7	6	5	4	3	2	1	LSB 0
IP	0B8H	PCT	–	PT2	PS	PT1	PX1	PT0	PX0

BIT LOCATION	FLAG	FUNCTION
IP.0	PX0	Interrupt priority bit for external interrupt 0. Priority is assigned when bit is "1".
IP.1	PT0	Interrupt priority bit for timer interrupt 0. Priority is assigned when bit is "1".
IP.2	PX1	Interrupt priority bit for external interrupt 1. Priority is assigned when bit is "1".
IP.3	PT1	Interrupt priority bit for timer interrupt 1. Priority is assigned when bit is "1".
IP.4	PS	Interrupt priority bit for serial port. Priority is assigned when bit is "1".
IP.5	PT2	Interrupt priority bit for timer interrupt 2. Priority is assigned when bit is "1".
IP.6	–	Reserved bit. The output data is "1" if the bit is read.
IP.7	PCT	Priority interrupt circuit control bit. The priority register contents are valid and priority assigned interrupts can be processed when this bit is "0". When the bit is "1", the priority interrupt circuit is stopped, and interrupts can only be controlled by the interrupt enable register (IE).

Program status word register (PSW)

NAME	ADDRESS	MSB 7	6	5	4	3	2	1	LSB 0
PSW	0D0H	CY	AC	F0	RS1	RS0	OV	F1	P

BIT LOCATION	FLAG	FUNCTION
PSW.0	P	Accumulator (ACC) parity indicator. "1" when the "1" bit number in the accumulator is an odd number, and "0" when an even number.
PSW.1	F1	User flag which may be set to "0" or "1" as desired by the user.
PSW.2	OV	Overflow flag which is set if the carry C6 from bit 6 of the ALU or CY is "1" as a result of an arithmetic operation. The flag is also set to "1" if the resultant product of executing a multiplication instruction (MUL AB) is greater than 0FFH, but is reset to "0" if the product is less than or equal to 0FFH.
PSW.3	RS0	RAM register bank switch
PSW.4	RS1	

RS1	RS0	BANK	RAM ADDRESS
0	0	0	00H – 07H
0	1	1	08H – 0FH
1	0	2	10H – 17H
1	1	3	18H – 1FH

PSW.5	F0	User flag which may be set to "0" or "1" as desired by the user.
PSW.6	AC	Auxiliary carry flag. This flag is set to "1" if a carry C_3 is generated from bit 3 of the ALU as a result of executing an arithmetic operation instruction. In all other cases, the flag is reset to "0".
PSW.7	CY	Main carry flag. This flag is set to "1" if a carry C_7 is generated from bit 7 of the ALU as result of executing an arithmetic operation instruction. If a carry C_7 is not generated, the flag is reset to "0".

I/O control register (IOCON)

NAME	ADDRESS	MSB 7	6	5	4	3	2	1	LSB 0
IOCON	0F8H	WDT	T32	SERR	IZC	P3HZ	P2HZ	P1HZ	ALF

BIT LOCATION	FLAG	FUNCTION
IOCON.0	ALF	If CPU power down mode (PD, HPD) is activated with this bit set to "1", the outputs from ports 0, 1, 2, and 3 are switched to floating status. When this bit is "0", ports 0, 1, 2, and 3 are in output mode.
IOCON.1	P1HZ	Port 1 becomes a high impedance input port when this bit is "1".
IOCON.2	P2HZ	Port 2 becomes a high impedance input port when this bit is "1".
IOCON.3	P3HZ	Port 3 becomes a high impedance input port when this bit is "1".
IOCON.4	IZC	The 10 kohm pull-up resistance for ports 1, 2, and 3 is switched off when this bit is "1", leaving only the 100 kohm pull-up resistance.
IOCON.5	SERR	Serial port reception error flag. This flag is set to "1" if an overrun or framing error is generated when data is received at a serial port. The flag is reset by software.
IOCON.6	T32	Timer/counters 0 and 1 are connected serially to form a 32-bit timer/counter when this bit is set to "1". TF1 of TCON is set if a carry is generated in the 32-bit timer/counter.
IOCON.7	WDT	Watchdog timer mode is set when this bit is set to "1". And if TF1 is set to "1" after watchdog timer mode has been set, the CPU is reset and the program is executed from address 0.

Timer 2 control register (T2CON)

NAME	ADDRESS	MSB 7	6	5	4	3	2	1	LSB 0
T2CON	0C8H	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$

BIT LOCATION	FLAG	FUNCTION
T2CON.0	CP/$\overline{RL2}$	Capture mode is set when TCLK + RCLK = "0" and CP/$\overline{RL2}$ = "1". 16-bit auto reload mode is set when TCLK + RCLK = "0" and CP/$\overline{RL2}$ = "0". CP/$\overline{RL2}$ is ignored when TCLK + RCLK = "1".
T2CON.1	C/$\overline{T2}$	Timer/counter 2 count clock designation control bit. The internal clocks (XTAL1 • 2 ÷ 12, XTAL1 • 2 ÷ 2) are used when this bit is "0", and the external clock applied to the T2 pin is passed to timer/counter 2 when the bit is "1".
T2CON.2	TR2	Timer/counter 2 counting start and stop control bit. Timer/counter 2 commences counting when this bit is "1" and stops counting when "0".
T2CON.3	EXEN2	T2EX timer/counter 2 external control signal control bit. Input of the T2EX signal is disabled when this bit is "0", and enabled when "1".
T2CON.4	TCLK	Serial port transmit circuit drive clock control bit. Timer/counter 2 is switched to baud rate generator mode when this bit is "1", and the timer/counter 2 carry signal becomes the serial port transmit clock. Note, however, that the serial ports can only use the timer/counter 2 carry signal in serial port modes 1 and 3.
T2CON.5	RCLK	Serial port receive circuit drive clock control bit. Timer/counter 2 is switched to baud rate generator mode when this bit is "1", and the timer/counter 2 carry signal becomes the serial port receive clock. Note, however, that the serial ports can only use the timer/counter 2 carry signal in serial port modes 1 and 3.
T2CON.6	EXF2	Timer/counter 2 external flag. This bit is set to "1" when the T2EX timer/counter 2 external control signal level is changed from "1" to "0" while EXEN2 = "1". This flag serves as the timer interrupt 2 request signal. If an interrupt is generated, EXF2 must be reset to "0" by software.
T2CON.7	TF2	Timer/counter 2 carry flag. This bit is set to "1" by a carry signal when timer/counter 2 is in 16-bit auto reload mode or in capture mode. This flag serves as the timer interrupt 2 request signal. If an interrupt is generated, TF2 must be reset to "0" by software.

LIST OF INSTRUCTIONS

LIST OF INSTRUCTION SYMBOLS

A : Accumulator
AB : Register pair
AC : Auxiliary carry flag
B : Arithmetic operation register
C : Carry flag
DPTR : Data pointer
PC : Program counter
Rr : Register indicator (r = 0 ~ 7)
SP : Stack pointer
AND : Logical product
OR : Logical sum
XOR : Exclusive OR
+ : Addition
− : Subtraction
X : Multiplication
/ : Division
(X) : Denotes the contents of X
((X)) : Denotes the contents of address determined by the contents of X
: Denotes the immediate data
@ : Denotes the indirect address
= : Equality
≠ : Non equality
← : Substitution
→ : Substitution
¯ : Negation
< : Smaller than
> : Larger than
bit address : RAM and the special function register bit specifier address ($b_0 \sim b_7$)
code address : Absolute address ($A_0 \sim A_{15}$)
data : Immediate data ($I_0 \sim I_7$)
relative offset: Relative jump address offset value ($R_0 \sim R_7$)
direct address: RAM and the special function register byte specifier address ($a_0 \sim a_7$)

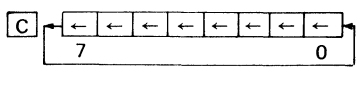
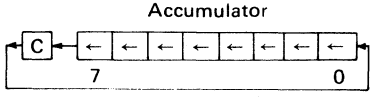
MSM80C154/MSM83C154 INSTRUCTION TABLE

L / H	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111
0 0000	NOP	AJMP address 11 (Page 0)	LJMP address 16	RR A	INC A	INC direct	INC @R0	INC @R1
1 0001	JBC bit, rel	ACALL address 11 (Page 0)	LCALL address 16	RRC A	DEC A	DEC direct	DEC @R0	DEC @R1
2 0010	JB bit, rel	AJMP address 11 (Page 1)	RET	RL A	ADD A, #data	ADD A, direct	ADD A, @R0	ADD A, @R1
3 0011	JNB bit, rel	ACALL address 11 (Page 1)	RETI	RLC A	ADDC A, #data	ADDC A, direct	ADDC A, @R0	ADDC A, @R1
4 0100	JC bit, rel	AJMP address 11 (Page 2)	ORL direct, A	ORL direct, #data	ORL A, #data	ORL A, direct	ORL A, @R0	ORL A, @R1
5 0101	JNC rel	ACALL address 11 (Page 2)	ANL direct, A	ANL direct, #data	ANL A, #data	ANL A, direct	ANL A, @R0	ANL A, @R1
6 0110	JZ rel	AJMP address 11 (Page 3)	XRL direct, A	XRL direct, #data	XRL A, #data	XRL A, direct	XRL A, @R0	XRL A, @R1
7 0111	JNZ rel	ACALL address 11 (Page 3)	ORL C, bit	JMP @A+DPTR	MOV A, #data	MOV direct, #data	MOV @R0, #data	MOV @R1, #data
8 1000	SJMP rel	AJMP address 11 (Page 4)	ANL C, bit	MOVC A, @A+PC	DIV AB	MOV direct 1, direct 2	MOV direct, @R0	MOV direct, @R1
9 1001	MOV DPTR #data 16	ACALL address 11 (Page 4)	MOV bit, C	MOVC A, @A+DPTR	SUBB A, #data	SUBB A, direct	SUBB A, @R0	SUBB A, @R1
A 1010	ORAL C, bit	AJMP address 11 (Page 5)	MOV C, bit	INC DPTR	MUL AB		MOV @R0, direct	MOV @R1, direct
B 1011	ANL C, bit	ACALL address 11 (Page 5)	CPL bit	CPL C	CJNE A, #data, rel	CJNE A, direct, rel	CJNE @R0, #data, rel	CJNE @R1, #data, rel
C 1100	PUSH direct	AJMP address 11 (Page 6)	CLR bit	CLR C	SWAP A	XCH A, direct	XCH A, @R0	XCH A, @R1
D 1101	POP direct	ACALL address 11 (Page 6)	SETB bit	SETB C	DA A	DJNZ direct, rel	XCHD A, @R0	XCH A, @R1
E 1110	MOVX A, @DPTR	AJMP address 11 (Page 7)	MOVX A, @R0	MOVX A, @R1	CLR A	MOV A, direct	MOV A, @R0	MOV A, @R1
F 1111	MOVX @DPTR, A	ACALL address 11 (Page 7)	MOVX @R0, A	MOVX @R1, A	CPL A	MOV direct, A	MOV @R0, A	MOV @R1, A

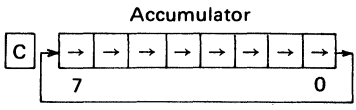
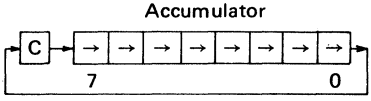
2 BYTE		3 BYTE
	MNEMONIC	
2 CYCLE		4 CYCLE

L / H	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111
0 0000	INC R0	INC R1	INC R2	INC R3	INC R4	INC R5	INC R6	INC R7
1 0001	DEC R0	DEC R1	DEC R2	DEC R3	DEC R4	DEC R5	DEC R6	DEC R7
2 0010	ADD A, R0	ADD A, R1	ADD A, R2	ADD A, R3	ADD A, R4	ADD A, R5	ADD A, R6	ADD A, R7
3 0011	ADDC A, R0	ADDC A, R1	ADDC A, R2	ADDC A, R3	ADDC A, R4	ADDC A, R5	ADDC A, R6	ADDC A, R7
4 0100	ORL A, R0	ORL A, R1	ORL A, R2	ORL A, R3	ORL A, R4	ORL A, R5	ORL A, R6	ORL A, R7
5 0101	ANL A, R0	ANL A, R1	ANL A, R2	ANL A, R3	ANL A, R4	ANL A, R5	ANL A, R6	ANL A, R7
6 0110	XRL A, R0	XRL A, R1	XRL A, R2	XRL A, R3	XRL A, R4	XRL A, R5	XRL A, R6	XRL A, R7
7 0111	MOV R0, #data	MOV R1, #data	MOV R2, #data	MOV R3, #data	MOV R4, #data	MOV R5, #data	MOV R6, #data	MOV R7, #data
8 1000	MOV direct, R0	MOV direct, R1	MOV direct, R2	MOV direct, R3	MOV direct, R4	MOV direct, R5	MOV direct, R6	MOV direct, R7
9 1001	SUBB A, R0	SUBB A, R1	SUBB A, R2	SUBB A, R3	SUBB A, R4	SUBB A, R5	SUBB A, R6	SUBB A, R7
A 1010	MOV R0, direct	MOV R1, direct	MOV R2, direct	MOV R3, direct	MOV R4, direct	MOV R5, direct	MOV R6, direct	MOV R7, direct
B 1011	CJNE R0, #data, rel	CJNE R1, #data, rel	CJNE R2, #data, rel	CJNE R3, #data, rel	CJNE R4, #data, rel	CJNE R5, #data, rel	CJNE R6, #data, rel	CJNE R7, #data, rel
C 1100	XCH A, R0	XCH A, R1	XCH A, R2	XCH A, R3	XCH A, R4	XCH A, R5	XCH A, R6	XCH A, R7
D 1101	DJNZ R0 rel	DJNZ R1 rel	DJNZ R2, rel	DJNZ R3, rel	DJNZ R4, rel	DJNE R5, rel	DJNE R6, rel	DJNE R7, rel
E 1110	MOV A, R0	MOV A, R1	MOV A, R2	MOV A, R3	MOV A, R4	MOV A, R5	MOV A, R6	MOV A, R7
F 1111	MOV R0, A	MOV R1, A	MOV R2, A	MOV R3, A	MOV R4, A	MOV R5, A	MOV R6, A	MOV R7, A

INSTRUCTION SET DETAILS

Type	Mnemonic	Instruction Code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Bytes	Cycles	Description
Airthmetic operation instructions	ADD A, Rr	0 0 1 0 1 r_2 r_1 r_0	1	1	(AC), (0V), (C), (A) ← (A)+(Rr)
	ADD A, direct	0 0 1 0 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	1	(AC), (0V), (C), (A) ← (A)+(direct address)
	ADD A, @Rr	0 0 1 0 0 1 1 r_0	1	1	(AC), (0V), (C), (A) ← (A)+((Rr))
	ADD A, #data	0 0 1 0 0 1 0 0 I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0	2	1	(AC), (0V), (C), (A) ← (A)+#data
	ADDC A, Rr	0 0 1 1 1 r_2 r_1 r_0	1	1	(AC), (0V), (C), (A) ← (A)+(C)+(Rr)
	ADDC A, direct	0 0 1 1 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	1	(AC), (0V), (C), (A) ← (A)+(C)+(direct address)
	ADDC A, @Rr	0 0 1 1 0 1 1 r_0	1	1	(AC), (0V), (C), (A) ← (A)+(C)+((Rr))
	ADDC A, #data	0 0 1 1 0 1 0 0 I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0	2	1	(AC), (0V), (C), (A) ← (A)+(C)+#data
	SUBB A, Rr	1 0 0 1 1 r_2 r_1 r_0	1	1	(AC), (0V), (C), (A) ← (A)−((C))+((Rr))
	SUBB A, direct	1 0 0 1 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	1	(AC), (0V), (C), (A) ← (A)−((C)+(direct address))
	SUBB A, @Rr	1 0 0 1 0 1 1 r_0	1	1	(AC), (0V), (C), (A) ← (A)−((C)+((Rr))
	SUBB A, #data	1 0 0 1 0 1 0 0 I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0	2	1	(AC), (0V), (C), (A) ← (A)−((C)+#data)
	MUL AB	1 0 1 0 0 1 0 0	1	4	(AB) ← (A) x (B)
	DIV AB	1 0 0 0 0 1 0 0	1	4	(A) quotient, (B) remainder ← (A)/(B)
	DA A	1 1 0 1 0 1 0 0	1	1	When the contents of accumulator bits 0 thru 3 are greater than 9, or when auxiliary carry (AC) is 1, 6 is added to bits 0 thru 3. Bits 4 thru 7 are then examined, and when bits 4 thru 7 following compensation of lower bits 0 thru 3 is greater than 9, or when carry (C) is 1, 6 is added to bits 4 thru 7. As a result, the carry flag can be set, but cannot be cleared.
Accumulation operation instructions	CLR A	1 1 1 0 0 1 0 0	1	1	(A) ← 0
	CPL A	1 1 1 1 0 1 0 0	1	1	(A) ← $\overline{(A)}$
	RL A	0 0 1 0 0 0 1 1	1	1	
	RLC A	0 0 1 1 0 0 1 1	1	1	

INSTRUCTION SET DETAILS (CONT.)

Type	Mnemonic	Instruction Code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Bytes	Cycles	Description
Accumulator operation instructions	RR A	0 0 0 0 0 0 1 1	1	1	
	RRC A	0 0 0 1 0 0 1 1	1	1	
	SWAP A	1 1 0 0 0 1 0 0	1	1	$(A_4 \sim {}_7) \rightleftharpoons (A_0 \sim {}_3)$
Increment/decrement	INC A	0 0 0 0 0 1 0 0	1	1	(A) ← (A)+1
	INC Rr	0 0 0 0 1 r_2 r_1 r_0	1	1	(Rr) ← (Rr)+1
	INC direct	0 0 0 0 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	1	(direct address) ← (direct address)+1
	INC @Rr	0 0 0 0 0 1 1 r_0	1	1	((Rr)) ← ((Rr))+1
	INC DPTR	1 0 1 0 0 0 1 1	1	2	(DPTR) ← (DPTR)+1
	DEC A	0 0 0 1 0 1 0 0	1	1	(A) ← (A)−1
	DEC Rr	0 0 0 1 1 r_2 r_1 r_0	1	1	(Rr) ← (Rr)−1
	DEC direct	0 0 0 1 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	1	(direct address) ← (direct address)−1
	DEC @Rr	0 0 0 1 0 1 1 r_0	1	1	((Rr)) ← ((Rr))−1
Logical operation instructions	ANL A, Rr	0 1 0 1 1 r_2 r_1 r_0	1	1	(A) ← (A) AND (Rr)
	ANL A, direct	0 1 0 1 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	1	(A) ← (A) AND (direct address)
	ANL A, @Rr	0 1 0 1 0 1 1 r_0	1	1	(A) ← (A) AND ((Rr))
	ANL A, #data	0 1 0 1 0 1 0 0 I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0	2	1	(A) ← (A) AND #data
	ANL direct, A	0 1 0 1 0 0 1 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	1	(direct address) ← (direct address) AND (A)
	ANL direct, #data	0 1 0 1 0 0 1 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0 I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0	3	2	(direct address) ← (direct address) AND #data
	ORL A, Rr	0 1 0 0 1 r_2 r_1 r_0	1	1	(A) ← (A) OR (Rr)
	ORL A, direct	0 1 0 0 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	1	(A) ← (A) OR (direct address)
	ORL A, @Rr	0 1 0 0 0 1 1 r_0	1	1	(A) ← (A) OR ((Rr))
	ORL A, #data	0 1 0 0 0 1 0 0 I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0	2	1	(A) ← (A) OR #data

INSTRUCTION SET DETAILS (CONT.)

Type	Mnemonic	Instruction Code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Bytes	Cycles	Description
Logical operation instructions	ORL direct, A	0 1 0 0 0 0 1 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	1	(direct address) ← (direct address) OR (A)
	ORL direct, #data	0 1 0 0 0 0 1 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0 I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0	3	2	(direct address) ← (direct address) OR #data
	XRL A, Rr	0 1 1 0 1 r_2 r_1 r_0	1	1	(A) ← (A) XOR (Rr)
	XRL A, direct	0 1 1 0 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	1	(A) ← (A) XOR (direct address)
	XRL A, @Rr	0 1 1 0 0 1 1 r_0	1	1	(A) ← (A) XOR ((Rr))
	XRL A, #data	0 1 1 0 0 1 0 0 I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0	2	1	(A) ← (A) XOR #data
	XRL direct, A	0 1 1 0 0 0 1 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	1	(direct address) ← (direct address) XOR (A)
	XRL direct, #data	0 1 1 0 0 0 1 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0 I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0	3	2	(direct address) ← (direct address) XOR #data
Immediate data setting instructions	MOV A, #data	0 1 1 1 0 1 0 0 I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0	2	1	(A) ← #data
	MOV Rr, #data	0 1 1 1 1 r_2 r_1 r_0 I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0	2	1	(Rr) ← #data
	MOV direct, #data	0 1 1 1 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0 I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0	3	2	(direct address) ← #data
	MOV @Rr, #data	0 1 1 1 0 1 1 r_0 I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0	2	1	(Rr)) ← #data
	MOV DPTR, #data 16	1 0 0 1 0 0 0 0 I_{15} I_{14} I_{13} I_{12} I_{11} I_{10} I_9 I_8 I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0	3	2	(DPTR) ← #data 16
Carry flag operation instructions	CLR C	1 1 0 0 0 0 1 1	1	1	(C) ← 0
	SETB C	1 1 0 1 0 0 1 1	1	1	(C) ← 1
	CPL C	1 0 1 1 0 0 1 1	1	1	(C) ← $\overline{(C)}$
	ANL C, bit	1 0 0 0 0 0 1 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	2	2	(C) ← (C) AND (bit address)
	ANL C,/bit	1 0 1 1 0 0 0 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	2	2	(C) ← (C) AND $\overline{\text{(bit address)}}$
	ORL C, bit	0 1 1 1 0 0 1 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	2	2	(C) ← (C) OR (bit address)
	ORL C,/bit	1 0 1 0 0 0 0 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	2	2	(C) ← (C) OR $\overline{\text{(bit address)}}$
	MOV C, bit	1 0 1 0 0 0 1 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	2	1	(C) ← (bit address)

6

INSTRUCTION SET DETAILS (CONT.)

Type	Mnemonic	Instruction Code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Bytes	Cycles	Description
	MOV bit, C	1 0 0 1 0 0 1 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	2	2	(bit address) ← (C)
Bit operation instructions	SETB bit	1 1 0 1 0 0 1 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	2	1	(bit address) ← 1
	CLR bit	1 1 0 0 0 0 1 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	2	1	(bit address) ← 0
	CPL bit	1 0 1 1 0 0 1 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0	2	1	(bit address) ← $\overline{\text{(bit address)}}$
Data transfer instructions	MOV A, Rr	1 1 1 0 1 r_2 r_1 r_0	1	1	(A) ← (Rr)
	MOV A, direct	1 1 1 0 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	1	(A) ← (direct address)
	MOV A, @Rr	1 1 1 0 0 1 1 r_0	1	1	(A) ← ((Rr))
	MOV Rr, A	1 1 1 1 1 r_2 r_1 r_0	1	1	(Rr) ← (A)
	MOV Rr, direct	1 0 1 0 1 r_2 r_1 r_0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2	(Rr) ← (direct address)
	MOV direct, A	1 1 1 1 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	1	(direct address) ← (A)
	MOV direct, Rr	1 0 0 0 1 r_2 r_1 r_0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2	(direct address) ← (Rr)
	MOV direct, @Rr	1 0 0 0 0 1 1 r_0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2	(direct address) ← ((Rr))
	MOV @Rr, A	1 1 1 1 0 1 1 r_0	1	1	((Rr)) ← (A)
	MOV @Rr, direct	1 0 1 0 0 1 1 r_0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2	((Rr)) ← (direct address))
Constant code instructions	MOVC A, @A+DPTR	1 0 0 1 0 0 1 1	1	2	(A) ← ((A)+(DPTR))
	MOVC A,@A+PC	1 0 0 0 0 0 1 1	1	2	(PC) ← (PC) + 1 (A) ← ((A) + (PC))
Data exchange instructions	XCH A, Rr	1 1 0 0 1 r_2 r_1 r_0	1	1	(A) ⇌ (Rr)
	XCH A, direct	1 1 0 0 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	1	(A) ⇌ (direct address)
	XCH A, @Rr	1 1 0 0 0 1 1 r_0	1	1	(A) ⇌ ((Rr))
	XCHD A, @Rr	1 1 0 1 0 1 1 r_0	1	1	($A_0 \sim_3$) ⇌ (($Rr_0 \sim_3$))

INSTRUCTION SET DETAILS (CONT.)

Type	Mnemonic	Instruction Code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Bytes	Cycles	Description
Subroutine instructions	PUSH direct	1 1 0 0 0 0 0 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2	(SP) ← (SP)+1 ((SP)) ← (direct address)
	POP direct	1 1 0 1 0 0 0 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2	(direct address) ← ((SP)) (SP) ← (SP)−1
	ACALL addr 11	A_{10} A_9 A_8 1 0 0 0 1 A_7 A_6 A_5 A_4 A_3 A_2 A_1 A_0	2	2	(PC) ← (PC)+2 (SP) ← (SP)+1 ((SP)) ← ($PC_0 \sim_7$) (SP) ← (SP)+1 ((SP)) ← ($PC_8 \sim_{15}$) ($PC_0 \sim_{10}$) ← $A_0 \sim_{10}$
	LCALL addr 16	0 0 0 1 0 0 1 0 A_{15} A_{14} A_{13} A_{12} A_{11} A_{10} A_9 A_8 A_7 A_6 A_5 A_4 A_3 A_2 A_1 A_0	3	2	(PC) ← (PC)+3 (SP) ← (SP)+1 ((SP)) ← ($PC_0 \sim_7$) (SP) ← (SP)+1 ((SP)) ← ($PC_8 \sim_{15}$) ($PC_0 \sim_{15}$) ← $A_0 \sim_{15}$
	RET	0 0 1 0 0 0 1 0	1	2	($PC_8 \sim_{15}$) ← ((SP)) (SP) ← (SP)−1 ($PC_0 \sim_7$) ← ((SP)) (SP) ← (SP)−1
	RETI	0 0 1 1 0 0 1 0	1	2	($PC_8 \sim_{15}$) ← ((SP)) (SP) ← (SP)−1 ($PC_0 \sim_7$) ← ((SP)) (SP) ← (SP)−1
Jump instructions	AJMP addr 11	A_{10} A_9 A_8 0 0 0 0 1 A_7 A_6 A_5 A_4 A_3 A_2 A_1 A_0	2	2	(PC) ← (PC)+2 ($PC_0 \sim_{10}$) ← $A_0 \sim_{10}$
	LJMP addr 16	0 0 0 0 0 0 1 0 A_{15} A_{14} A_{13} A_{12} A_{11} A_{10} A_9 A_8 A_7 A_6 A_5 A_4 A_3 A_2 A_1 A_0	3	2	($PC_0 \sim_{15}$) ← $A_0 \sim_{15}$
	SJMP rel	1 0 0 0 0 0 0 0 R_7 R_6 R_5 R_4 R_3 R_2 R_1 R_0	2	2	(PC) ← (PC)+2 (PC) ← (PC)+relative offset
	JMP @A+DPTR	0 1 1 1 0 0 1 1	1	2	(PC) ← (A)+(DPTR)

Branch instructions	CJNE A, direct, rel	1 0 1 1 0 1 0 1
		a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0
		R_7 R_6 R_5 R_4 R_3 R_2 R_1 R_0
			3	2	(PC) ← (PC)+3
					IF (A) ≠ (direct address)
					THEN
					(PC) ← (PC)+relative offset
					IF (A) < (direct address)
					THEN
					(C) ← 1
					ELSE
					(C) ← 0

	CJNE A, #data, rel	1 0 1 1 0 1 0 0
		I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0
		R_7 R_6 R_5 R_4 R_3 R_2 R_1 R_0
			3	2	(PC) ← (PC)+3
					IF (A) ≠ #data
					THEN
					(PC) ← (PC)+relative offset
					IF (A) < #data
					THEN
					(C) ← 1
					ELSE
					(C) ← 0

6

INSTRUCTION SET DETAILS (CONT.)

Type	Mnemonic	Instruction Code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Bytes	Cycle	Description

Branch instructions	CJNE Rr, #data, rel	1 0 1 1 1 r_2 r_1 r_0
		I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0
		R_7 R_6 R_5 R_4 R_3 R_2 R_1 R_0
			3	2	(PC) ← (PC)+3
					IF ((Rr)) ≠ #data
					THEN
					(PC) ← (PC)+relative offset
					IF ((Rr)) < #data
					THEN
					(C) ← 1
					ELSE
					(C) ← 0

	CJNE @Rr, #data, rel	1 0 1 1 0 1 1 r_0
		I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0
		R_7 R_6 R_5 R_4 R_3 R_2 R_1 R_0
			3	2	(PC) ← (PC)+3
					IF ((Rr)) ≠ #data
					THEN
					(PC) ← (PC)+relative offset
					IF ((Rr)) < #data
					THEN
					(C) ← 1
					ELSE
					(C) ← 0

	DJNZ Rr, rel	1 1 0 1 1 r_2 r_1 r_0 R_7 R_6 R_5 R_4 R_3 R_2 R_1 R_0	2	2	(PC) ← (PC)+2 (Rr) ← (Rr)−1 IF (Rr) ≠ 0 THEN (PC) ← (PC)+relative offset
	DJNZ direct, rel	1 1 0 1 0 1 0 1 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0 R_7 R_6 R_5 R_4 R_3 R_2 R_1 R_0	3	2	(PC) ← (PC)+3 (direct address) ← (direct address)−1 IF (direct address) ≠ 0 THEN (PC) ← (PC)+relative offset
	JZ rel	0 1 1 0 0 0 0 0 R_7 R_6 R_5 R_4 R_3 R_2 R_1 R_0	2	2	(PC) ← (PC)+2 IF (A) = 0 THEN (PC) ← (PC)+relative offset
	JNZ rel	0 1 1 1 0 0 0 0 R_7 R_6 R_5 R_4 R_3 R_2 R_1 R_0	2	2	(PC) ← (PC)+2 IF (A) ≠ 0 THEN (PC) ← (PC)+relative offset
	JC rel	0 1 0 0 0 0 0 0 R_7 R_6 R_5 R_4 R_3 R_2 R_1 R_0	2	2	(PC) ← (PC)+2 IF (C) = 1 THEN (PC) ← (PC)+relative offset
	JNC rel	0 1 0 1 0 0 0 0 R_7 R_6 R_5 R_4 R_3 R_2 R_1 R_0	2	2	(PC) ← (PC)+2 IF (C) = 0 THEN (PC) ← (PC)+relative offset
	JB bit, rel	0 0 1 0 0 0 0 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0 R_7 R_6 R_5 R_4 R_3 R_2 R_1 R_0	3	2	(PC) ← (PC)+3 IF (bit address) = 1 THEN (PC) ← (PC)+relative offset
	JNB bit, rel	0 0 1 1 0 0 0 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0 R_7 R_6 R_5 R_4 R_3 R_2 R_1 R_0	3	2	(PC) ← (PC)+3 IF (bit address) = 0 THEN (PC) ← (PC)+relative offset

INSTRUCTION SET DETAILS (CONT.)

Type	Mnemonic	Instruction Code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Bytes	Cycle	Description
Branch instructions	JBC bit, rel	0 0 0 1 0 0 0 0 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0 R_7 R_6 R_5 R_4 R_3 R_2 R_1 R_0	3	2	(PC) ← (PC)+3 IF (bit address) = 1 THEN (bit address) ← 0 (PC) ← (PC)+relative offset
External memory instructions	MOVX A, @Rr	1 1 1 0 0 0 1 r_0	1	2	(A) ← ((Rr)) EXTERNAL RAM
	MOVX A, @DPTR	1 1 1 0 0 0 0 0	1	2	(A) ← ((DPTR)) EXTERNAL RAM
	MOVX @Rr, A	1 1 1 1 0 0 1 r_0	1	2	(Rr) ← (A) EXTERNAL RAM
	MOVX @DPTR, A	1 1 1 1 0 0 0 0	1	2	((DPTP)) ← (A) EXTERNAL RAM
Other instructions	NOP	0 0 0 0 0 0 0 0	1	1	(PC) ← (PC)+1

6

ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings

Parameter	Symbol	Conditions	Rating	Unit
Supply voltage	Vcc	Ta = 25 °C	−0.5 ~ 7	V
Input voltage	V_I	Ta = 25 °C	−0.5 ~ Vcc + 0.5	V
Storage temperature	Tstg		−55 ~ + 150	°C

Operational Range

- MSM80C154/83C154 ...DC to 12 MHz, VCC = ±20%
- MSM80C154-1/83C154-1 ...DC to 16 MHz, VCC = ±10%

Parameter	Symbol	Conditions	Rating	Unit
Supply voltage	Vcc	*1 fosc = DC-16 MHz	2.5 ~ 6	V
Memory hold voltage	Vcc		2 ~ 6	V
Ambient temperature	Ta		−40 ~ + 85	°C

*1: 2.5 V ≤ Vcc < 4 V DC characteristics will be specified elsewhere.
16 MHz version of MSM83C154 (12 MHz < XTAL 1·2 ≤ 16 MHz) is being developed.

DC Characteristics

(V_{CC} = 5V±10%, Ta = −40 to +85°C)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit	Measuring circuit
Input Low Voltage	V_{IL}		−0.5		0.2 V_{CC}−0.1	V	1
Input High Voltage	V_{IH}	Except XTAL1 and RESET	0.2 V_{CC} +0.9		V_{CC} + 0.5	V	
Input High Voltage	V_{IHI}	XTAL1 and RESET	0.7 V_{CC}		V_{CC} + 0.5	V	
Output Low Voltage (PORT 1, 2, 3)	V_{OL}	I_{OL} = 1.6 mA			0.45	V	
Output Low Voltage (PORT 0, ALE, PSEN)	V_{OL1}	I_{OL} = 3.2 mA			0.45	V	
Output High Voltage (PORT 1, 2, 3)	V_{OH}	I_{OH} = −60 μA V_{CC} = 5 V ± 10%	2.4			V	
		I_{OH} = −30 μA	0.75 V_{CC}			V	
		I_{OH} = −10 μA	0.9 V_{CC}			V	
Output High Voltage (PORT 0, ALE, PSEN)	V_{OHI}	I_{OH} = −400 μA V_{CC} = 5 V ± 10%	2.4			V	
		I_{OH} = −150 μA	0.75 V_{CC}			V	
		I_{OH} = −40 μA	0.9 V_{CC}			V	
Logical 0 Input Current (PORT 1, 2, 3)	I_{IL}	V_I = 0.45 V	−10		−200	μA	2
Logical 1 to 0 Transition Current (PORT 1, 2, 3)	I_{TL}	V_I = 2.0 V			−500	μA	
Input Leakage Current (PORT 0 floating, EA)	I_{LI}	$V_{SS} < V_I < V_{CC}$			± 10	μA	3
RESET Pulldown Resistor	R_{RST}		20	40	125	KΩ	2
Pin Capacitance	C_{IO}	T_A = 25°C, f = 1 MHz 5 V (except XTAL1)			10	pF	
Power Down Current	I_{PD}	V_{CC} = 2 ~ 6 V		1	50	μA	4

Maximum Power Supply Current
Normal Operation I_{CC} (mA)

V_{CC}	4 V	5 V	6 V
Freq			
0.5 MHz	1.6	2.2	3
3.5 MHz	4.3	5.7	7.5
8 MHz	8.3	11	14
12 MHz	12	16	20

Maximum Power Supply Current
Idle Mode I_{CC} (mA)

V_{CC}	4 V	5 V	6 V
Freq			
0.5 MHz	0.6	0.9	1.2
3.5 MHz	1.1	1.6	2.2
8 MHz	1.8	2.7	3.7
12 MHz	2.5	3.7	5

*1: **2.5 V** $\leqslant V_{CC} < 4$ V DC characteristics will be specified elsewhere.

Maximum Power Supply Current
Normal Operation I_{CC} (mA)

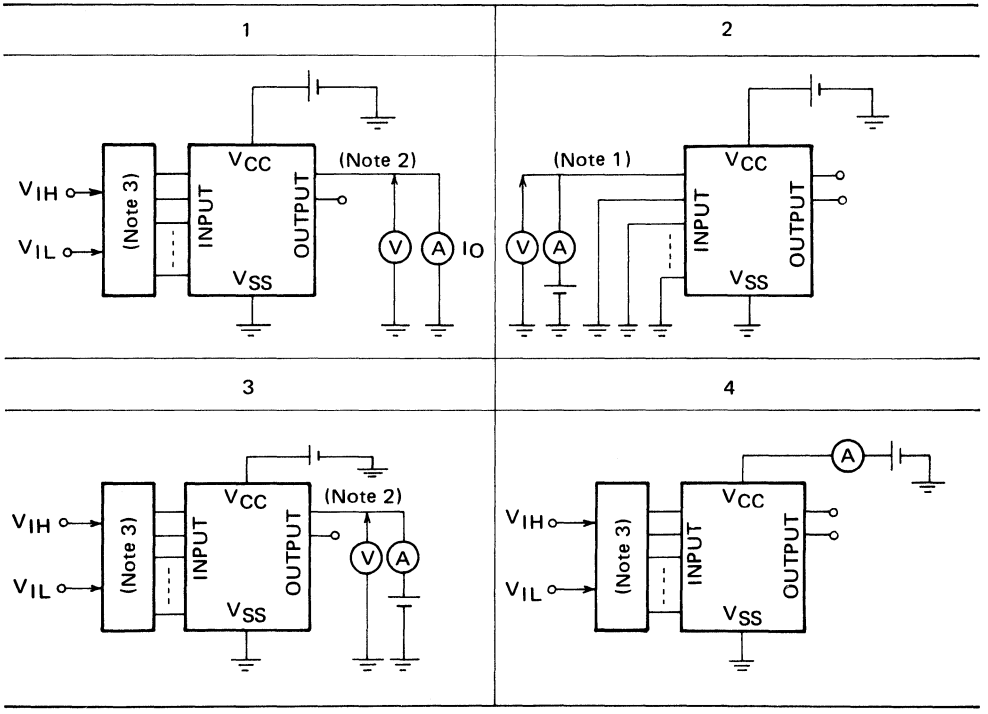
V_{CC}	4.5 V	5 V	5.5 V
Freq.			
1.2 MHz	2.0	2.3	2.6
8 MHz	10	11	12.5
12 MHz	14	16	18
16 MHz*	18	20	23

Maximum Power Supply Current
Idle Mode I_{CC} (mA)

V_{CC}	4.5 V	5 V	5.5 V
Freq.			
1.2 MHz	1.4	1.5	1.6
8 MHz	2.3	2.7	3.2
12 MHz	3.0	3.7	5.0
16 MHz*	4.0	5.0	6.0

*2: MSM80C154-1/MSM83C154-1

Measuring Circuits



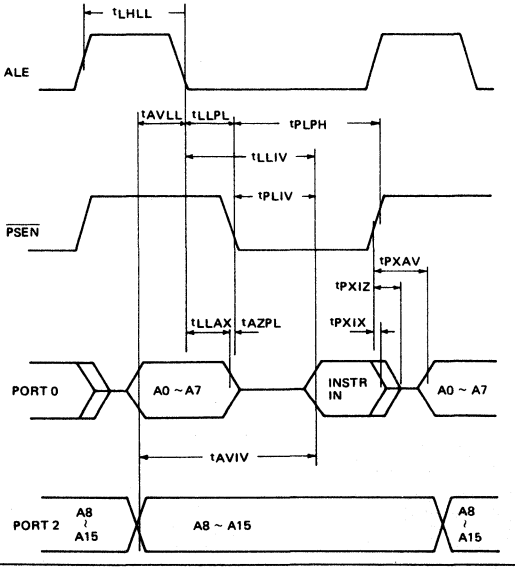
Note 1. Repeated for specified input pins.
2. Repeated for specified output pins.
3. Input logic for specified status.

External Program Memory Access AC Characteristics

(V_{CC} = 5 V ± 20%, V_{SS} = 0 V, XTAL1·2 = 12 MHz, Ta = −40 °C to 85 °C
V_{CC} = 5 V ± 10%, V_{SS} = 0 V, 12 MHz < XTAL1·2 ≤ 16 MHz, Ta = −40 °C to 85 °C
PORT 0, ALE, and $\overline{PSEN}$ connected with 100 pF load, other connected with 80 pF load)

Parameter	Symbol	Ratings				Unit
		16 MHz clock		Variable clock from DC to 16 MHz		
		Min.	Max.	Min.	Max.	
XTAL1·2 Oscillator Period	tCLCL	62.5		62.5		ns
ALE Pulse Width	tLHLL	85		2tCLCL-40		ns
Address Valid to ALE Low	tAVLL	18.5		1tCLCL-44		ns
Address Hold After ALE Low	tLLAX	27.5		1tCLCL-35		ns
ALE Low to Valid Instr In	tLLIV		150		4tCLCL-100	ns
ALE Low to $\overline{PSEN}$ Low	tLLPL	32.5		1tCLCL-30		ns
$\overline{PSEN}$ Pulse Width	tPLPH	152.5		3tCLCL-35		ns
$\overline{PSEN}$ Low to Valid Instr In	tPLIV		82.5		3tCLCL-105	ns
Input Instr Hold After $\overline{PSEN}$	tPXIX	0		0		ns
Input Instr Float After $\overline{PSEN}$	tPXIZ		42.5		1tCLCL-20	ns
$\overline{PSEN}$ to Address Valid	tPXAV	42.5		1tCLCL-20		ns
Address to Valid Instr In	tAVIV		207.5		5tCLCL-105	ns
Address Float to $\overline{PSEN}$ Low	tAZPL	0		0		ns

External program memory read cycle

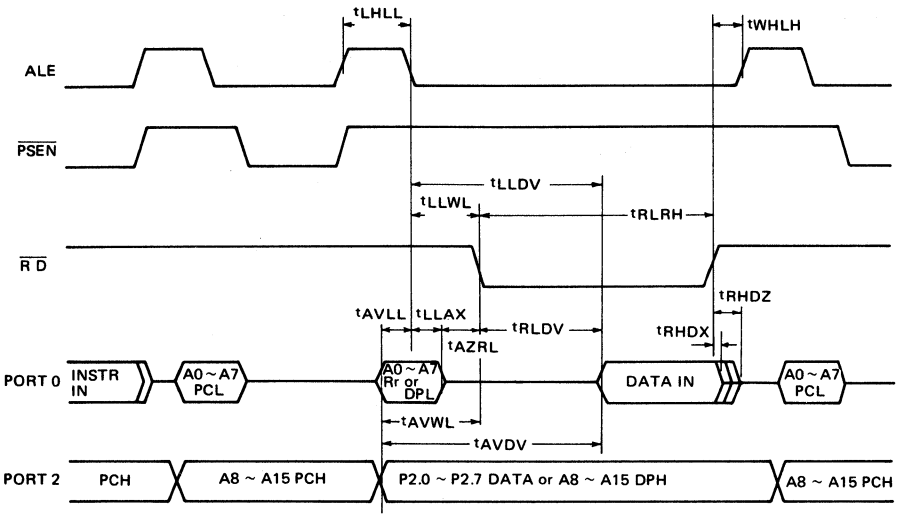


External Program Memory Access AC Characteristics

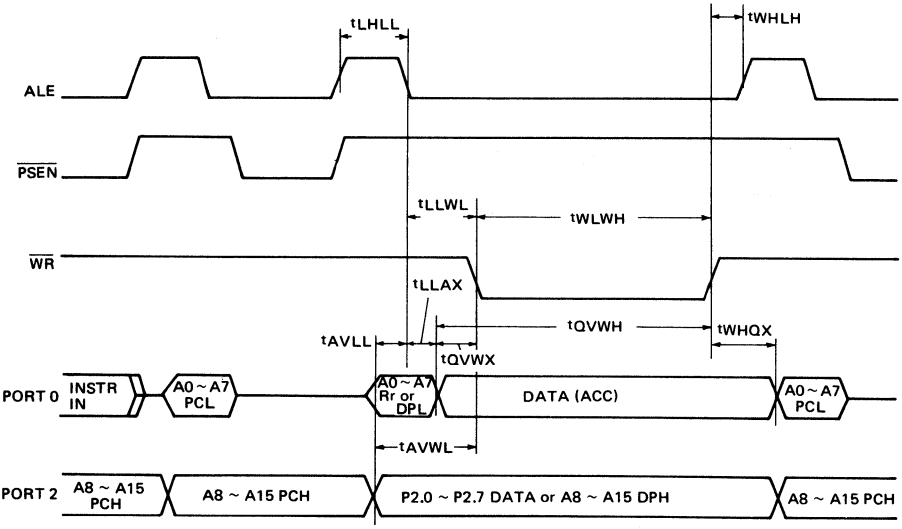
(V_{CC} = 5 V ± 20%, V_{SS} = 0 V, XTAL1·2 = 12 MHz, Ta = −40 °C to 85 °C
V_{CC} = 5 V ± 10%, V_{SS} = 0 V, 12 MHz < XTAL1·2 ≤ 16 MHz, Ta = −40 °C to 85 °C
PORT 0, ALE, and $\overline{PSEN}$ connected with 100 pF load, other connected with 80 pF load)

Parameter	Symbol	Ratings				Unit
		16 MHz clock		Variable clock from DC to 16 MHz		
		Min.	Max.	Min.	Max.	
XTAL1·2 Oscillator Period	tCLCL	62.5		62.5		ns
ALE Pulse Width	tLHLL	85		2tCLCL-40		ns
Address Valid to ALE Low	tAVLL	18.5		1tCLCL-44		ns
Address Hold After ALE Low	tLLAX	27.5		1tCLCL-35		ns
$\overline{RD}$ Pulse Width	tRLRH	275		6tCLCL-100		
$\overline{WR}$ Pulse Width	tWLWH	275		6tCLCL-100		ns
$\overline{RD}$ Low to Valid Data In	tRLDV		207.5		5tCLCL-105	ns
Data Hold After $\overline{RD}$	tRHDX	0		0		ns
Data Float After $\overline{RD}$	tRHDZ		55		2tCLCL-70	ns
ALE Low to Valid Data In	tLLDV		400		8tCLCL-100	ns
Address to Valid Data In	tAVDV		457.5		9tCLCL-105	ns
ALE Low to $\overline{RD}$ or $\overline{WR}$ Low	tLLWL	147.5	227.5	3tCLCL-40	3tCLCL+40	ns
Address to $\overline{RD}$ or $\overline{WR}$ Low	tAVWL	180		4tCLCL-70		ns
Data Valid to $\overline{WR}$ Transition	tQVWX	22.5		1tCLCL-40		ns
Data Valid to $\overline{WR}$ High	tQVWH	332.5		7tCLCL-105		ns
Data Hold After $\overline{WR}$	tWHQX	75		2tCLCL-50		ns
Address Float to $\overline{RD}$ Low	tAZRL		0		0	ns
$\overline{RD}$ or $\overline{WR}$ High to ALE High	tWHLH	32.5	102.5	1tCLCL-30	1tCLCL+40	ns

External data memory read cycle



External data memory write cycle

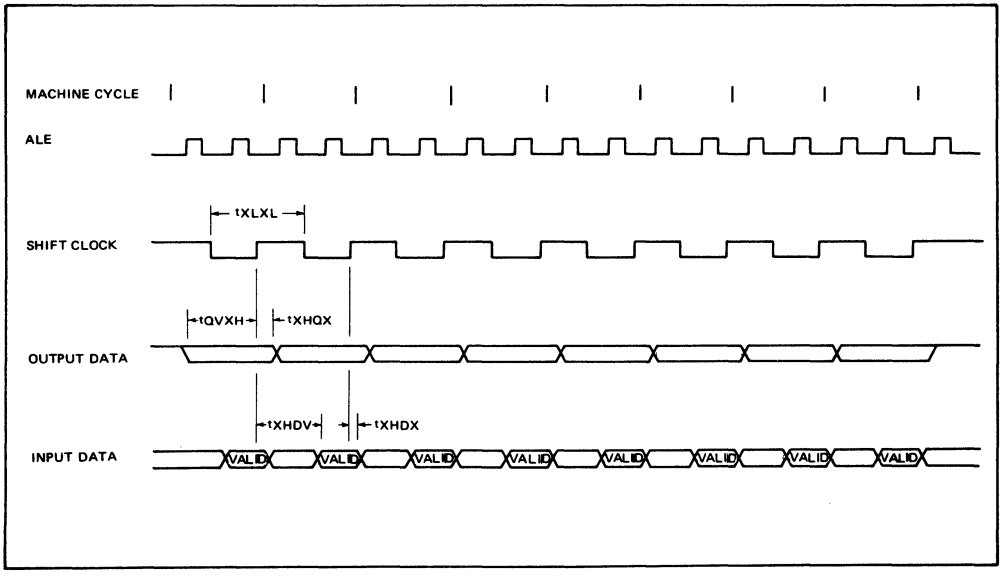


6

Serial Port (I/O Extension Mode) AC Characteristics

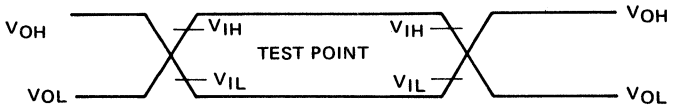
V_{CC} = 5 V ±20%, V_{SS} = 0 V, XTAL1·2 = 12 MHz, Ta = –40 °C to 85 °C
V_{CC} = 5 V ±10%, V_{SS} = 0 V, 12 MHz < XTAL1·2 ≤ 16 MHz, Ta = –40 °C to 85 °C

Parameter	Symbol	Min.	Max.	Unit
Serial Port Clock Cycle Time	tXLXL	12tCLCL		ns
Output Data Setup to Clock Rising Edge	tQVXH	10tCLCL-133		ns
Output Data Hold After Clock Rising Edge	tXHQX	2tCLCL-75		ns
Input Data Hold After Clock Rising Edge	tXHDX	0		ns
Clock Rising Edge to Input Data Valid	tXHDV		10tCLCL-133	ns



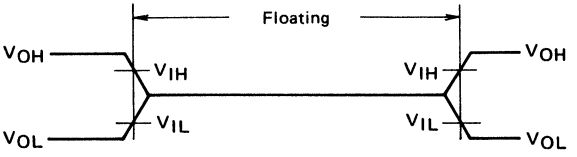
AC Characteristics Measuring Conditions

1. Input/output signal



* The input signals in AC test mode are either V_{OH} (logic "1") or V_{OL} (logic "0") input signals where logic "1" corresponds to a CPU output signal waveform measuring point in excess of V_{IH}, and logic "0" to a point below V_{IL}.

2. Floating



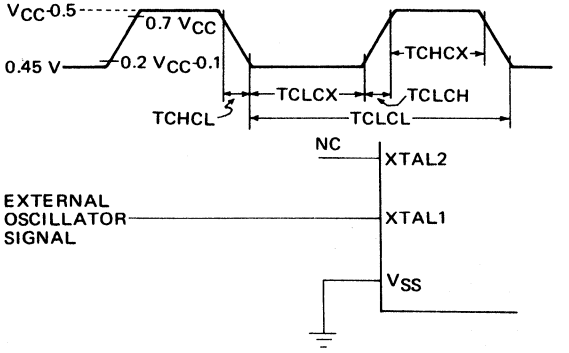
* The port 0 floating interval is measured from the time the port 0 pin voltage drops below V_{IH} after sinking to GND at 2.4 mA when switching to floating status from a "1" output, and from the time the port 0 pin voltage exceeds V_{IL} after connecting to a 400 μA source when switching to floating status from a "0" output.

6

XTAL1 External Clock Input Waveform Conditions

Parameter	Symbol	Min.	Max.	Units
Oscillator Freq.	1/tCLCL	DC	16	MHz
High Time	tCHCX	20		ns
Low Time	tCLCX	20		ns
Rise Time	tCLCH		20	ns
Fall Time	tCHCL		20	ns

EXTERNAL CLOCK DRIVE WAVEFORM

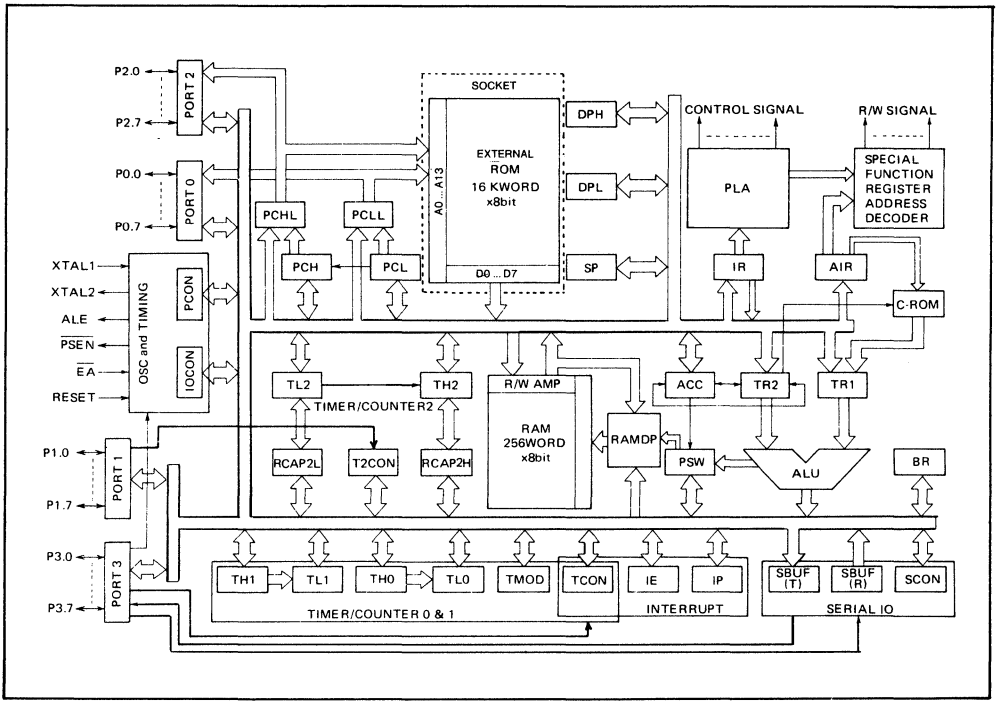


MSM85C154VS

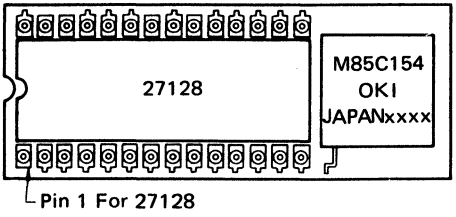
M83C154/M80C51F PIGGY BACK

GENERAL DESCRIPTION

The MSM85C154 is a device whose built-in ROM is replaced by external EPROM using the piggy-back method. External EPROM capacity is up to 16K bytes. It can be used for evaluation of programs for MSM83C154 and MSM80C51F.



INSTALLATION METHOD FOR EXTERNAL ROM



* The MSM85C154VS pin layout of bottom side is the same as the pin layout for MSM83C154-XXRS.

*NOTE

MSM85C154VS piggy back is originally designed for the programming of MSM83C154 and it covers the function as the piggy back for MSM80C51F.
Please be careful not to use additional function which dedicated to MSM83C154 in usig the piggy back for MSM80C51. The function, flag, and resistor listed below are dedicated to MSM83C154.

- ICON (0F8H) : I/O CONTROL RESISTOR
- TH2 (0CDH) : TIMER 2. UPPER SIDE RESISTOR
- TL2 (0CCH) : TIMER 2. LOWER SIDE RESISTOR
- RCAP2H (0CBH) : CAPTURE RESISTOR. UPPER SIDE
- RCAP2L (0CAH) : CAPTURE RESISTOR. LOWER SIDE
- T2CON (0C8H) : TIMER CONTROL RESISTOR 2
- IP (0B8H) : INTERRUPT PRIORITY RESISTOR 2
  bit 5 (Bit address BDH) PT2
  bit 7 (Bit address BFH) PCT
- IE (0A8H) : INTERRUPT ENABLE RESISTOR
  bit 5 (Bit address ADH) ET2
- PCON (087H) : POWER CONTROL RESISTOR
  bit 5 (Bit address Nil) RPD
  bit 6 (Bit address Nil) HPD

In using this piggy back for MSM80C51F, do not set the above items (Control bit should not be "1").
All bits are set to "0" at initial reset.

6

In high temperature atmosphere, malfunction may happen (output latches of the Port 0 are set when interrupt occurs) in writing the instruction code of which LSB is "0" at address 0 of the EPROM.
To avoid this problem, please be sure to write AJMP instruction (operational code is X1) at address 0 instead of LJMP instruction (operational code is 02).

Operating frequency is from DC to 12MHz.

The MSM85C154VS has been developed assuming that it is used for evaluation of program. Please use the MSM83C154 (mask ROM version) as the devices installed on a product.

PROGRAM DEVELOPMENT SUPPORT SYSTEMS

7

EASE40 PROGRAM DEVELOPMENT SUPPORT SYSTEM

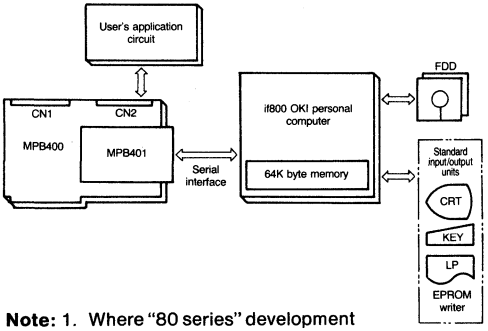
The OKI EASE40 Program Development Support System designed for use with the CMOS 4-bit, 1-chip microcontroller OLMS40 Series consists of the MPB400 (evaluation board) and the MPB400DS (key & display board), plus a number of EASE40 microcontroller interface boards. These components can also be connected by MULTIBUS interface or serial I/O interface to the "80 series" of development tools or serial I/O terminal units for even more efficient program development.

- Since the MPB400 includes a MULTIBUS interface, it can be connected online to "80 series" development systems equipped with an existing MULTIBUS interface to enable direct use of the system console and I/O units.
- With an MPB401 connected to the MPB400, online connections are also possible to serial interface systems (such as the if800 personal computer).
- A floppy disk based assembler/debugger is available for use in online type systems. (CP/M®, ISIS-II base)

Note: (CP/M® is the registered trademark of Digital Research Inc. (USA).
ISIS-II and MULTIBUS are registered trademarks of Intel Corp (USA).

SYSTEM CONFIGURATION

[1] SERIAL INTERFACE TYPES



Note: 1. Where "80 series" development systems other than the if800 are used, the system must be capable of using CP/M.
2. The MPB401 is connected directly to the MPB400 via a socket.

FEATURES

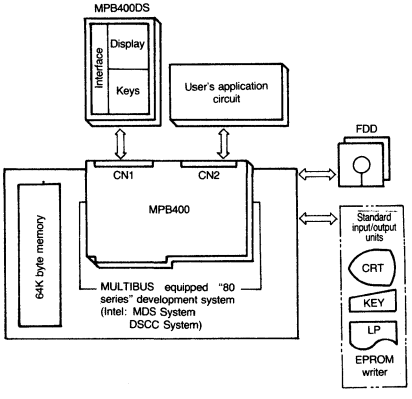
The serial interface type employs serial interface equipped development tools available on the market or personal computers, together with the MPB400 board and MPB401.

HOST COMPUTER REQUIREMENTS

1. CP/M-80 version 2.0 or 2.2 must be operable as the operating system.
2. The host computer memory capacity must be large enough to operate at least 52K bytes of CP/M.
3. Interface must be RS232C, TTL, or 20 mA current loop.
4. The RS232C serial port must be accessible when the CP/M system calls function No. 6 (direct console I/O), on condition that a logical device CON: is assigned to a physical device TTY.

Oki personal computers if800 model 20 and model 30 are standard host computers which satisfy these requirements.

[2] BUS INTERFACE TYPES



Note: For Bus interface types, an online/offline switch located on MPB400DS is set to "online" position.

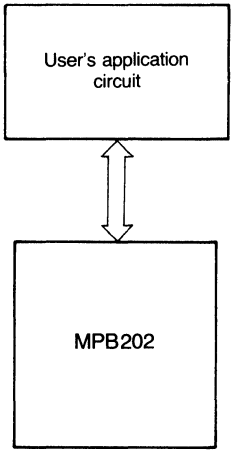
FEATURES

Bus interface types employ MULTIBUS equipped development tools available on the market, e.g., the MPB400, and MPB400DS. Use of an FDD unit (floppy disk drive) which the development system has as standard equipment, enables disk base editing, assembling, and debugging. When debugging especially, a disk based "debugger" enables loading, excuting and correcting of users' application programs from a development system console.

HOST COMPUTER REQUIREMENTS

1. Intel 8080A microprocessor or equivalent processor is used as the CPU.
2. Intel MULTIBUS or equivalent bus system is used.
3. ISIS-II or CP/M-80 msut be operable as the operating system.
4. 00 is not used as the 8080A (or equivalent) microprocessor I/O address.

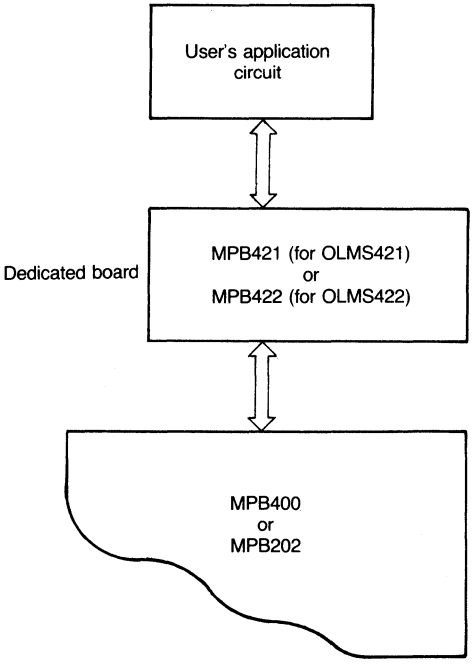
[3] FIELD DEBUGGING TYPES



FEATURES

Field debugging types are used for final field evaluation of application circuits and programs after program debugging with serial interface or bus interface types has been completed. The small and compact MPB202 board used for this purpose is equipped with a socket for program EPROMs.

[4] OLMS40 SERIES EVALUATION



FEATURES

The dedicated boards indicated in the diagram are required for a program evaluation of the OLMS421 or OLMS422 microcontroller.

7

ASSEMBLER

ASM40 is a floppy disk based high-performance assembler which operates under CP/M and ISIS-II. This assembler is used to translate source files generated on disk by using an editor (available on the market), thereby generating object files (Intel HEX format), assembly list files and symbol files in the specified devices.

FEATURES

1. Free descriptive format source files
2. Enables description of up to nine types arithmetic expressions in the source program operand field
3. Symbol file generation capacity
4. Six types of pseudo-instructions available
5. A type setting pseudo-instruction enables checking the limitation of basic instruction sets for each OLMS40 microcontroller.

LIST OF PSEUDO-INSTRUCTIONS

Pseudo-instruction	Function
EQU	Assignment of operand value to name.
ORG	Setting of program start address
END	Indication of end of program
NSE	Setting of 0 in the 4 lower order bits of the assembler location counter, and addition of 16. The NOP instruction is assigned to blank areas where no machine language instruction has been assigned.
TYP	Indication of the OLMS40 series type

DEBUGGER

The EASE40 debugger can be used for effective debugging in both serial interface and bus interface type systems.

FEATURES

1. Simple input format for debugging commands entered from the console of the connected development tool (personal computer)
2. Efficient loading/saving and display/change of object codes
3. Real-time execution and step execution
4. Display/change of microcontroller contents
5. Use of assembler symbols permitted (only in bus interface types)

EXAMPLE OF USE OF COMMANDS

```
LOAD△H△B:TEST.HEX
LIST△:F1:LIST1.001
DEFINE△.START=1AC
REMOVE△.JOB1,.END2
URAM△0△TO△1FF
PC=789
URAM△100=48△49△12
GO△URAM△FROM△123△TILL△PC=7A4△&10
GO△TILL△PF=4
STEP△FROM△500△COUNT△30
VERIFY△O△A:SAMPL2.OBJ△UROM△400△TO△7FF
SAVE△:F1:JOB1.002△0△TO△7FF
```

Command name	Command input format
LOAD	LOAD [△ H/O] △ Pathname [△ EXT]
LIST	LIST △ Pathname/OFF
DEFINE	DEFINE △ Symbol = expr
REMOVE	REMOVE △ Symbol [, Symbol, ...]/SYMBOLS
Display	Keyword [△ address]
Change	Keyword = data·ref Keyword △ addr = data [, data,data....,data]
ESET	RESET
EXIT	EXIT
GO	GO [△ URAM/UROM] [△ FROM △ addr·exp] [△ TILL △ PC = /</> addr·exp [△ & n]] [△ ASM] GO [△ URAM/UROM] [△ FROM △addr·exp] [△ TILL △ Port·exp = data·exp] [△ ASM]
STEP	STEP [△ URAM/UROM][△FROM△addr·exp] [△ COUNT △ n] [△ ASM]
@	@
VERIFY	VERIFY [△ H/O] △ Pathname [△ EXT] [△ URAM/UROM] [address]
SAVE	SAVE [△ H/O] △ Pathname [△ URAM/UROM] [address]

△ ..... Space, [...]..... May be omitted, /..... Slash denoting "or".

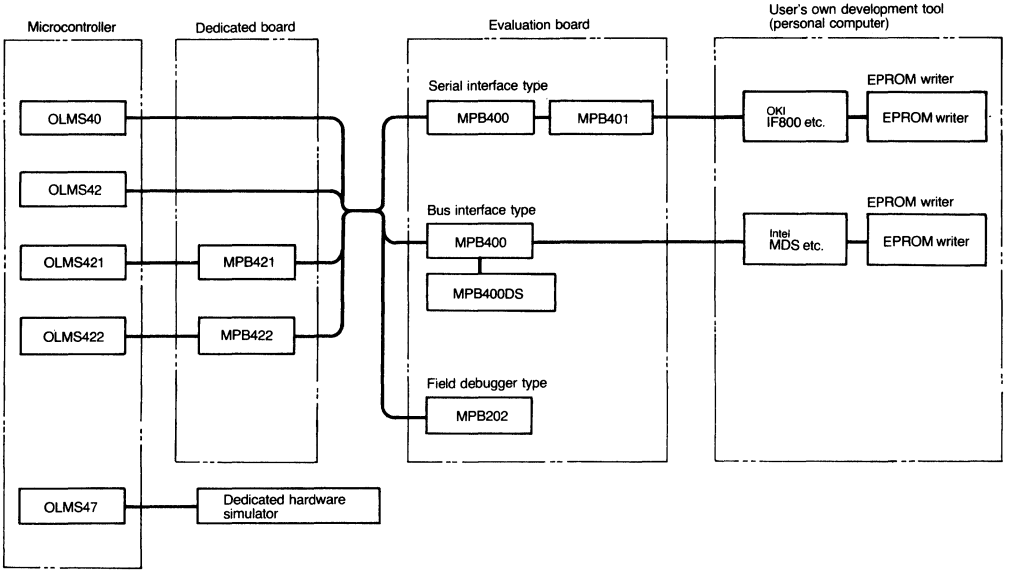
7

DEBUGGER COMMAND SAMPLES

```
DB400

<< DB400 >>
OLMS40 SERIES ON LINE DEBUGGER
VERS 1.3
*LOAD A:SAMPL1.HEX
*IB
    << INITIAL BUFFER >>
    F  E  D  C  B  A  9  8  7  6  5  4  3  2  1  0
0:  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
1:  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
2:  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
3:  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
4:  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
5:  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
6:  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
7:  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
RW=0 RX=0 RY=0 RZ=0 XCH=0 A=0
ACC=0 CY=0 DPH=0 DPL=0 PP=0
PF=0 PG=0
*GO URAM FROM 7A1 TILL PC=7CF &2
MODE: IB TO FB      Y/N/E ?    Y
PC=07CF 74      PA=F PB=F PD=C PE=5 PF=0 PG=0 PK=B PI=A
PC=07CF 74      PA=F PB=F PD=0 PE=5 PF=0 PG=0 PK=B PI=A
*GO FROM 7A1 TILL PF=9
MODE: IB TO FB      Y/N/E ?    N
MODE=NON TO FB
PC=07CF 74      PA=F PB=F PD=F PE=F PF=9 PG=0 PK=B PI=A
*FB
    << FINAL  BUFFER >>
    F  E  D  C  B  A  9  8  7  6  5  4  3  2  1  0
0:  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
1:  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
2:  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
3:  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
4:  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
5:  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
6:  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
7:  0  F  *  *  9  E  4  F  0  0  0  0  0  0  0  0
RW=F RX=4 RY=E RZ=9 XCH=* A=*
ACC=9 CY=1 DPH=2 DPL=0 PP=*
PA=F PB=F PD=F PE=F PF=9 PG=0
PK=B PH=2 CIN=0 INT=1
*STEP FROM 7A1 COUNT 6
PC=07A1 3700    PA=F PB=F PD=C PE=0 PF=9 PG=0 PK=B PI=A
PC=0700 AC      PA=F PB=F PD=C PE=0 PF=9 PG=0 PK=B PI=A
PC=0702 AD      PA=F PB=F PD=C PE=0 PF=9 PG=0 PK=B PI=A
PC=0704 65      PA=F PB=F PD=C PE=0 PF=9 PG=0 PK=B PI=A
PC=0705 26      PA=F PB=F PD=C PE=0 PF=9 PG=0 PK=B PI=A
PC=0706 37A3    PA=F PB=F PD=C PE=0 PF=9 PG=0 PK=B PI=A
*EXIT
```

SELECTION OF MICROCONTROLLER AND EVALUATION BOARDS/SOFTWARE



EVALUATION MICROCONTROLLERS

OLMS40	MSM5840	MPB400 board accessories
OLMS42	MSM5840	MPB400 board accessories
OLMS421	MSM5840	MPB421 board accessories
OLMS422	MSM5840	MPB422 board accessories

SOFTWARE

Serial interface type	FD40S·CP/M	Assembler/(serial) debugger CP/M single side single density 8-inch Two (serial) monitor software EPROMs
Bus interface type	FD40B·ISIS	Assembler/(bus) debugger ISIS-II single sided single density 8-inch Two (bus) monitor software EPROMSs
	PD40B·CP/M	Assembler/(bus) debugger CP/M single sided single density 8-inch Two (bus) monitor software EPROMS

MPB400 Board

The MPB400 board is used for OLMS-40 Series microcontroller program generation and evaluation. With the MSM8085A used for control purposes, several system configurations can be achieved in combination with other boards and software.
Dimensions: 220 × 305 (mm)

MPB401 Board

The MPB401 board, used together with the MPB400 board, consists of interface circuit with the development system and transfer speed converter unit.
Dimensions: 105 × 205 (mm)
80 × 115 (mm)

MPB202 Board

The MPB202 board is equipped with an evaluation microcontroller and 4K bytes of program EPROM sockets. This miniature size board is used effectively in mounting evaluations when built into application equipment for final testing of debugged programs.
Dimensions: 95 × 120 (mm)

MPB421

The MPB421 board has been designed for OLMS-421 microcontroller program evaluation, and is used connected to the MPB400. This board is equipped with LCD driver circuits and PLA EPROM sockets.
Dimensions: 145 × 180 (mm)

EASE6400 PROGRAM DEVELOPMENT SUPPORT SYSTEM

The EASE6400 Program Development Support System has been specifically designed for rapid and efficient program development of Oki's OLMS-64 series of CMOS 4-bit, 1-chip microcontroller. (Target chips: MSM6404, MSM6402, MSM6422, MSM6411, MSM6431, MSM6442 and MSM6408)

FEATURES

1. In-circuit emulation when connected to an RS232C interface equipped a host computer or to an input/output device such as a CRT terminal and the EASE8 (option).
3. Six types of break conditions (for interruption of. emulation) including execution address and machine cycle counter.
4. EPROM writer (for 2732, 2732A, 2764) to enable programming, transfer, and verification of user program area contents.
5. User program debugging operations directed by debug commands entered from the keyboard of the connected input/output device.
6. Program evaluation by use of dedicated evaluation chip (MSM6400E), thereby enabling the same operations to be executed as when the MSM6404 chip is used.
7. Built-in system diagnostic program.

HOST COMPUTER REQUIREMENTS

1. Operating system is one of the follows.
   (1) CP/M®-80 (ver 2.0 or later)
   memory capacity of the host computer must be sufficient to run at least 52K CP/M.
   (2) MS-DOS® (ver2.11 or later)
   (3) PC-DOS® (ver2.11 or later)
2. At least one RS232C communication port is implemented.
3. Data transfer is performed through RS232C communication port using BDOS function call 6 on condition that console device is assigned to TTY:.

SYSTEM CONFIGURATION

The EASE6400 Program Development Support System consists of the EASE6400 Emulator (a high performance program emulator which includes the EASE Host Monitor, the EASE6400 Evaluation, and the host computer) and the ASM6404 Assembler. With the EASE6400 Evaluation connected online to the host computer equipped with an RS232C interface, the system covers all operations from assembly of the source program through program evaluation and debugging.

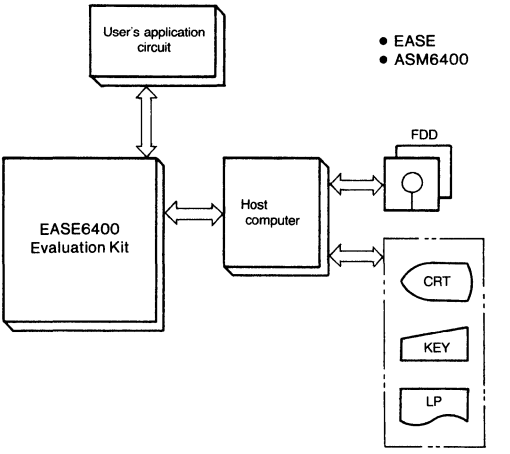
COMMUNICATION WITH HOST COMPUTER

ICs used

- Communication interface MSM82C51A
- Driver/receiver SN75188N, SN75189N

Transmission format

- 110 to 9600 bps (switchable)
- 8 bits, 2 stop bits, non-parity
- 7 bits, 2 stop bits, even parity
- 7 bits, 2 stop bits, odd parity (switchable)



Note: CP/M is a registered trademark of Digital Research Inc. (U.S.A.)

EASE and ASM6404 for PC-DOS, MS-DOS are optional softwares.

7

ASM6400 ASSEMBLER

The ASM6400 is a floppy disk based high-performance assembler which operates on CP/M-80, MS-DOS, or PC-DOS.
This assembler is used to translate source files generated on disk by using an editor (available on the market), thereby generating object files (Intel HEX format), assemble list files, and cross-reference list files in the specified devices.

FEATURES

1. Free descriptive format source files
2. Capacity to describe up to ten types of operators in the source program operand column
3. Ability to specify the number of characters per line and the number of lines per page in assembly list files
4. Cross-reference list file generation capacity
5. 13 types of powerful pseudo-instructions available
6. Object codes where internal page jump instructions and all page jump instructions are assigned automatically can be obtained by branch pseudo-instructions
7. Control of assembly list file outputs by LIST or NLST pseudo-instruction

LIST OF PSEUDO-INSTRUCTIONS

Pseudo-instruction	Function
TYP	Specifies a type of target chip
EQU	Assignment of operand value to name.
SET	Same as EQU pseudo-instruction, but with redefinition capacity
ORG	Setting of program start address
END	Indication of end of program
B	Automatic conversion to internal page or all page jump instruction after checking branch destination
DB	8-bit data or ASCII character definition
DS	Reserves memory area for specified number of bytes
NSE	Setting of 0 in the 4 lower order bits of the assembler location counter, and addition of 16. The NOP instruction is assigned to blank areas where no machine language instruction has been assigned.
DATE	Insertion of date in assemble list title
PAGE	Execution of assemble list page feed
TITL	Insertion of assemble list title
LIST	Designation of assemble list output
NLST	Inhibition of assemble list output

EASE6400 EMULATOR

Consisting of a host computer and the EASE6400 Program Evaluation Kit, the EASE6400 Emulator supports a wide range of development debugging operations efficiently and effectively. OLMS-64 application programs can be debugged without user application circuits just as easily as completed systems.

FEATURES

1. Real-time tracing function does not effect execution time

The EASE6400 Evaluation Kit incorporates a realtime trace area for 2048 machine cycles. When the tracing for a particular user program area address has been set, the address, operation code, port, and probe status are traced each time the specified address is executed.

2. Execution time measurement

The user program execution time can be measured by using the cycle counter on the EASE6400 Evaluation Kit. (Max. 16,777,215 cycles)

This counter can also be used as a pass counter to indicate the number of times a specified address is executed. And emulation can be stopped after a specified address has been executed a specified number of times.

3. Mass storage

With an 8192-bytes of static RAM area in the EASE6400 Evaluation Kit for use as a user program area, there is no problem with inadequate memory area during debugging. And, needless to say, user programs can be loaded/saved from the host computer. Furthermore, with an EPROM writer included on this board, the user program area contents can be written into EPROMs, and the EPROM contents can be read out.

4. Ample break functions

The emulator can suspend (break) program execution by any of the following six break conditions.

a) Breakpoint break
   Break upon execution of specified address. (Any address may be specified.)
b) External break
   Break by application of external break signal.
c) Halt/stop instruction break
   Break by execution of HALT or STOP instruction.
d) Trace buffer full break
   Break when overflow of trace area occurs.
e) Cycle counter overflow break
   Break when overflow of cycle counter occurs.
f) Probe match break
   Break when probe data matches set data.

5. Extensive range of debugging commands

In addition to display/updating of all register, port, and RAM contents, the following commands enable all debugging operations to be executed efficiently.

a) User program execution

Input format:
STP number of instructions, start address

Input of this command results in the specified number of instructions in the user program being executed from the specified address (start address).

Input format:
G start address, break address (n)

Input of this command results in the user program being executed from the specified address (start address). Emulation is subsequently stopped when the specified address (break address) is executed n times.

Note: Program execution is suspended temporarily each time the specified address is executed.

Input format:
G start address, break address RAM (address–n)

Input of this command results in the user program being executed from the specified address (start address). Emulation is subsequently stopped if the contents of the specified RAM address are n when the specified address (break address) is executed. In this case, too, program execution is suspended temporarily whenever the specified address is executed.

b) Use of floppy disks

Input format:
LOD filename

Input of this command enables the specified file contents (user program) to be loaded into the EASE6400 Evaluation Kit code memory which serves as the OLMS-64 masked ROM area.

Input format:
SAV filename start address, end address

Input of this command enables the contents of the specified range of code memory to be saved at the specified filename.

Input format:
DIAG 2

Input of this command enables execution of commands within the specified file.
This execution can be suspended temporarily by using the PAUSE command.

c) Instruction executed bit memory
The EASE6400 Emulator includes an instruction executed bit memory which indicates which user program addresses have been executed. Program flow can be determined by examination of the contents of this memory.

6. Easy to remember command format

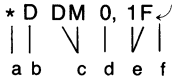
The EASE6400 Emulator debugging commands consist of command mnemonics followed by parameters (address or mnemonic).

Command mnemonic configuration

a) First character
Denotes the function to be executed by the emulator.

b) Second and following characters
Name of function to be executed by the emulator– that is, MSM6400E evaluation chip or EASE6400 Evaluation Kit register, memory, or port name (abbreviation).

Example:

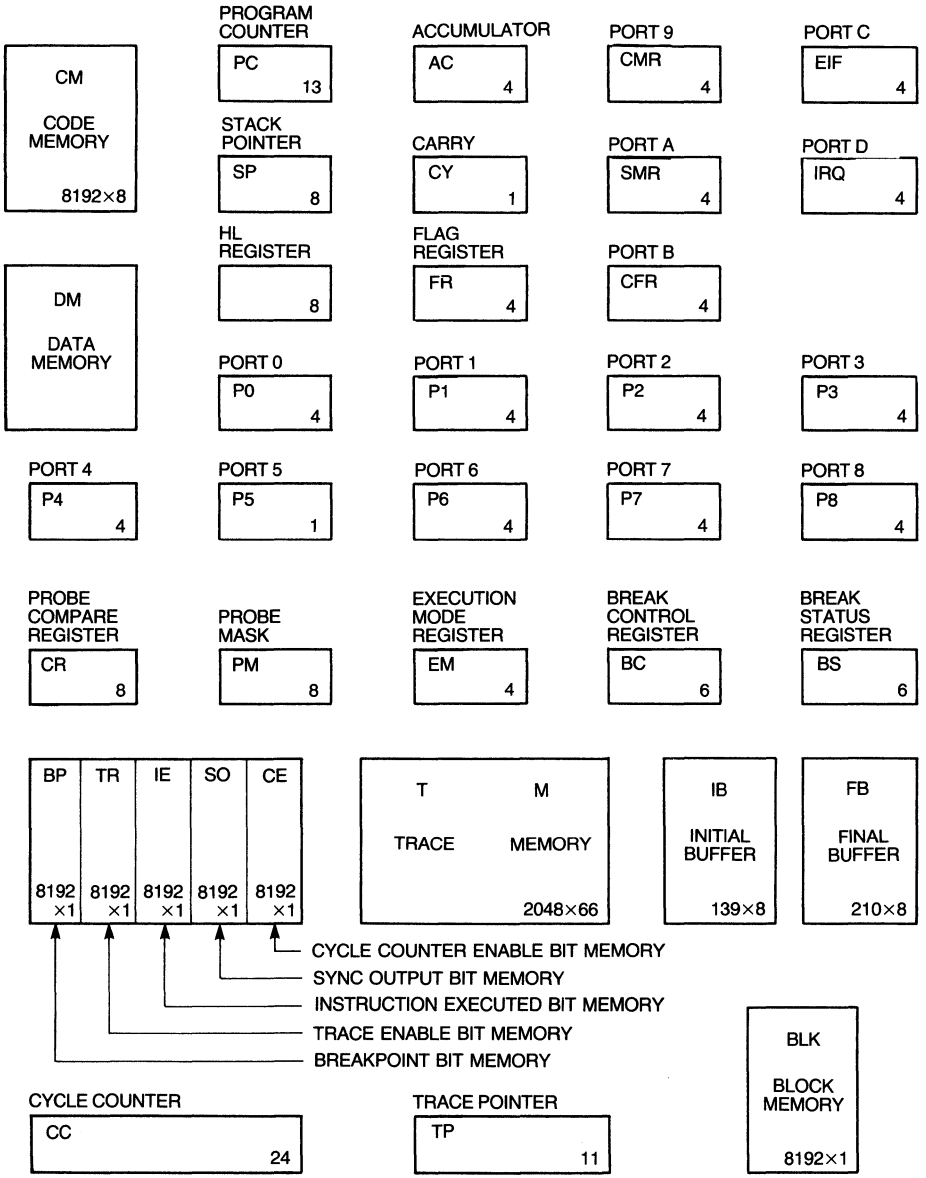


a: denotes wait for command input from emulator
b: denotes display of contents of specified object
c: data memory (equivalent MSM6400E RAM area) designation
d: start address of contents to be displayed
e: end address of contents to be displayed
f: carriage return denoting end of command input

7

EASE6400 EMULATOR MEMORY CONFIGURATION

The EASE6400 emulator memory area which can be used by the user, is outlined in the following diagram. The sections enclosed in boxes represent the register and memory areas which can be accessed from the host computer keyboard, and which can be displayed/updated by debugging command.



MAJOR MEMORY OPERATIONS

1. Code memory
The code memory is an 8192 × 8-bit RAM area which corresponds to the OLMS-64 masked ROM area where user programs are stored.
(The MSM6404 ROM capacity is 4000 bytes)

2. Data memory
The data memory is a 256 × 4-bit RAM area corresponding to the MSM6400E RAM area.

3. Instruction executed bit memory
This memory is an 8192 × 1-bit RAM area which has addresses identical to the code memory. When a code memory program is executed, the bits corresponding to the excuted addresses are set to "1". The program flow can thus be determined by checking the contents of this memory.

4. Break point bit memory
The break point bit memory is also an 8192 × 1-bit RAM area which has addresses identical to the code memory. If a particular bit in this memory is "1", program execution is suspended immediately after the code memory contents corresponding to that bit are executed.

5. Trace enable bit memory
The trace enable bit memory is another 8192 × 1-bit RAM area which has addresses identical to the code memory. If a particular bit in this memory is "1", port/register contents, etc., are stored in the trace memory when the code memory contents corresponding to that bit are executed.

6. Sync ouput bit memory
The sync output bit memory is another 8192 × 1-bit RAM area which has addresses identical to the code memory. When the code memory contents are executed, the corresponding bits in the sync output memory are checked, and if a "1" bit is found, an output sync signal (active LOW) is passed to the probe terminal.

7. Cycle counter enable bit
The cycle counter enable bit memory is also another 8192 × 1-bit RAM area which has addresses identical to the code memory. When the code memory contents are executed, the corresponding bits in the cycle counter enable bit memory are checked, and if a "1" bit is found, the cycle counter is counted up in step with that machine cycle.

8. Probe comparison register/probe mask
The conditions for generating a break by input data from the probe (probe match break) can be changed by altering these two settings.

9. Initial/final buffers
These two memories are 210 × 8-bit RAM areas used to store the initialized settings of the MSM6400E evaluation chip or the MSM6400E status immediately after a break when real-time emulation is executed.

10. Trace memory
The trace memory is a 2048 × 66-bit RAM area used to store traced data.
The trace instruction is given by the trace enable bit.

LIST OF DEBUGGING COMMANDS

Load, save, and Verify Commands		
	LOD [dr :] filename ↵ SAV [dr :] filename [address, address] ↵ VER [dr :] filename [address, address] ↵	Load programs into code memory Save code memory program Verify file with code memory
EPROM Commands		
	PPR address, address [, address] ↵ TPR address, address [, address] ↵ VPR address, address [, address] ↵	Program Code Memory into EPROM Transfer EPROM into Code Memory Verify EPROM with Code Memory
Commands Used to Display/Change Internal Status of Evachip		
PC	DPC CPC address ↵	Display Program Counter Change Program Counter
SP	DSP ↵ CSP data ↵	Display Stack Pointer Change Stack Pointer
AC	DAC ↵ CAC data ↵	Display Aoc Change Acc
CY	DCY ↵ CCY data ↵	Display Carry Flag Change Carry Flag
HL	DHL ↵ CHL data ↵	Display H-L Registers Change H-L Registers
FR	DFR ↵ CFR data ↵	Display Flag Register Change Flag Register
P0 ~ P8	DPn ↵ (n = 0, 1, 2, 3,... 8) CPn data ↵ (n = 0, 1, 2, 3,... 8)	Display Port n Change Port n
P9	DCMR ↵ CCMR data ↵	Display Counter Mode Register (port 9) Change Counter-Mode Register
PA	DSMR ↵ CSMR data ↵	Display Shift Mode Register (port A) Change Shift Mode Register
PB	DCFR * Change CFR command is not permitted	Display Control Flag Register (port B)
PC	DEIF ↵ CEIF data ↵	Display Enable Interrupt Flag (port C) Change Enable Interrupt Flag
PD	DIRQ * Change IRQ command is not permitted	Display Interrupt Request Flag (port D)
ME I	DME I ↵ CME I data ↵	Display Master Enable Interrupt Flag Change Master Enable Interrupt Flag
	D ↵	Display all internal status
Data & Code Memory Display, Change, and Fill Commands		
DM	DDM address [, address] ↵ CDM address ↵ data data... ↵ FDM address, address, data ↵	Display Data Memory Change Data Memory Fill Data Memory
CM	DCM address [, address] ↵ CCM address ↵ data data... ↵ FCF address, address, data ↵	Display Code Memory Change Code Memory Fill Code Memory

Attribute Memory Display, Enable, and Reset Commands		
BP	DBP address [, address] ↵ EBP address [, address] ↵ RBP address [, address] ↵	Display Break Point Bits Memory Enable Break Point Memory Reset Break Point Bits Memory
TR	DTR address [, address] ↵ ETR address [, address] ↵ RTR address [, address] ↵	Display Trace Enable Bits Memory Enable Trace Enable Bits Memory Reset Trace Enable Bits Memory
IE	DIE address [, address] ↵ RIE address [, address] ↵	Display Instruction Executed Bits Memory Reset Instruction Executed Bits Memory
CE	DCE address [, address] ↵ ECE address [, address] ↵ RCE address [, address] ↵	Display Cycle Counter Enable Bits Memory Enable Cycle Counter Enable Bits Memory Reset Cycle Counter Enable Bits Memory
SO	DSO address [, address] ↵ ESO address [, address] ↵ RSO address [, address] ↵	Display Sync Out Enable Bits Memory Enable Sync Out Enable Bits Memory Reset Sync Out Enable Bits Memory
Trace Memory Display Commands		
TM	DTM DTL	Display Trace Memory Display Trace List
Other Hardware Display and Change Commands		
CC	DCC ↵ CCC number ↵	Display Cycle Counter Change Cycle Counter
BC	DBC ↵ SBC [[±] mnemonic, ...]	Display Break Condition Register Set Break Condition Register mnemonic means one of following key words here. BB ..... Break at Break Point XB ..... External Break CO ..... Cycle Counter Over Flow TF ..... Trace Memory Full PM ..... Probe Match HS ..... HALT/STOP Instruction Executed
BS	DBS ↵	Display Break Status Register
PM	DPM ↵ CPM data ↵	Display Probe Mask Register Change Probe Mask Register
CR	DCR ↵ CCR data ↵	Display Probe Compare Register Change Probe Compare Register
EM	DEM CEM mode-ref ↵	Display Execution Mode Register Change Execution Mode Register mode-ref means one of following keywords here. IF ..... IB TO FB IN ..... IB TO NON NN ..... NON TO NON NF ..... NON TO FB FF ..... FB TO FB
BLK	DBL address [, address] ↵ SPB address [, address] ↵ SDB address [, address] ↵	Display Block Memory Set Block Memory into Program Block Set Block Memory into Data Block
BANK	BANK 1 or 2	Set Attribute Memory Bank Register

Initial Buffer & Final Buffer		
	DIB↵ DFB↵ CI key-word data↵	Display Initial Buffer Display Final Buffer Change elements of Initial Buffer key-word...AC, HL, FR, SMR, CMR, MEI, EIF, CY, Pn (n=0, 1, 3, 4, ..., 8)
IB	CIDM address↵ data data...↵	Change Initial Buffer Data Memory
FB	CF key-word data↵	Change elements of Final Buffer key-word...AC, HL, FR, SMR, CMR, MEI, EIF, CY, Pn (n=0, 1, 3, 4, ..., 8)
	CFDM address data data...↵ FIDM address, address, data↵ FFDM address, address, data↵	Change Final Buffer Data Memory Fil Initial Buffer Data Memory Fill Final Buffer Data Memory
Assemble & Disassemble Commands		
	ASM address↵ mnemonic...↵ DASM address [, address]↵	Assemble to Code Memory Disassemble to Console
Emulation Commands		
	G [start-address] [, break-parameter, ...]↵ GD [start-address] [, break-parameter, ...]↵	Go (Start Real Time Emulation) Go Direct (Start Real Time Emulation) break-parameter format is shown below 1) address 2) asddress (n) 3) addres (n) & address (m) 4) addres RAM (address – data) $1 < n,\ m < 65535$
	STP [number] [, address]↵	Start Step Execution
Other Commands		
	@ [–n]↵ (n = 1, 2, ..., 9) HELP↵ EXIT↵ RES↵ RES E↵ SIO H or F↵ DEV↵ ST mnemonic↵ FDD [dr ... dr]↵ DIR [dr:]↵	Command – Repeat Display Help File (EASE 64, HLP Exit to CP/M Reset EASE 6400 system Reset Eva – chip (MSM 6400E) Set I/O Mode CH2 port Output Control Family Set Flopy Disk Unit Set Display file directory

EASE6502 PROGRAM DEVELOPMENT SUPPORT SYSTEM

The EASE6502 Program Development Support System has been specifically designed for rapid and efficient program development of Oki's MSM6502 CMOS 4-bit, 1-chip microcontroller.

FEATURES

1. Connecting the MPB6502 Evaluation Board to a host computer or to an I/O terminal such as a CRT terminal includes an RS232C interface circuit. All debugging operations can be entered from the console keyboard.
2. Executes user programs with the trace in real time or in single step mode.
3. Mass storage user program area (4K bytes).
4. Six types of break conditions, including execution address and machine cycle counter.
5. EPROM programmer (for 2716, 2732, 2732A) to enable programming, transfer, and verifing with user program area contents.
6. The same operation as the MSM6502, because OKi designed the original evaluation chip (MSM6502E).
7. Built-in system diagnostic program.

HOST COMPUTER REQUIREMENTS

1. Operating system is one of the follows.
   (1) CP/M®-80 (ver 2.0 or later)
   memory capacity of the host computer must be sufficient to run at least 52K CP/M.
   (2) MS-DOS® (ver 2.11 or later)
   (3) PC-DOS® (ver 2.11 or later)
2. At least one RS232C communication port is implemented.
3. Data transfer is performed through RS232C communication port using BDOS function call 6 on condition that console device is assigned to TTY:.

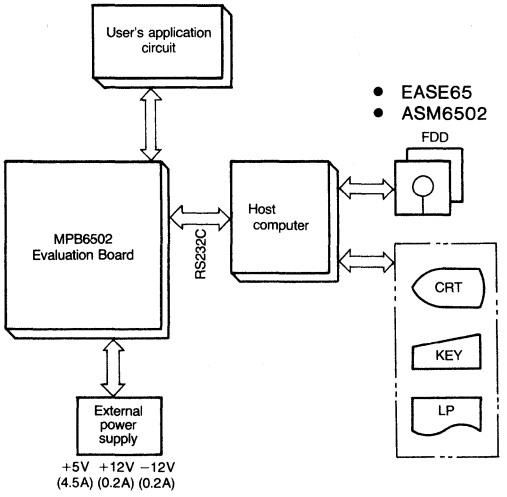
COMMUNICATION WITH HOST COMPUTER

ICs used
- Communication interface MSM82C51A
- Driver/receiver SN75188N, SN75189N

Transmission format
- 110 to 9600 bps (selectable)
- 8 bits, 2 stop bits, non-parity
  7 bits, 2 stop bits, even parity
  7 bits, 2 stop bits, odd parity (selectable)

SYSTEM CONFIGURATION

The EASE6502 Program Development Support System consists of the EASE6502 Emulator (a high performance program emulator which includes the EASE65 Host Monitor, the MPB6502 Evaluation Board, and the host computer) and the ASM6502 Assembler. With the MPB6502 Evaluation Board connected online to the host computer equipped with an RS232C interface, the system covers all operations from assembly of the source program through to program evaluation and debugging.



Note: CP/M is the registered trademark of Digital Research Inc. (U.S.A.)

EASE65 and ASM6502 for MS-DOS, PC-DOS are optional softwares.

ASM6502 ASSEMBLER

The ASM6502 is a floppy disk based high-performance assembler.
This assembler is used to translate source files generated on disk by using an editor (available on the market), thereby generating object files (Intel HEX format), assembly list files, and cross-reference list files on the specified devices.

FEATURES

1. Free descriptive format source files
2. Capacity to describe up to ten types of arithmetic in the source program operand field
3. Ability to specify the number of characters per line and the number of lines per page in assembly list files
4. Cross-reference list file generation capacity
5. 13 types of powerful pseudo-instructions available
6. Object codes where all page jump instructions and inter-page jump instructions are assigned automatically can be obtained by branch pseudo-instructions
7. Control of assemble list file outputs by LIST or NLST pseudo-instruction

LIST OF PSEUDO-INSTRUCTIONS

Pseudo-instruction	Function
EQU	Assign the operand value to the name.
SET	Same as EQU pseudo-instruction, but with redefinition capacity
ORG	Define assembler location counter.
END	Terminte assembly
B	Automatic conversion to all-page or inter-page jump instruction after checking branch destination
DB	8-bit data or ASCII character string definition
DS	Reserve n bytes area of uninitialized storage
NSE	Setting of 0 in the 4 lower order bits of the assembler location counter, and addition of 16. The NOP instruction is assigned to blank areas where no machine language instruction has been assigned.
DATE	Insertion of date in assemble list title
PAGE	Execute page eject
TITL	Insert assemble list title
LIST	Turn on assemble list output
NLST	Turn off assemble list output

EASE6502 EMULATOR

Consisting of a host computer and the MPB6502 Program Evaluation Board, the EASE6502 Emulator supports a wide range of development debugging operations efficiently and effectively. MSM6502 application programs can be debugged without user application circuits just as easily as completed systems.

FEATURES

1. Real-time tracing function does not effect execution time
The MPB6502 Evaluation Board incorporates a real-time trace area for 2048 machine cycles. When tracing for a particular user program area address has been set, the port, HL register, and MEI flag status are traced each time the specified address is executed.

2. Execution time measurement
The user program execution time can be measured by using the cycle counter on the MPB6502 Evaluation Board. (Max. 16,777,215 cycles)
This counter can also be used as a pass counter to indicate the number of times a specified address is executed. Emulation can be stopped after a specified address has been executed a specified number of times.

3. Mass storage user program area
With a 4096-byte static RAM area in the MPB6502 Evaluation Board for use as a user program area, there is no problem with inadequate memory area during debugging. And, needless to say, user programs can be loaded/saved from the host computer. Furthermore, with an EPROM programmer included on this board, the user program area contents can be written into EPROMs, and the EPROM contents can be read out.

4. Ample break functions
The emulator can suspend (break) program execution by any of the following six brteak conditions.

a) Breakpoint break
   Break upon execution of specified address. (Any address may be specified.)
b) External break
   Break by application of external break signal.
c) Halt instruction break
   Break by execution of HALT instruction.
d) Trace buffer full break
   Break when overflow of trace area occurs.
e) Cycle counter overflow break
   Break when overflow of cycle counter occurs.
f) Probe match break
   Break when probe data matches set data.

5. Extensive range of debugging commands
In addition to display/updating of all register, port, and RAM contents, the following commands enable all debugging operations to be executed efficiently.

a) User program execution

Input format:
STP number of instructions, start address

Input of this command results in the specified number of instructions in the user program being executed from the specified address (start address).
And if the terminal to be used to display the contents has been specified by SDF command at this time, the display can be put into an easy to read format.

Input format:
G start address, break address (n)

Input of this command results in the user program being executed from the specified address (start address). Emulation is subsequently stopped when the specified address (break address) is executed n times.

Note: Program execution is suspended temporarily each time the specified address is executed.

Input format:
G start address, break address RAM (address–n)

Input of this command results in the user program being executed from the specified address (start address). Emulation is subsequently stopped if the contents of the specified RAM address are n when the specified address (break address) is executed. In this case, too, program execution is suspended temporarily whenever the specified address is executed.

b) Use of floppy disks

Input format:
LOD filename

Input of this command enables the specified file contents (user program) to be loaded into the MPB6502 Evaluation Board code memory which serves as the MSM6502 masked ROM area.

Input format:
SAV filename start address, end address

Input of this command enables the contents of the specified range of code memory to be saved at the specified filename.

Input format:
DIAG filename

Input of this command enables execution of commands within the specified file automatically. This execution can be suspended temporarily by using the PAUSE command.

Input format:
LIST filename
NLIST

If the list command is entered, the emulator creat the CP/M file and write into it the any characters which are output to the console till entering the NLST command.

c) Instruction executed bit memory
The EASE6502 Emulator includes an instruction executed bit memory which indicates which user program addresses have been executed. Program flow can be known by examination of the contents of this memory.

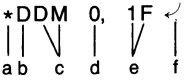
6. Easy to remember command format

The MPB6502 Emulator debugging commands consist of command mnemonics followed by parameters (address or mnemonic).
Command mnemonic configuration

a) First character
Stand for the emulator function (display/update)

b) Second and following characters
Represent one of the MPB6502 Evaluation Board or the MSM6502E evaluation chip element (register, memory or port name).

Example:

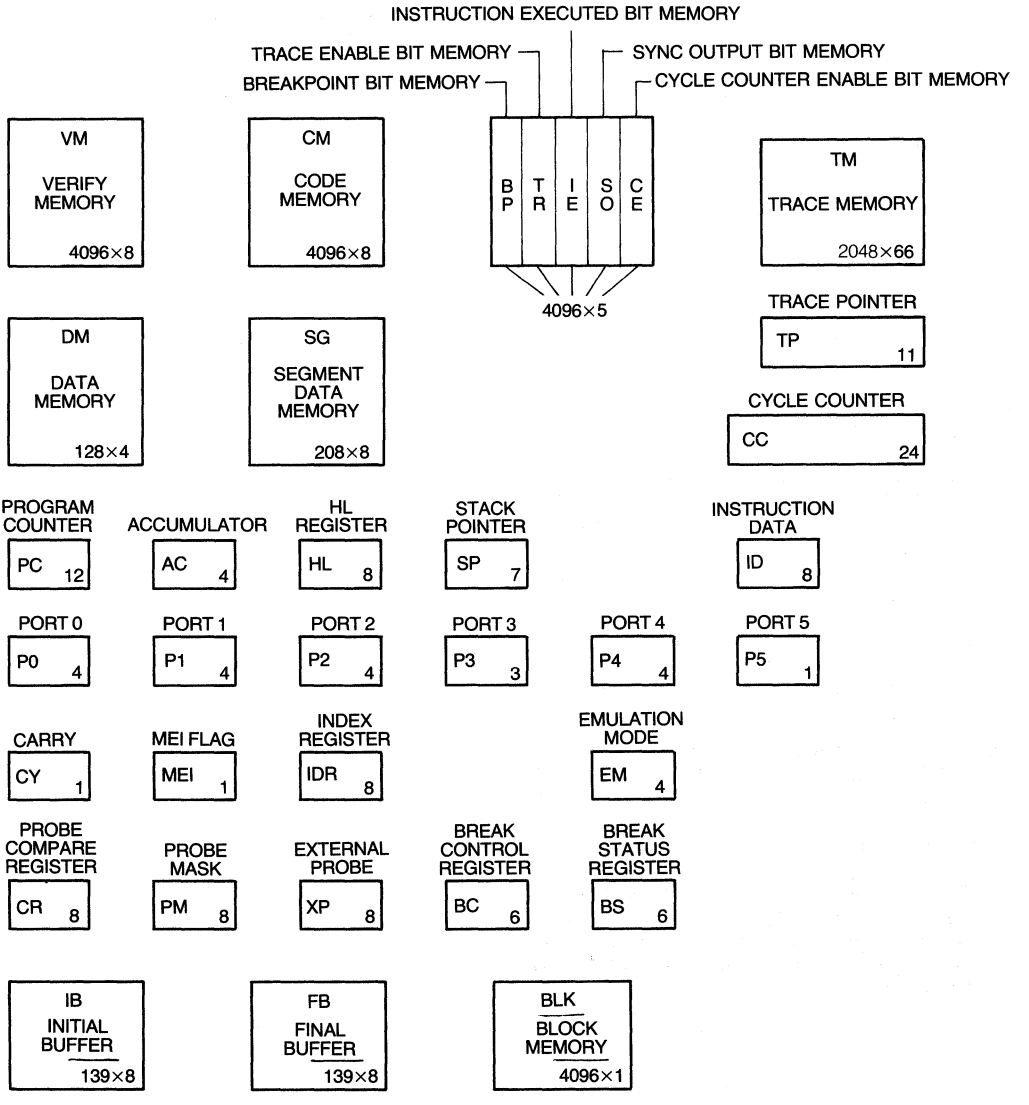


a: denotes wait for command input from emulator
b: denotes display of contents of specified registter/ memory/board
c: data memory (equivalent MSM6502 RAM area) designation
d: start address of contents to be displayed
e: end address of contents to be displayed
f: carriage return denoting end of command input

7

EASE6502 EMULATOR MEMORY CONFIGURATION

The EASE6502 emulator memory area which can be used by the user is outlined in the following diagram. The sections enclosed in boxes represent the register and memory areas which can be accessed from the host computer keyboard, and which can be displayed/updated by debugging command.



MAJOR MEMORY OPERATIONS

1. Code memory
The code memory is an 4096 × 8-bit RAM area which corresponds to the MSM6502 masked ROM area where user programs are stored.
(The MSM6502 ROM capacity is 2000 bytes)

2. Data memory
The data memory is a 128 × 4-bit RAM area corresponding to the MSM6502 masked RAM area.

3. Verify memory
The verify memory is a RAM area with addresses identical to the code memory. When a user program is loaded into the code memory identical contents are also set in this memory.
Comparrison of this memory with the code memory enables the operator to determine what sections of the user program have been changed after loading.

4. Instruction executed bit memory
This memory is an 4096 × 1-bit RAM area which has addresses identical to the code memory. When a code memory program is executed, the bits corresponding to the executed addresses are set to "1". The program flow can thus be known by checking the contents of this memory.

5. Break point bit memory
The break point bit memory is also an 4096 × 1-bit RAM area which has addresses identical to the code memory. If a particular bit in this memory is "1", program execution is suspended immediately after the code memory contents corresponding to that bit are executed.

6. Trace enable bit memory
The trace enable bit memory is another 4096 × 1-bit RAM area which has addresses identical to the code memory. If a particular bit in this memory is "1", port/register contents, etc., are stored in the trace memory when the code memory contents corresponding to that bit are executed.

7. Sync ouput bit memory
The sync output bit memory is another 4096 × 1-bit RAM area which has addresses identical to the code memory. When the code memory contents are executed, the corresponding bits in the sync output memory are checked, and if a "1" bit is found, an output sync signal (active LOW) is passed to the probe terminal.

8. Cycle counter enable bit
The cycle counter enable bit memory is also another 4096 × 1-bit RAM area which has addresses identical to the code memory. When the code memory contents are executed, the corresponding bits in the cycle counter enable bit memory are checked, and if a "1" bit is found, the cycle counter is counted up in step with that machine cycle.

9. Probe comparl register/probe mask
The conditions for generating break by input data from the probe (probe match break) can be changed by altering these two settings.

10. Segment data memory
The segment data memory is a 208 × 8-bit RAM area with coordinates 0 thru 7 on the vertical axis and A thru Z on the horizontal axis. This memory is used to display the status of each bit of the MSM6502 RAM liquid crystal display area at the specified coordinate and by the specified character.
Displaying the status of this memory provides an image of the liquid crystal display status.

11. Initial/final buffers
These two memories are 139 × 8-bit RAM areas used to store the initialized settings of the MSM6502E evaluation chip or the MSM6502E status immediately after a break when real-time emulation is executed.

12. Trace memory
The trace memory is a 2048 × 66-bit RAM area used to store traced data.
The trace instruction is given by the trace enable bit.

DEBUGGING COMMAND TABLE

Load, Save, and Verify Commands	
1. LOD [dr :] filename	Load Program into Code Memory
2. SAV [dr :] filename [address, address]	Save Code Memory Program
3. VER [dr :] filename [adress, address]	Verifty File with Code Memory
4. LSG [dr :] filename [address, address]	Load Segment data
5. SSG [dr :] filename	Save Segment data
6. VCM address [, address]	Verify Code Memory with Verify Memory
EPROM Commands	
1. PPR address, address [, address]	Program Code Memory into EPROM
2. TPR address, address	Transfer EPROM into Code Memory
3. VPR address, address [, address]	Verify EPROM with Code Memory
Display Commands	
1. D	All Register, Flag, Port
2. DAC	Acc Register
3. DHL	HL Register
4. DCY	Carry Flag
5. DSP	Stack Pointer
6. DLD	LCD Driver ON/OFF status (Port 4-0 bit)
7. DEI	Enable Interupt Flag (MEI, TBC EI, ext. INT EI)
8. DPn	Port N (n=0, 1, 2, 3, 4)
9. DPC	Program Counter
10. DCC	Cycle Counter
11. DCR	Probe Compare Register
12. DPM	Probe Mask Register
13. DBC	Break Condition
14. DBS	Break Status
15. DEM	Emulation Mode
16. DIB	Initial buffer & Final buffer
17. DFB	Final buffer & Initial buffer
18. DSG	Segment Data Status
19. DBP address [, address]	Break Point Bit Memory
20. DCE address [, address]	Cycle Counter Enable Bit Memory
21. DSO address [, address]	Sync Output Enable Bit Memory
22. DIE address [, address]	Instruction Executed Bit Memory
23. DTR address [, address]	Trace Enable Bit Memory
24. DDM address [, address]	Data Memory (M6502E Internal RAM)
25. DCM address [, address]	Code Memory
26. DVM address [, address]	Verify Memory
27. DTM tp-address, line-number	Trace Memory
28. DTL -number, line-number	Trace List
Change Commands	
1. CAC data	Acc Register
2. CHL data	HL Register
3. CCY 0 or 1	Carry Flag
4. CSP data	Stack Pointer
5. CLD 0 or 1	LCD Driver ON/OFF Status
6. CEI [data]	Enable Interrupt Flag
7. CIR data	Index Register
8. CPn data	Port n (n=0, 1, 3, 4, 5)
9. CPC address	Program Counter
10. CCC (+ or −) data	Cycle Counter
11. CCR data	probe Compare Register
12. CPM data	Probe Mask Register
13. CBZ 0 or 1	Buzzer Output Select Hard or Soft
14. CEM mode	Emulation Mode *mode* IF IB to FB IN IB to NON NN NON to NON NF NON to FB FN FB to NON
15. CI key-word data	Initial Buffer
16. CF key-word data	Final Buffer
17. CSG	Segment data
18. CCM address	Code Memory
19. CDm address	Data Memory (M6502E Internal RAM)
20. BANK 0 or 1	Attribute Memory Bank

Enable Commands

1. EBP address [, address] Break Point Bit Memory
2. ECE address [, address] Cycle Counter Enable Bit Memory
3. ESO address [, address] Sync Output Enable Bit Memory
4. ETR address [, address] Trace Enable Bit Memory

Reset Commands

1. RBP address [, address] Break Point Bit Memory
2. RCE address [, address] Cycle Counter Enable Bit Memory
3. RIE address [, address] Instruction Executed Bit Memory
4. RSO address [, address] Sync Output Enable Bit
5. RTR address [, address] Trace Enable Memory
6. RES E M6502E Eva-chip
7. RES MPB6502 Board

Fill Commands

1. FCM address, address, data Code Memory
2. FDM address, address, data Data Memory
3. FIDM address, address, data Initial Data Memory
4. FFDM address, address, data Final Data Memory

Emulation Commands

1. G [st-address] [, break-parameter, ----] Begin Emulation
2. GD [st-address] [, break-parameter, ----] Begin Emulation

If the optional st-address is given, emulator will begin emulation from st-address. And if optional break-parameter is given, emulation will break on the first break-parameter to be satisfied.

Break-parameter = break-address
= break-address (pass count)
= break-address (pass count) & break-address (pass count)
= break-address RAM (RAM address-data)

3. STP [number] [, st-address] Single Step

Other Commands

1. ASM address Assemble to Code Memory
2. DASM address [, address] Disassemble to Console
3. LIST [dr :] filename List Console Output into Disk File
4. NLST End listing
5. DBLK address [, address] Display Block Memory
6. SPB address [, address] Set Program Block
7. SDB addres [, address] Set DB Block
8. SIO F or H Set Emulator to I/O terminal mode
9. EXIT Return to CP/M
10. PAUSE Stop Command File Execution
11. DIAGI Start Sell Check Program
12. DIAG [dr :] filename Execute Command File
13. @ [−n] Repeat Command
14. HELP Display HELP File
15. SBC (+ or −) mnemonic [, (+ or −) mnemonic. ---]
Set/Reset Break Condition Register

* mnemonic*

BB	Break when Break Point Reached
XB	External Break
CO	Break on Cycle Counter Overflow
TF	Break when Trace Buffer Full
PM	Break on Probe Match
HT	Break on HALT Instruction

16. SDF (+ or −) mnemonic [, (+ or −) mnemonic, ---]
Set/Reset Dump Format

<CONTROL KEY>

1. DEL Delete the last character entered
2. Cntrl/C Return to CP/M
3. Cntrl/P Copy all subsequent console output to the currently assigned list device
4. Cntrl/Q Continue normal dispaly
5. Cntrl/S Stop display
6. Cntrl/R Echoe current input line
7. ESC Abort any command inprogress
8. @ [−n] Repeat the command

Note: The MSM6502 RAM area corresponds to the EASE6502 Emulator data memory, and the mask ROM corresponds to the code memory.

SAMPLES

```
A>EASE65

      << EASE65 >>   Preliminary #1.0

   Sep. 3,1983  OKI electric ind.co.,ltd

        Now, start up  MPB6502 board.


           **** HARD WARE  SELF CHECK ****
                    [ NORMAL END ]




      <<   MPB6502 EMULATOR  PRELIMINARY  #1.5   >>
              COPYRIGHT OKI ELECTRIC IND. CO., LTD. 1984

   ACTIVE BREAK CONDITION ---> BREAK POINT BREAK
                               HALT INSTRUCTION BREAK




     ** NOW DEVICE NO. = A,B **
     [ KEY IN OTHER DEVICE NO. ]
     [PRESENTED DEVICE NO.=A:B:]

 * DIAG INIT1
 * ; INITIALIZATION START
 * FCM *
 * LSG PIC.DAT
   SGRAM SET-UP COMPLETED
 * LOD RAMSET
   LOAD COMPLETED    NEXT ADDRESS=0020
 * LIST CRT.LST
 * ; NOW INITIALIZATION END AND LIST START
 *
   DIAG COMMAND END
 * GD 0,1F
   BANK : 1    EXECUTION MODE : NON TO NON
   (RESET TRACE POINTER)
   ** EMULATION GO **
              PARAMETER=001F(     0)
                   ADDRESS BREAK
           [BREAK PC=01F          NEXT PC=020]
                   [NEXT TRACE POINTER=17]
 * DDM 0,3
   LOC=03-00    5 5 5 2
 * DTM 4,3
   LOC=012 82       LAI  2       PORT(0-2)=FF0 HL=00 CY=0 MEI=0   XP=FF   TP=4
   LOC=013 B4       LMA           PORT(0-2)=FF0 HL=00 CY=0 MEI=0   XP=FF   TP=5
   LOC=014 11       INL           PORT(0-2)=FF0 HL=00 CY=0 MEI=0   XP=FF   TP=6
 * DIE 0,20
   LOC=000    1100 0000 0000 0000    1111 1111 1111 1101
   LOC=020    0
 * DASM 0,13
   LOC=000 5F       DI
   LOC=001 D0       JCP  010
   LOC=002          DB   00 00 00 00 00 00 00 00  00 00 00 00 00 00
   LOC=010 6A00     LHLI 00
   LOC=012 82       LAI  2
   LOC=013 B4       LMA
 * NLST
 * EXIT
A>
```

EASE6502 PROGRAM DEVELOPMENT SUPPORT SYSTEM for the MSM6502 CMOS 4-BIT 1-CHIP MICROCONTROLLER

Category	Model	Title
Hardware	MPB6502	4-bit 1-chip microcontroller evaluation board
Software	EASE65	Floppy disk based host monitor*
	ASM6502	Floppy disk assembler*
Manual	TM-6502	EASE6502 Development Support System – User's Manual
Accessories	TCU-6502	User application circuit connection cables
	TCS-1	Host CP/M® computer connecting cables (for if800 model 10/20/30)
	TCP-1	Power supply cables (+5V, 4.5A) (+12V, 0.2A) (−12V, 0.2A)
	TCX-1	External probe cables
	TCC-1	Board connecting cables

* Available under following operating system
- CP/M-80 (ver 2.0 or later)
- MS-DOS (ver2.11 or later)
- PC-DOS (ver2.11 or later)

OPTIONS

Option name	Remarks
MPB6502 EVA Board	A simplified evaluation board consisting of the MSM6502E plus EPROM sockets, and designed for the MSM6502. Programs are evaluated by inserting the EPROM where the user program is stored into an EPROM socket. Dimensions: 160 × 127 (mm)

7

EASE 80C49 PROGRAM DEVELOPMENT SUPPORT SYSTEM

for

MSM80C48RS/MSM80C49RS/MSM80C50RS CMOS 8-Bit Microcontrollers

EASE80C49 PROGRAM DEVELOPMENT SUPPORT SYSTEM

The EASE80C49 Program Development Support System has been specifically designed for rapid and efficient program development of Oki's MSM80C48RS, MSM80C49RS and MSM80C50RS CMOS 8-bit, 1-chip microcontrollers.

FEATURES

1. In-circuit emulation when connected to an RS232C interface equipment host computer or to an input/output device such as a CRT terminal.
2. 4K byte user program area available for code transfer to/from floppy disk, collation/updating of area contents, and continuous/step execution.
3. Eight types of break conditions (for interruption of emulation) including execution address and machine cycle counter.
4. EPROM programmer (for 2716, 2732, 2732A) to enable writing, transfer, and comparison of user program area contents.
5. User program debugging operations directed by debug commands entered from the keyboard of the connected input/output device.
6. Built-in system diagnostic program.

HOST COMPUTER REQUIREMENTS

1. Operating system is one of the follows.
   (1) CP/M®-80 (ver 2.0 or later)
   memory capacity of the host computer must be sufficient to run at least 52K CP/M.
   (2) MS-DOS® (ver 2.11 or later)
   (3) PC-DOS® (ver 2.11 or later)
2. At least one RS232C communication port is implemented.
3. Data transfer is performed through RS232C communication port using BDOS function call 6 on condition that console device is assigned to TTY:.

MPB800 EVALUATION BOARD DATA TRANSFER SYSTEM

ICs used

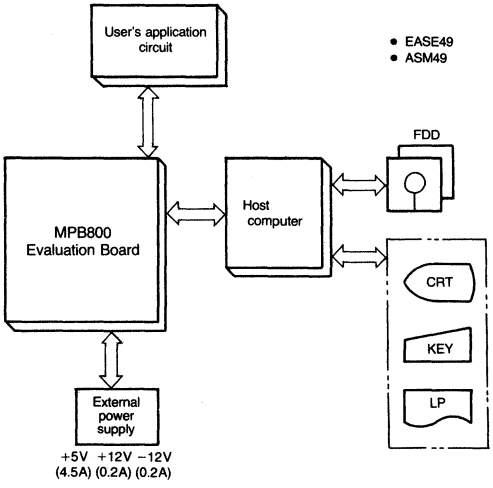
- Communication interface MSM82C51A
- Driver/receiver SN75188N, SN75189N

Transmission format

- 110 to 9600 bps (Switchable)
- 8 bits, 2 stop bits, non-parity

SYSTEM CONFIGURATION

The EASE80C49 Program Development Support System consists of the EASE80C49 Emulator (a high performance program emulator which includes the EASE49 Host Monitor, the MPB800 Evaluation Board, and the host computer) and the ASM-49 Assembler.
With the MPB800 Evaluation Board connected online to the host CP/M computer (such as an if800) equippped with an RS232C interface, the system covers all operations from assembly of the source program through to program evaluation and debugging.



Note: CP/M is a registered trademark of Digital Research Inc. (U.S.A.)
When the EASE80C49 is used on the MSM80C50, the MSM80C39 evaluation chip on the emulation board is to be replaced by MSM80C40.
EASE49 and ASM-49 for PC-DOS, MS-DOS are optional softwares.

ASM-49 CROSS-ASSEMBLER

The ASM-49 is a floppy disk based high-performance assembler.
It translates source files generated on disk by using an editor, thereby generating object code files and assembly list files.

FEATURES

1. Free descriptive format source files
2. 35 types of powerful pseudo-instructions available for assembler control purposes
3. Macro definition ability
4. 11 types of pseudo-instructions to enable conditional assembly
5. Assembly repetition processing (loop processing)
6. Determination of variable values in the source program by input from a console during assembly, and linking to existing source programs. (INPUT and LINK pseudo-instructions)
7. Capacity to describe up to 13 types of operators in the source program operand column

LIST OF PSEUDO-INSTRUCTIONS

Pseudo-instructions	Function
ORG Expression	Location counter value defined as nnn. (Expression=nnn)
DS Expression	Reserves memory area for n number of bytes. The first and last byte values may be changed. Use ORG $+n to prevent change of values. (Expression=n)
DW Expression	16-bit data definition
DB Expression	8-bit data or ASCII character string definition
EQU	Assignment of operand value to name
SET	Same as EQU pseudo-instruction, but with redefinition capacity
IF Expression	Evaluate expression value, and skip to next ENDIF, END or EDF (end of file) if result is zero. Assemble the next instruction if result is not zero.
NIF Expression	Evaluate expression value, and skip to next ENDIF, END or EDF (end of file) if result is not zero.
END Expression	End of assembly
MACRO	Macro definition
GO TO Label	Branch the subsequent assembly to the destination indicated in the label
REPT Expression	Denotes repetition block. Assembly is executed the number of repetitions indicated by the expression value.
REPND	Definition of end of repetition block
LIST	Pass option invalidated, and output of full assembly list
NOLST	Inhibition of assembly list outputs apart from error messages
TITLE	Allocation of title at head of assembly list page

EASE80C49 EMULATOR

Consisting of a host computer and the MPB800 Program Evaluation Board, the EASE80C49 Emulator supports a wide range of development debugging operations efficiently and effectively. MSM80C49RS and MSM80C50RS application programs can be debugged without user application circuits just as easily as completed systems.

FEATURES

1. **Real-time tracing function does not effect execution time**
   The MPB800 Evaluation Board incorporates a real-time trace area for 1024 machine cycles. When tracing for a particular user program area address has been set, the port, program counter, and probe data status are traced each time the specified address is executed.

2. **Execution time measurement**
   The user program execution time can be measured by using the cycle counter on the MPB800 Evaluation Board. (Max. 16,777, 215 cycles)
   This counter can also be used as a pass counter to indicate the number of times a specified address is executed.

3. **4096-byte user program area**
   A 4096-byte static RAM area in the MPB800 Evaluation Board is used as a user program area, and user programs can be loaded/saved from the host computer into this area. Furthermore, with an EPROM programmer included on this board the user program area contents can be written into EPROMs, and EPROM contents can be read.

4. **Ample break functions**
   The emulator can suspend (break) program execution by any of the following eight break conditions.
   a) BB (Breakpoint Break)
      Break upon execution of specified address. (Any address may be specified.)
   b) EB (External Break)
      Break by application of external break signal.
   c) CO (Cycle counter Overflow break)
      Break when overflow of cycle counter occurs.
   d) TF (Trace Full break)
      Break when 4097 tracings executed.
   e) PM (Probe Match break)
      Break when probe data matches set data
   f) IF (Instruction Fetch)
      Break when machine code is fetched.
   g) MX (MovX)
      Break by MOVX instruction.
   h) ER (Error code)
      Break if incorrect machine code is fetched.

5. **Extensive range of debugging commands**
   In addition to display/updating of all register, port, and RAM contents, the following commands enable all debugging operations to be executed efficiently.
   a) User program execution

   Input format:
   STP number of instructions, start address

   Input of this command results in the specified number of instructions in the user program being executed from the specified address (start address).

   Input format:
   G [, start address]

   Input of this command results in the user program being executed from the specified address (start address). If no start address is specified, execution is started from the address indicated by the program counter at that time.

   b) Use of floppy disks

   Input format:
   LOA filename

   Input of this command enables the specified file contents (user program) to be loaded into the MPB800 Evaluation Board code memory which serves as the MSM80C49RS/MSM80C50RS masked ROM area.
   The same contents can also be loaded into the utility buffer by using the LDU command.

   Input format:
   SAV, start address, end address, filename

   Input of this command enables the contents of the specified range of code memory to be saved at the specified filename.

   c) Instruction excuted bit memory
   The EASE80C49 Emulator includes an instruction executed bit memory which indicates which user program addresses have been executed. Program flow can be determined by examination of the contents of this memory.

6. Easy to remember command format

The MPB800 Emulator debugging commands consist of command mnemonics followed by parameters (address or mnemonic).

Command mnemonic configuration

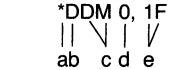
a) First character

Denotes the function to be executed by the emulator.

b) Second and following characters

Name of function to be executed by the emulator–that is, MSM80C39 microcomputer or MPB800 Evaluation Board register, memory, or port name (abbreviation).

Example:



a: denotes wait for command input from emulator
b: denotes display of contents of specified register/memory/board
c: data memory (equivalent MSM80C49RS/MSM80C50RS RAM area) designation
d: start address of contents to be displayed
e: end address of contents to be displayed
f: carriage return denoting end of command input

MAJOR MEMORY OPERATIONS

1. Code memory

The code memory is an 4096 × 8-bit RAM area which corresponds to the MSM80C49RS/MSM80C50RS program area (masked ROM) where user programs are stored.

2. Data memory

The data memory is a 128 × 8-bit RAM area corresponding to the data memory (RAM) on the MSM80C49RS chip. The data memory area for MSM80C50RS is 256 × 8-bits.

3. Utility buffer

The utility buffer is a 4096 × 8-bit RAM area used in temporary saving from the emulator when a load, save, or transfer command is entered.

4. Break point bit memory

The break point bit memory is also an 4096 × 1-bit RAM area which has addresses identical to the code memory. If a particular bit in this memory is "1", program execution is suspended immediately after the code memory contents corresponding to that bit are executed.

5. Trace enable bit memory

The trace enable bit memory is another 4096 × 1-bit RAM area which has addresses identical to the code memory. If a particular bit in this memory is "1", port and probe data is stored in the trace memory when the code memory contents corresponding to that bit are executed.

6. Sync output bit memory

The sync output bit memory is another 4096 × 1-bit RAM area which has addresses identical to the code memory. When the code memory contents are executed, the corresponding bits in the sync output memory are checked, and if a "1" bit is found, an output sync signal (active LOW) is passed to the probe terminal.

7. Cycle counter enable bit

The cycle counter enable bit memory is also another 4096 × 1-bit RAM area which has addresses identical to the code memory. When the code memory contents are executed, the corresponding bits in the cycle counter enable bit memory are checked, and if a "1" bit is found, the cycle counter is counted up in step with that machine cycle.

8. Probe comparison register/probe mask register

The conditions for generating a break by input data from the probe (probe match break) can be changed by altering these two settings.

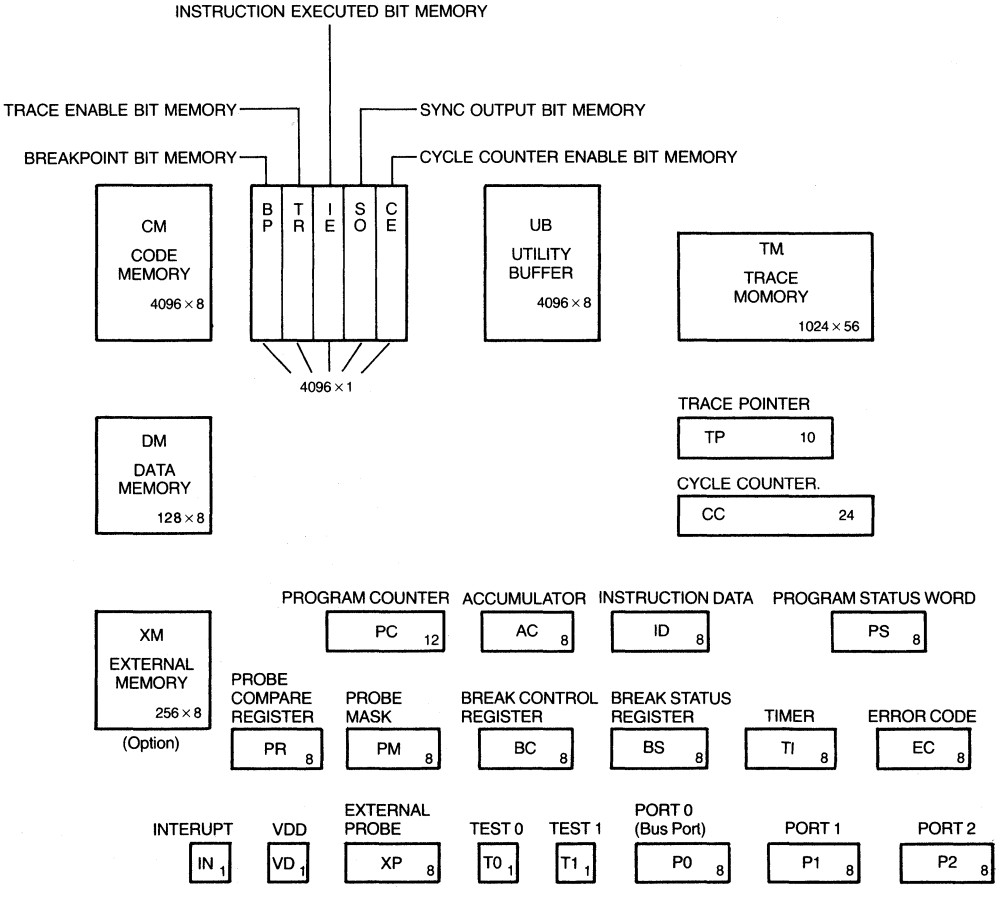
9. Trace memory

The trace memory is a 1024 × 56-bit RAM area used to store traced data.

The trace instruction is given by the trace enable bit.

EASE80C49 EMULATOR MEMORY CONFIGURATION

The EASE80C49 emulator which consists of a host CP/M® computer and the MPB800 Evaluation Board has been designed for efficient debugging of MSM80C49RS and MSM80C50RS application programs.
The various sections of the EASE80C49 Emulator which can be used are outlined in the following diagram. The sections enclosed in boxes represent the register and memory areas which can be accessed from the host computer keyboard, and which can be displayed/update by emulator command.



Emulator memory configuration

EXECUTION EXAMPLE

In this sample program, 1 byte is read from port 1, and a parity check is executed on the first 7 bits. The MSB (most significant bit) is set to "1" if parity is odd parity, and reset to "0" if even parity. The result is then passed to port 2. Note that errors have been intentionally included in this program for demonstration purposes. Various emulator commands are used to correct the errors.
It is assumed that the program has already been assembled, and the object code has been stored in hexadecimal format in a file called "PARITY.HEX" in drive A.

Program list

```
0000                ;  * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
0000                ;
0000                ;
0000                ;
0000                ;       This program reads a byte from Port 1 and checks
0000                ;     bits 0 thru 6 for parity. If Parity is odd, then
0000                ;     bit 7 in the byte will be set and the result will
0000                ;     be output to Port 2. If the Parity is even, bit 7
0000                ;     will be reset before the byte is output to Port 2.
0000                ;  * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
0000                ;
0000                ;                       **************
0000                ;                       *   MAIN PROGRAM   *
0000                ;                       **************
0000                ;
0000                ;
0000                PARITY      ORG         0
0000   09                       IN          A, P1               ; Input byte from P1
0001   537F                     ANL         A, #7FH             ; Mask out bit 7
0003   140C                     CALL        PAR0
0005   E609                     JNC         PAR1                ; Jump if even parity
0007   4308                     ORL         A, #8H              ; Set bit 7
0009   3A          PAR1        OUTL        P2, A               ; Output byte to P2
000A   040A        WAIT        JMP         WAIT
000C                ;                       **************
000C                ;                       *   SUBROUTINES    *
000C                ;                       **************
000C   BA08        PAR0        MOV         R2, #8              ; Set loop count
000E   97                       CLR         C                   ; Initialize Carry Flag
000F   1212        PAR2        JB          0, PAR3             ; Jump if bit 0 = 1
0011   A7                       CPL         C                   ; Toggle Carry Flag
0012   77          PAR3        RR          A
0013   EA0F                     DJNZ        R2, PAR2            ; Loop 8 times and ret
0015   83                       RET
0016                            END
0000   ERRORS
SYMBOL TABLE
PAR0        000C  PARI        0009  PAR2        000F  PAR3        0012
```

```
A>EASE49
MPB800 80C49 EMULATOR, VER 4.3
COPYRIGHT JUNE.1982, OKI SEMICONDUCTOR
*FCM 0 FFF FF ——— Code memory is filled with FFH
*LOA PARITY ——— Contents of the "PARITY.HEX" file are transferred to the code memory
*DCM 0 1F ——— Code memory location contents from address 0H thru 1FH are displayed

000=09 53 7F 14 0C E6 09 43 08 3A 04 0A BA 08 97 12
010=12 A7 77 EA 0F 83 FF FF FF FF FF FF FF FF FF FF
*EBP A ——— Break point at address AH is set to "1".
*EBP 17 FFF ——— Break point bits from address 17H thru FFFH are st to "1"

*DBP 0 2F ——— The break point bit status from address 0H thru 2FH is displayed

000=0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
010=0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1
020=1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

*CP1 0 ——— Port 1 is set to "0" (even parity)
*CP2 0 ——— Port 2 is set to "0"
*G 0 ——— Start of emulation from address 0H

EMULATION BEGUN
PC=00A TERMINATED P0=FF P1=00 P2=00 T0=1 T1=1 XP=FF AC=00 R0=
R1=CA ——— Port 2 is correct in respect to even parity

*CP1 1 ——— Port 1 is set to "1" (odd parity)
*SDF -AL P1 P2 AC ——— Dump format is changed
*G 0 ——— Start of emulation from address 0H
EMULATION BEGUN
PC=00A TERMINATED P1=01 P2=09 AC=09 ——— Port 2 is not correct in respect to odd parity
*CP1 3 ——— Port 1 is set to "3" (even parity)
*CP2 0 ——— Port 2 is set to "0"
*G 0 ——— Start of emulation from address 0H
EMULATION BEGUN
PC=00A TERMINAT
*CP1 4 ——— Port 1 is set to "4" (odd parity)
*CP2 0 ——— Port 2 is set to "0"
*G 0 ——— Start of emulation from address 0H
```

```
EMULATIOM BEGUN
PC=00A  TERMINATED P1=04 P2=0C AC=0C — Port 2 is not correct in respect to odd parity

*DTM -10 10 ——— Trace contents of the last ten cycles are displayed
PC=015 RET          P0=FF P1=04 P2=00 T0=1 T1=1 IN=1 XP=FF TP=0058
PC=005 JNC  09      P0=FF P1=04 P2=00 T0=1 T1=1 IN=1 XP=FF TP=0060
POSSIBLE INTERRUPT ····· SEE APPENDIX3
PC=007 ORL  A,#08 P0=FF P1=04 P2=00 T0=1 T1=1 IN=1 XP=FF TP=0062
PC=009 OUTL P2,A  P0=FF P1=04 P2=0C T0=1 T1=1 IN=1 XP=FF TP=0064
PC=00A JMP  00A   P0=FF P1=04 P2=0C T0=1 T1=1 IN=1 XP=FF TP=0066
*EBP 5          The break point bit at address 5H is set to "1"
*CP2 0          Port 2 is set to "0"
*G 0            Start of emulation from address 0H
EMULATION BEGUN
PC=007 TERMINATED     P1=04 P2=00 AC=04

*STP 5 ———————— Five steps (instructions) are executed
PC=007 ORL  A,#08  P1=04 P2=00 AC=0C
PC=009 OUTL P2,A   P1=04 P2=0C AC=0C
PC=00A JMP  00A    P1=04 P2=0C AC=0C
PC=00A JMP  00A    P1=04 P2=0C AC=0C
PC=00A JMP  00A    P1=04 P2=0C AC=0C
*DCM, 0 F ———————— Code memory location contents from address 0H thru FH are displayed

000=09 53 7F 14 0C E6 09 43 08 3A 04 0A BA 08 97 12
*CCM 8 80 ———————— Code memory location contents at address 8H are changed to "80H"
*DCM 0 F ———————— Code memory location contents from address 0H thru FH are displayed
000=09 53 7F 14 0C E6 09 43 80 3A 04 0A BA 08 97 12
*RBP 5 ———————— The break point bit at address 5H is set to "0" (reset)
*CP1 1 ———————— Port 1 is set to "1" (execution of odd parity again)
*CP2 0
*G 0
EMULATION BEGUN
PC=00A TERMINATED P1=01 P2=81 AC=81—
*CP1 7F ———————— Port 1 is set to "7F" (execution of odd parity again)
*CP2 0
*G 0
EMULATION BEGUN
PC=00A TERMINATED  P1=7F P2=FF AC=FF— Port 2 is correct in respect to odd parity
*SAV 0 1F PRTY1— Program with file name of "PRTY1.HEX" is stored
*FCM 0 FFF FF—— Code memory is filled with FFH
*VPR 0 FFF 0 —— EPROM is inserted into an EPROM socket and a blank check is
                executed
*LOA PRTY1 ———— Program with file name of "PRTY.HEX" is transferred to code memory
*PPR 0 1F 0 ——— Transferred contents are written in the EPROM
*VPR 0 1F 0 ——— EPROM contents are compared with code memory contents
*↑C ————————— No error message (EPROM and code memory contents are identical)
A>  ————————— Return to CP/M(R) control
```

LIST OF DEBUGGING COMMANDS

Characters which can be used	
AaBbCcDdEeFfGgHhIiJjKkLlMmNnOoPpQqRrSsTtUuVvWwXxYyZz 1234567890.,@+− Cntl/C Cntl/P Cntl/Q Cntl/R Cntl/T Escape Space	
Display Commands	
1. DAC	Display Accumulator
2. DBP, 000 [,FFF]	Display Breakpoint Bit(s) Status
3. DBS	Display Break Status
4. DCC	Display Cycle Counter
5. DCE, 000 [,FFF]	Display Cycle Counter Enable Bit(s) Status
6. DCM, 000 [,FFF]	Display Code Memory
7. DDM, 00 [,7F]	Display Data Memory
8. DIE, 000 [,FFF]	Display Instruction Executed Bit(s) Status
9. DPO	Display Port 0 (Bus Port)
10. DP1	Display Port 1
11. DP2	Display Port 2
12. DPC	Display Program Counter
13. DPS	Display Program Status Word, Test 0, Test 1, Interrupt Pin
14. DRG	Display Registers R0 thru R7
15. DSK	Display Stack
16. DSO, 000 [,FFF]	Display Sync Output Bit Memory
17. DTI	Display Timer
18. DTM, XXX, YYYY	Move Trace Pointer (XXXX) and display Trace Memory (YYYY)
19. DTP	Display Trace Pointer
20. DTR, 000 [,FFF]	Display Trace Enable Bit(s) Status
21. DVD	Display Vdd Pin
22. DXM, 00 [,FF]	Display External Memory
23. DXP	Display External Probe Byte, Probe Mask and Probe Compare Register
Change Commands	
1. CAC, FF	Change Accumulator
2. CCC, ZZZZZZZZ	Change Cycle Counterr (ZZZZZZZZ is a positive or negative decimal number)
3. CCM, FFF, FF	Change Code Memory
4. CDM, 7F, FF	Change Data Memory
5. CPO, FF	Change Port 0 (Bus Port)
6. CP1, FF	Change Port 1
7. CP2, FF	Change Port 2
8. CPC, FFF	Change Program Counter
9. CPM, BBBBBBBB	Change Probe Mask (BBBBBBBB is a binary number)
10. CPR, BBBBBBBB	Change Probe Compare Register (BBBBBBBB is a binary number)
11. CPS, FF	Change Program Status Word
12. CTI, FF	Change Timer
13. CXM, FF, FF	Change External Memory
Fill Commands	
1. FCM, 000, FFF, FF	Fill Code Memory
2. FDM, 00, 7F, FF	Fill Data Memory
3. FXM, 00, FF, FF	Fill External Memory
Enable Commands	
1. EBP, 000 [,FFF]	Enable Breakpoint Bit(s)
2. ECE, 000 [,FFF]	Enable Cycle Counter Bit(s)
3. ESO, 000 [,FFF]	Enable Sync Output Bit(s)
4. ETR, 000 [,FFF]	Enable Trace Bit(s)

Reset Commands	
1. RBP, 000 [,FFF]	Reset Breakpoint Bit(s)
2. RCE, 000 [,FFF]	Reset Cycle Counter Enable Bit(s)
3. RIE, 000 [,FFF]	Reset Instruction Executed Bit(s)
4. RSO, 000 [,FFF]	Reset Sync Output Bit(s)
5. RTR, 0000 [,FFF]	Reset Trace Enable Bit(s)
6. RES	Reinitialize MPB800 System
Utility & Disk Input/Output Commands	
1. CUB, FFF, FF	Change Utility Buffer
2. DUB, 000 [,FFF]	Display Utility Buffer
3. FUB, 000, FFF, FF	Fill Utility Buffer
4. TCM, 000, FFF	Transfer Code Memory into Utility Buffer
5. TUB, 000, FFF	Transfer Utility Buffer into Code Memory
6. TST	Test
7. LOA, filename	Load Program into Code Memory
8. LDU, Filename	Load Program into Utility Buffer
9. SAV, 000, FFF, filename	Save Code Memory Program
10. SVU, 000, FFF, filename	Save Utility Buffer Program
EPROM Writer Commands	
1. PPR, 000, FFF, FF	Pogram Code Memory onto EPROM
2. VPR, 000, FFF, FF	Verify EPROM with Code Memory
3. TPR, 000, FFF	Transfer EPROM into Code Memory
Emulation Commands	
1. G [,HHH]	Begin real time emulation. HHH is the start address (hex).
2. STP [,EEEE] [,HHH]	Step emulation, where EEEE is the decimal number of instruction to execute, and HHH is the start address (hex).
3. SBC, mnemonic [,mnemonic]	Set/Reset Break Control Bit
4. SDF, mnemonic [,mnemonic]	Set/Reset Dump Format
Command Line Editing & Keyboard Operation	
1. Rubout	Delete the last character entered
2. Cntl/C	Return to CP/M
3. Cntl/P	Copy all subsequent console output to the currently assigned list device. Output is sent to both the list device untill the next Cnt/P is typed.
4. Cntl/Q	Continue normal display
5. Cntl/R	Echoe current input line
6. Cntl/S	Stop display
7. Escape	Abort any command inprogress
8. @	Repeat last command

EASE80C49 8-BIT 1-CHIP MICROCONTROLLER PROGRAM DEVELOPMENT SUPPORT SYSTEM

Category	Model	Title
Hardware	MPB800	8-bit 1-chip microcontroller evaluation board
Software	EASE-49	Floppy disk based emulator*
	ASM-49	Floppy disk based assembler*
Manual	TM-800	Program Development Support System–User's Manual
Accessories	TCU-800	User application system connecting cables
	TCS-8	Host CP/M(R) computer connecting cables (for if800 model 20/30)
	TCP-8	Power supply cables (+5V,3.5A) (+12V, 0.2A) (−12V,0.2A)
	TCX-1	External probe emulation purposes

* Available under following operating system
- CP/M-80 (ver 2.0 or later)
- MS-DOS (ver2.11 or later)
- PC-DOS (ver2.11 or later)

7

EASE80C51mkII PROGRAM DEVELOPMENT SYSTEM

The EASE80C51mkII Program Development System is a high-performance dedicated system featuring Oki's exclusive technology, and which has been specifically designed for rapid and efficient program development of the Oki MSM83C154/80C51 8-bit single-chip microcontroller.

SYSTEM CONFIGURATION

The EASE80C51mkII Program Development System consists of the EASE80C51mkII Emulator (a high performance program emulator which includes the EASE Host Monitor, the EASE8051mkII Emulation Kit, and the host CP/M® or PC-DOS computer) and the ASM51 Assembler (a powerful assembler operated in CP/M® or PC-DOS. With the EASE80C51mkII Emulation Kit connected online to the host CP/M computer (such as the if800) equipped with an RS232C interface, the system covers all operations from assembly of the source program through to program evaluation and debugging.

HOST COMPUTER REQUIREMENTS

1. Operating system is one of the follows.
   (1) CP/M®-80 (ver 2.0 or later)
   memory capacity of the host computer must be sufficient to run at least 52K CP/M.
   (2) MS-DOS® (ver2.11 or later)
   (3) PC-DOS® (ver2.11 or later)
2. At least one RS232C communication port is implemented.
3. Data transfer is performed through RS232C communication port using BDOS function call 6 on condition that console device is assigned to TTY:.

EMULATOR DATA TRANSFER

ICs used

- Communication interface MSM82C51A
- Driver/receiver SN75188N, SN75189N

Transmission format

- 300 to 19200 bps (Switchable)
- 8 bits, 2 stop bits, non-parity
- Asynchronous

7

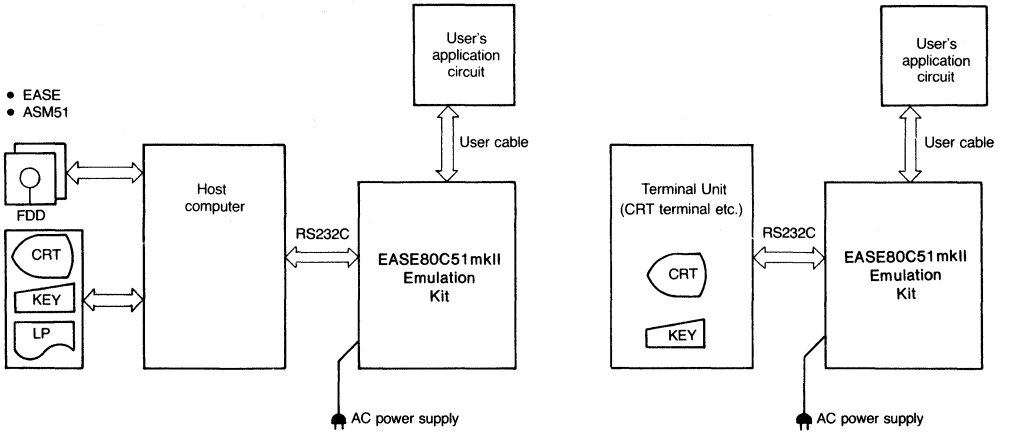


Fig. 1 System configuration

Note 1. CP/M is a registered trademark of Digital Research Inc. (U.S.A.)
2. The AC power supply can be switched to 90–132V and 180V–264V at a terminal inside the emulation kit.
3. PC-DOS stands for IBM personal computer DOS.

7

ASM51 ASSEMBLER

The ASM51 is a floppy disk based high-performance assembler.
This assembler is used to translate source files generated on disk by using an editor (available on the market), thereby generating object files (Intel HEX format), assemble list files, cross-reference list files, and symbol list files in the specified devices.

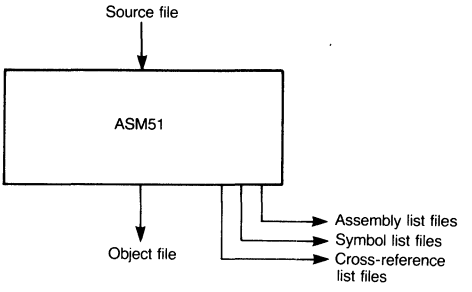
OUTPUT FILES

1. Object files (Intel HEX format)
2. Assembly list files
3. Cross-reference list files
4. Symbolic list files

FEATURES

1. Free descriptive format source files
2. Capacity to describe up to ten types of operators in the source program operand column
3. Ability to specify the number of characters per line and the number of lines per page in assemble list files
4. 24 powerful pseudo-instructions
5. Control of assemble list file outputs by LIST or NOLIST pseudo-instruction
6. Output file and output device can be specified when the assembler is started.

ASM51 ASSEMBLER FUNCTIONAL BLOCK DIAGRAM



PART OF PSEUDO-INSTRUCTIONS LIST

Pseudo-instructions	Function
EQU	Assignment of operand value to name
SET	Same as EQU pseudo-instruction, but with redefinition capacity
ORG	Setting of program start address
END	Indication of end of program
JMP	Automatic change to relative, internal 2K page, or all pages jump instruction after checking branch destination
CALL	Automatic chang to relative, internal 2K page, or all pages call instruction after checking branch destination
RADIX	Radix changed to 2, 8, 10, or 16 depending on value of operand
DB	8-bit data or ASCII character definition
DW	16-bit data definition
DS	Reserves memory area for specified number of bytes
NSE	Setting of 0 in the 4 lower order bits of the assembler location counter, and addition of 16. The NOP instruction is assigned to blank areas where no machine language instruction has been assigned.
DATE	Insertion of date in assemble list title
EJECT	Assemble list page feed operation
TITLE	Insertion of assemble list title
LIST	Designation of assemble list output
NOLIS	Zuhibition of assemble list output

EASE80C51mkII EMULATOR

Connected to the host computer via an RSS232C interface, the EASE80C51mkII Emulator supports a wide range of development debugging operations efficiently and effectively.
MSM83C154/80C51 application programs can be debugged without user application circuits just as easily as completed systems.

FEATURES

1. Real-time emulation
Real-time emulation without insertion of a wait state is possible because of the MSM83C154E evachip.

2. Execution time measurement
The user program execution time can be measured with the cycle counter in the Emulation Kit. (Max. 4,294,836,225 cycles)
This cycle counter start/stop is effected according to the cycle counter start address/stop address set by the command.

3. Mass storage user program area
This emulator provides a 64K-byte RAM area, (which is the entire address space of MSM83C154/80C51) as the code memory (user program area). So, the largest program for MSM83C154/80C51 can be fully loaded in this RAM area.
It is also possible to assign 4K-byte units of program memory area to the RAM area on the emulation kit or the ROM on the user's application circuit by using the mapping command.
This emulator further provides the EPROM programmer to enable the user program area contents to be written to the EPROM or the EPROM contents to be read.
(EPROMs supported: Intel 2732, 2732A, 2764 or 27128 or equivalent:)

4. Ample break functions
The emulator can suspend (break) program execution by any of the following break conditions. All conditions can be set and cancelled as desired.

a) Breakpoint break
Break upon execution of the address where a break point has been set. (Any address to be specified.)

b) Address break
Break execution of the address specified when the emulation command input was applied. It is also possible to specify a break after the specified address has been executed n times.

c) Power down break
Break occured when evachip going to the power down mode.

d) Break by external forced break signal
Break by input of low level break signal from outside via the attached probe cable.

e) Break by trace memory overflow
Break occurs when the trace memory is filled up with trace data.

f) Break by cycle counter overflow

g) Break by internal RAM/SFR area contents
Break occurs when the contents of the specified RAM/SFR in the MSM83C154E matched with the specified contents the specified number of times upon execution of the program at the specified address.

5. Comprehensive real-time trace functions
This emulator has the following two real time trace areas not affecting the execution time.
1. Trace memory
This is the memory to trace (store) the status of the ports, carry flag and accumulator of the MSM83C154E when an instruction in the program memory area is executed. (Up to 2048 machine cycles)
Tracing is instructed in three ways as shown below.

a) To start tracing each time the specified address is executed.

b) To start tracing upon execution of a special instruction (ACALL, AJMP, LCALL, LJMP, RET, RETI, PUSH or POP)

c) To cause tracing by trace start/stop bit

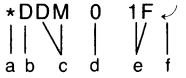
2. Flash trace memory
The memory to trace status of the whole internal RAM or SFR area in the MSM83C154E, when the instruction at the address specified by the emulator command (debug command) being executed.
The contents of the internal RAM or SFR area can be stored up to 16 times in this memory.

6. Easy-to-use emulator commands
The emulator command (debug command) of this emulator consists of the command mnemonic and succeeding parameter(s) (address or mnemonic).
Command mnemonic structure

a) First letter
Represents the function to be performed by the emulator.

b) Second letter and on
Represent MSM83C154E evachip (or EASE80C51mkII emulation kit) register, memory, or port name.

Example:



a: Indication of waiting for command input from emulator
b: Instruction of displaying contents of specified object
c: Instruction of data memory (equivalent to internal RAM area in MSM83C154E)
d: Start address of contents to be displayed
e: End address of contents to be displayed
f: Carriage return indicating end of command input

This emulator provides various emulator commands not only for display/modification of the contents of each register and port of the special evaluation chip MSM83C154E but also for efficient debugging operation.

Input format:
STP number-of-instructions start-address

Inputting the above command causes the user program to be executed for as much as the specified number of instructions from the specified addresses (start address).
It is possible to modify the display format to be easier to read by specifying the object of contents display by the SSF command.

Input format:
G start-address, parameter
Parameter input format
(1) Break-address, . . . , break-address
(2) Break-address (n)
(3) Break-address RAM ram-address (byte-n)
(4) Break-address sfr-mnemonic (byte-n)

Inputting the command as shown in (1) causes the user program to be executed from the specified address (start address), and breaks the program execution upon execution of any of the specified break addresses.
Inputting the command as shown in (2) causes the user program to be executed from the specified start address, and breaks the program execution when it has passed the specified break address in times.
Inputting the command, as shown in (3), causes the program to be executed from the specified address (start address). When the user program at the specified break address is executed, the contents of the specified address (ram address) of the internal RAM in the MSM83C154E are compared with the specified contents (byte) and the program execution is broken if they agree the specified number of times (n).
Inputting the command, as shown in (4), causes the user program to be executed from the specified address (start address). When the user program at the specified break address is executed, the contents of the specified SFR in the MSM83C154E are checked. If the contents agree with the specified value (byte) the specified number of times (n), breaking occurs.

Input format:
(1) LOD filename
(2) SAV filename start-address end-address
(3) VER filename start-address end-address

Inputting the command, as shown in (1), enables the contents (user program) of the specified file on the host computer to be loaded into the code memory on the EASE80C51mkII emulation kit corresponding to the program memory area (user program area) of the MSM83C154/80C51.
Inputting the command, as shown in (2), enables the block of code in the specified range of the code memory to be saved using the specified file name.
Inputting the command, as shown in (3), enables the block of code in the specified file to be compared with the block of code in the code memory.

Input format:
(1) DIAG filename
(2) M
(3) @

Inputting the command, as shown in (1), enables the command in the specified file to be executed automatically. Use of the PAUSE command in combination enables execution of the command in the file to be suspended temporarily.
Inputting the command, as shown in (2), enables the command line defined by the MAC command to be executed.
Inputting the command, as shown in (3), enables the last command to be executed again.

Input format:
(1) LIST filename
(2) NLST

If LIST command is entered, the emulator create the CP/M file on the host computer and writes into it any characters which are output to the console until entering the NLST command.
Use of the above "DIAG" command in combination enable the debugging work to be executed and stored in the file automatically.

Input format:
S tm-mnemonic data number

Using the S command, you can search out the trace information in the trace memory.
Where the "tm-mnemonic" is the element of the trace information in the trace memory that you wish to search, and "data" is a value of that element. "number" is the time of an agreement.
For example, Entering "S" PO 2 3", the emulator searches through the trace memory from the top and finds the position where the trace PO data is equal to 2, three times, and displays the trace information at that position.

DISK ACCESS COMMANDS

1.	LOD	[dr:] filename	Load Program into Code Memory
2.	SAV	[dr:] filename [address adress]	Save Code Memory Program
3.	VER	[dr:] filename [address adress]	Verify File with Code Memory
4.	DIAG	[dr:] filename	Execute command file
5.	LIST	[dr:] filename	List console Output into Disk file
6.	NLST		End listing
7.	HELP		Display HELP File
8.	FDD	dr: ... dr:	Set Disk Drive

EPROM PROGRAMMER COMMANDS

1.	PPR address address [address]	Program Code Memory into EPROM
2.	TPR address address [address]	Transfer EPROM into Code Memory
3.	VPR address address [address]	Verify EPROM with Code Memory
4.	TYPE mnemonic	Set EPROM type

ASSEMBLE COMMANDS

1.	ASM	address	Assemble to Code Memory
2.	DASM	address	Disassemble to Console
3.	DBLK	address [address]	Display Block Memory
4.	SPB	address [address]	Set Program Block
5.	SDB	address [address]	Set Data Block

TRACE COMMANDS

1.	DTM-number number	Display Trace Memory
2.	DFTM number number	Display Flash Trace Memory
3.	DTG	Display Trigger Mode
4.	DFA	Display Flash Trace address
5.	S mnemonic data number	Search Trace Memory data
6.	SFF mnemonic	Set Flash Trace Display format
7.	STG mnemonic	Set Trigger Mode
8.	SFA address [... address]	Set Flash Trace address
9.	RFA address [... address]	Reset Flash Trace address
10.	RTG	Reset Trigger Mode

EMULATION COMMANDS

1. STP number [number]
2. G[st-address] [,break-parameter]
   If the optional st-address if given, emulator will begin emulation from st-address.
   And if optional break-parameter is given.
   emulation will break on the first break-parameter to be satisfied.

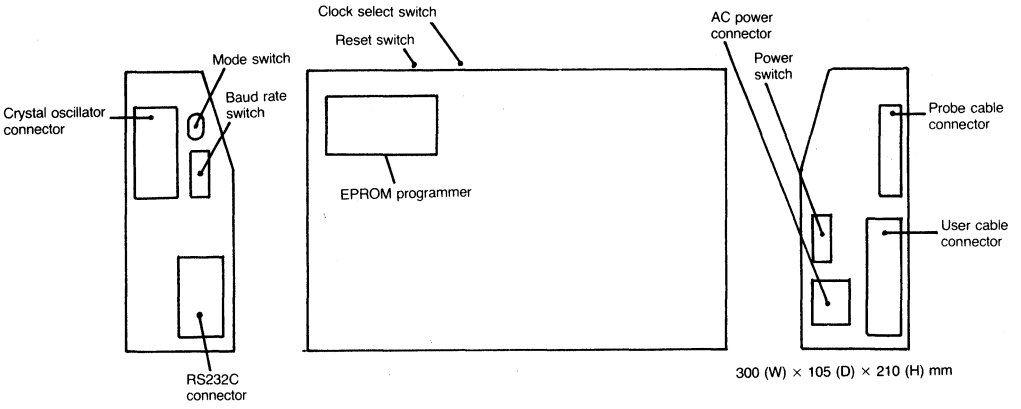
   break-parameter = break-address, ..., break-address (max. 10)
   break-address (pass count)
   break-address RAM ram-address (byte-pass count)
   break-address sfr-mnemonic (byte-pass count)
3. SSF (+/–) mnemonic [...(+/–) mnemonic]
   Set STP Command Display format

DISPLAY COMMANDS

1.	D	Display Register Flag Port
2.	Dsfr-mnemonic	Display sfr-mnemonic data
3.	DSFR mnemonic [...mnemonic]	Display mnemonic data
4.	DPC	Display Program Counter
5.	DREG	Display all Register Bank
6.	DSBUF	Display receiver data

CHANGE COMMANDS	
1. Csfr-mnemonic data	Change sfr-mnemonic
2. CSFR [mnemonic...mnemonic]	Change SFR
3. CPC data	Change Program Counter
4. CREG data	Change Register bank
5. CSBUF data	Change Transmitter data
CODE MEMORY & DATA MEMORY COMMANDS	
1. DCM address [address]	Display Code Memory
2. DDM address [address]	Display Data Memory
3. DXDM address [address]	Display External Data Memory
4. CCM address	Change Code Memory
5. CDM address	Change Data Memory
6. CXDM address	Change External Data Memory
7. FCM address address byte	Fill Code Memory with byte
8. FDM address address byte	Fill Data Memory with byte
9. FXDM address address byte	Fill External Data Memory with byte
ATTRIBUTE MEMORY COMMANDS	
1. DBP address [address]	Display Break Point Bit Memory
2. DTR address [address]	Display Trace Enable Bit Memory
3. DSO address [address]	Display Sync Output Enable Bit Memory
4. DIE address [address]	Display Instruction Executed Bit Memory
5. EBP address ... address	Enable Brak Point Bit
6. ETR address ... address	Enable Trace Enable Bit
7. ESO address ... address	Enable Sync Output Enable Bit
8. FPB address address byte	Fill Break Point Bit Memory with byte
9. FTR address address byte	Fill Trace Enable Bit Memory with byte
10. FSO address address byte	Fill Sync Output Enable Bit Memory with byte
11. RBP address ... address	Reset Break Point Bit
12. RTR address ... address	Reset Trace Enable Bit
13. RSO address ... address	Reset Sync Output Enable Bit
14. RIE	Reset Instruction Executed Bit Memory
BUFFER MEMORY COMMANDS	
1. DBUF	Display Buffer Memory
2. TBUF	Transfer Data Memory & SFR Data into Buffer Memory
3. LBUF	Load Buffer Memory into Data Memory & SFR
CYCLE COUNTER COMMANDS	
1. DCC	Display Cycle Counter
2. CCC data	Change Cycle Counter
3. TIME data	Set 1 cycle time
BREAK CONDITION & STATUS COMMANDS	
1. DBC	Display Break Condition
2. DBS	Display Break Status
3. SBC (+/−) mnemonic.. (+/−) mnemonic	Set Break
OTHER COMMANDS	
1. RES	EASE80C51mKII System initialization
2. RES E	MSM83C154E Evachip reset
3. SIO F or H	Set Emulator to I/O terminal mode
4. PAUSE	Stop command file execution and wait key-in
5. EXIT	Return Host computer's OS
6. DIR [dr:]	Display file directory
7. MAP [address]	Mapping
8. MAC	Define command execution
9. M	Execute Defined command
10. @	Repeat front command
11. M80C51	Set Emulator to the MSM80C51 checking mode
12. M83C154	Set Emulator to the MSM83C154 checking mode

EASE80C51mkII PROGRAM DEVELOPMENT SUPPORT SYSTEM for the MSM83C154/80C51 CMOS 8-BIT 1-CHIP MICROCONTROLLER



Category	Model name	Component name
Hardware	EASE80C51mkII Emulation kit	Emulation kit for MSM83C154/80C51
Software	EASE ASM 51	Host monitor for floppy disk based (Note 2) Assembler for floppy disk based (Note 2)
Manuals	TM-80C51 AM-80C51	EASE80C51mkII Emulator User's Manual ASM51 Assembler User's Manual
Accesories	TCU-80C51 TCS-n TCP-2 TCX-2	User application circuit connection cable Host computer connection cable (3) (Note 3) AC power supply cable Emulation probe cable

Note 1. Refer to the ASM51 Assembler User's Manual for assembler details.

Note 2. PCDOS 51/4 floppy disk, PC format, double-side double-density (8 sectors × 40 tracks × 2)
PC DOS stands for IBM personal computer DOS.
CP/M-80 8-inch floppy disk, IBM3740 soft-sectored format, single-side single-density.
CP/M-80 is the registered trademark of Digital Research Co.

Note 3. Two cables are used for connecting the if800 or equivalent computer to the EASE80C51mkII Emulation Kit-one to the CH1 port, and the other to the CH2 port. (Model name, TCS-2)
The third cable is for connection of an IBM PC or equivalent computer to the EASE80C51mkII Emulation Kit. This calbe is connected to the CH1 port. (Model name, TSC-3).

OPTIONAL SOFTWARE

Name	OS					Supplied software
	CP/M-80	MS-DOS	PC-DOS	VMS	UNIX	
PASM	○	○	○	*	*	Pre-Processor
MAC51	○	○	○	*	*	Macro-Assembler
RL51	○	○	○	*	*	Linker
LIB51	○	○	○	*	*	Librarian
SID51	*	○	○	*	*	Symbolic Debugger
OBJHEX	*	○	○	*	*	Object Converter

*·····Being developed

OPTIONAL HARDWARE

Type	Remarks
EASE8	Handy I/O terminal EASE8 can be used as a terminal unit for the EASE80C51mkII Emulation Kit. (Convenient portable model)

MAC51 SUPPORT SOFTWARE PACKAGE
for the
MSM80C51 CMOS 8-BIT MICROCONTROLLER

FEATURES

1. Symbolic relocatable assembly language programming for 80C51/154 microcontrollers
2. Produces Relocatable Object Code which is linkable to other 8051 Object Modules
3. Encourages modular program design for maintainability and reliability
4. Macro Assembler features conditional assembly and macro capabilities

MAC51 SUPPORT SOFTWARE PACKAGE

The following MAC51 programs are available to develop user programs.

PASM Pre-processor

PASM is used to expand macro calls, conditional assembly statements, and INCLUDE statements included in user generated source programs, thereby generating expanded source programs. A number of items of development information are inserted in these developed source programs for MAC51.

MAC51 Assembler

MAC51 converts source programs into relocatable codes to form relocatable object files (OBJ files). Print, symbol, cross reference, and error files are generated as assembly information.

LIB51 Librarian

The LIB51 program manages OBJ files for each module. Files consisting of a number of relocatable object modules (OBJ modules) generated by LIB51 are called the object library. LIB51 handles object library generation, and OBJ module addition, deletion, and upgrading.

RL51 Linker

RL51 links and relocates one or more OBJ modules to generate one absolute object module. RL51 also generates a list file consisting of symbol table and link map as link information. OBJ modules which serve as the RL51 input can be OBJ files generated by MAC51, and OBJ modules located within the object library generated by LIB51.

SID51 Symbolic debugger

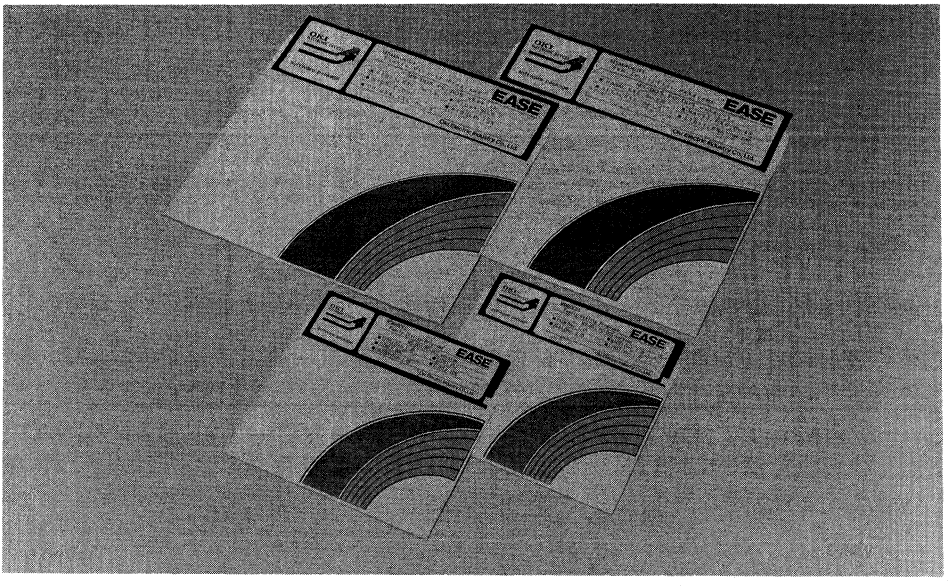
SID51 is used when a symbol using debugger is selected. Absolute object files are converted to Intellec HEX format files.

OBJHEX Converter

OBJHEX converts Object file into Intellect HEX file.

Outline of Program Development

The procedures involved in the processing from source program generation through ROM loading are described below in the sequence indicated in Figure 1-1. For further details of individual utilities, refer to the respective manuals.



1. Generation of MAC51 assembly language source program by the editor.
   Source programs can contain basic and pseudo instructions, and assembler control, macro call, conditional assembly, and INCLUDE statements.
2. When macros are used, macro definitions are generated in macro library files (with MAC extension) by the editor. Macro call statements are described within source programs.
3. Where macro call, conditional assembly statement, and INCLUDE statement descriptions are included in a source program, there are expanded by PASM to form an expanded source program.
4. Source programs or expanded source programs are assembled by MAC51 to form relocatable object files.
5. A group of relocatable object files can be managed by LIB51 together in a relocatable object library files (LIB extension). When required, object modules can be called from this object library by RL51.
6. Relocatable object files are converted to absolute object files by RL51. At this stage, one or more relocatable object files can be linked to relocatable object modules in the library file.
7. ABS modules are converted by SID51 or OBJHEX to Intellec HEX format files. With EASE80C51mkII, HEX file contents can be written into EPROM devices .

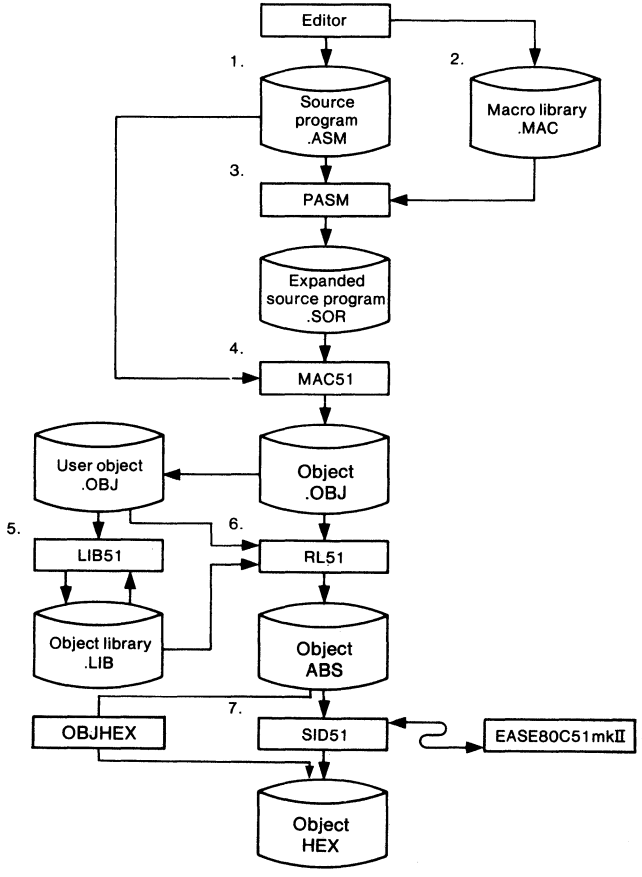


Fig. 1-1 Program development flow

MEMO

MEMO

MEMO

OKI

Oki Semiconductor

650 North Mary Avenue, Sunnyvale, CA94086, U.S.A.
Tel: (408) 720-1900 Fax: (408) 720-1918 Telex: 296687 OKI SNTA

For further information please contact:

Oki Electric Industry Co., Ltd,
Head Office Annex
10-3, Shibaura, 4-chome,
Minato-ku, Tokyo 108, Japan
Tel : 3-454-2111
Fax : 3-798-7643
Telex : J22627
Electronic Devices Group
Overseas Marketing Dept.

Oki Semiconductor
650 North Mary Avenue,
Sunnyvale, CA 94086,U.S.A.
Tel : (408) 720-1900
Fax : (408) 720-1918
Telex : 296687 OKI SNTA

Oki Electric Europe GmbH
Niederkasseler Lohweg 8,
D-4000 Düsseldorf 11,
Fed. Rep. of Germany
Tel : 211-59550
Fax : 211-591669
Telex : 853-4312 OKI-D

Oki Electronics (Hong Kong) Ltd.
16th Floor, Fairmont House,
8 Cotton Tree Drive, Hong Kong
Tel : 5-263111
Fax : 5-200102
Telex : 62459 OKIHK HX

Printed in USA

REF NO.: MCU10-4A24-58