MAGNACHIP SEMICONDUCTOR LTD.
8-BIT SINGLE-CHIP MICROCONTROLLERS
WITH EMBEDDED FLASH

HMS99C51S
HMS99C52S
HMS99C54S
HMS99C56S
HMS99C58S

User's Manual (Ver. 1.01)

MagnaChip

Revision History

Ver 1.01 (Sep, 10, 2004) this book
The company name, Hynix Semiconductor Inc. changed to MagnaChip Semiconductor Ltd.

Ver 1.0 (Dec, 01, 2003)
The first released document.

Version 1.01

Published by
MCU Application Team

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

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

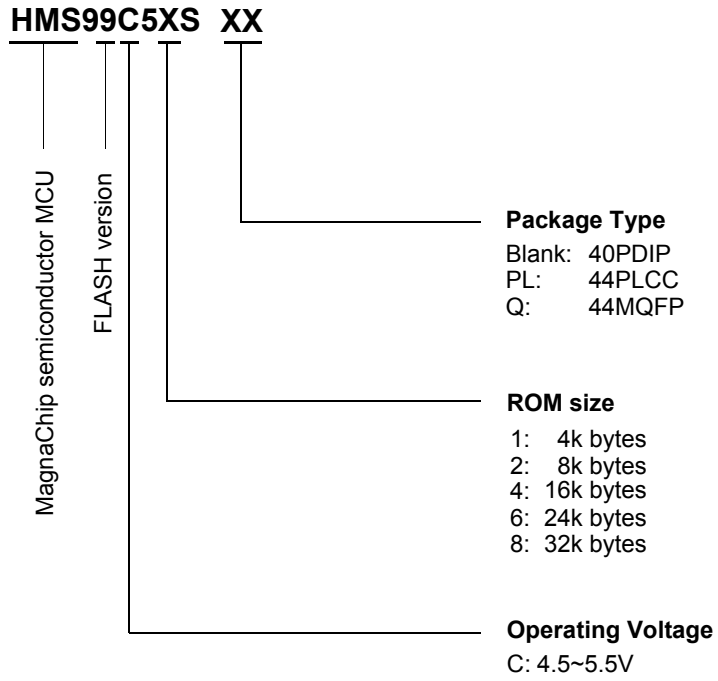
The information, diagrams and other data in this manual are correct and reliable; however, MagnaChip semiconductor is in no way responsible for any violations of patents or other rights of the third party generated by the use of this manual.

CONTENTS

Sep. 2004 Ver 1.01

Sep. 2004 Ver 1.01

DEVICE NAMING STRUCTURE



HMS99C51S/52S/54S/56S/58S SELECTION GUIDE

Operating Voltage (V)	**ROM size** (bytes)	**RAM size** (bytes)	**Device Name**	**Operating Frequency** (MHz)
	FLASH			
4.5~5.5	4K 8K 16K 24K 32K	256	HMS99C51S HMS99C52S HMS99C54S HMS99C56S HMS99C58S	40

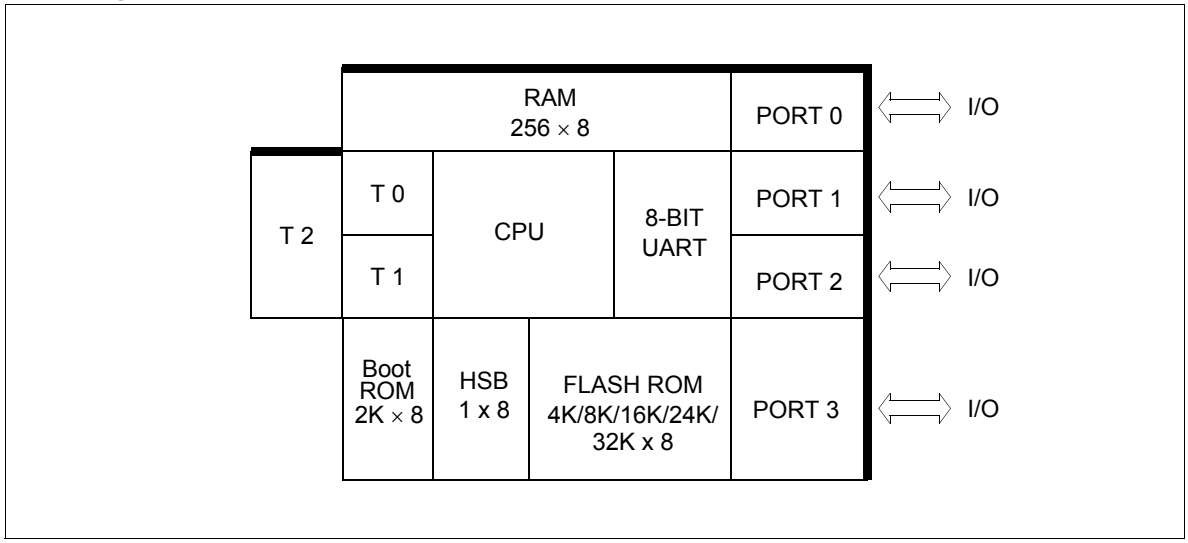
FEATURE

- Fully compatible to standard MCS-51 microcontroller
- Wide operating frequency up to 40MHz
  (for more detail, See “HMS99C51S/52S/54S/56S/58S SELECTION GUIDE” on page 1)
- X2 Speed Improvement capability (**X2 Mode : 6 clocks/machine cycle**)
  20MHz @5V (Equivalent to 40MHz @5V)
- ISP(In-System Programming) using Standard V_{CC} Power Supply
- Boot ROM Contains Low Level FLASH Programming Routines and a Default Serial Loader
- 4K/8K/16K/24K/32K bytes FLASH user program memory
  - Byte Write and Block(2K, 8K Bytes) Erase
- 2K bytes FLASH boot loader
- 1 byte Hardware Security Byte (HSB)
- 256 bytes RAM
- 64K bytes external program memory space
- 64K bytes external data memory space
- Four 8-bit ports
- Three 16-bit Timers / Counters (Timer2 with up/down counter feature)
- UART
- One clock output port
- Programmable ALE pin enable / disable (Low **EMI**)
- Six interrupt sources, two priority levels
- Power saving Idle and power down mode
- P-DIP-40, P-LCC-44, P-MQFP-44 package
- Temperature Ranges : -40°C ~ 85°C

Description

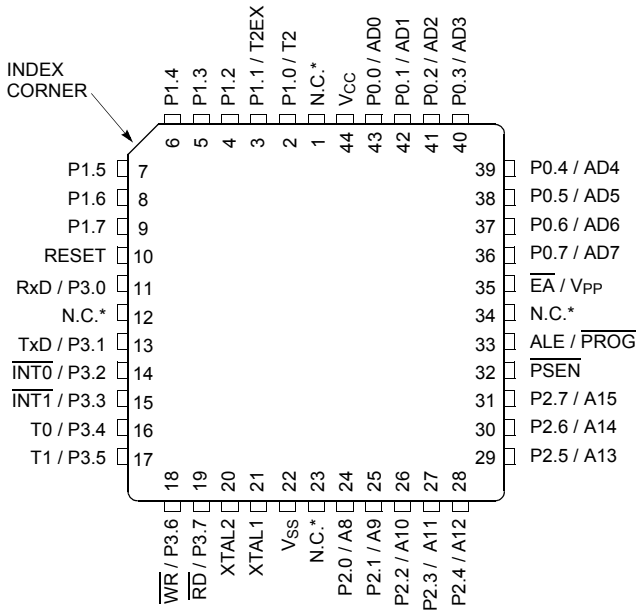
The Flash memory increases EPROM and ROM functionality with in-circuit electrical erasure and programming. It contains 4K, 8K, 16K, 24K or 32K bytes of program memory. This memory is both parallel and serial In-System Programmable(ISP). The ISP allows devices to alter their own program memory in the actual end product under software control through UART ports. A default serial loader(bootloader) program supports ISP of the Flash memory. The programming does not require external 12V programming voltage. The necessary high programming voltage is generated on-chip using the standard V_{CC} pins of the microcontroller.

Block Diagram



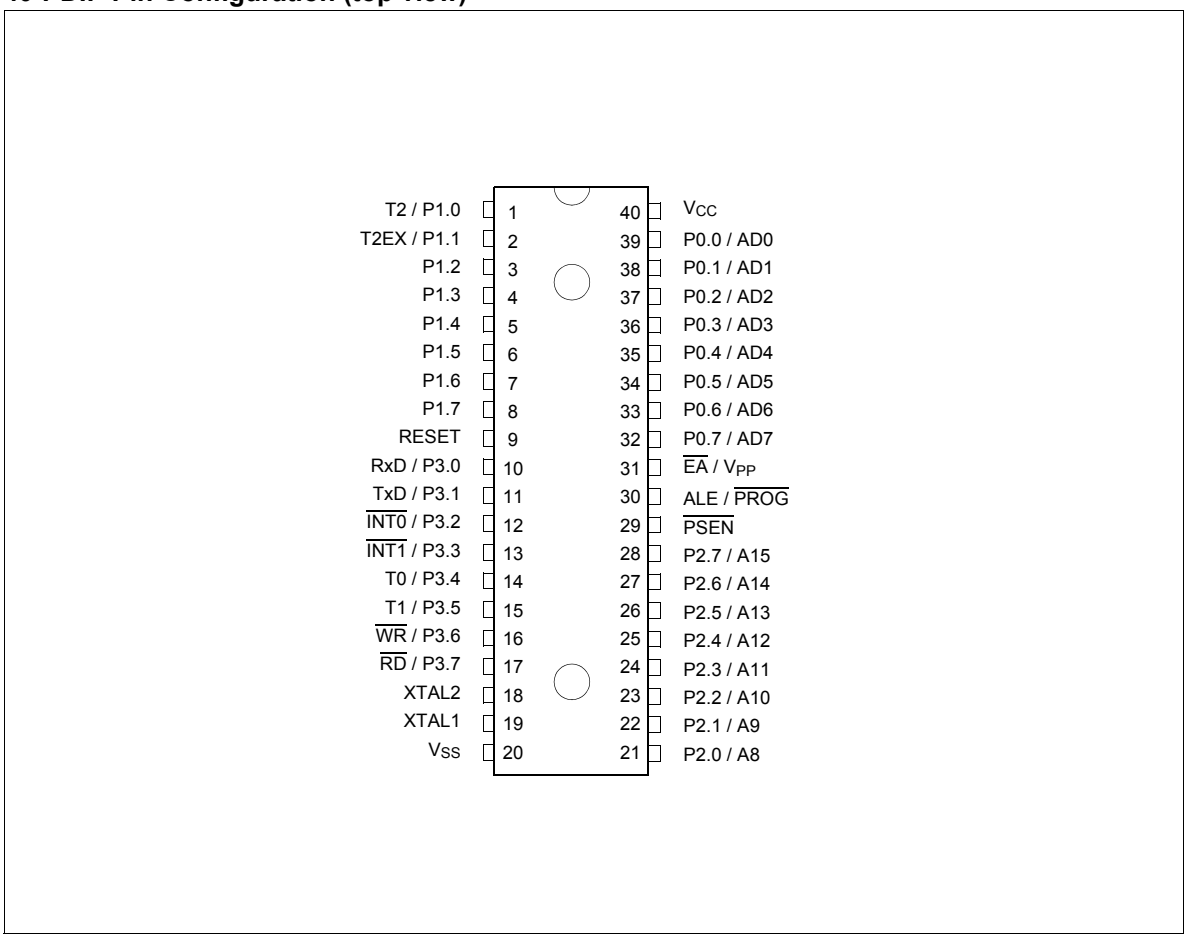
PIN CONFIGURATION

44-PLCC Pin Configuration (top view)

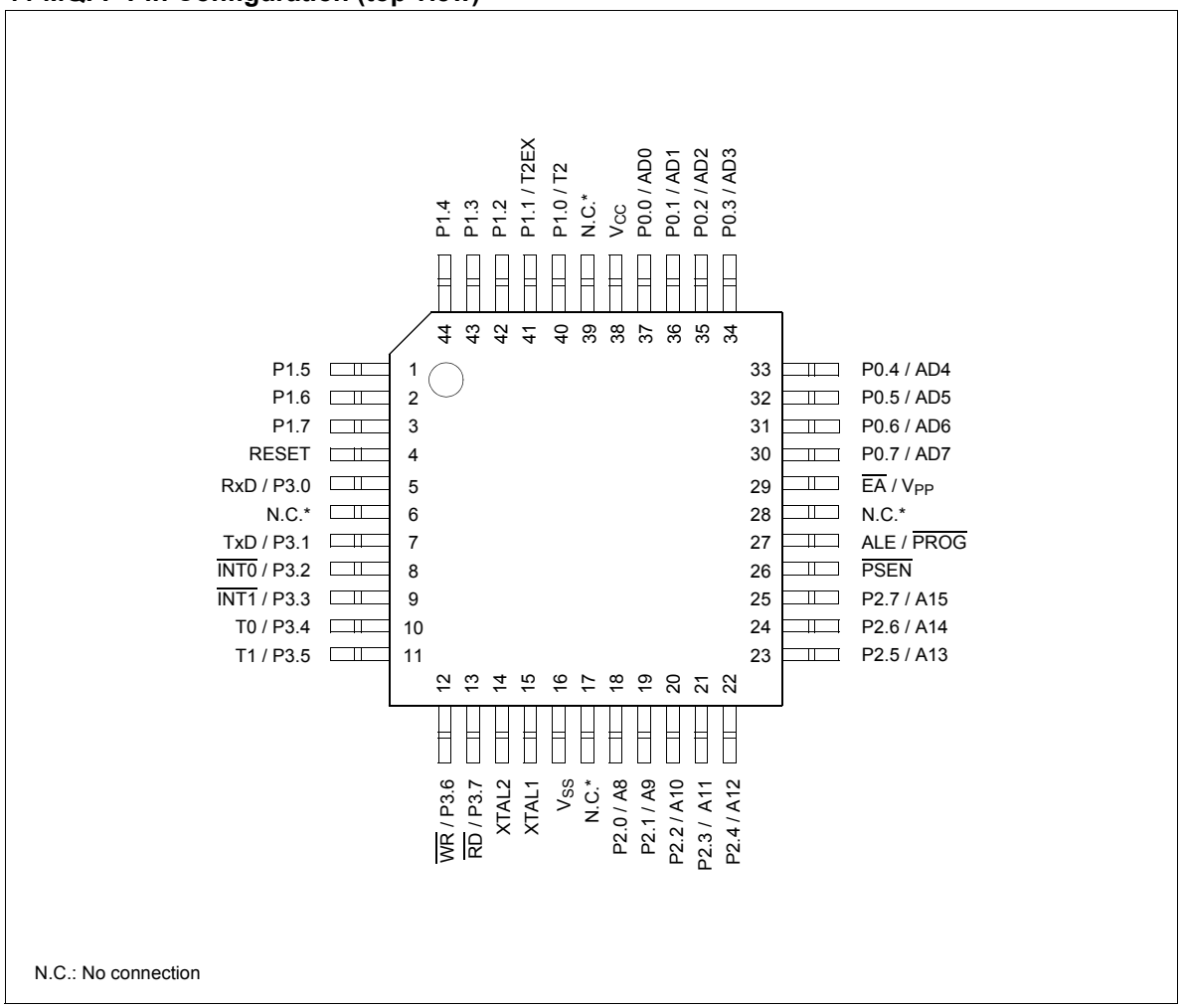


N.C.: No connection

40-PDIP Pin Configuration (top view)

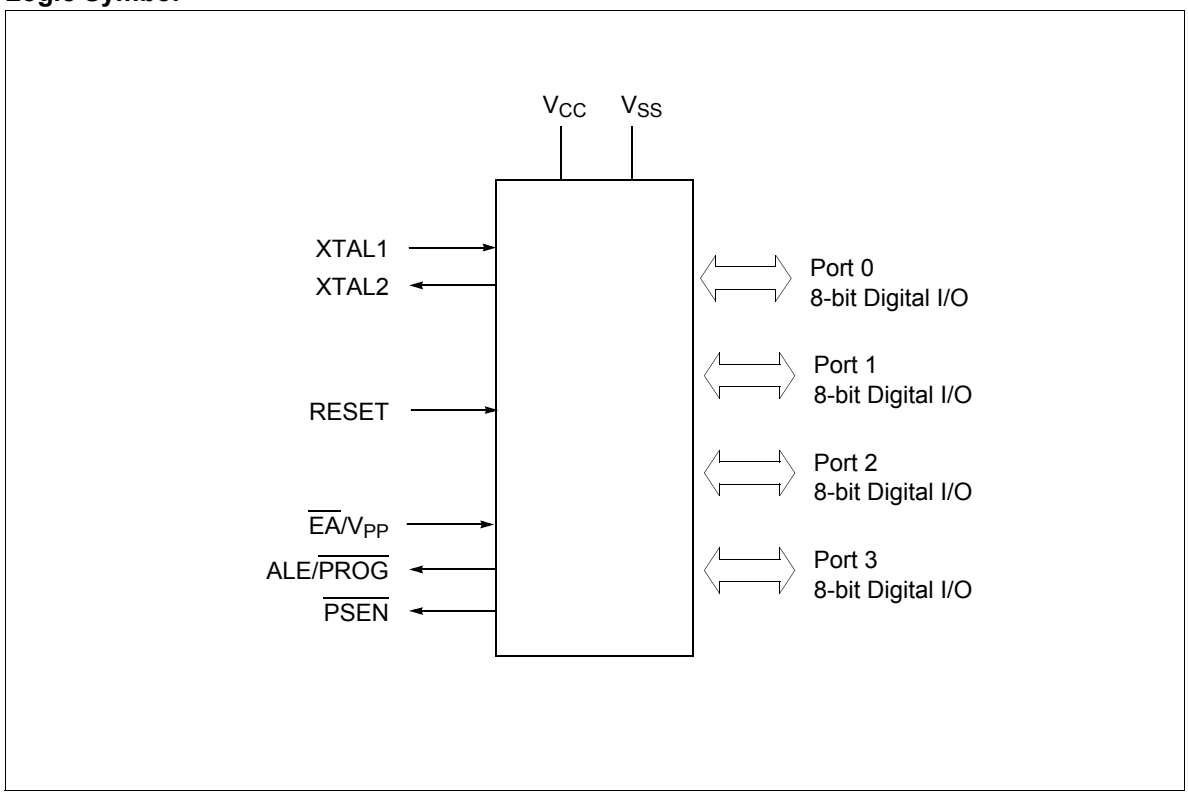


44-MQFP Pin Configuration (top view)



N.C.: No connection

Logic Symbol



PIN DEFINITIONS AND FUNCTIONS

Symbol	Pin Number			Input/ Output	Function
	PLCC-44	PDIP-40	MQFP-44		
P1.0-P1.7	2-9	1-8	40-44, 1-3	I/O	**Port1** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-up resistors and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the pulls-ups (I_{IL}, in the DC characteristics). Port1 also serves alternate functions of Timer 2 as follows. P1.0 / T2, Clock Out : Timer/counter 2 external count input, Clock Out P1.1 / T2EX :Timer/counter 2 trigger input Port1 receives the low-order address bytes during Flash programming and verifying.
	2	1	40		
	3	2	41		
P3.0-P3.7	11, 13-19	10-17	5, 7-13	I/O	**Port 3** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-up resistors and can be used as inputs. As inputs, port 3 pins that are externally pulled low will source current because of the pulls-ups (I_{IL}, in the DC characteristics). Port 3 also serves the special features of the 80C51 family, as listed below.
	11	10	5		P3.0 / RxD receiver data input (asynchronous) or data input/output(synchronous) of serial interface 0
	13	11	7		P3.1 / TxD transmitter data output (asynchronous) or clock output (synchronous) of the serial interface 0
	14	12	8		P3.2 /$\overline{INT0}$ interrupt 0 input/timer 0 gate control
	15	13	9		P3.3 / $\overline{INT1}$ interrupt 1 input/timer 1 gate control
	16	14	10		P3.4 /T0 counter 0 input
	17	15	11		P3.5 /$\overline{T1}$ counter 1 input
	18	16	12		P3.6 / $\overline{WR}$ the write control signal latches the data byte from port 0 into the external data memory
	19	17	13		P3.7 /$\overline{RD}$ the read control signal enables the external data memory to port 0
XTAL2	20	18	14	O	**XTAL2** Output of the inverting oscillator amplifier.

Symbol	Pin Number			Input/ Output	Function
	PLCC-44	PDIP-40	MQFP-44		
XTAL1	21	19	15	I	**XTAL1** Input to the inverting oscillator amplifier and input to the internal clock generator circuits. To drive the device from an external clock source, XTAL1 should be driven, while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is divided down by a divide-by-two flip-flop. Minimum and maximum high and low times as well as rise and fall times specified in the AC characteristics must be observed.
P2.0-P2.7	24-31	21-28	18-25	I/O	**Port 2** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-up resistors and can be used as inputs. As inputs, port 2 pins that are externally pulled low will source current because of the pulls-ups (I_{IL}, in the DC characteristics). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application it uses strong internal pull-ups when outputting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @Ri), port 2 emits the contents of the P2 special function register. Some Port 2 pins receive the high-order address bits during flash program, verify, and erase operations.
$\overline{\text{PSEN}}$	32	29	26	O	**The $\overline{\text{Program Store Enable}}$** The read strobe to external program memory when the device is executing code from the external program memory. $\overline{\text{PSEN}}$ is activated twice each machine cycle, except that two $\overline{\text{PSEN}}$ activations are skipped during each access to external data memory. $\overline{\text{PSEN}}$ is not activated during fetches from internal program memory.
RESET	10	9	4	I	**RESET** A high level on this pin for two machine cycles while the oscillator is running resets the device. The port pins will be driven to their reset condition when a minimum V_{IH} voltage is applied whether the oscillator is running or not. An internal diffused resistor to V_{SS} permits power-on reset using only an external capacitor to V_{CC}.

Symbol	Pin Number			Input/ Output	Function
	PLCC-44	PDIP-40	MQFP-44		
ALE / $\overline{PROG}$	33	30	27	O	**The Address Latch Enable / $\overline{\text{Program pulse}}$** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. This pin is also the program pulse input ($\overline{PROG}$) during EPROM programming. If desired, ALE operation can be disabled by setting bit 0 of SFR location $8E_H$. With this bit set, the pin is weakly pulled high. The ALE disable feature will be terminated by reset. Setting the ALE-disable bit has no affect if the microcontroller is in external execution mode.
$\overline{EA}$ / V_{PP}	35	31	29	I	External Access Enable / Program Supply Voltage $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000_H to $FFFF_H$. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than its internal memory size. This pin also receives the 12.75V programming supply voltage (V_{PP}) during EPROM programming. Note; however, that if any of the Lock bits are programmed, $\overline{EA}$ will be internally latched on reset.
P0.0-P0.7	36-43	32-39	30-37	I/O	**Port 0** Port 0 is an 8-bit open-drain bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application it uses strong internal pull-ups when emitting 1s. Port 0 also receives and outputs the code bytes during program and verification respectively in the GMS99X5X. External pull-up resistors are required during program verification.
V_{SS}	22	20	16	-	**Circuit ground potential**
V_{CC}	44	40	38	-	**Supply terminal** for all operating modes
N.C.	1,12 23,34	-	6,17 28,39	-	**No connection**

FUNCTIONAL DESCRIPTION

The HMS99C51S/52S/54S/56S/58S are fully compatible to the standard 8051 microcontroller family.

It is compatible with the general 8051 family, while maintaining all architectural and operational characteristics of the general 8051 family.

Figure 1 shows a block diagram of the HMS99C51S/52S/54S/56S/58S.

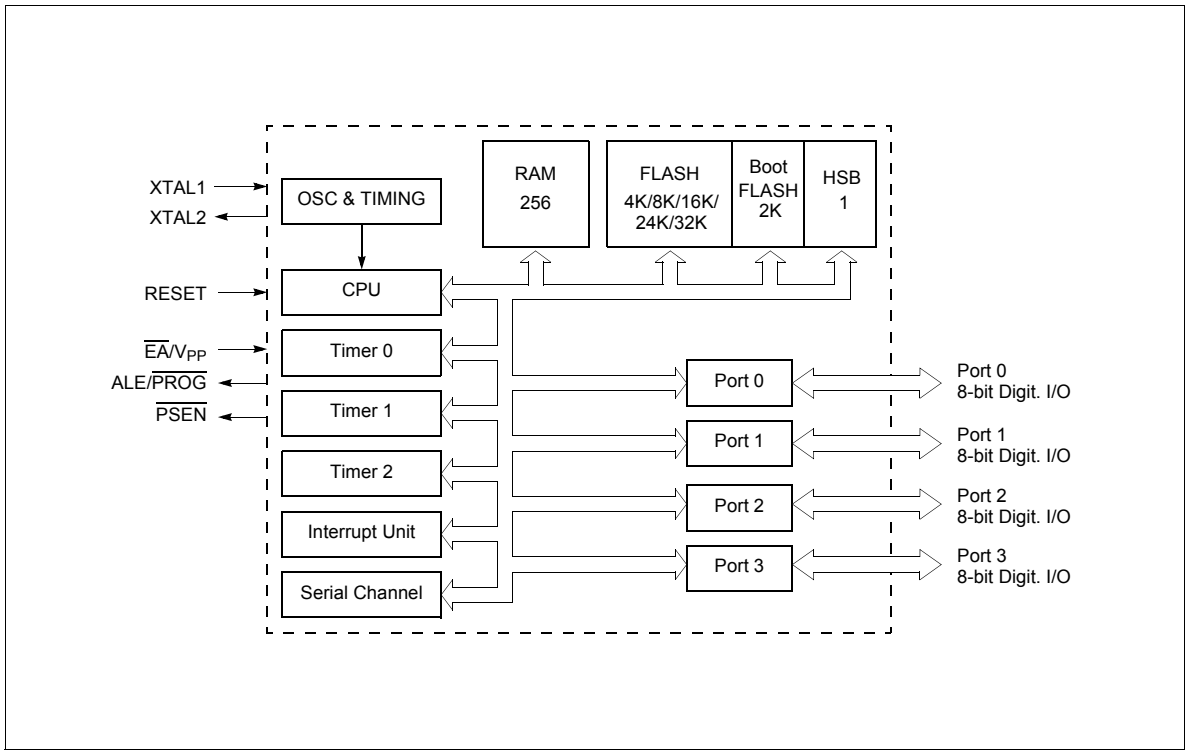


Figure 1. Block Diagram of the HMS99C51S/52S/54S/56S/58S

CPU

The HMS99C51S/52S/54S/56S/58S are efficient both as a controller and as an arithmetic processor. It has extensive facilities for binary and BCD arithmetic and excels in its bit-handling capabilities. Efficient use of program memory results from an instruction set consisting of 44% one-byte, 41% two-byte, and 15% three-byte instructions. With a 12 MHz crystal, 58% of the instructions are executed in 1.0μs (40MHz: 300ns).

Special Function Register PSW

	MSB							LSB	
Bit No.	7	6	5	4	3	2	1	0	
Addr. $D0_H$	**CY**	**AC**	**F0**	**RS1**	**RS0**	**OV**	**F1**	**P**	PSW

Bit	Function
CY	**Carry Flag**
AC	**Auxiliary Carry Flag** (for BCD operations)
F0	**General Purpose Flag**
RS1 ***RS0*** *0* *0* *0* *1* *1* *0* *1* *1*	**Register Bank select control bits** Bank 0 selected, data address 00_H - 07_H Bank 1 selected, data address 08_H - $0F_H$ Bank 2 selected, data address 10_H - 17_H Bank 3 selected, data address 18_H - $1F_H$
OV	**Overflow Flag**
F1	**General Purpose Flag**
P	**Parity Flag** Set/cleared by hardware each instruction cycle to indicate an odd/even number of "one" bits in the accumulator, i.e. even parity.

Reset value of PSW is 00_H.

SPECIAL FUNCTION REGISTERS

All registers, except the program counter and the four general purpose register banks, reside in the special function register area.

The 28 special function registers (SFR) include pointers and registers that provide an interface between the CPU and the other on-chip peripherals. There are also 128 directly addressable bits within the SFR area.

All SFRs are listed in Table 1, Table 2, and Table 3.

In Table 1 they are organized in numeric order of their addresses. In Table 2 they are organized in groups which refer to the functional blocks of the HMS99C51S/52S/54S/56S/58S. Table 3 illustrates the contents of the SFRs

Address	Register	Contents after Reset	Address	Register	Contents after Reset
80_H	**P0** [1]	**FF_H**	**88_H**	**TCON** [1]	**00_H**
81_H	SP	07_H	89_H	TMOD	00_H
82_H	DPL	00_H	$8A_H$	TL0	00_H
83_H	DPH	00_H	$8B_H$	TL1	00_H
84_H	reserved	XX_H [2]	$8C_H$	TH0	00_H
85_H	reserved	XX_H [2]	$8D_H$	TH1	00_H
86_H	reserved	XX_H [2]	$8E_H$	AUXR0	XX_H [2]
87_H	PCON	$0XXX0000_B$ [2]	$8F_H$	CKCON	$XXXXXXX0_B$ [2]
90_H	**P1** [1]	**FF_H**	**98_H**	**SCON** [1]	**00_H**
91_H	reserved	00_H	99_H	SBUF	XX_H [2]
92_H	reserved	XX_H [2]	$9A_H$	reserved	XX_H [2]
93_H	reserved	XX_H [2]	$9B_H$	reserved	XX_H [2]
94_H	reserved	XX_H [2]	$9C_H$	reserved	XX_H [2]
95_H	reserved	XX_H [2]	$9D_H$	reserved	XX_H [2]
96_H	reserved	XX_H [2]	$9E_H$	reserved	XX_H [2]
97_H	reserved	XX_H [2]	$9F_H$	reserved	XX_H [2]
$A0_H$	**P2** [1]	**FF_H**	**$A8_H$**	**IE** [1]	**$0X000000_B$** [2]
$A1_H$	reserved	XX_H [2]	$A9_H$	reserved	XX_H [2]
$A2_H$	reserved	XX_H [2]	AA_H	reserved	XX_H [2]
$A3_H$	reserved	XX_H [2]	AB_H	reserved	XX_H [2]
$A4_H$	reserved	XX_H [2]	AC_H	reserved	XX_H [2]
$A5_H$	reserved	XX_H [2]	AD_H	reserved	XX_H [2]
$A6_H$	reserved	XX_H [2]	AE_H	reserved	XX_H [2]
$A7_H$	reserved	XX_H [2]	AF_H	reserved	XX_H [2]
$B0_H$	**P3** [1]	**FF_H**	**$B8_H$**	**IP** [1]	**$XX000000_B$** [2]
$B1_H$	reserved	XX_H [2]	$B9_H$	reserved	XX_H [2]
$B2_H$	reserved	XX_H [2]	BA_H	reserved	XX_H [2]
$B3_H$	reserved	XX_H [2]	BB_H	reserved	XX_H [2]
$B4_H$	reserved	XX_H [2]	BC_H	reserved	XX_H [2]
$B5_H$	reserved	XX_H [2]	BD_H	reserved	XX [2]
$B6_H$	reserved	XX_H [2]	BE_H	reserved	XX_H [2]
$B7_H$	reserved	XX_H [2]	BF_H	reserved	XX_H [2]

Table 1. Special Function Registers in Numeric Order of their Addresses (cont'd)

Address	Register	Contents after Reset	Address	Register	Contents after Reset
$C0_H$	reserved	**XX_H**	**$C8_H$**	**T2CON** [1]	**00_H**
$C1_H$	reserved	XX_H [2]	$C9_H$	T2MOD	$XXXXXX00_B$ [2]
$C2_H$	reserved	XX_H [2]	CA_H	RC2L [1]	00_H
$C3_H$	reserved	XX_H [2]	CB_H	RC2H [1]	00_H
$C4_H$	reserved	XX_H [2]	CC_H	TL2 [1]	00_H
$C5_H$	reserved	XX_H [2]	CD_H	TH2 [1]	00_H
$C6_H$	reserved	XX_H [2]	CE_H	reserved	XX_H [2]
$C7_H$	reserved	XX_H [2]	CF_H	reserved	XX_H [2]
$D0_H$	**PSW** [1]	**FF_H**	**$D8_H$**	reserved	XX_H [2]
$D1_H$	**FCON** [3]	$XXXX0000_B$ [2]	$D9_H$	reserved	XX_H [2]
$D2_H$	reserved	XX_H [2]	DA_H	reserved	XX_H [2]
$D3_H$	reserved	XX_H [2]	DB_H	reserved	XX_H [2]
$D4_H$	reserved	XX_H [2]	DC_H	reserved	XX_H [2]
$D5_H$	reserved	XX_H [2]	DD_H	reserved	XX_H [2]
$D6_H$	reserved	XX_H [2]	DE_H	reserved	XX_H [2]
$D7_H$	reserved	XX_H [2]	DF_H	reserved	XX_H [2]
$E0_H$	**ACC** [1]	**00_H**	**$E8_H$**	reserved	XX_H [2]
$E1_H$	reserved	XX_H [2]	$E9_H$	reserved	XX_H [2]
$E2_H$	reserved	XX_H [2]	EA_H	reserved	XX_H [2]
$E3_H$	reserved	XX_H [2]	EB_H	reserved	XX_H [2]
$E4_H$	reserved	XX_H [2]	EC_H	reserved	XX_H [2]
$E5_H$	reserved	XX_H [2]	ED_H	reserved	XX_H [2]
$E6_H$	reserved	XX_H [2]	EE_H	reserved	XX_H [2]
$E7_H$	reserved	XX_H [2]	EF_H	reserved	XX_H [2]
$F0_H$	**B** [1]	**00_H**	**$F8_H$**	reserved	XX_H [2]
$F1_H$	reserved	XX_H [2]	$F9_H$	reserved	XX_H [2]
$F2_H$	reserved	XX_H [2]	FA_H	reserved	XX_H [2]
$F3_H$	reserved	XX_H [2]	FB_H	reserved	XX_H [2]
$F4_H$	reserved	XX_H [2]	FC_H	reserved	XX_H [2]
$F5_H$	reserved	XX_H [2]	FD_H	reserved	XX_H [2]
$F6_H$	reserved	XX_H [2]	FE_H	reserved	XX_H [2]
$F7_H$	reserved	XX_H [2]	FF_H	reserved	XX_H [2]

Table 1. Special Function Registers in Numeric Order of their Addresses (cont'd)

1) Bit-addressable Special Function Register.

2) X means that the value is indeterminate and the location is reserved.

3) FCON access is reserved for the ISP software.

Block	Symbol	Name	Address	Contents after Reset
CPU	ACC	Accumulator	**$E0_H$** 1)	00_H
	B	B-Register	**$F0_H$** 1)	00_H
	DPH	Data Pointer, High Byte	83_H	00_H
	DPL	Data Pointer, Low Byte	82_H	00_H
	PSW	Program Status Word Register	**$D0_H$** 1)	00_H
	SP	Stack Pointer	81_H	07_H
Interrupt System	IE	Interrupt Enable Register	**$A8_H$** 1)	$0X000000_B$ 2)
	IP	Interrupt Priority Register	**$B8_H$** 1)	$XX000000_B$ 2)
Ports	P0	Port 0	**80_H** 1)	FF_H
	P1	Port 1	**90_H** 1)	FF_H
	P2	Port 2	**$A0_H$** 1)	FF_H
	P3	Port 3	**$B0_H$** 1)	FF_H
Serial Channels	PCON 3)	Power Control Register	87_H	$0XXX0000_B$ 2)
	SBUF	Serial Channel Buffer Reg.	99_H	XX_H 2)
	SCON	Serial Channel 0 Control Reg.	**98_H** 1)	00_H
Timer 0/ Timer 1	TCON	Timer 0/1 Control Register	**88_H** 1)	00_H
	TH0	Timer 0, High Byte	$8C_H$	00_H
	TH1	Timer 1, High Byte	$8D_H$	00_H
	TL0	Timer 0, Low Byte	$8A_H$	00_H
	TL1	Timer 1, Low Byte	$8B_H$	00_H
	TMOD	Timer Mode Register	89_H	00_H
Timer 2	T2CON	Timer 2 Control Register	**$C8_H$** 1)	00_H
	T2MOD	Timer 2 Mode Register	$C9_H$	00_H
	RC2H	Timer 2 Reload Capture Reg., High Byte	CB_H	00_H
	RC2L	Timer 2 Reload Capture Reg., Low Byte	CA_H	00_H
	TH2	Timer 2, High Byte	CD_H	00_H
	TL2	Timer 2, Low Byte	CC_H	00_H
	AUXR0	Aux. Register 0	$8E_H$	$XXXXXXX0_B$ 2)
Power Saving Modes	PCON 3)	Power Control Register	87_H	$0XXX0000_B$ 2)
FLASH	FCON 4)	Flash Control Register	$D1_H$	$XXXX0000_B$ 2)

Table 2. Special Function Registers - Functional Blocks

1) Bit-addressable Special Function register

2) X means that the value is indeterminate and the location is reserved

3) This special function register is listed repeatedly since some bit of it also belong to other functional blocks

4) This special function register is reserved for the ISP software.

Address	Register	Bit 7	6	5	4	3	2	1	0
80_H	P0								
81_H	SP								
82_H	DPL								
83_H	DPH								
87_H	PCON	SMOD	-	-	-	GF1	GF0	PDE	IDLE
88_H	TCON	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
89_H	TMOD	GATE	C/$\overline{T}$	M1	MT	GATE	C/$\overline{T}$	M1	M0
$8A_H$	TL0								
$8B_H$	TL1								
$8C_H$	TH0								
$8D_H$	TH1								
$8E_H$	AUXR0	-	-	-	-	-	-	-	A0
$8F_H$	CKCON	-	-	-	-	-	-	-	X2
90_H	P1								
98_H	SCON	SM0	SM1	SM2	REN	TB8	RB8	TI	RI
99_H	SBUF								
$A0_H$	P2								
$A8_H$	IE	EA	-	ET2	ES	ET1	EX1	ET0	EX0
$B0_H$	P3								
$B8_H$	IP	-	-	PT2	PS	PT1	PX1	PT0	PX0

Table 3. Contents of SFRs, SFRs in Numeric Order

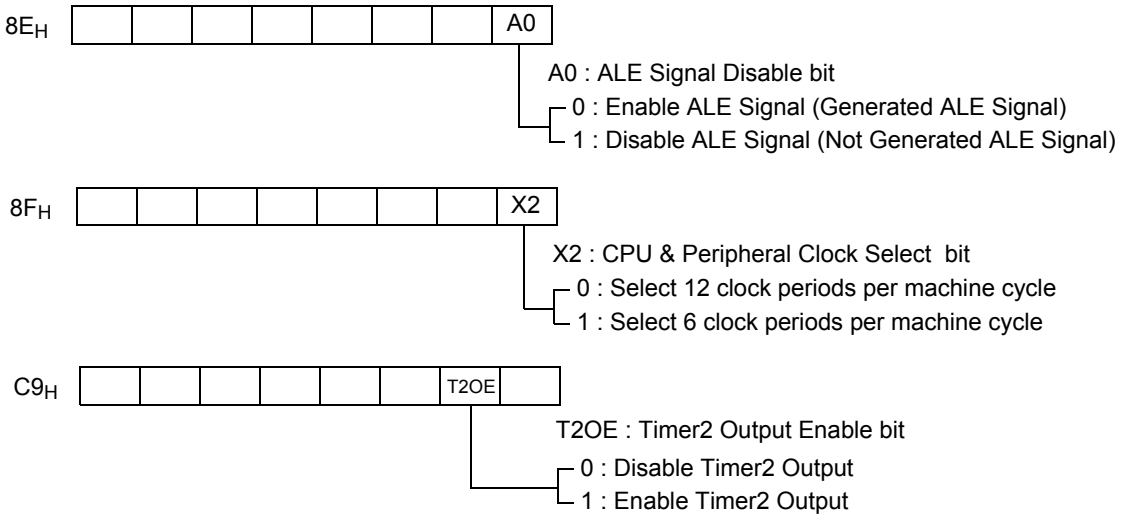
SFR bit and byte addressable

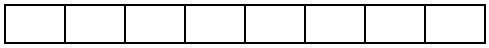
SFR not bit addressable

- : this bit location is reserved

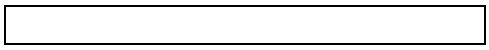
Address	Register	Bit 7	6	5	4	3	2	1	0
$C8_H$	T2CON	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$
$C9_H$	T2MOD	-	-	-	-	-	-	T2OE †	DCEN
CA_H	RC2L								
CB_H	RC2H								
CC_H	TL2								
CD_H	TH2								
$D0_H$	PSW	CY	AC	F0	RS1	RS0	OV	F1	P
$D1_H$	FCON	FRSEL2	FRSEL1	FRSEL0	ERASESEL	ENBOOT	INTROM_EN	PGMSEL1	PGMSEL0
$E0_H$	ACC								
$F0_H$	B								

Table 3. Contents of SFRs, SFRs in Numeric Order (cont'd)





SFR bit and byte addressable



SFR not bit addressable

- : this bit location is reserved

X2 MODE

The HMS99C51S/52S/54S/56S/58S core needs only 6 clock periods per machine cycle in X2 mode. This feature called "X2" provides the following advantages:

- Divide frequency crystals by 2 (cheaper crystals) while keeping same CPU power.
- Save power consumption while keeping same CPU power (oscillator power saving).
- Save power consumption by dividing dynamically operating frequency by 2 in operating and idle modes.
- Increase CPU power by 2 while keeping same crystal frequency.

In order to keep the original C51 compatibility, a divider by 2 is inserted between the XTAL1 signal and the main clock input of the core (phase generator). This divider may be disabled by software.

X2 Mode Description

The clock for the whole circuit and peripheral is first divided by two before being used by the CPU core and peripherals. This allows any cyclic ratio to be accepted on XTAL1 input. In X2 mode, as this divider is bypassed, the signals on XTAL1 must have a cyclic ratio between 40 to 60%. Figure 2. shows the clock generation block diagram. X2 bit is validated on XTAL1÷2 rising edge to avoid glitches when switching from X2 to STD mode. Figure 3. shows the mode switching waveforms:

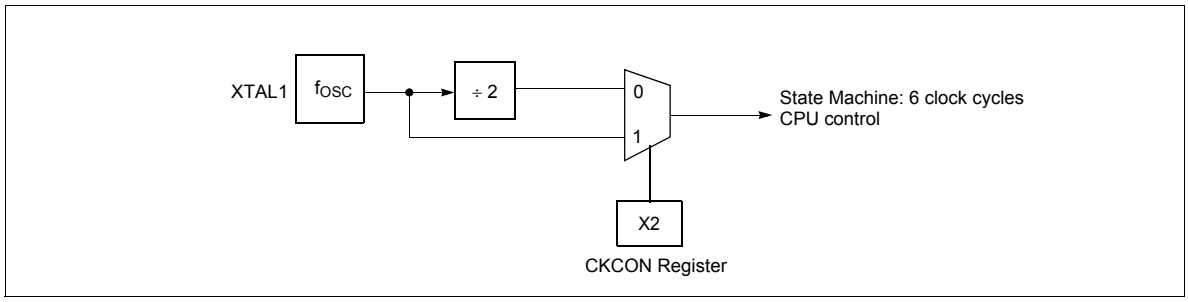


Figure 2. Clock Generation Diagram

The X2 bit in the CKCON register allows to switch from 12 clock cycles per instruction to 6 clock cycles and vice versa. At reset, the standard speed is activated (STD mode). Setting this bit activates the X2 feature(X2 mode).

CAUTION

In order to prevent any incorrect operation while operating in X2 mode, user must be aware that all peripherals using clock frequency as time reference (UART, timers) will have their time reference divided by two. For example a free running timer generating an interrupt every 30 ms will then generate an interrupt every 15 ms. UART with 2400 baud rate will have 4800 baud rate.

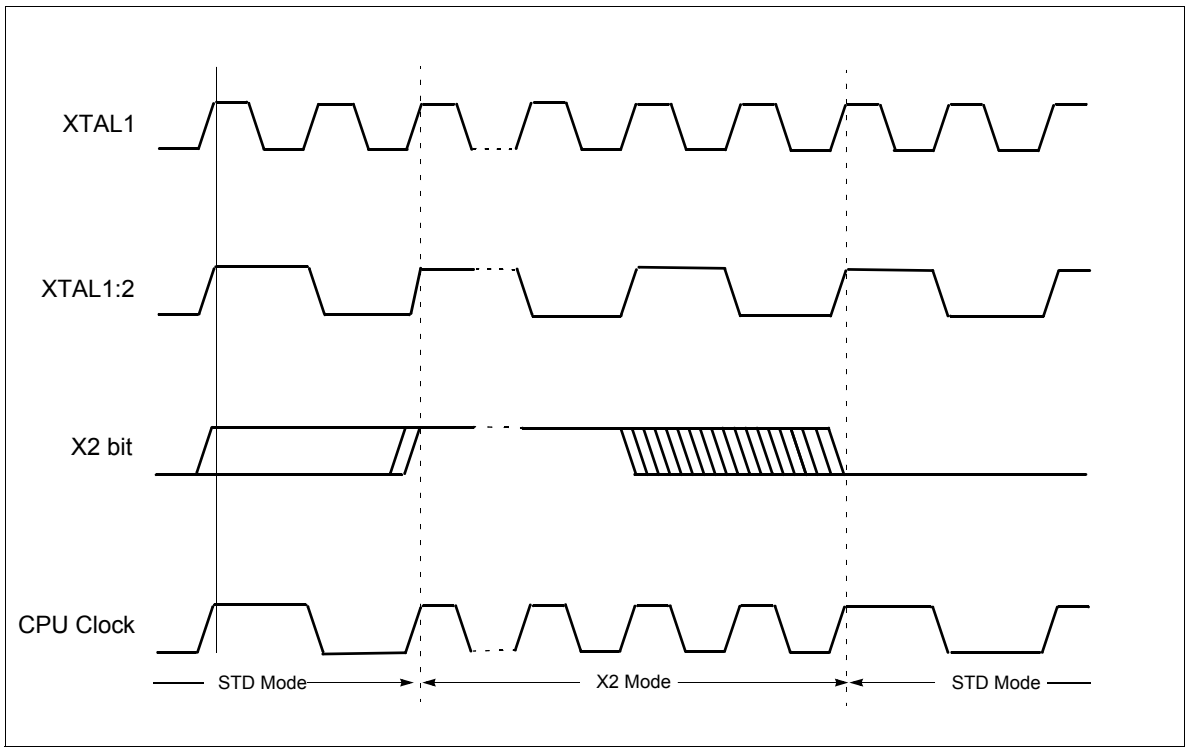


Figure 3. Mode Switching Waveforms

.

TIMER / COUNTER 0 AND 1

Timer/Counter 0 and 1 can be used in four operating modes as listed in Table 4:

Mode	Description	TMOD				Input Clock	
		Gate	C/$\overline{T}$	M1	M0	internal	external (Max.)
0	8-bit timer/counter with a divide-by-32 prescaler	X	X	0	0	$f_{OSC} \div (12 \times 32)$	$f_{OSC} \div (24 \times 32)$
1	16-bit timer/counter	X	X	0	1	$f_{OSC} \div 12$	$f_{OSC} \div 24$
2	8-bit timer/counter with 8-bit auto-reload	X	X	1	0	$f_{OSC} \div 12$	$f_{OSC} \div 24$
3	Timer/counter 0 used as one 8-bit timer/counter and one 8-bit timer Timer 1 stops	X	X	1	1	$f_{OSC} \div 12$	$f_{OSC} \div 24$

Table 4. Timer/Counter 0 and 1 Operating Modes

In the "timer" function (C/$\overline{T}$ = "0") the register is incremented every machine cycle. Therefore the count rate is $f_{OSC}/12$.

In the "counter" function the register is incremented in response to a 1-to-0 transition at its corresponding external input pin (P3.4/T0, P3.5/T1). Since it takes two machine cycles to detect a falling edge the max. count rate is $f_{OSC}/24$. External inputs $\overline{INT0}$ and $\overline{INT1}$ (P3.2, P3.3) can be programmed to function as a gate to facilitate pulse width measurements. Figure 4 illustrates the input clock logic.

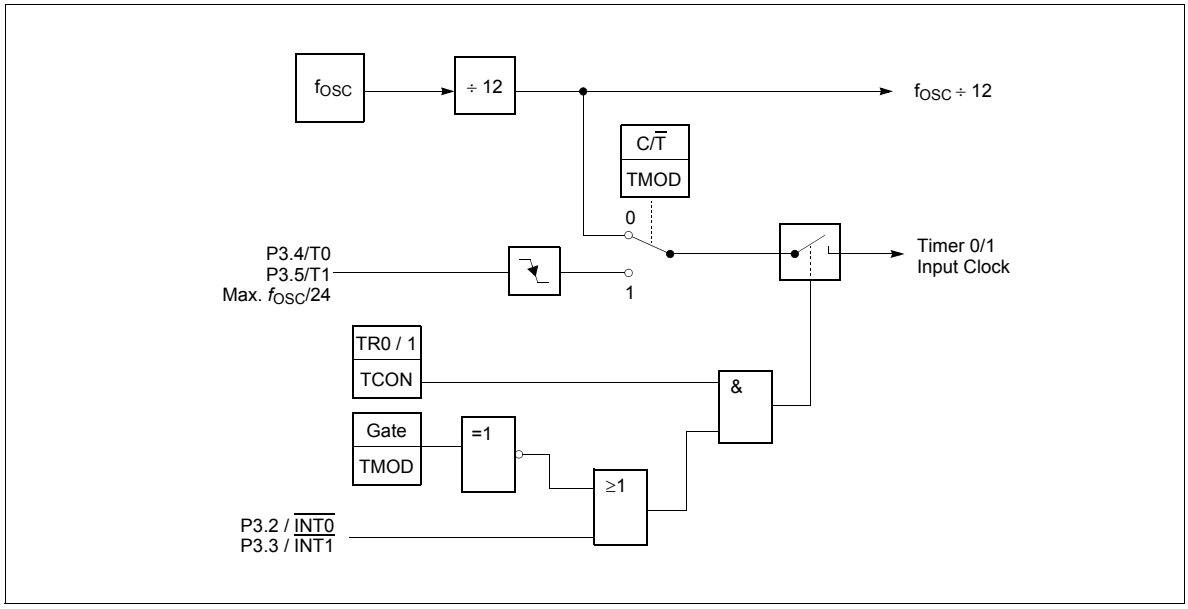


Figure 4. Timer/Counter 0 and 1 Input Clock Logic

TIMER 2

Timer 2 is a 16-bit timer/Counter with an up/down count feature. It can operate either as timer or as an event counter which is selected by bit C/$\overline{T2}$ (T2CON.1). It has three operating modes as shown in Table 5.

Mode	T2CON			T2MOD	T2CON	P1.1/T2EX	Remarks	Input Clock	
	RCLK or TCLK	CP/RL2	TR2	DCEN	EXEN2			internal	external (P1.0/T2)
16-bit Auto-Reload	0	0	1	0	0	X	reload upon over-flow	$f_{OSC} \div 12$	Max. $f_{OSC} \div 24$
	0	0	1	0	1	↓	reload trigger (falling edge)		
	0	0	1	1	X	0	Down counting		
	0	0	1	1	X	1	Up counting		
16-bit Capture	0	1	1	X	0	X	16 bit Timer/ Counter (only up-counting)	$f_{OSC} \div 12$	Max. $f_{OSC} \div 24$
	0	1	1	X	1	↓	capture TH2,TL2 → RC2H,RC2L		
Baud Rate Generator	1	X	1	X	0	X	no overflow interrupt request (TF2)	$f_{OSC} \div 12$	Max. $f_{OSC} \div 24$
	1	X	1	X	1	↓	extra external interrupt (“Timer 2”)		
Off	X	X	0	X	X	X	Timer 2 stops	-	-

Table 5. Timer/Counter 2 Operating Modes

Note: ↓ = falling edge

SERIAL INTERFACE (UART)

The serial port is full duplex and can operate in four modes (one synchronous mode, three asynchronous modes) as illustrated in Table 6. The possible baud rates can be calculated using the formulas given in Table 7.

Mode	SCON		Baudrate	Description
	SM0	SM1		
0	0	0	$\frac{f_{OSC}}{12}$	Serial data enters and exits through RxD. TxD outputs the shift clock. 8-bit are transmitted/received (LSB first)
1	0	1	Timer 1/2 overflow rate	8-bit UART 10 bits are transmitted (through TxD) or received (RxD)
2	1	0	$\frac{f_{OSC}}{32}$ or $\frac{f_{OSC}}{64}$	9-bit UART 11 bits are transmitted (TxD) or received (RxD)
3	1	1	Timer 1/2 overflow rate	9-bit UART Like mode 2 except the variable baud rate

Table 6. UART Operating Modes

Baud Rate derived from	Interface Mode	Baudrate
Oscillator	0	$\frac{f_{OSC}}{12}$
	2	$\frac{2^{SMOD}}{64} \times f_{OSC}$
Timer 1 (16-bit timer) (8-bit timer with 8-bit auto reload)	1,3	$\frac{2^{SMOD}}{32} \times (Timer\ 1\ overflow)$
	1,3	$\frac{2^{SMOD}}{32} \times \frac{f_{OSC}}{12 \times [256 - (TH1)]}$
Timer 2	1,3	$\frac{f_{OSC}}{32 \times [65536 - (RC2H, RC2L)]}$

Table 7. Formulas for Calculating Baud rates

INTERRUPT SYSTEM

The HMS99C51S/52S/54S/56S/58S provide 6 (above 8K bytes ROM version) interrupt sources with two priority levels. Figure 5 gives a general overview of the interrupt sources and illustrates the request and control flags.

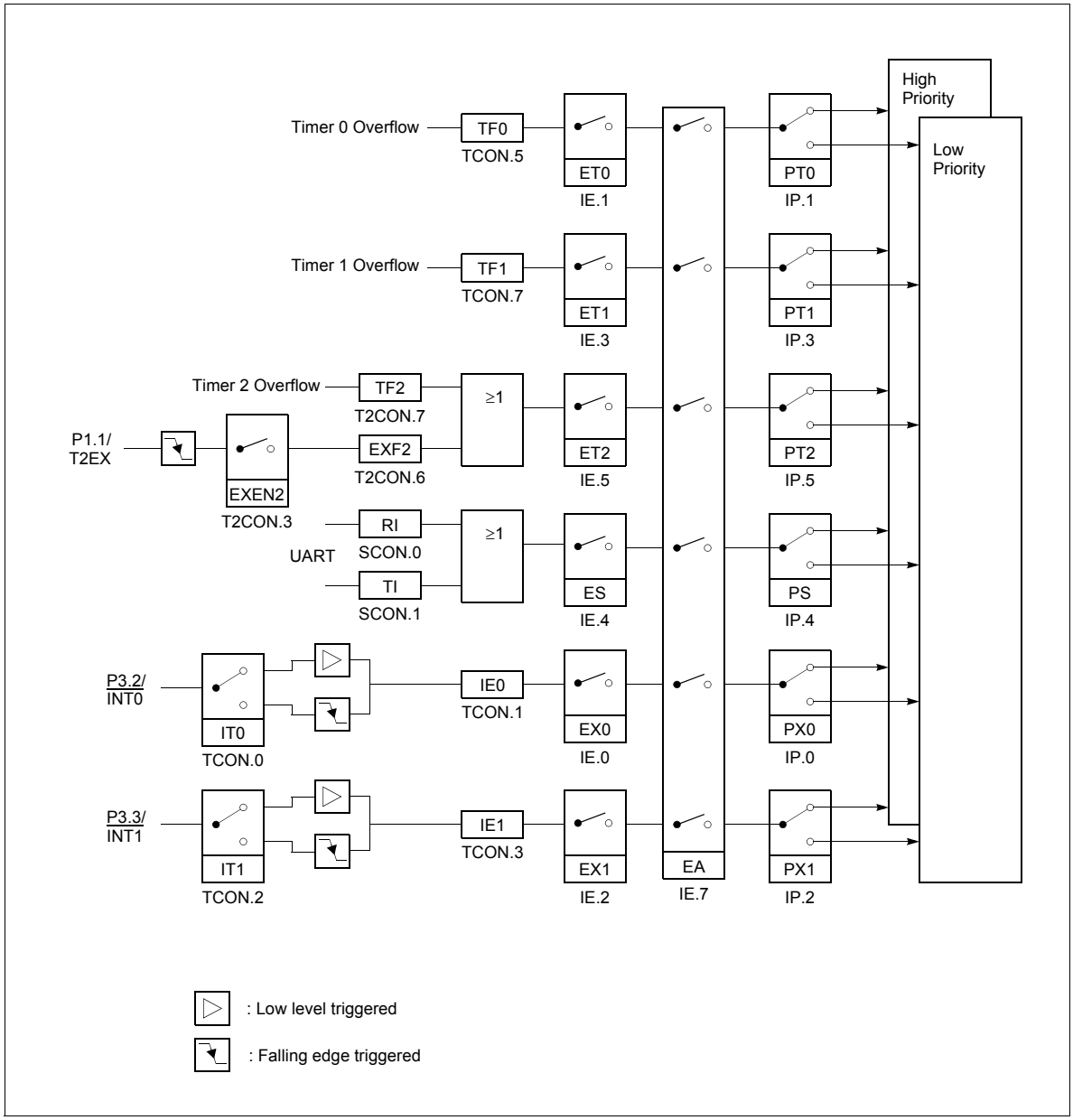


Figure 5. Interrupt Request Sources

Source (Request Flags)	Vectors	Vector Address
RESET	RESET	0000_H
IE0	External interrupt 0	0003_H
TF0	Timer 0 interrupt	$000B_H$
IE1	External interrupt 1	0013_H
TF1	Timer 1 interrupt	$001B_H$
RI + TI	Serial port interrupt	0023_H
TF2 + EXF2	Timer 2 interrupt	$002B_H$

Table 8. Interrupt Sources and their Corresponding Interrupt Vectors

A low-priority interrupt can itself be interrupted by a high-priority interrupt, but not by another low priority interrupt. A high-priority interrupt cannot be interrupted by any other interrupt source.

If two requests of different priority level are received simultaneously, the request of higher priority is serviced. If requests of the same priority are received simultaneously, an internal polling sequence determines which request is serviced. Thus within each priority level there is a second priority structure determined by the polling sequence as shown in Table 9.

Interrupt Source		Priority
External Interrupt 0	IE0	High
Timer 0 Interrupt	TF0	↓
External Interrupt 1	IE1	↓
Timer 1 Interrupt	TF1	↓
Serial Channel	RI + TI	↓
Timer 2 Interrupt	TF2 + EXF2	Low

Table 9. Interrupt Priority-Within-Level

Power Saving Modes

Two power down modes are available, the Idle Mode and Power Down Mode.

The bits PDE and IDLE of the register PCON select the Power Down mode or the Idle mode, respectively. If the Power Down mode and the Idle mode are set at the same time, the Power Down mode takes precedence. Table 10 gives a general overview of the power saving modes.

Mode	Entering Instruction Example	Leaving by	Remarks
Idle mode	ORL PCON, #01H	- Enabled interrupt - Hardware Reset	CPU is gated off CPU status registers maintain their data. Peripherals are active
Power-Down mode	ORL PCON, #02H	Hardware Reset	Oscillator is stopped, contents of on-chip RAM and SFR's are maintained (leaving Power Down Mode means redefinition of SFR contents).

Table 10. Power Saving Modes Overview

In the Power Down mode of operation, V_{CC} can be reduced to minimize power consumption. It must be ensured, however, that V_{CC} is not reduced before the Power Down mode is invoked, and that V_{CC} is restored to its normal operating level, before the Power Down mode is terminated. The reset signal that terminates the Power Down mode also restarts the oscillator. The reset should not be activated before V_{CC} is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize (similar to power-on reset).

ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings

Ambient temperature under bias (T_A)..... -40 to + 85 °C

Storage temperature (T_{ST})..... -65 to + 150 °C

Voltage on V_{CC} pins with respect to ground (V_{SS}) ..... -0.5V to 6.5V

Voltage on any pin with respect to ground (V_{SS}) .....-0.5V to V_{CC} + 0.5V

Input current on any pin during overload condition.....-10mA to +10mA

Absolute sum of all input currents during overload condition.....|100mA|

Power dissipation 200mW

Note: *Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage of the device. This is a stress rating only and functional operation of the 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 for longer periods may affect device reliability. During overload conditions (V_{IN} > V_{CC} or V_{IN} < V_{SS}) the Voltage on V_{CC} pins with respect to ground (V_{SS}) must not exceed the values defined by the absolute maximum ratings.*

DC Characteristics

DC Characteristics for HMS99C51S/52S/54S/56S/58S

V_{CC}= 5V + 10%, -15%, V_{SS}=0V, T_A= -40°C to 85 °C

Parameter	Symbol	Limit Values		Unit	Test Conditions
		Min.	Max.		
Input low voltage (except $\overline{EA}$, RESET)	V_{IL}	-0.5	$0.2V_{CC}$ - 0.1	V	V_{CC}= 5.5V
Input low voltage ($\overline{EA}$)	V_{IL1}	-0.5	$0.2V_{CC}$ - 0.1	V	V_{CC}= 5.5V
Input low voltage (RESET)	V_{IL2}	-0.5	$0.2V_{CC}$ + 0.1	V	V_{CC}= 5.5V
Input high voltage (except XTAL1, $\overline{EA}$, RESET)	V_{IH}	$0.2V_{CC}$ + 0.9	V_{CC} + 0.5	V	V_{CC}= 4.5V
Input high voltage to XTAL1	V_{IH1}	$0.7V_{CC}$	V_{CC} + 0.5	V	V_{CC}= 4.5V
Input high voltage to $\overline{EA}$, RESET	V_{IH2}	$0.6V_{CC}$	V_{CC} + 0.5	V	V_{CC}= 4.5V
Output low voltage (ports 1, 2, 3)	V_{OL}	-	0.45	V	V_{CC}= 5.5V, I_{OL}= 1.6mA [1]
Output low voltage (port 0, ALE, $\overline{PSEN}$)	V_{OL1}	-	0.45	V	V_{CC}= 5.5V, I_{OL}= 3.2mA [1]
Output high voltage (ports 1, 2, 3)	V_{OH}	2.4 $0.9V_{CC}$	-	V	V_{CC}= 4.5V, I_{OH}= -80μA V_{CC}= 4.5V, I_{OH}= -10μA
Output high voltage (port 0 in external bus mode, ALE, $\overline{PSEN}$)	V_{OH1}	2.4 $0.9V_{CC}$	-	V	V_{CC}= 4.5V, I_{OH}= -800μA [2] V_{CC}= 4.5V, I_{OH}= -80μA [2]
Logic 0 input current (ports 1, 2, 3)	I_{IL}	-10	-65	μA	V_{IN}= 0.45V
Logical 1-to-0 transition current (ports 1, 2, 3)	I_{TL}	-65	-650	μA	V_{IN}= 2.0V
Input leakage current (port 0, $\overline{EA}$)	I_{LI}	-	±1	μA	0.45 < V_{IN} < V_{CC}
Pin capacitance	C_{IO}	-	10	pF	f_C= 1MHz T_A= 25°C
Power supply current: Active mode, 4MHz [3] Idle mode, 4MHz [4] Active mode, 24 MHz [4] Idle mode, 24MHz [4] Active mode, 40 MHz [4] Idle mode, 40 MHz [4] Power Down Mode [4]	 I_{CC} I_{CC} I_{CC} I_{CC} I_{CC} I_{CC} I_{PD}	-	 8 4 25 10 30 15 50	 mA mA mA mA mA mA uA	 V_{CC}= 5V [4] V_{CC}= 5V [5] V_{CC}= 5V [4] V_{CC}= 5V [5] V_{CC}= 5V [4] V_{CC}= 5V [5] V_{CC}= 5V [6]

1) Capacitive loading on ports 0 and 2 may cause spurious noise pulses to be superimposed on the V_{OL} of ALE and port 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operation. In the worst case(capacitive loading: $>$ 50pF at 3.3V, $>$ 100pF at 5V), the noise pulse on ALE line may exceed 0.8V. In such cases it may be desirable to qualify ALE with a schmitt-trigger, or use an address latch with a schmitt-trigger strobe input.
2) Capacitive loading on ports 0 and 2 may cause the V_{OH} on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{CC}$ specification when the address lines are stabilizing.
3) I_{CC} Max. at other frequencies is given by:
   active mode: $I_{CC} = 1.27 \times f_{OSC} + 5.73$
   idle mode: $I_{CC} = 0.28 \times f_{OSC} + 1.45$ (except OTP devices)
   where f_{OSC} is the oscillator frequency in MHz. I_{CC} values are given in mA and measured at V_{CC} = 5V.
4) I_{CC} (active mode) is measured with:
   XTAL1 driven with t_{CLCH}, t_{CHCL} = 5ns, $V_{IL} = V_{SS}$ + 0.5V, $V_{IH} = V_{CC}$ - 0.5V; XTAL2 = N.C.;
   $\overline{EA}$ = Port0 = RESET = V_{CC}; all other pins are disconnected. I_{CC} would be slightly higher if a crystal oscillator is used (appr. 1mA).
5) I_{CC} (Idle mode) is measured with all output pins disconnected and with all peripherals disabled;
   XTAL1 driven with t_{CLCH}, t_{CHCL} = 5ns, $V_{IL} = V_{SS}$ + 0.5V, $V_{IH} = V_{CC}$ - 0.5V; XTAL2 = N.C.;
   RESET = $\overline{EA}$ = V_{SS}; Port0 = V_{CC}; all other pins are disconnected;
6) I_{PD} (Power Down Mode) is measured under following conditions:
   $\overline{EA}$ = Port0 = V_{CC}; RESET = V_{SS}; XTAL2 = N.C.; XTAL1 = V_{SS}; all other pins are disconnected.

AC Characteristics

Explanation of the AC Symbols

Each timing symbol has 5 characters. The first character is always a 't' (stand for time). The other characters, depending on their positions, stand for the name of a signal or the logical status of that signal. The following is a list of all the characters and what they stand for.

A: Address
C: Clock
D: Input Data
H: Logic level HIGH
I: Instruction (program memory contents)
L: Logic level LOW, or ALE
P: $\overline{\text{PSEN}}$
Q: Output Data
R: $\overline{\text{RD}}$ signal
T: Time
V: Valid
W: $\overline{\text{WR}}$ signal
X: No longer a valid logic level
Z: Float

For example,
t_{AVLL} = Time from Address Valid to ALE Low
t_{LLPL} = Time from ALE Low to $\overline{\text{PSEN}}$ Low

AC Characteristics for HMS99C51S/52S/54S/56S/58S (12MHz version)

V_{CC}= 5V : V_{CC}= 5V + 10%, – 15%, V_{SS}= 0V, T_A= -40°C to 85°C
(C_L for port 0, ALE and $\overline{\text{PSEN}}$ outputs = 100pF, C_L for all other outputs = 80pF)

Variable clock : Vcc = 5V : $1/t_{CLCL}$ = 3.5 MHz to 12 MHz

External Program Memory Characteristics

Parameter	Symbol	12 MHz Oscillator		Variable Oscillator $1/t_{CLCL}$ = 3.5 to 12MHz		Unit
		Min.	Max.	Min.	Max.	
ALE pulse width	t_{LHLL}	127	-	$2t_{CLCL}$-40	-	ns
Address setup to ALE	t_{AVLL}	43	-	t_{CLCL}-40	-	ns
Address hold after ALE	t_{LLAX}	30	-	t_{CLCL}-53	-	ns
ALE low to valid instruction in	t_{LLIV}	-	233	-	$4t_{CLCL}$-100	ns
ALE to $\overline{\text{PSEN}}$	t_{LLPL}	58	-	t_{CLCL}-25	-	ns
$\overline{\text{PSEN}}$ pulse width	t_{PLPH}	215	-	$3t_{CLCL}$-35	-	ns
$\overline{\text{PSEN}}$ to valid instruction in	t_{PLIV}	-	150	-	$3t_{CLCL}$-100	ns
Input instruction hold after $\overline{\text{PSEN}}$	t_{PXIX}	0	-	0	-	ns
Input instruction float after $\overline{\text{PSEN}}$	t_{PXIZ} †	-	63	-	t_{CLCL}-20	ns
Address valid after $\overline{\text{PSEN}}$	t_{PXAV} †	75	-	t_{CLCL}-8	-	ns
Address to valid instruction in	t_{AVIV}	-	302	-	$5t_{CLCL}$-115	ns
Address float to $\overline{\text{PSEN}}$	t_{AZPL}	0	-	0	-	ns

† Interfacing the HMS99C51S/52S/54S/56S/58S to devices with float times up to 75 ns is permissible. This limited bus contention will not cause any damage to port 0 Drivers.

AC Characteristics for HMS99C51S/52S/54S/56S/58S (12MHz)

External Data Memory Characteristics

Parameter	Symbol	12 MHz Oscillator		Variable Oscillator $1/t_{CLCL}$ = 3.5 to 12MHz		Unit
		Min.	Max.	Min.	Max.	
$\overline{RD}$ pulse width	t_{RLRH}	400	-	$6t_{CLCL}$-100	-	ns
$\overline{WR}$ pulse width	t_{WLWH}	400	-	$6t_{CLCL}$-100	-	ns
Address hold after ALE	t_{LLAX2}	53	-	t_{CLCL}-30	-	ns
$\overline{RD}$ to valid data in	t_{RLDV}	-	252	-	$5t_{CLCL}$-165	ns
Data hold after $\overline{RD}$	t_{RHDX}	0	-	0	-	ns
Data float after $\overline{RD}$	t_{RHDZ}	-	97	-	$2t_{CLCL}$-70	ns
ALE to valid data in	t_{LLDV}	-	517	-	$8t_{CLCL}$-150	ns
Address to valid data in	t_{AVDV}	-	585	-	$9t_{CLCL}$-165	ns
ALE to $\overline{WR}$ or $\overline{RD}$	t_{LLWL}	200	300	$3t_{CLCL}$-50	$3t_{CLCL}$+50	ns
Address valid to $\overline{WR}$ or $\overline{RD}$	t_{AVWL}	203	-	$4t_{CLCL}$-130	-	ns
$\overline{WR}$ or $\overline{RD}$ high to ALE high	t_{WHLH}	43	123	t_{CLCL}-40	t_{CLCL}+40	ns
Data valid to $\overline{WR}$ transition	t_{QVWX}	33	-	t_{CLCL}-50	-	ns
Data setup before $\overline{WR}$	t_{QVWH}	433	-	$7t_{CLCL}$-150	-	ns
Data hold after $\overline{WR}$	t_{WHQX}	33	-	t_{CLCL}-50	-	ns
Address float after $\overline{RD}$	t_{RLAZ}	-	0	-	0	ns

Advance Information (12MHz)

External Clock Drive

Parameter	Symbol	Variable Oscillator (Freq. = 3.5 to 12MHz)		Unit
		Min.	Max.	
Oscillator period (V_{CC}=5V)	t_{CLCL}	83.3	285.7	ns
High time	t_{CHCX}	20	t_{CLCL} - t_{CLCX}	ns
Low time	t_{CLCX}	20	t_{CLCL} - t_{CHCX}	ns
Rise time	t_{CLCH}	-	20	ns
Fall time	t_{CHCL}	-	20	ns

AC Characteristics for HMS99C51S/52S/54S/56S/58S (24MHz version)

V_{CC}= 5V + 10%, −15%, V_{SS}= 0V, T_A= -40°C to 85°C
(C_L for port 0. ALE and $\overline{PSEN}$ outputs = 100pF; C_L for all other outputs = 80pF)

External Program Memory Characteristics

Parameter	Symbol	24 MHz Oscillator		Variable Oscillator $1/t_{CLCL}$ = 3.5 to 24MHz		Unit
		Min.	Max.	Min.	Max.	
ALE pulse width	t_{LHLL}	43	-	$2t_{CLCL}$-40	-	ns
Address setup to ALE	t_{AVLL}	17	-	t_{CLCL}-25	-	ns
Address hold after ALE	t_{LLAX}	17	-	t_{CLCL}-25	-	ns
ALE low to valid instruction in	t_{LLIV}	-	80	-	$4t_{CLCL}$-87	ns
ALE to $\overline{PSEN}$	t_{LLPL}	22	-	t_{CLCL}-20	-	ns
$\overline{PSEN}$ pulse width	t_{PLPH}	95	-	$3t_{CLCL}$-30	-	ns
$\overline{PSEN}$ to valid instruction in	t_{PLIV}	-	60	-	$3t_{CLCL}$-65	ns
Input instruction hold after $\overline{PSEN}$	t_{PXIX}	0	-	0	-	ns
Input instruction float after $\overline{PSEN}$	t_{PXIZ} †	-	32	-	t_{CLCL}-10	ns
Address valid after $\overline{PSEN}$	t_{PXAV} †	37	-	t_{CLCL}-5	-	ns
Address to valid instruction in	t_{AVIV}	-	148	-	$5t_{CLCL}$-60	ns
Address float to $\overline{PSEN}$	t_{AZPL}	0	-	0	-	ns

† Interfacing the HMS99C51S/52S/54S/56S/58S to devices with float times up to 35 ns is permissible. This limited bus contention will not cause any damage to port 0 Drivers.

AC Characteristics for HMS99C51S/52S/54S/56S/58S(24MHz)

External Data Memory Characteristics

Parameter	Symbol	24 MHz Oscillator		Variable Oscillator $1/t_{CLCL}$ = 3.5 to 24MHz		Unit
		Min.	Max.	Min.	Max.	
$\overline{RD}$ pulse width	t_{RLRH}	180	-	$6t_{CLCL}$-70	-	ns
$\overline{WR}$ pulse width	t_{WLWH}	180	-	$6t_{CLCL}$-70	-	ns
Address hold after ALE	t_{LLAX2}	15	-	t_{CLCL}-27	-	ns
$\overline{RD}$ to valid data in	t_{RLDV}	-	118	-	$5t_{CLCL}$-90	ns
Data hold after $\overline{RD}$	t_{RHDX}	0	-	0	-	ns
Data float after $\overline{RD}$	t_{RHDZ}	-	63	-	$2t_{CLCL}$-20	ns
ALE to valid data in	t_{LLDV}	-	200	-	$8t_{CLCL}$-133	ns
Address to valid data in	t_{AVDV}	-	220	-	$9t_{CLCL}$-155	ns
ALE to $\overline{WR}$ or $\overline{RD}$	t_{LLWL}	75	175	$3t_{CLCL}$-50	$3t_{CLCL}$+50	ns
Address valid to $\overline{WR}$ or $\overline{RD}$	t_{AVWL}	67	-	$4t_{CLCL}$-97	-	ns
$\overline{WR}$ or $\overline{RD}$ high to ALE high	t_{WHLH}	17	67	t_{CLCL}-25	t_{CLCL}+25	ns
Data valid to $\overline{WR}$ transition	t_{QVWX}	5	-	t_{CLCL}-37	-	ns
Data setup before $\overline{WR}$	t_{QVWH}	170	-	$7t_{CLCL}$-122	-	ns
Data hold after $\overline{WR}$	t_{WHQX}	15	-	t_{CLCL}-27	-	ns
Address float after $\overline{RD}$	t_{RLAZ}	-	0	-	0	ns

Advance Information (24MHz)

External Clock Drive

Parameter	Symbol	Variable Oscillator (Freq. = 3.5 to 24MHz)		Unit
		Min.	Max.	
Oscillator period	t_{CLCL}	41.7	285.7	ns
High time	t_{CHCX}	12	t_{CLCL} - t_{CLCX}	ns
Low time	t_{CLCX}	12	t_{CLCL} - t_{CHCX}	ns
Rise time	t_{CLCH}	-	12	ns
Fall time	t_{CHCL}	-	12	ns

AC Characteristics for HMS99C51S/52S/54S/56S/58S(40MHz version)

V_{CC}= 5V + 10%, – 15%, V_{SS}= 0V, T_A= -40°C to 85°C
(C_L for port 0, ALE and $\overline{PSEN}$ outputs = 100pF, C_L for all other outputs = 80pF)

External Program Memory Characteristics

Parameter	Symbol	40 MHz Oscillator		Variable Oscillator $1/t_{CLCL}$ = 3.5 to 40MHz		Unit
		Min.	Max.	Min.	Max.	
ALE pulse width	t_{LHLL}	40	-	$2t_{CLCL}-20$	-	ns
Address setup to ALE	t_{AVLL}	10	-	$t_{CLCL}-20$	-	ns
Address hold after ALE	t_{LLAX}	10	-	$t_{CLCL}-20$	-	ns
ALE low to valid instruction in	t_{LLIV}	-	56	-	$4t_{CLCL}-65$	ns
ALE to $\overline{PSEN}$	t_{LLPL}	15	-	$t_{CLCL}-15$	-	ns
$\overline{PSEN}$ pulse width	t_{PLPH}	80	-	$3t_{CLCL}-20$	-	ns
$\overline{PSEN}$ to valid instruction in	t_{PLIV}	-	35	-	$3t_{CLCL}-55$	ns
Input instruction hold after $\overline{PSEN}$	t_{PXIX}	0	-	0	-	ns
Input instruction float after $\overline{PSEN}$	t_{PXIZ} †	-	20	-	$t_{CLCL}-10$	ns
Address valid after $\overline{PSEN}$	t_{PXAV} †	25	-	$t_{CLCL}-5$	-	ns
Address to valid instruction in	t_{AVIV}	-	91	-	$5t_{CLCL}-60$	ns
Address float to $\overline{PSEN}$	t_{AZPL}	0	-	0	-	ns

† Interfacing the HMS99C51S/52S/54S/56S/58S to devices with float times up to 20 ns is permissible. This limited bus contention will not cause any damage to port 0 Drivers.

AC Characteristics for HMS99C51S/52S/54S/56S/58S(40MHz)

External Data Memory Characteristics

Parameter	Symbol	at 40 MHz Clock		Variable Clock $1/t_{CLCL}$ = 3.5 to 40MHz		Unit
		Min.	Max.	Min.	Max.	
$\overline{RD}$ pulse width	t_{RLRH}	132	-	$6t_{CLCL}$-50	-	ns
$\overline{WR}$ pulse width	t_{WLWH}	132	-	$6t_{CLCL}$-50	-	ns
Address hold after ALE	t_{LLAX2}	10	-	t_{CLCL}-20	-	ns
$\overline{RD}$ to valid data in	t_{RLDV}	-	81	-	$5t_{CLCL}$-70	ns
Data hold after $\overline{RD}$	t_{RHDX}	0	-	0	-	ns
Data float after $\overline{RD}$	t_{RHDZ}	-	46	-	$2t_{CLCL}$-15	ns
ALE to valid data in	t_{LLDV}	-	153	-	$8t_{CLCL}$-90	ns
Address to valid data in	t_{AVDV}	-	183	-	$9t_{CLCL}$-90	ns
ALE to $\overline{WR}$ or $\overline{RD}$	t_{LLWL}	71	111	$3t_{CLCL}$-20	$3t_{CLCL}$+20	ns
Address valid to $\overline{WR}$ or $\overline{RD}$	t_{AVWL}	66	-	$4t_{CLCL}$-55	-	ns
$\overline{WR}$ or $\overline{RD}$ high to ALE high	t_{WHLH}	10	40	t_{CLCL}-20	t_{CLCL}+20	ns
Data valid to $\overline{WR}$ transition	t_{QVWX}	5	-	t_{CLCL}-25	-	ns
Data setup before $\overline{WR}$	t_{QVWH}	142	-	$7t_{CLCL}$-70	-	ns
Data hold after $\overline{WR}$	t_{WHQX}	10	-	t_{CLCL}-20	-	ns
Address float after $\overline{RD}$	t_{RLAZ}	-	0	-	0	ns

Advance Information (40MHz)

External Clock Drive

Parameter	Symbol	Variable Oscillator (Freq. = 3.5 to 40MHz)		Unit
		Min.	Max.	
Oscillator period	t_{CLCL}	30.3	285.7	ns
High time	t_{CHCX}	11.5	t_{CLCL} - t_{CLCX}	ns
Low time	t_{CLCX}	11.5	t_{CLCL} - t_{CHCX}	ns
Rise time	t_{CLCH}	-	5	ns
Fall time	t_{CHCL}	-	5	ns

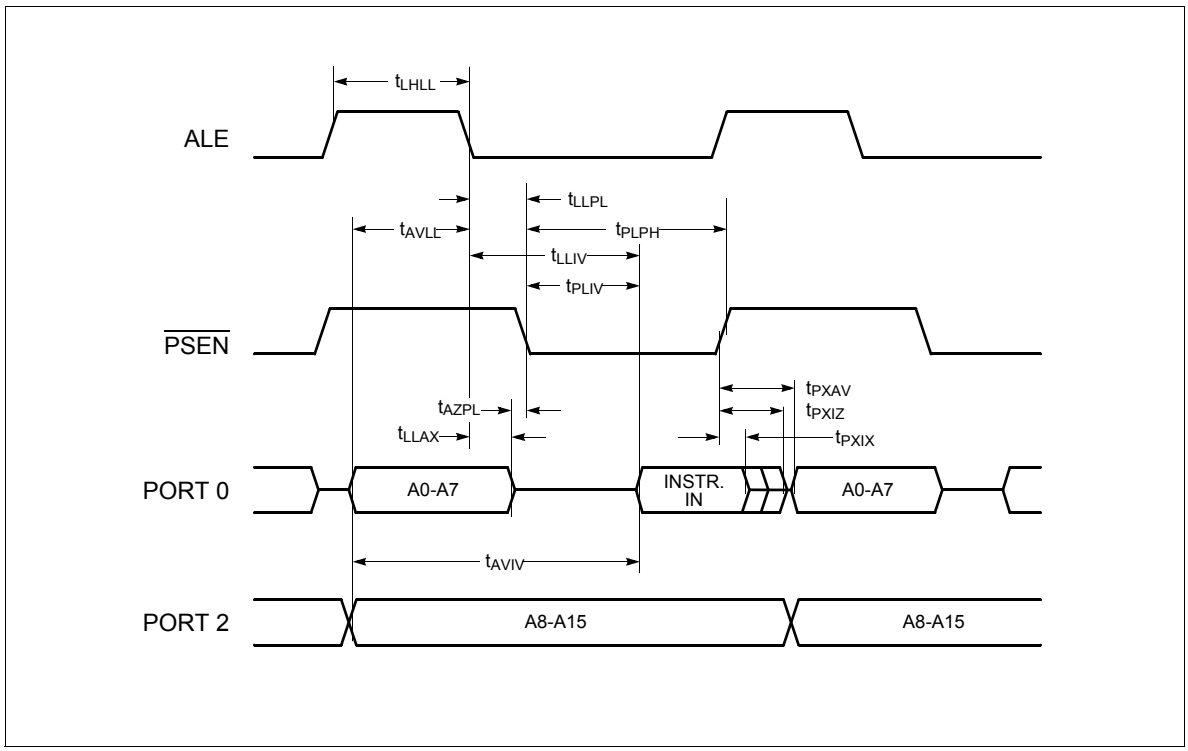


Figure 6. External Program Memory Read Cycle

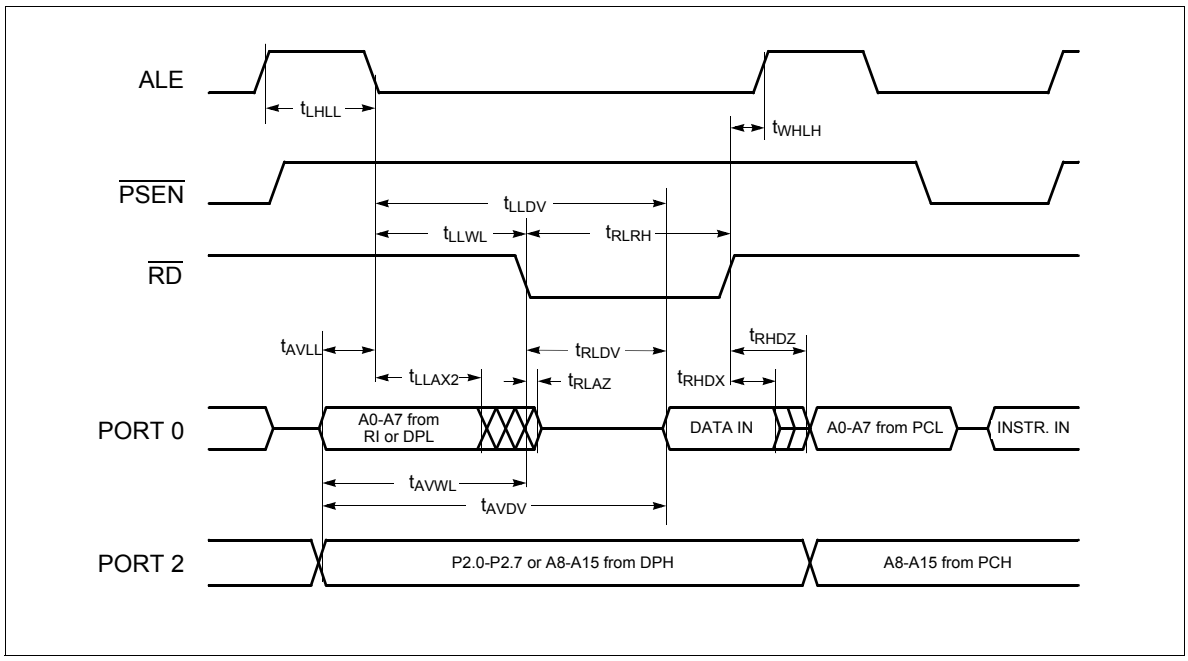


Figure 7. External Data Memory Read Cycle

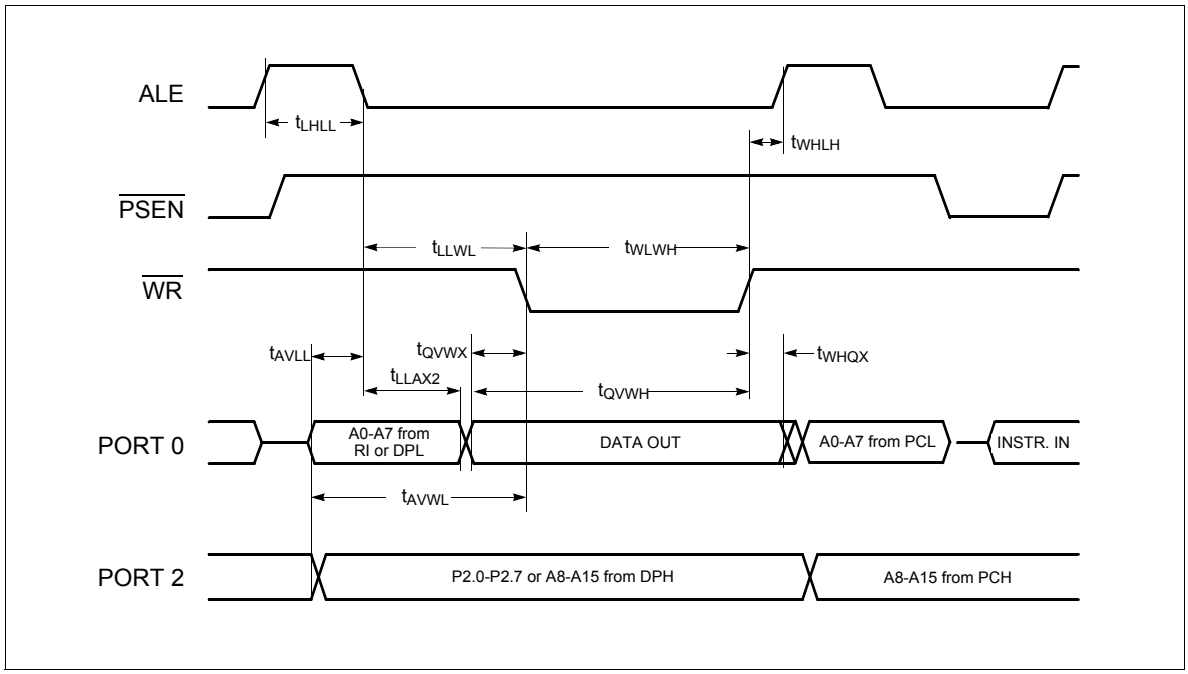
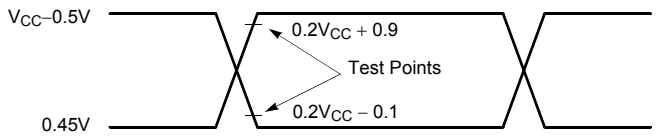
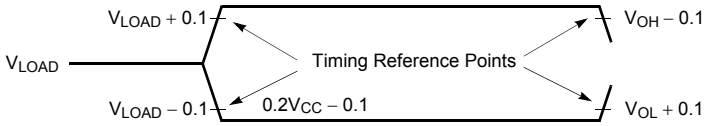


Figure 8. External Data Memory Write Cycle



AC Inputs during testing are driven at V_{CC}–0.5V for a logic '1' and 0.45V for a logic '0'. Timing measurements are made a V_{IHmin} for a logic '1' and V_{ILmax} for a logic '0'.

Figure 9. AC Testing: Input, Output Waveforms



For timing purposes a port pin is no longer floating when a 100mV change from load voltage occurs and begins to float when a 100mV change from the loaded V_{OH} / V_{OL} level occurs. I_{OL} / $I_{OH} \geq$ 20mA.

Figure 10. Float Waveforms

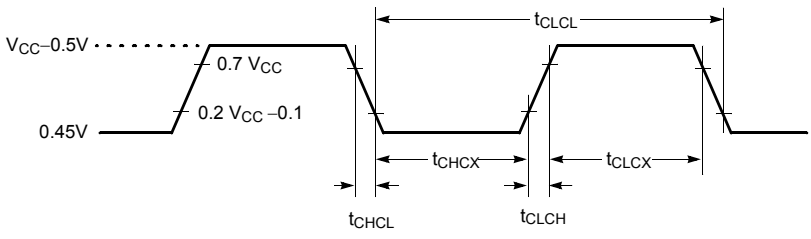


Figure 11. External Clock Cycle

OSCILLATOR CIRCUIT

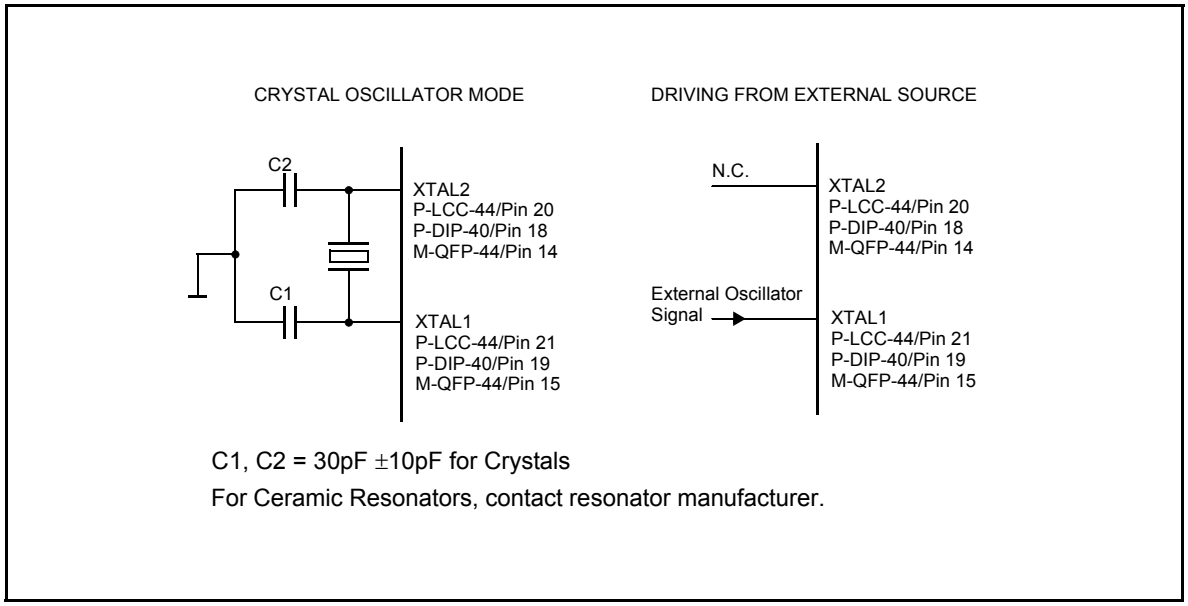


Figure 12. Recommended Oscillator Circuits

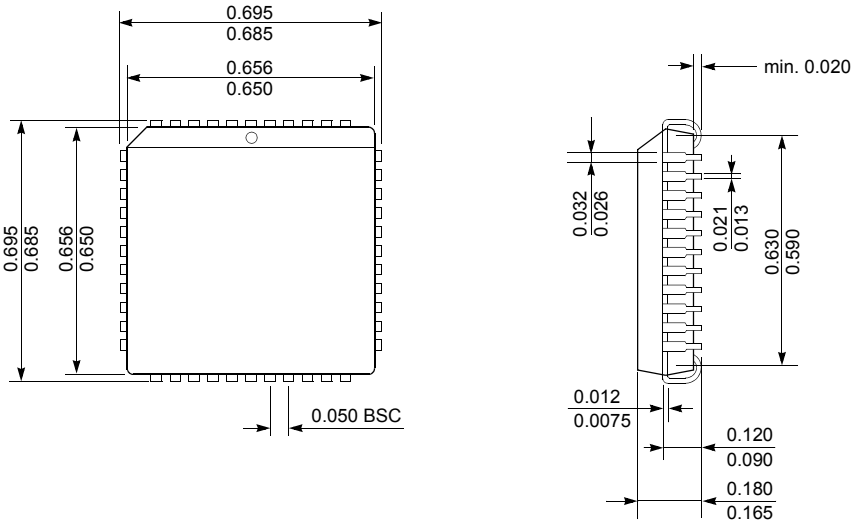
Oscillation circuit is designed to be used either with a ceramic resonator or crystal oscillator. Since each crystal and ceramic resonator have their own characteristics, the user should consult the crystal manufacturer for appropriate values of external components.

Plastic Package P-LCC-44

(Plastic Leaded Chip-Carrier)

44PLCC

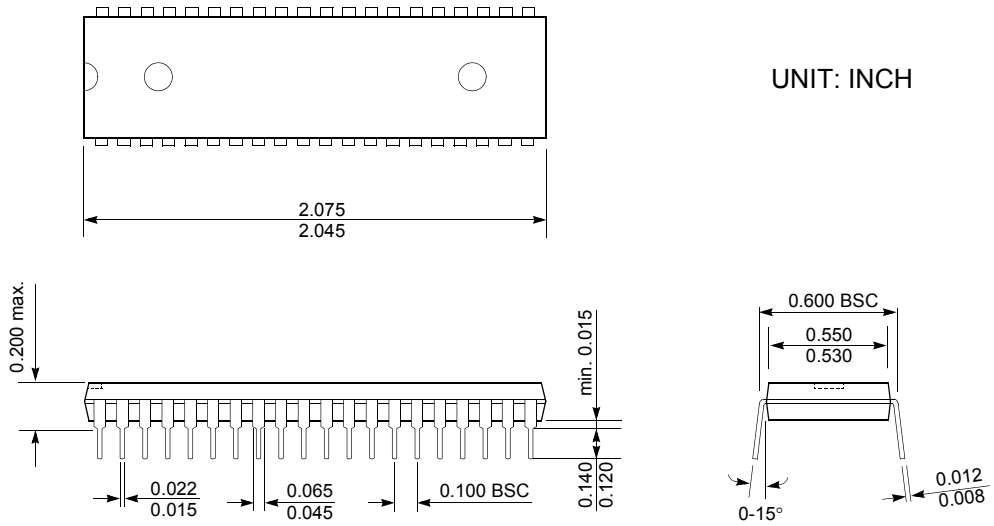
UNIT: INCH



Plastic Package P-DIP-40

(Plastic Dual in-Line Package)

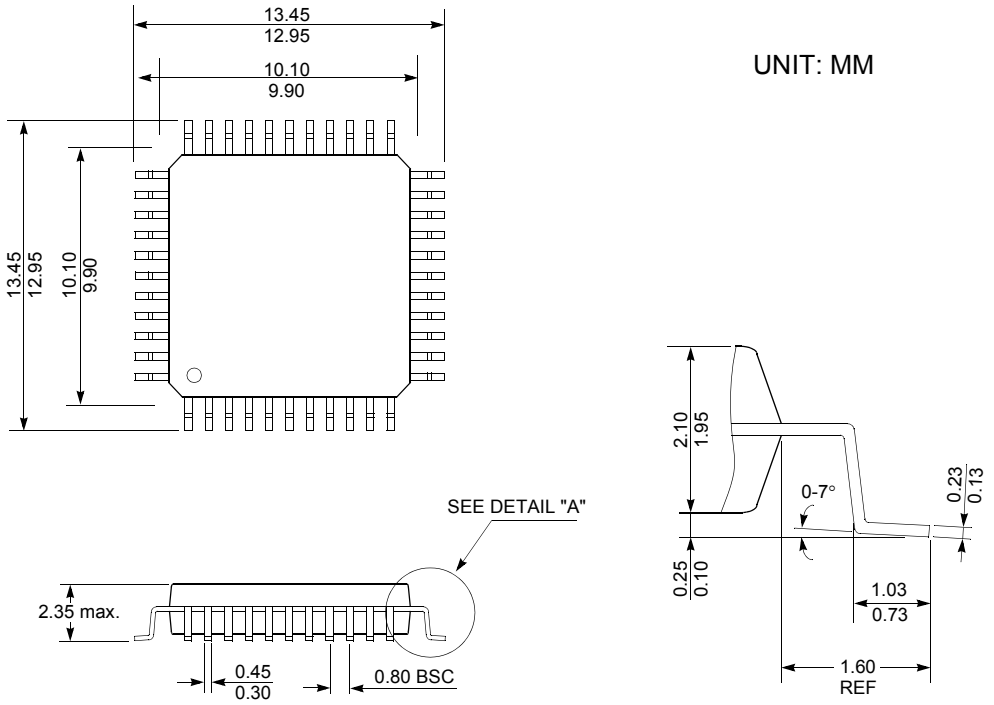
40DIP



Plastic Package P-MPQF-44

(Plastic Metric Quad Flat Package)

44MQFP



FLASH MEMORY

Overview

The Flash memory increases EPROM and ROM functionality with in-circuit electrical erasure and programming. It contains 4K, 8K, 16K, 24K or 32K bytes of program memory. This memory is both parallel and serial In-System Programmable (ISP). ISP allows devices to alter their own program memory in the actual end product under software control. A default serial loader (bootloader) program allows ISP of the Flash. The programming does not require 12V external programming voltage. The necessary high programming voltage is generated on-chip using the standard V_{CC} pins of the microcontroller.

Features

- Flash memory internal program memory.
- Default loader in Boot ROM allows programming via the serial port without the need of a user provided loader.
- Up to 64K byte external program memory if the internal program memory is disabled (EA = 0).
- Programming and erase voltage with standard 5V V_{CC} supply.
- Read/Programming/Erase:
  - Programming time per byte : 20us (TBD)
  - Block erase/Total Erase time : 200ms (TBD)
  - Typical programming time (32K bytes) is 10s at ISP mode (TBD)
- Parallel programming with Atmel/Philips chip compatible hardware interface to programmer
- Programmable security for the code in the Flash
- Endurance : 10,000 cycles (TBD)
- Data retention : 10 years (TBD)

Flash Programming and Erasure

There are three methods of programming the Flash memory:

- First, the on-chip ISP bootloader may be invoked which will use low level routines to program. The interface used for serial downloading of Flash memory is the UART.
- Second, the Flash may be programmed or erased in the end-user application by calling low level routines through a common entry point in the Boot ROM.
- Third, the Flash may be programmed using the parallel method by using conventional EPROM programmer. The commercially available programmers need to have support for the HMS99C51S/52S/54S/56S/58S. The bootloader routines are located in the Boot ROM.

Flash Memory Architecture

HMS99C51S/52S/54S/56S/58S feature two on-chip Flash memories:

- Flash memory FM0:

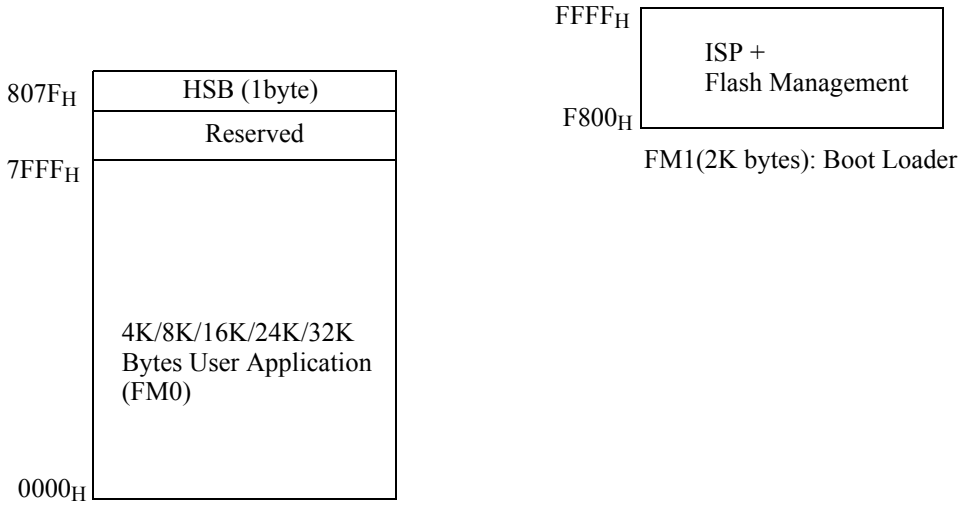
4K/8K/16K/24K/32K bytes user program memory

- Flash memory FM1:

2K bytes for bootloader.

The FM0 and FM1 can be programmed by both parallel programming and Serial In-System Programming.

The ISP mode is detailed in the "In-System Programming" section.



FM0 Memory Architecture

- 4K/8K/16K/24K/32K bytes User Program Memory
- Hardware Security Bits (HSB)

User Space

This space is composed of a 4K/8K/16K/24K/32K bytes Flash memory. HMS99C51S/52S has only sectors of 2K byte unit block, and HMS99C54S/56S/58S has 4 sectors of 2K byte unit block and other sectors of 8K byte unit block. It contains the user's application code.

Hardware Security Byte

The Hardware Security Byte space is a part of HSB and has a size of 1 byte.

Cross Flash Memory Access Description

The FM0 and FM1 memory can be programmed by parallel programming.

The FM0 memory can be programmed from FM1. But, programming FM1 from FM0 or from external memory is impossible.

FM1 memory can be programmed only by parallel programming.

Code Executing from	Action	FM0 (User Flash)	FM1 (Boot Flash)
FM0	Read	OK	-
	Write/Erase	-	-
FM1	Read	OK	OK
	Write/Erase	OK	-

Table 11. Cross Flash Memory Access

Overview of FM0 Operations

The CPU interfaces to the Flash memory through the FCON register of SFR FCON register is used to:

- Select Register for operation of Flash Access (FRSEL[2:0])
- Erase Mode Select (ERASESEL)
- Enable Boot Flash (ENBOOT)
- 64K Bytes Internal Rom Access (INTROM_EN)
- Program Mode Select (PGMSEL)

Mapping of the Memory Space

By default, the user space is accessed by MOVC instruction for read only.

The other memory spaces (user, BOOT, HSB) are made accessible in the code segment by programming bits INTROM_EN, ENBOOT in FCON register in advance. A MOVC instruction is then used for reading these spaces in accordance with address of Table 12.

Region	Addr. 15	Addr. 14~11	Addr. 10~7	Addr. 6~0
HSB(1 Bytes)	1	0000	0111	1111
Boot(2K Bytes)		1111	Variable	Variable
User(32K Bytes)	0	Variable	Variable	Variable

Table 12. FM0 Blocks Select Bits

Flash Registers and Memory Map

The HMS99C51S/52S/54S/56S/58S Flash memory uses several registers for its management:

- Hardware registers can only be accessed through the parallel programming modes which are handled by the parallel programmer.

Hardware Register

The only hardware register of the HMS99C51S/52S/54S/56S/58S is called Hardware Security Byte(HSB).

-	-	-	BLJB_EN	BLJB	LB2	LB1	LB0

Table 13. Hardware Security Byte(HSB)

Bit No	Bit Mnemonic	Description
7~5	-	Reserved
4	**BLJN_EN**	**Enable BLJB Bit** 1 : BLJB is enabled for ISP Mode 0 : BLJB is disabled. (After finishing of download, must be programmed.)
3	**BLJB**	**Boot Loader Jump Bit** 1 : Start the user's application on next reset at address 0000_H 0 : Start the boot loader at address $F800_H$(Default).
2~0	**LB2~0**	**User Memory Lock Bits** See Table 14

1 : Unprogrammed
0 : Programmed

Note: *HSB can be read but can not be programmed in ISP Mode and only programmable by specific tools.*

Flash Memory Lock Bits

The three lock bits provide different levels of protection for the on-chip code and data, when programmed as shown in Table 14.

Program Lock Bit				Protection Description
Security Level	LB0	LB1	LB2	
1	U	U	U	No program lock features enabled.
2	P	U	U	MOVC instruction executed from external program memory is disabled from fetching code bytes from internal memory, EA is sampled and latched on reset, and further parallel programming of the Flash is disabled. ISP and software programming with ISP are still allowed.
3	X	P	U	Same as 2, also verify through parallel programming interface is disabled.
4	X	X	P	Same as 3, also external execution is disable.

Table 14. Program Lock Bits

Note: *U : unprogrammed or "1", P : programmed or "0", X: don't care*

Note: *Security level 2 and 3 should only be programmed after Flash and code verification.*

These security bits protect the code access through the parallel programming interface. They are set by default to level 1. Though at level 2, 3 and 4, the code access through the ISP is still possible.

Default Values

The default value of the HSB provides parts ready to be programmed with ISP

- BLJB_EN: BLJB bit is enabled or disabled.(default : disabled)
- BLJB: Programmed force ISP operation(Default : ISP inactivated)).
- LB2-0: Security level four to protect the code from a parallel access with maximum security.(Default : Level 1)

Software Security

The software security provide two different levels of protection for the on-chip code and data,

- Level 1 : No program lock features enabled.
- Level 2 : ISP programming and verify of the Flash is disabled.

Flash Memory Status

HMS99C51S/52S/54S/56S/58S themselves are delivered in standard with the ISP boot code in the Flash memory. After ISP or parallel programming, the possible contents of the Flash memory are summarized on Figure 13.

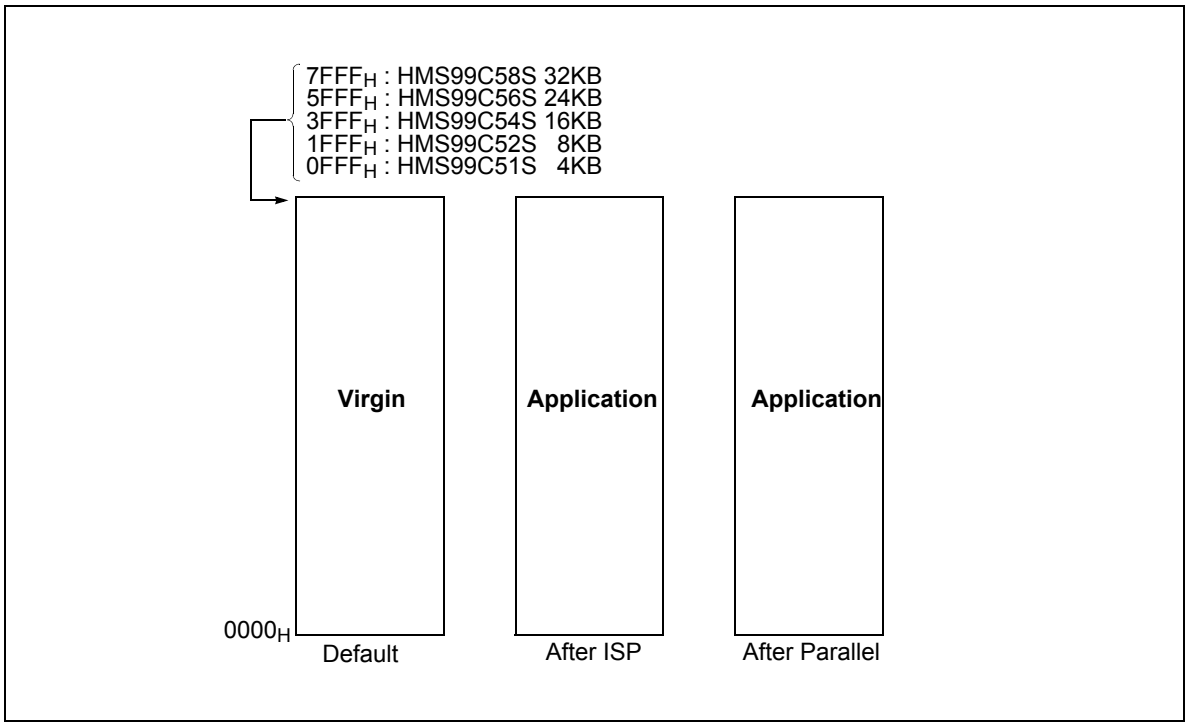


Figure 13. Flash Memory Possible Contents

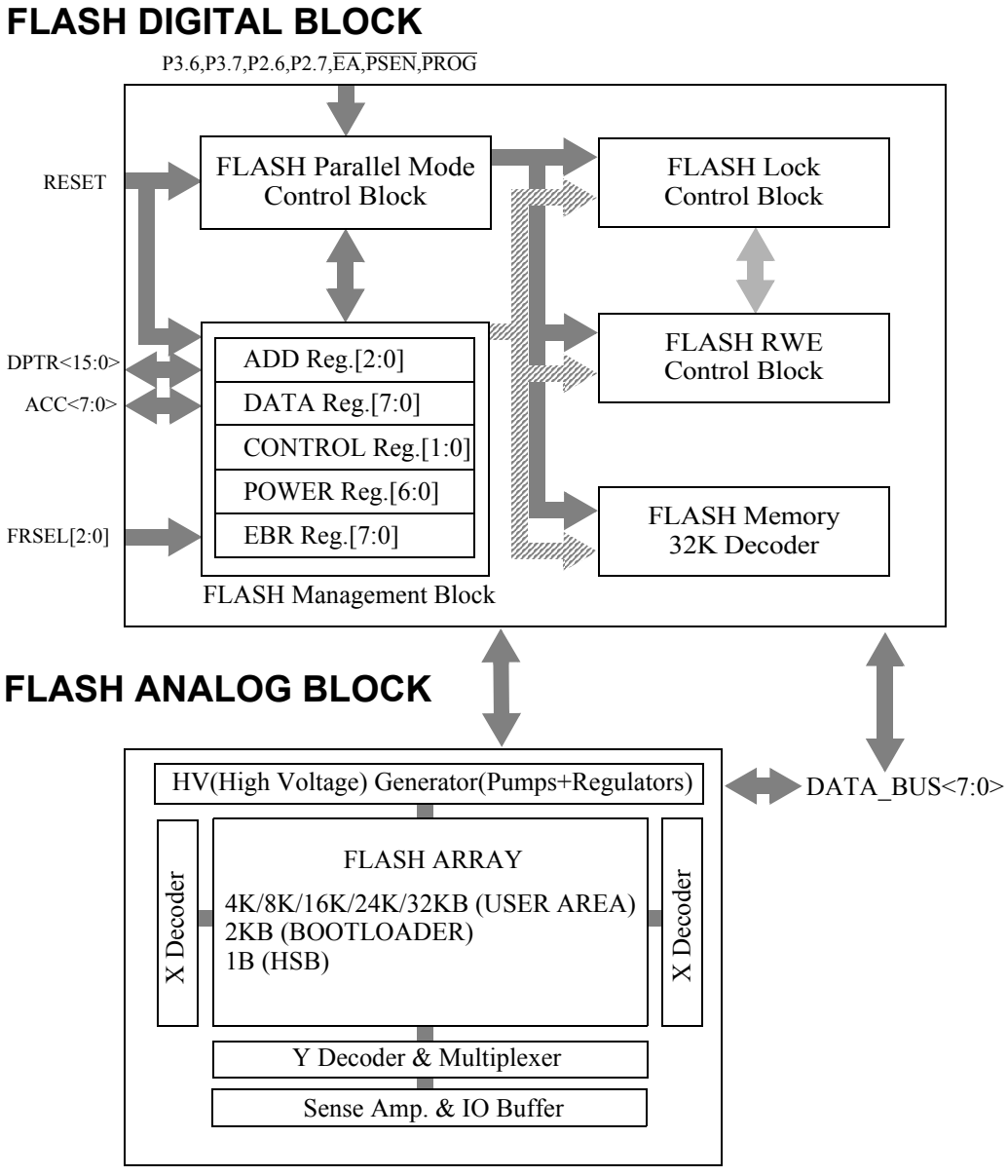
Memory Organization

In the HMS99C51S/52S/54S/56S/58S, the lowest 4K, 8K, 16K, 24K or 32K of the 64 KB program memory address space is filled by internal Flash cells. When the $\overline{EA}$ pin is high, the processor fetches instructions from internal program Flash memory. Bus expansion for accessing program memory from 4K, 8K, 16K, 24K or 32K is upward since external instruction fetches occur automatically when the program counter exceeds 0FFFh(4K), 1FFFh(8K), 3FFFh(16K), 5FFFh(24K) or 7FFFh (32K).

If the $\overline{EA}$ pin is tied low, all program memory fetches are from external memory.

Flash Management Block

- Flash Management Block is controlled by ISP Command
-



SFR Register for a ISP Mode

FCON register exists in $D1_H$ in SFR region and defines selection of Flash register operation, R/W for a Flash Registers, BOOT Flash usage, selection of Flash Memory Space and selection of Program location.

$D0_H$	PCON 00000000_B	**FCON** **00000000_B**	-	-	-	-	$D7_H$

Table 15. SFR Register for a Flash memory

FCON (Flash Control) Register : $D1_H$

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FRSEL2	FRSEL1	FRSEL0	ERASESEL	ENBOOT	INTROM_EN	PGMSEL1	PGMSEL0

Bit No	Bit Mnemonic	Description
7~5	FRSEL[2:0]	**Select Register Operation for Flash Access** This bits define register operation for Flash Memory Access See Figure 16.
4	ERASESEL	**ERASESEL** 0 : Erase Mode is deselected 1 : Erase Mode is selected
3	ENBOOT	**Enable Boot Flash** Cleared to disable boot ROM Set to map the boot ROM between $F800_H$ ~ $0FFFF_H$.
2	INTROM_EN	**Internal ROM Access Enable bit** 0 : External Memory Access over 32K bytes 1 : Internal Memory Access to use Boot ROM
1~0	PGMSEL[1:0]	**The Program Location Select** 00 : Reserved 01 : 1 Byte Program 10 : 4 Byte Program 11 : 8 Byte Program

FRSEL[2:0]	Operation	
0 (000_B) Default	Verify / Read	Reset CONTREG [7:0] DATA_BUS [7:0] → ACC[7:0]
4 (100_B)	Write Address and Data	DPTR[14:0] → ADDREG[14:0] ACC[7:0] → DATAREG[7:0]
5 (101_B)	Write CONTREG	ACC[7:0] → CONTREG[7:0]
6 (110_B)	Write EBR	ACC[7:0] → EBR[6:0]
7 (111_B)	Write PWR	ACC[7:0] → PWR[7:0]

Table 16. Register Operation Table for Flash Access

Bit Position	Name	Function	
Bit0	PGM_SET	Program Power Setup	Positive Gate Pump Setup
Bit1	ER_SET	Erase Power Setup	Negative/Positive Gate Pump Setup
Bit2~Bit7	-	-	Reserved

Table 17. Control Register

Bit Position	Name	Function	Function Effect
Bit7 ~ Bit6	VEEIOPT[1:0]	Define VEEI (Negative Pump Output Value)	00 : VEEI → -09V 01 : VEEI → -10V 10 : VEEI → -11V
Bit5 ~ Bit4	VPPIOPT[1:0]	Define VPPI (Positive Gate Pump Output Value)	00 : VPPI → 09V 01 : VPPI → 10V 10 : VPPI → 11V
Bit3	DNWOPT	Define DNWELL Bias	0 : DNWELL Bias = V_{CC}-V_T 1 : DNWELL Bias = V_{PP}-V_T
Bit2	I_ER_VFY	ER_VFY	1: Down the level to check a erased cell (around 1V) 0: default(around 2V)
Bit1	I_PGM_VFY	PGM_VFY	1: Up the level to check a programmed cell (around 6V) 0: default(around 5V)
Bit0	-	Reserved	For Other Test

Table 18. Power Register

Bit Position	Name	Function	Function Effect
Bit0	EBR0	Erase Block (0000_H~$07FF_H$)	Erase 2K Bytes
Bit1	EBR1	Erase Block (0800_H~$0FFF_H$)	Erase 2K Bytes
Bit2	EBR2	Erase Block (1000_H~$17FF_H$)	Erase 2K Bytes
Bit3	EBR3	Erase Block (1800_H~$1FFF_H$)	Erase 2K Bytes
Bit4	EBR4	Erase Block (2000_H~$3FFF_H$)	Erase 8K Bytes
Bit5	EBR5	Erase Block (4000_H~$5FFF_H$)	Erase 8K Bytes
Bit6	EBR6	Erase Block (6000_H~$7FFF_H$)	Erase 8K Bytes

Table 19. Erase Block Register(EBR)

Bootloader Architecture

Introduction

The bootloader manages a communication according to a specific defined protocol to provide the whole access and service on Flash memory. Furthermore, all accesses and routines can be called from the user application

The Flash bootloader includes:

- The serial communication protocol
- The ISP command decoder
  In order to access User FLASH area at a custom bootloader, User must modify the related FLASH registers directly.

This may be necessary in case of :

- Another communication interface
- Different protocol (other data format, encrypted data, etc.)
- Flash areas protection
- Flash areas checks (CRC, etc.)

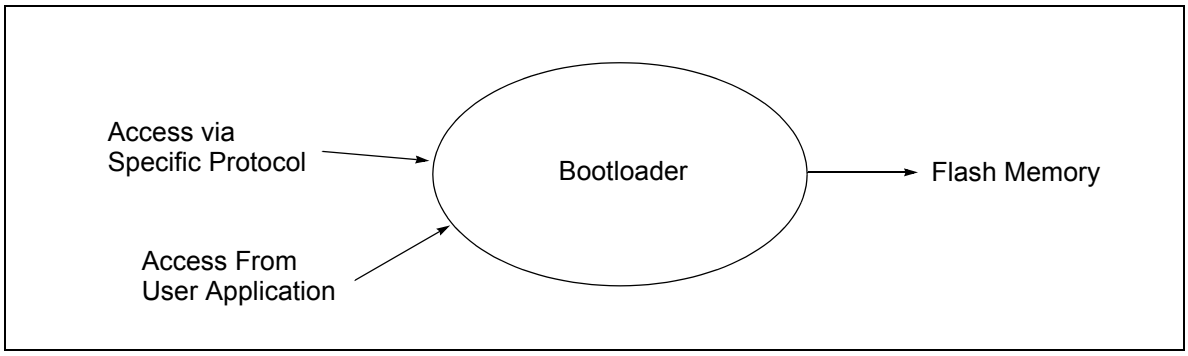


Figure 14. Diagram Context Description

Bootloader Functional Description

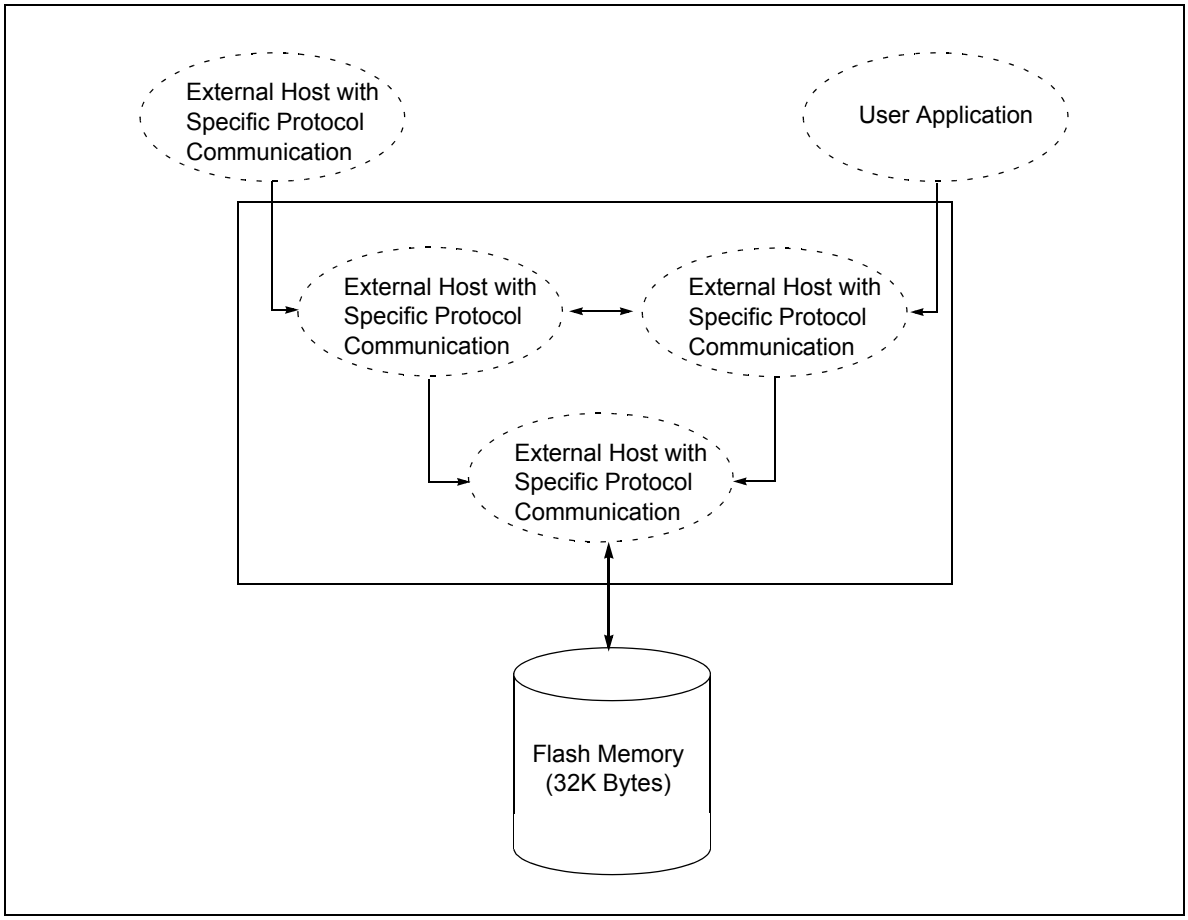


Figure 15. Bootloader Functional Description

On the above diagram, the on-chip bootloader processes are:

- ISP Communication Management

The purpose of this process is to manage the communication and its protocol between the on-chip bootloader and a external device. The on-chip boot ROM implement a serial protocol. This process translate serial communication frame (UART) into Flash memory access (read, write, erase, ...)

- Flash Memory Management

This process manages low level access to Flash memory (performs read, write and erase access).

Bootloader Process

The bootloader can be activated by Hardware conditions. The Hardware conditions ($\overline{PSEN}$ = 0, $\overline{EA}$ = 1, ALE = 1) during the Reset falling edge force the on-chip bootloader execution. This allows an application to be built that will normally execute the end user's code but can be manually forced into default ISP operation. As $\overline{PSEN}$ is a an output port in normal operating mode (running user application or bootloader code) after reset, it is recommended to release $\overline{PSEN}$ after falling edge of reset signal. The hardware conditions are sampled at reset signal falling edge, thus they can be released at any time when reset input is low. The on-chip bootloader boot process is shown in Figure 16.

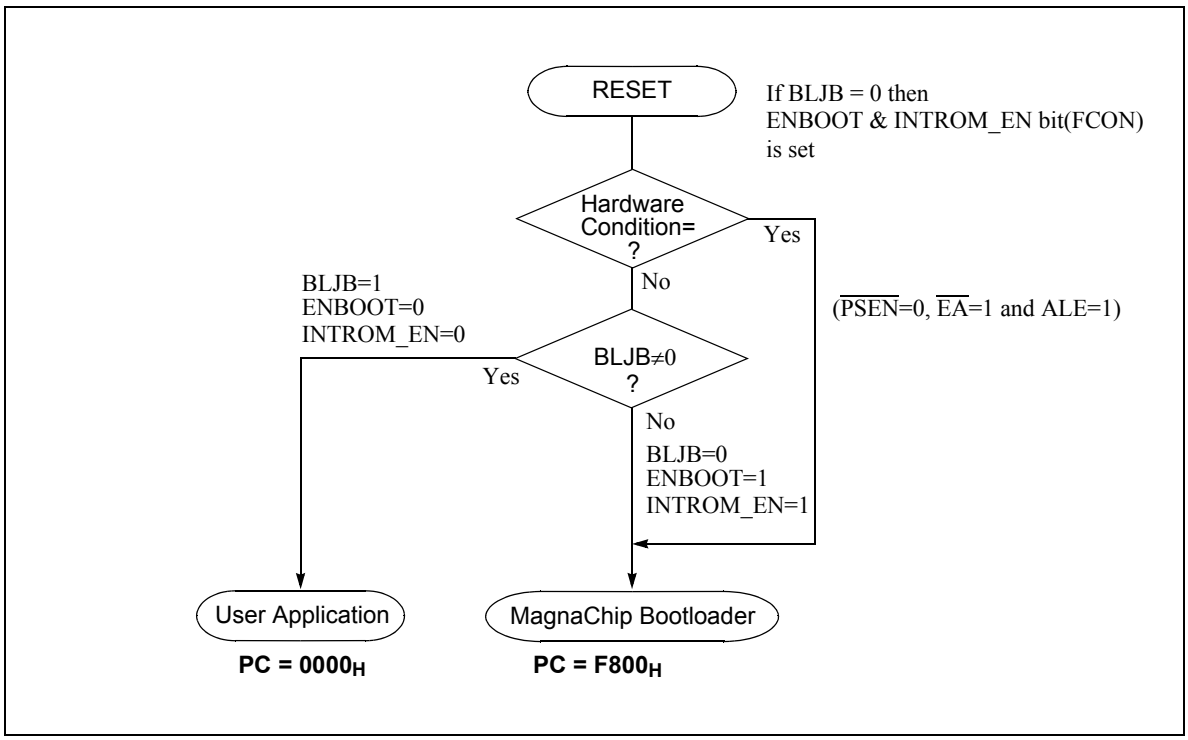


Figure 16. Bootloader Process by hardware

•

IN-SYSTEM PROGRAMMING (ISP)

The In-System Programming (ISP) is performed without removing the microcontroller from the system. The ISP facility consists of a series of internal hardware resources coupled with internal firmware to facilitate remote programming of the HMS99C51S/52S/54S/56S/58S through the serial port. The MagnaChip Microcontrollers ISP facility has made in-circuit programming in an embedded application possible with a minimum of additional expense in components and circuit board area. The ISP function through UART uses four pins: TxD, RxD, V_{SS}, and V_{CC}. Only a small connector needs to be available to interface the application to an external circuit in order to use this feature.

Using In-System Programming (ISP)

The ISP feature allows a wide range of baud rates in the user application. It is also adaptable to a wide range of oscillator frequencies. This is accomplished by measuring the bit-time of a single bit in a received character. This information is then used to program the baud rate in terms of timer counts based on the oscillator frequency.

The ISP feature requires that an initial character(an uppercase U) be sent to the HMS99C51S/52S/54S/56S/58S to establish the baud rate. The ISP firmware provides auto-echo of received characters. Once baud rate initialization has been performed, the ISP firmware will only accept Intel Hex-type records. Intel Hex records consist of ASCII characters used to represent hexadecimal values and are summarized below:

:NNAAAARRDD...DDCC

HMS99C51S/52S/54S/56S/58S will accept up to 16(10_H) data bytes. The “AAAA” string represents the address of the first byte in the record. If there are zero bytes in the record, this field is often set to “0000”. The “RR” string indicates the record type. A record type of “00” is a data record. A record type of “01” indicates the end-of-file mark. In this application, additional record types will be added to indicate either commands or data for the ISP facility. The “DD” string represents the data bytes. The maximum number of data bytes in a record is limited to 16(decimal). The “CC” string represents the checksum byte. ISP commands are summarized in Table 22. As a record is received by the HMS99C51S/52S/54S/56S/58S, the information in the record is stored internally and a checksum calculation is performed and compared to “CC”. The operation indicated by the record type is not performed until the entire record has been received. Should an error occur in the checksum, the HMS99C51S/52S/54S/56S/58S will send an “X” out the serial port indicating a checksum error. If the checksum calculation is found to match the checksum in the record, then the command will be executed. In most cases, successful reception of the record will be indicated by transmitting a “.” character out the serial port(displaying the contents of the internal program memory is an exception). In the case of a Data Record(record type “00”), an additional check is made. A “.” character will NOT be sent unless the record checksum matched the calculated checksum and all of the bytes in the record were successfully programmed.

For a data record, an “X” indicates that the checksum failed to match, and an “R” character indicates that one of the bytes did not properly program.

MagnaChip ISP, a software utility to implement ISP programming with a PC, is available from the MagnaChip web site. The users of this ISP function should use this MagnaChip ISP software for proper flash ROM control and operation.

RECORD TYPE	COMMAND/DATA FUNCTION
00	Program Data Record :nnaaaa00dd. . . . ddcc Where: nn = number of bytes(hex) in record aaaa = memory address of first byte in record dd....dd=databytes cc = checksum Example: :05008000AF5F67F060B6
01	End of File (EOF), no operation :xxxxxx01cc Where: xxxxxx = required field, but value is a “don’t care” cc = checksum Example: :00000001FF
02	Specify Erase/Write Pulse :03xxxx02wweellcc Where: xxxx = required field, but value is a “don’t care” ww = write pulse ee = erase pulse high byte ll = erase pulse low byte cc = checksum Example: :03000002789C40A7
03	Miscellaneous Write Functions :nnxxxx03ffssddcc Where: nn = number of bytes(hex) in record xxxx = required field, but value is a “don’t care” 03 = Write Function ff = subfunction code ss = selection code dd = data input(as needed) cc = checksum Subfunction Code = 01(Erase Block) ff = 01 ss = block index in bits 6:0 (block number is designated by bit position) Example: :020000030122D8 erase block 1 and 5 (position of bit 1 and 5) Subfunction Code = 05 (Program Software Security Bits) ff = 05 program software security bit (Level 2 inhibit reading/writing to Flash) Example: :0100000305F7 (program security bit)

Table 20. Intel-Hex Records Used by In-system Programming

RECORD TYPE	COMMAND/DATA FUNCTION
04	Display Device Data or Blank Check Record type 04 causes the contents of the entire Flash array to be sent out the serial port in a formatted display. This display consists of an address and the contents of 16 bytes starting with that address. No display of the device contents will occur if security bit 2 has been programmed. The dumping of the device data to the serial port is terminated by the reception of any character. General Format of Function 04 :05xxxx04sssseeeeffcc Where: 05 = number of bytes (hex) in record xxxx = required field, but value is a “don’t care” 04 = “Display Device Data or Blank Check” function code ssss = starting address eeee = ending address ff = subfunction 00 = display data 01 = blank check cc = checksum Example: :0500000440004FFF0069 (display 4000_H ~ $4FFF_H$)
05	Miscellaneous Read Functions General Format of Function 05 :02xxxx05ffsscc Where: 02 = number of bytes (hex) in record xxxx = required field, but value is a “don’t care” 05= “Miscellaneous Read” function code ffss = subfunction and selection code 0001 = read copy of the signature byte - device ID (Family code) 0700 = read the software security bits 0703 = read the oscillation information cc = checksum Example: :020000050001F0 (read copy of the signature byte - device id)

Table 20. Intel-Hex Records Used by In-system Programming

Command	Command Name	Data[0]	Data[1]	Command Effect
00_H	Program Data	-	-	Program Data Byte Bootloader will accept up to 128 data bytes.
01_H	End Of File	-	-	End Of File
02_H	Specify E/W Pulse	Data[0] = write times low Data[1] = erase times high Data[2] = erase times low		Erase/Write pulse information setup
03_H	Write function	01H	Block index	Bit 0 : Erase Blk0(0000_H~$07FF_H$) ; 2K bytes Bit 1 : Erase Blk1(0800_H~$0FFF_H$) ; 2K bytes Bit 2 : Erase Blk2(1000_H~$17FF_H$) ; 2K bytes Bit 3 : Erase Blk3(1800_H~$1FFF_H$) ; 2K bytes Bit 4 : Erase Blk4(2000_H~$3FFF_H$) ; 8K bytes Bit 5 : Erase Blk5(4000_H~$5FFF_H$) ; 8K bytes Bit 6 : Erase Blk0(6000_H~$7FFF_H$) ; 8K bytes Bit 7 : not used
		05H	-	Program Security Lockbit
		07H	-	Erase User Memory fully(max. 32K bytes)
04_H	Display function	Data[0:1] = start address Data[2:3] = end address Data[4] = 00h(Display) Data[4] = 01h(Blank check)		Display Data/Blank Check
05_H	Read Function*	00_H	01_H	Read Device id
		07_H	00_H	Read Security Information
			03_H	Read Oscillator Information

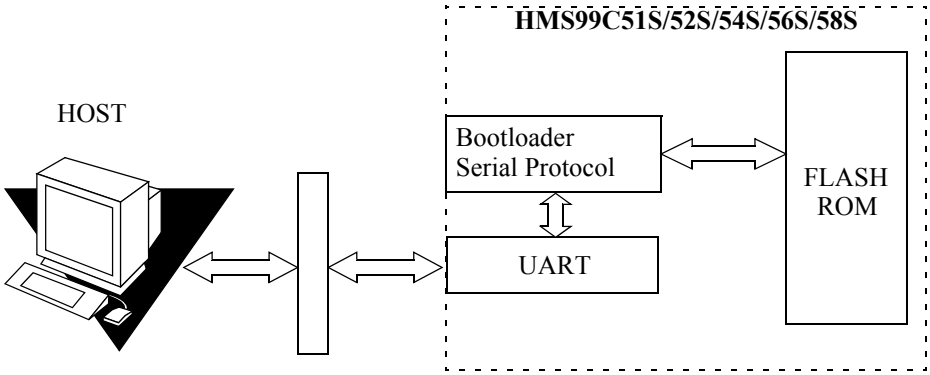
Table 21. ISP Commands Summary

Serial Protocol

This application note describes the Serial Protocol used to program the Flash code memory from MagnaChip microcontrollers. Commands sent over the serial line are interpreted by the on-chip bootloader program.

This applied for HMS99C51S/52S/54S/56S/58S.

This protocol is a serial UART protocol.



Protocol Configuration

1. Physical Layer

The UART used to transmit information has the following configuration:

- Character: 8-bit data
- Parity: none
- Stop: 1 bit
- Flow control: none
- Baudrate: autobaud is performed by the bootloader to compute the baudrate chosen by the host.

2. Frame Description

The Serial Protocol is based on the Intel Hex-type records.

Intel Hex records consist of ASCII characters used to represent hexadecimal values and are summarized below:

Record Mark ‘ : ’	Reclen	Load Offset	Record Type	Data or Info	Checksum
1 Byte	1 Byte	2 Byte	1 Byte	n Byte	1 Byte

- Record Mark:

  Record Mark is the start of frame. This field must contain ':'.

- Reclen:

  Reclen specifies the number of bytes of information or data which follows the Record Type field of the record.

- Load Offset:

  Load Offset specifies the 16-bit starting load offset of the data bytes, therefore this field is used only for Data Program Record(see Table 20).

- Record Type:

  Record Type specifies the command type. This field is used to interpret the remaining information within the frame. The encoding for all the current record types is described in Table 20.

- Data/Info:

  Data/Info is a variable length field. It consists of zero or more bytes encoded as pairs of hexadecimal digits. The meaning of data depends on the Record Type.

- Checksum:

  The two's complement of bytes that result from converting each pair of ASCII hexadecimal digits to one byte of binary, and including the Reclen field to and including the last byte of the Data/Info field. Therefore, the sum of all the ASCII pairs in a record after converting to binary, from the Reclen field to and including the Checksum field, is zero.

Note: *1. A data byte is represented by two ASCII characters*
2. When the field Load Offset is not used, it should be coded as four ASCII zero characters ('0').

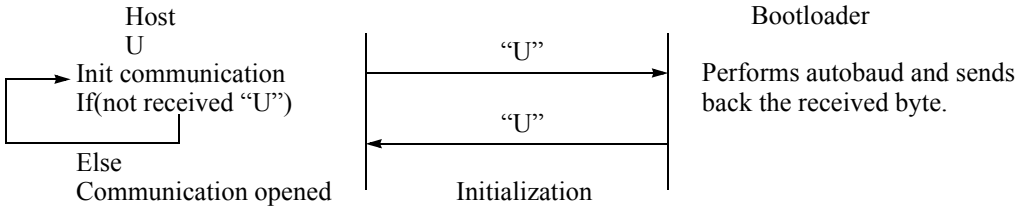
Protocol Description

1.Overview

An initialization step must be performed after each Reset. After microcontroller reset, the bootloader waits for an autobaud sequence. When the communication is initialized, the protocol depends on the record type requested by the host.

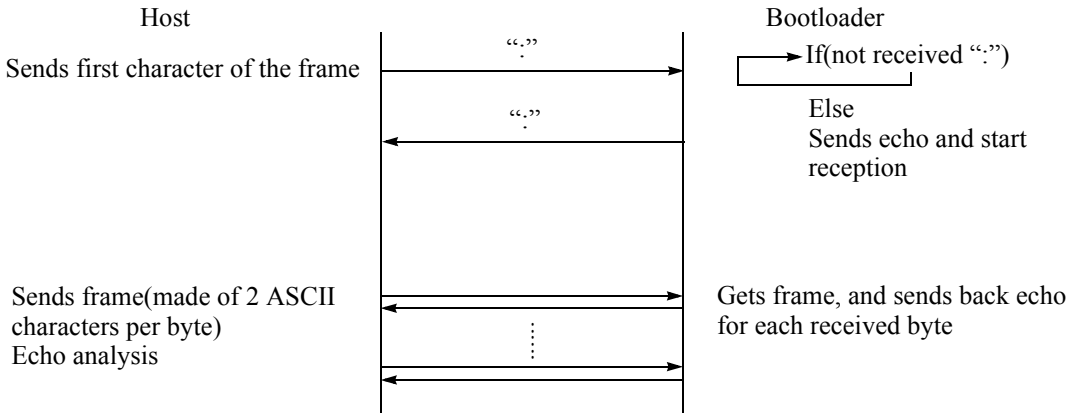
2.Communication Initialization

The host initializes the communication by sending a “U” character to help the bootloader to compute the baudrate (autobaud).



3.Command Data Stream Protocol

All commands are sent using the same flow.



Command Flow

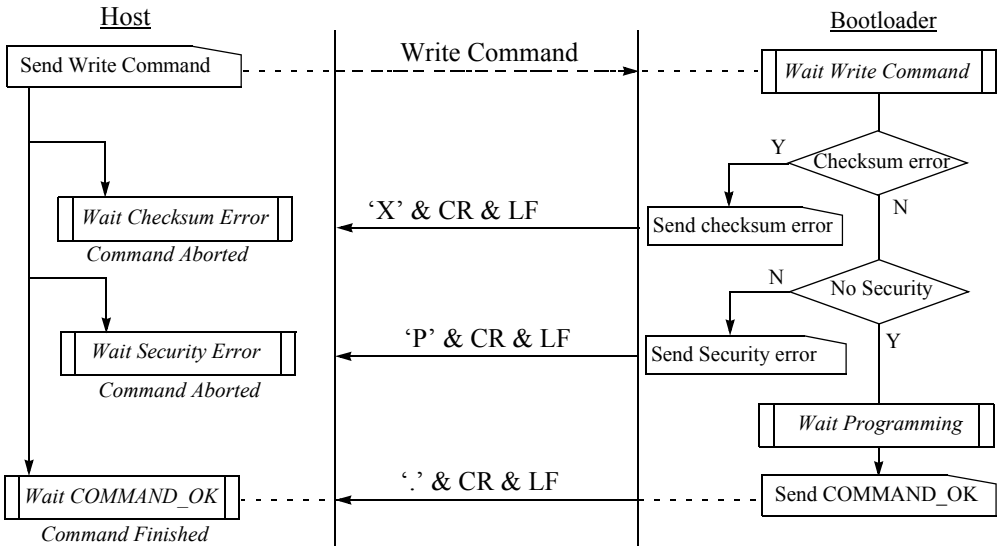
Note: All commands sent with the echo mechanism will be represented by:

3.1 Write / Program Commands

This flow is common to the following frames:

- Flash Programming Data Frame
- EOF or MagnaChip Frame (only Programming MagnaChip Frame)
- Erase/Write Timing Frame
- Lockbit Programming Data Frame

a. Description



Erase/Write Flow

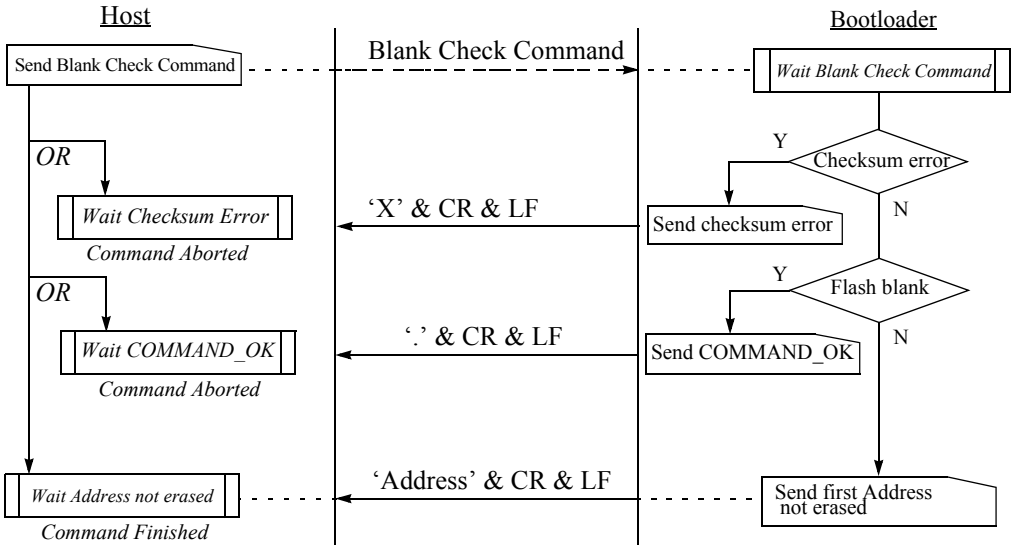
b. Example

HOST : 01 0010 00 55 9A
BOOTLOADER : 01 0010 00 55 9A . CR LF
Programming Data (write 55_H at address 0010_H in the Flash)

HOST : 02 0000 03 05 01 F5
BOOTLOADER : 02 0000 03 05 01 F5. CR LF
Programming Lockbit function (write Software Security to level 2)

3.2 Blank Check Command

a. Description



Blank Check Flow

b. Example

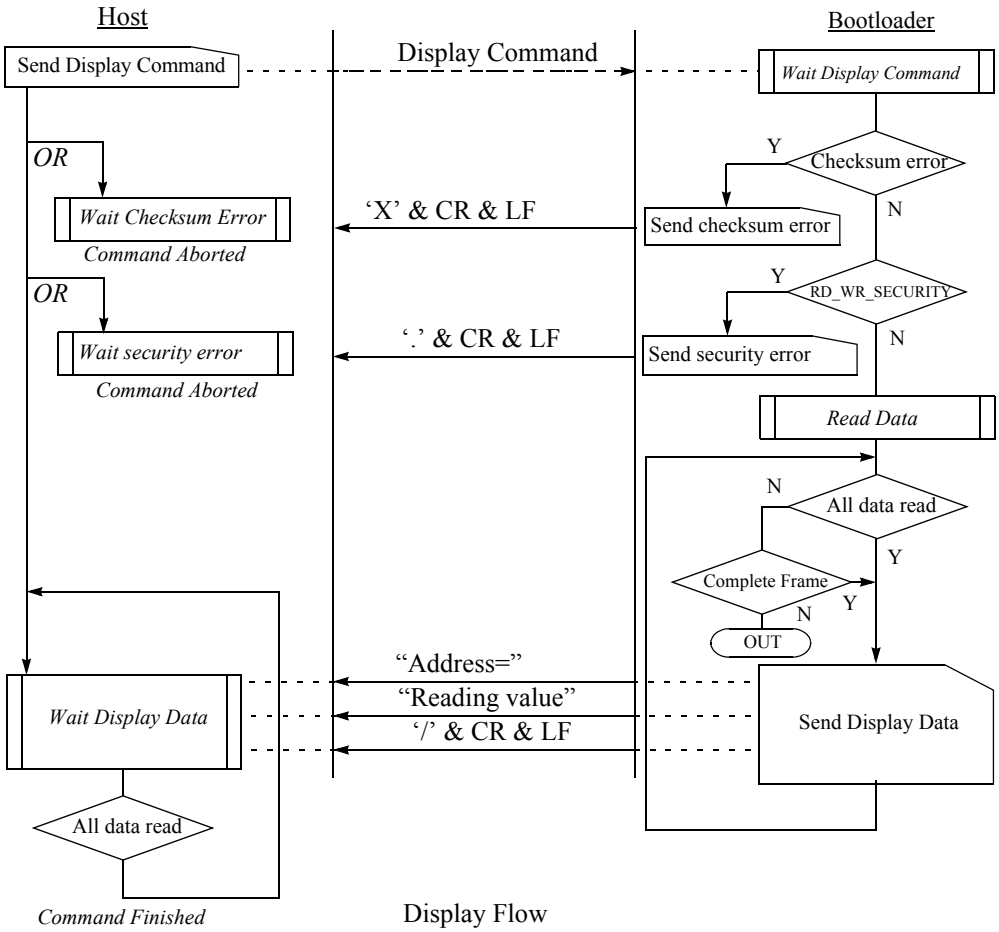
HOST : 05 0000 04 0000 7FFF 01 78
BOOTLOADER : 05 0000 04 0000 7FFF 01 78 . CR LF
Blank Check ok

BOOTLOADER : 05 0000 04 0000 7FFF 01 70 X CR LF CR LF
Blank Check with checksum error

HOST : 05 0000 04 0000 7FFF 01 70
BOOTLOADER : 05 0000 04 0000 7FFF 01 78 xxxx CR LF
Blank Check failure at address xxxx

3.3 Display Data

a. Description



Display Flow

Note: The maximum size of display block is equal to the Flash ROM size.

b. Example

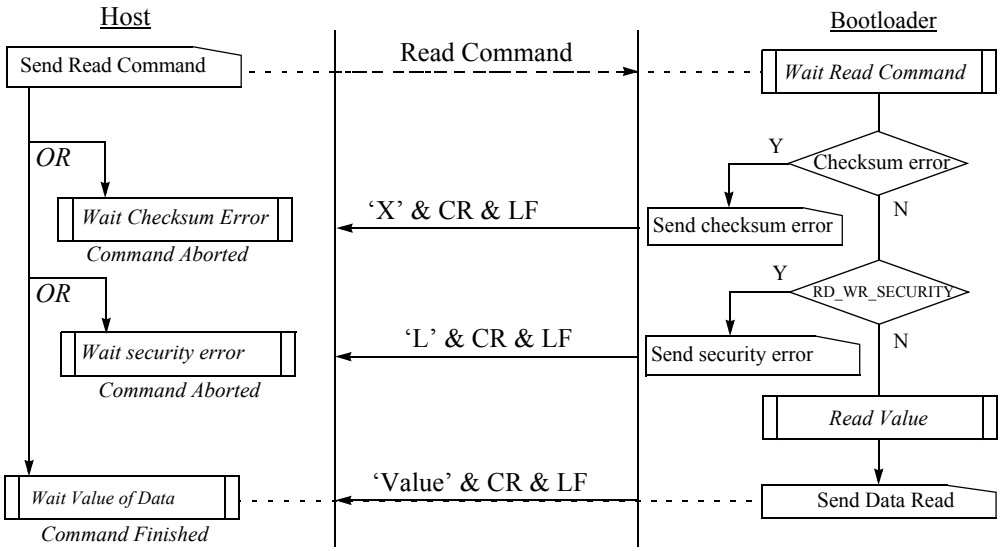
HOST : 05 0000 04 0000 0020 00 D7
BOOTLOADER : 05 0000 04 0000 0020 00 D7
BOOTLOADER 0000=-----data------ / CR LF (16 data)
BOOTLOADER 0010=-----data------ / CR LF (16 data)
BOOTLOADER 0020=data CR LF (1 data)
Display data from address 0000_H to 0020_H

3.4. Read Function

This flow is similar for the following frames:

- Reading Frame
- EOF Frame/ MagnaChip Frame (only reading MagnaChip Frame)

a. Description



Read Flow

b. Example

HOST : 02 0000 05 00 01 F8
BOOTLOADER : 02 0000 05 00 01 F8 Value . CR LF
Read function (read device ID)

ISP METHOD FOR PC(MAGNACHIP WINISP)

Getting Started / Installation

The following section details the procedure for accomplishing the installation procedure.

1. Connect the serial(RS-232C) cable between a target board and the COM1 serial port of your PC.

2. Configure the COM1 serial port of your PC as following.

- Baudrate : 115,200
- Data bit : 8
- Parity : No
- Stop bit : 1
- Flow control : No

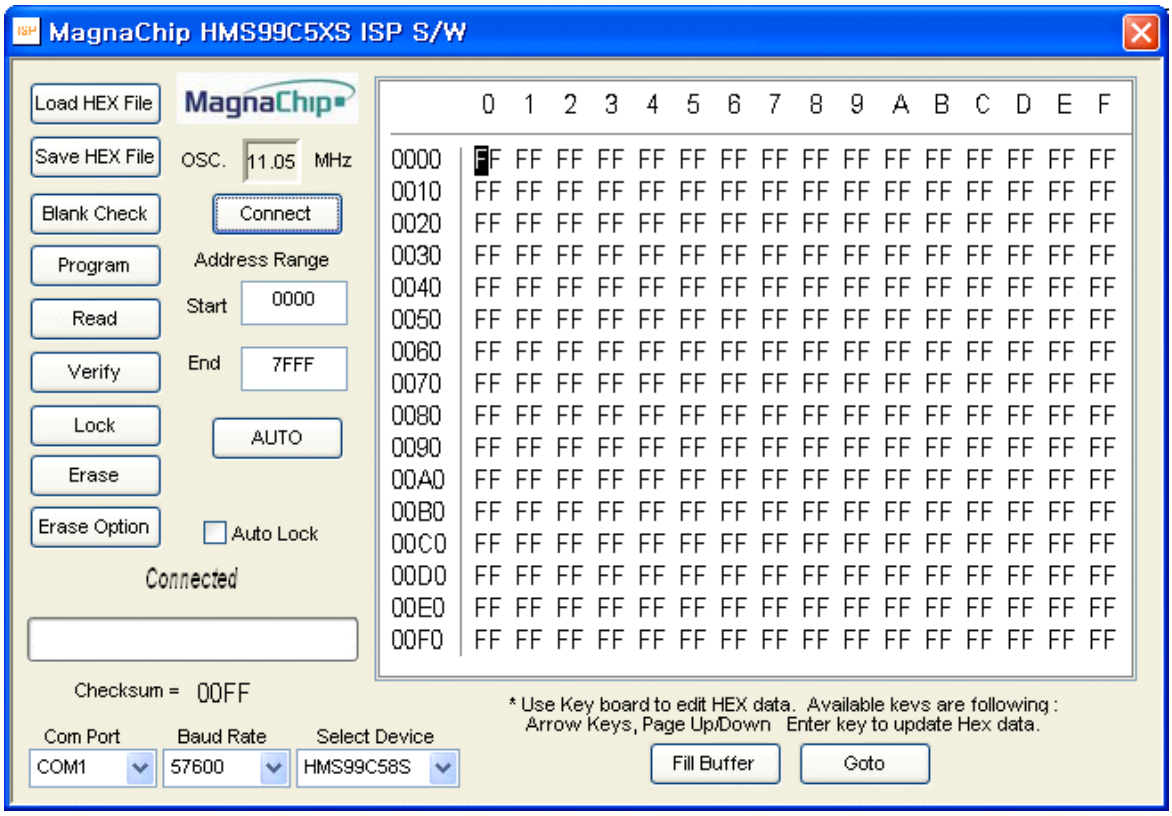
3. Turn your target B/D power switch ON.
Your target B/D must be configured to enter the ISP mode.

4. Run the MagnaChip ISP software.

5. Press the Reset Button in the ISP S/W.

If the status windows shows a message as "Connected", all the conditions for ISP are provided.
If you press the Reset button again after connected, the status windows will show the message as "Disconnected". Please discard it because the HMS99C51S/52S/54S/56S/58S can not check the reset button after connected successfully.

Basic Information



Function	Description
Load HEX File	Load the data from the selected file storage into the memory buffer.
Save HEX File	Save the current data in your memory buffer to a disk storage by using the Intel HEX format.
Erase	Erase the data in your target MCU before programming it.
Blank Check	Verify whether or not a device is in an erased or unprogrammed state.
Program	This button enables you to place new data from the memory buffer into the target device.
Read	Read the data in the target MCU into the buffer for examination. The checksum will be displayed on the checksum box.
Verify	Assures that data in the device matches data in the memory buffer. If your device is secured, a verification error is detected.
Lock	Secures devices so that their content can no longer be examined or modified.

Table 22. ISP Function Description

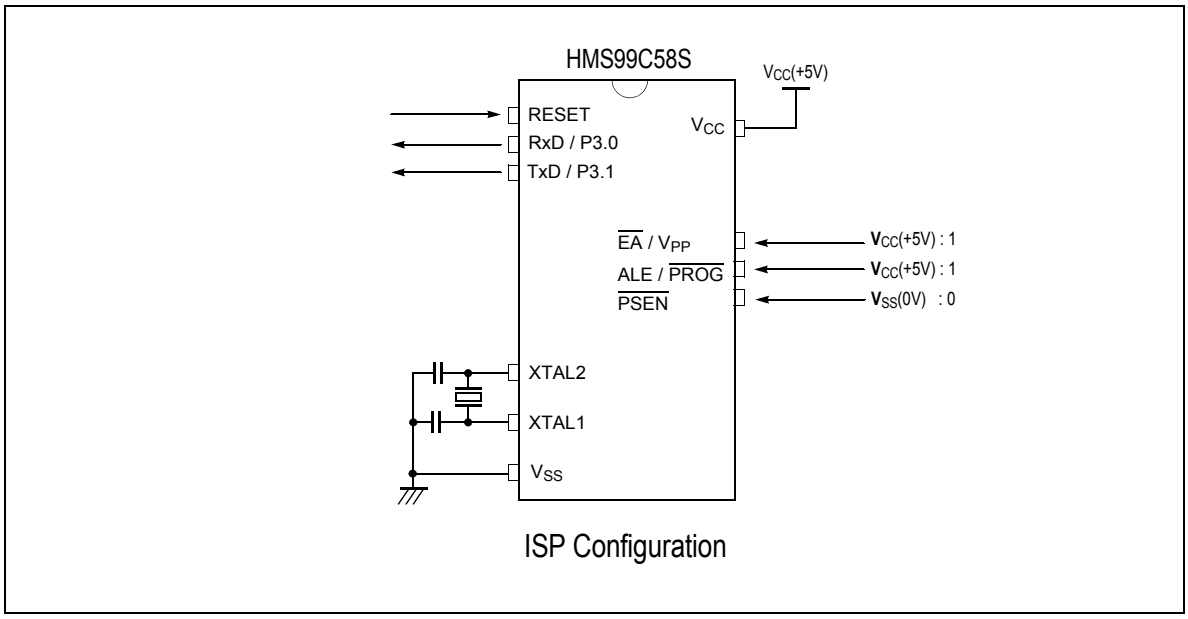
Function	Description
Erase Option	Select blocks for Erasure.
AUTO	Blank Check & Program & Verify
Auto Lock	If selected with check mark, the security locking is performed after erasure.
Connect	Connect a MCU in your target Board with displaying as “Connected” in the status box. Users have to press this button at least one time to initialize a target MCU for entering the ISP mode. If failed to enter the ISP mode, all the buttons are unavailable. And, after entering successfully, the Connect button will be unavailable.
Edit Buffer	Modify the data in the selected address in your buffer memory
Fill Buffer	Fill the selected area with a data.
Goto	Display the selected page.
OSC. _____ MHz	Display your target system’s oscillator value with discarding below point.
Start _____	Starting address
End _____	End address
Checksum=8000	Display the checksum(Hexdecimal) after reading the target device.
Com Port	Select serial port.
Baud Rate	Select UART baud rate.
Select Device	Select target device.
Page Up Key	Display the previous page of your memory buffer.
Page Down Key	Display the higher page than the current location.

Table 22. ISP Function Description

Hardware Conditions to Enter the ISP Mode

The In-System Programming (ISP) is performed without removing the microcontroller from the system. The In-System Programming(ISP) facility consists of a series of internal hardware resources coupled with internal firmware through the serial port. The In-System Programming (ISP) facility make in-circuit programming in an embedded application possible with a minimum of additional expense in components and circuit board area.

The bootloader can be executed by holding $\overline{PSEN}$ LOW, $\overline{EA}/V_{PP}$ greater than V_{IH} (such as +5V), and ALE/$\overline{PROG}$ HIGH at the falling edge of RESET. The ISP function block uses four pins: TxD, RxD, V_{SS}, and V_{CC}. Only a small connector needs to be available to interface your application to an external circuit in order to use this feature.



ISP Configuration