No. 5486

LC66E2316

Four-Bit Single-Chip Microcontroller with 16 KB of On-Chip EPROM

Preliminary

Overview

The LC66E2316 is an on-chip EPROM version of the LC6623XX Series CMOS 4-bit single-chip microcontrollers. The LC66E2316 provides the same functions as the LC662316A, and is pin compatible with that product. Since the LC66E2316 is provided in a window package, it can be reprogrammed repeatedly and is thus optimal for program development.

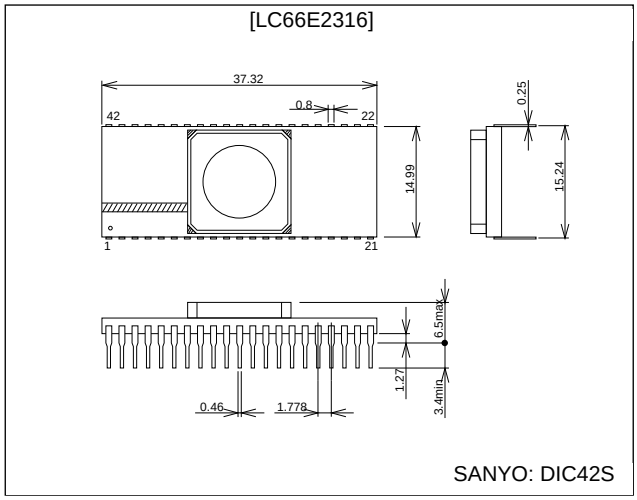
Features and Functions

- On-chip EPROM capacity of 16 kilobytes, and an on-chip RAM capacity of 512 × 4 bits.
- Fully supports the LC66000 Series common instruction set (128 instructions).
- I/O ports: 36 pins
- DTMF generator
  This microcontroller incorporates a circuit that can generate two sine wave outputs, DTMF output, or a melody output for software applications.
- 8-bit serial interface: one circuit
- Instruction cycle time: 0.95 to 10 µs (at 4.5 to 5.5 V)
- Powerful timer functions and prescalers
  — Time limit timer, event counter, pulse width measurement, and square wave output using a 12-bit timer.
  — Time limit timer, event counter, PWM output, and square wave output using an 8-bit timer.
  — Time base function using a 12-bit prescaler.
- Powerful interrupt system with 10 interrupt factors and 7 interrupt vector locations.
  — External interrupts: 3 factors/3 vector locations
  — Internal interrupts: 4 factors/4 vector locations
  (Waveform output internal interrupts: 3 factors and 1 vector; shared with external expansion interrupts)
- Flexible I/O functions
  Selectable options include 20-mA drive outputs, inverter circuits, pull-up and open-drain circuits.
- Optional runaway detection function (watchdog timer)
- 8-bit I/O functions
- Power saving functions using halt and hold modes.
- Packages: DIC42S (window), QFC48 (window)
- Evaluation LSIs: LC66599 (evaluation chip) + EVA800/850 - TB662YXX2

Package Dimensions

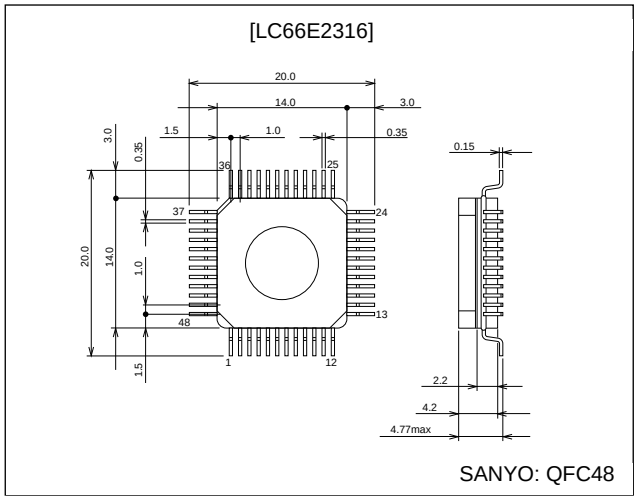
unit: mm

3127-DIC42S



unit: mm

3157-QFC48



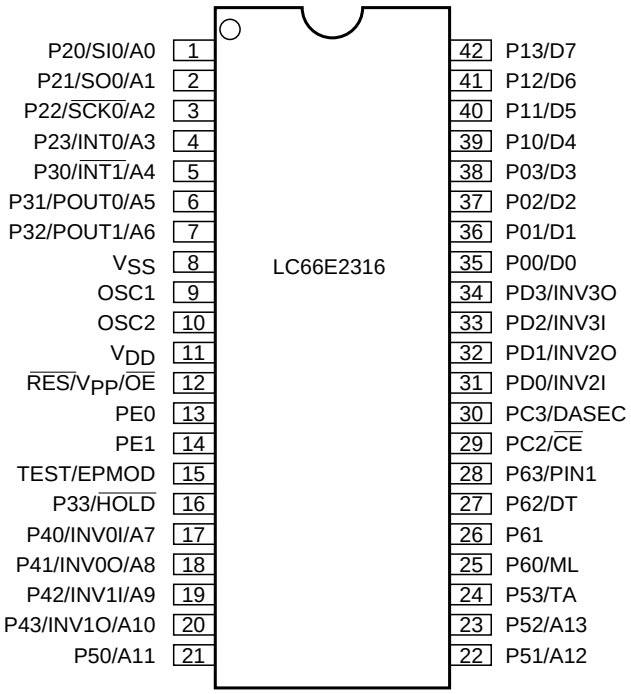
Series Organization

Type No.	No. of pins	ROM capacity	RAM capacity	Package		Features
LC66304A/306A/308A	42	4 K/6 K/8 KB	512 W	DIP42S	QFP48E	Normal versions 4.0 to 6.0 V/0.92 µs
LC66404A/406A/408A	42	4 K/6 K/8 KB	512 W	DIP42S	QFP48E	
LC66506B/508B/512B/516B	64	6 K/8 K/12 K/16 KB	512 W	DIP64S	QFP64A	
LC66354A/356A/358A	42	4 K/6 K/8 KB	512 W	DIP42S	QFP48E	Low-voltage versions 2.2 to 5.5 V/3.92 µs
LC66354S/356S/358S	42	4 K/6 K/8 KB	512 W		QFP44M	
LC66556A/558A/562A/566A	64	6 K/8 K/12 K/16 KB	512 W	DIP64S	QFP64E	
LC66354B/356B/358B	42	4 K/6 K/8 KB	512 W	DIP42S	QFP48E	Low-voltage high-speed versions 3.0 to 5.5 V/0.92 µs
LC66556B/558B/562B/566B	64	6 K/8 K/12 K/16 KB	512 W	DIP64S	QFP64E	
LC66354C/356C/358C	42	4 K/6 K/8 KB	512 W	DIP42S	QFP48E	2.5 to 5.5 V/0.92 µs
LC662104A/06A/08A	30	4 K/6 K/8 KB	384 W	DIP30SD	MFP30S	On-chip DTMF generator versions 3.0 to 5.5 V/0.95 µs
LC662304A/06A/08A/12A/16A	42	4 K/6 K/8 K/12 K/16 KB	512 W	DIP42S	QFP48E	
LC662508A/12A/16A	64	8 K/12 K/16 KB	512 W	DIP64S	QFP64E	
LC665304A/06A/08A/12A/16A	48	4 K/6 K/8 K/12 K/16 KB	512 W	DIP48S	QFP48E	Dual oscillator support 3.0 to 5.5 V/0.95 µs
LC66E308	42	EPROM 8 KB	512 W	DIC42S with window	QFC48 with window	Window and OTP evaluation versions 4.5 to 5.5 V/0.92 µs
LC66P308	42	OTPROM 8 KB	512 W	DIP42S	QFP48E	
LC66E408	42	EPROM 8 KB	512 W	DIC42S with window	QFC48 with window	
LC66P408	42	OTPROM 8 KB	512 W	DIP42S	QFP48E	
LC66E516	64	EPROM 16 KB	512 W	DIC64S with window	QFC64 with window	
LC66P516	64	OTPROM 16 KB	512 W	DIP64S	QFP64E	
LC66E2108*	30	EPROM 8 KB	384 W			Window evaluation versions 4.5 to 5.5 V/0.92 µs
LC66E2316	42	EPROM 16 KB	512 W	DIC42S with window	QFC48 with window	
LC66E2516	64	EPROM 16 KB	512 W	DIC64S with window	QFC64 with window	
LC66E5316	52/48	EPROM 16 KB	512 W	DIC52S with window	QFC48 with window	
LC66P2108*	30	OTPROM 8 KB	384 W	DIP30SD	MFP30S	OTP 4.0 to 5.5 V/0.95 µs
LC66P2316*	42	OTPROM 16 KB	512 W	DIP42S	QFP48E	
LC66P2516	64	OTPROM 16 KB	512 W	DIP64S	QFP64E	
LC66P5316	48	OTPROM 16 KB	512 W	DIP48S	QFP48E	

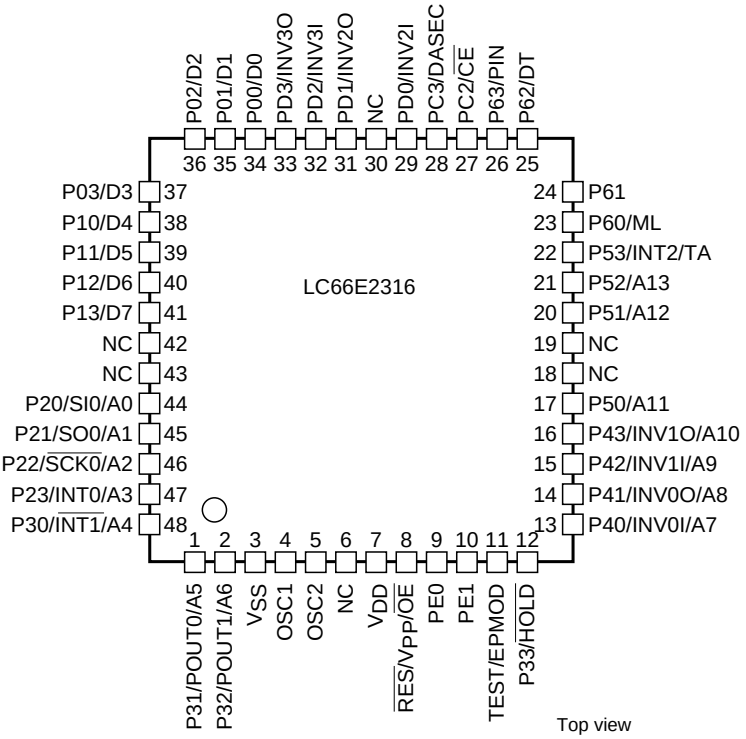
Note: * Under development

Pin Assignments

DIC42S



QFC48

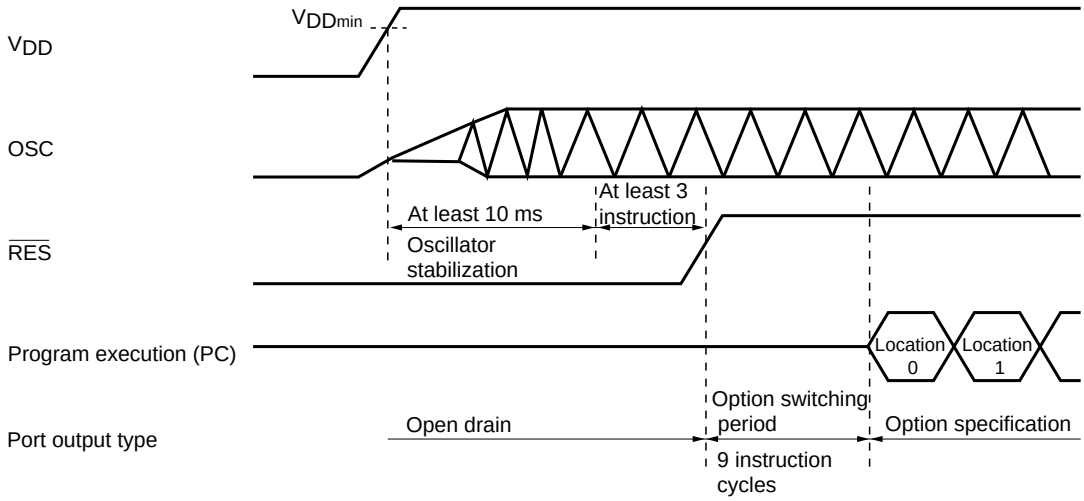


Top view

Usage Notes

The LC66E2316 was created for program development, product evaluation, and prototype development for products based on the LC6623XX Series microcontrollers. Keep the following points in mind when using this product.

1. After a reset
   The $\overline{\text{RES}}$ pin must be held low for an additional 3 instruction cycles after the oscillator stabilization period has elapsed. Also, the port output circuit types are set up during the 9 instruction cycles immediately after $\overline{\text{RES}}$ is set high. Only then is the program counter set to 0 and program execution started from that location. (The port output circuits all revert to the open-drain type during periods when $\overline{\text{RES}}$ is low.)



2. Notes on LC6623XX evaluation
   The high end of the EPROM area (locations 3FF0H to 3FFFH) are the option specification area. Option specification data must be programmed for and loaded into this area. The Sanyo specified cross assembler for this product is the program LC66S.EXE. Also, insert JMP instructions so that user programs do not attempt to execute addresses that exceed the capacity of the mask ROM, and write zeros (00H) to areas (other than 3FF0H to 3FFFH) that exceed the actual capacity of the mask ROM.
3. Always apply an opaque seal to the window on the LC66E2316 package when actually using the device.

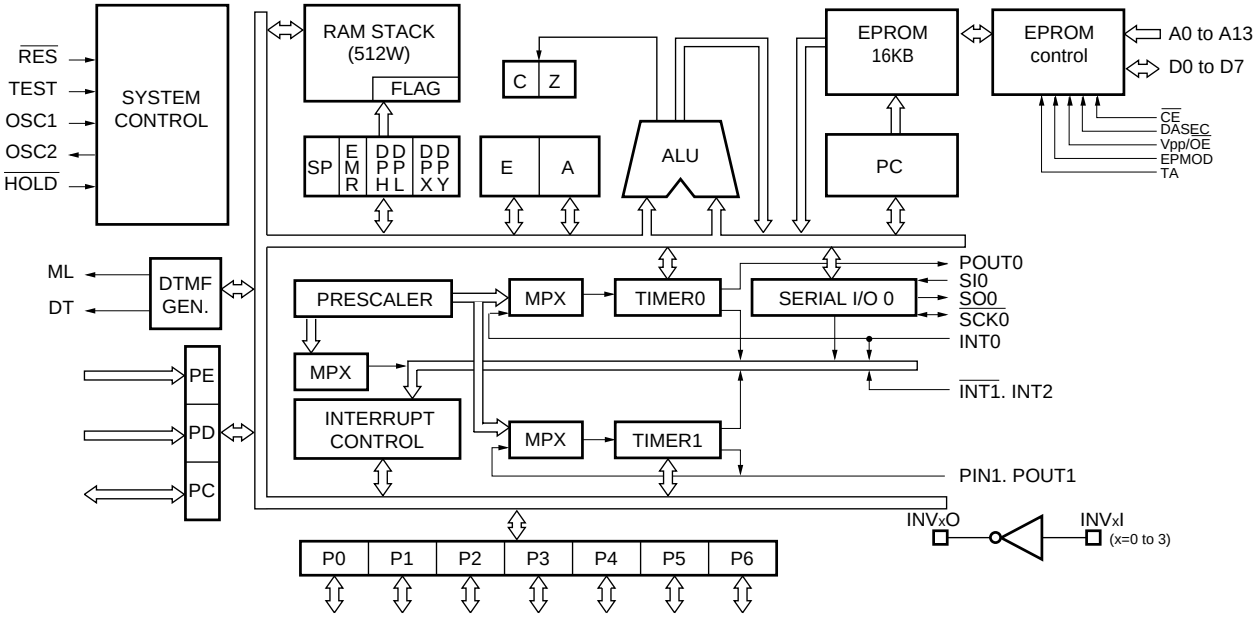
Main differences between the LC66E2316, LC66P2316, and LC6623XX Series

Item	LC6623XX Series (mask version)	LC66E2316	LC66P2316
Differences in the main characteristics • Operating temperature range	–30 to +70°C	+10 to +40°C	–30 to +70°C
• Operating supply voltage/operating frequency (cycle time)	3.5 to 5.5 V/0.95 to 10 µs	4.5 to 5.5 V/0.95 to 10 µs	4.0 to 5.5 V/0.95 to 10 µs
• Input high-level current ($\overline{\text{RES}}$)	Maximum: 1 µA	Typical: 10 µA (normal operation and halt mode) Hold mode: 1 µA maximum	Typical: 10 µA (normal operation and halt mode) Hold mode: 1 µA maximum
• Input low-level current ($\overline{\text{RES}}$)	Maximum: 1 µA	Typical: 100 µA	Typical: 100 µA
• Current drain (Operating at 4 MHz) (Halt mode at 4 MHz) (Hold mode)	Typical: 10 nA, maximum: 10 µA	Larger than that for the mask versions Typical: 10 nA, maximum: 10 µA*	Larger than that for the mask versions Typical: 10 nA, maximum: 10 µA*
Port output types at reset	The output type specified in the options	Open-drain outputs	Open-drain outputs
Package	• DIP42S • QFP48E	• DIC42S window package • QFC48 window package	• DIP42S • QFP48E

Note: * Although the microcontroller will remain in hold mode if the $\overline{\text{RES}}$ pin is set low while it is in hold mode, always use the reset start sequence (after switching $\overline{\text{HOLD}}$ from low to high, switch $\overline{\text{RES}}$ from low to high) when clearing hold mode. Also not that a current of about 100 µA flows from the $\overline{\text{RES}}$ pin when it is low. This increases the hold mode current drain by about 100 µA.

See the data sheets for the individual products for details on other differences.

System Block Diagram



Pin Function Overview

Pin	I/O	Overview	Output driver type	Options	State after a reset	Standby mode operation
P00/D0 P01/D1 P02/D2 P03/D3	I/O	I/O ports P00 to P03 • Input or output in 4-bit or 1-bit units • P00 to P03 support the halt mode control function (This function can be specified in bit units.) • Used as data pins in EPROM mode	• Pch: Pull-up MOS type • Nch: Intermediate sink current type	• Pull-up MOS or Nch OD output • Output level on reset	High or low (option)	Hold mode: Output off Halt mode: Output retained
P10/D4 P11/D5 P12/D6 P13/D7	I/O	I/O ports P10 to P13 • Input or output in 4-bit or 1-bit units • Used as data pins in EPROM mode	• Pch: Pull-up MOS type • Nch: Intermediate sink current type	• Pull-up MOS or Nch OD output • Output level on reset	High or low (option)	Hold mode: Output off Halt mode: Output retained
P20/SI0/A0 P21/SO0/A1 P22/$\overline{SCK0}$/A2 P23/INT0/A3	I/O	I/O ports P20 to P23 • Input or output in 4-bit or 1-bit units • P20 is also used as the serial input SI0 pin. • P21 is also used as the serial output SO0 pin. • P22 is also used as the serial clock $\overline{SCK0}$ pin. • P23 is also used as the INT0 interrupt request pin, and also as the timer 0 event counting and pulse width measurement input. • Used as address pins in EPROM mode	• Pch: CMOS type • Nch: Intermediate sink current type	CMOS or Nch OD output	H	Hold mode: Output off Hold mode: Output off

Continued on next page.

Continued from preceding page.

Pin	I/O	Overview	Output driver type	Options	State after a reset	Standby mode operation
P30/$\overline{\text{INT1}}$/A4 P31/POUT0/A5 P32/POUT1/A6	I/O	I/O ports P30 to P32 • Input or output in 3-bit or 1-bit units • P30 is also used as the $\overline{\text{INT1}}$ interrupt request. • P31 is also used for the square wave output from timer 0. • P32 is also used for the square wave and PWM output from timer 1. • P31 and P32 also support 3-state outputs. • Used as address pins in EPROM mode	• Pch: CMOS type • Nch: Intermediate sink current type	CMOS or Nch OD output	H	Hold mode: Output off Halt mode: Output retained
P33/$\overline{\text{HOLD}}$	I	Hold mode control input • Hold mode is set up by the HOLD instruction when $\overline{\text{HOLD}}$ is low. • In hold mode, the CPU is restarted by setting $\overline{\text{HOLD}}$ to the high level. • This pin can be used as input port P33 along with P30 to P32. • When the P33/$\overline{\text{HOLD}}$ pin is at the low level, the CPU will not be reset by a low level on the $\overline{\text{RES}}$ pin. Therefore, applications must not set P33/$\overline{\text{HOLD}}$ low when power is first applied.				
P40/INV0I/A7 P41/INV0O/A8 P42/INV1I/A9 P43/INV1O/A10	I/O	I/O ports P40 to P43 • Input or output in 4-bit or 1-bit units • Input or output in 8-bit units when used in conjunction with P50 to P53. • Can be used for output of 8-bit ROM data when used in conjunction with P50 to P53. • Dedicated inverter circuit (option) • Used as address pins in EPROM mode	• Pch: Pull-up MOS type • CMOS type when the inverter circuit option is selected • Nch: Intermediate sink current type	• Pull-up MOS or Nch OD output • Output level on reset • Inverter circuit	• High or low (option) • Inverter I/O is set to the output off state.	Hold mode: Port output off, inverter output off Halt mode: Port output retained, inverter output continues
P50/A11 P51/A12 P52/A13 P53/INT2/TA	I/O	I/O ports P50 to P53 • Input or output in 4-bit or 1-bit units • Input or output in 8-bit units when used in conjunction with P40 to P43. • Can be used for output of 8-bit ROM data when used in conjunction with P40 to P43. • P53 is also used as the INT2 interrupt request. • Used as address pins in EPROM mode	• Pch: Pull-up MOS type • Nch: Intermediate sink current type	• Pull-up MOS or Nch OD output • Output level on reset	High or low (option)	Hold mode: Output off Halt mode: Output retained
P60/ML P61 P62/DT P63/PIN1	I/O	I/O ports P60 to P63 • Input or output in 4-bit or 1-bit units • P60 is also used as the melody output ML pin. • P62 is also used as the tone output DT pin. • P63 is also used for the event count input to timer 1.	• Pch: CMOS type • Nch: Intermediate sink current type	CMOS or Nch OD output	H	Hold mode: Output off Halt mode: Output retained

Continued on next page.

Continued from preceding page.

Pin	I/O	Overview	Output driver type	Options	State after a reset	Standby mode operation
PC2/$\overline{CE}$ PC3/DASEC	I	I/O ports PC2 to PC3 • Output in 2-bit or 1-bit units • PC3 is also used as the control $\overline{CE}$ and DASEC pin in EPROM mode.	• Pch: CMOS type • Nch: Intermediate sink current type	CMOS or Nch OD output	H	Hold mode: Port output off Halt mode: Port output retained
PD0/INV2I PD1/INV2O PD2/INV3I PD3/INV3O	I	Dedicated input ports PD0 to PD3 Dedicated inverter circuits (option)	• When the inverter circuit option is selected. • Pch: CMOS type • Nch: Intermediate sink current type	Inverter circuits	Normal input Inverter I/O goes to the output off state.	Hold mode: Inverter Output off Halt mode: Inverter output continues
PE0 PE1	I	Dedicated input ports				Hold mode: input disabled Halt mode: inputs accepted
OSC1 OSC2	I O	System clock oscillator connections When an external clock is used, leave OSC2 open and connect the clock signal to OSC1.		Ceramic oscillator or external clock selection	Option selection	Hold mode: oscillator stops Halt mode: oscillator continues
$\overline{RES}$/V_{PP}/ $\overline{OE}$	I	System reset input • When the P33/$\overline{HOLD}$ pin is at the high level, a low level input to the $\overline{RES}$ pin will initialize the CPU. • This pin is also used as the V_{PP}/$\overline{OE}$ pin in EPROM mode.				
TEST/ EPMOD	I	CPU test pin This pin must be connected to V_{SS} during normal operation. Setting this pin to +12 V switches the LC66E2316 to EPROM mode.				
V_{DD} V_{SS}		Power supply pins				

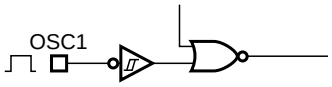
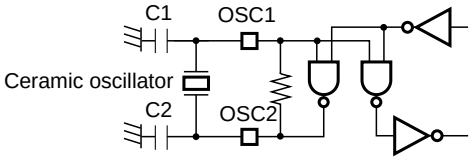
Note: Pull-up MOS type: The output circuit includes a MOS transistor that pulls the pin up to V_{DD}.
CMOS output: Complementary output.
OD output: Open-drain output.

User Options

1. Port 0, 1, 4, and 5 output level at reset option
   The output levels at reset for I/O ports 0, 1, 4, and 5, in independent 4-bit groups, can be selected from the following two options.

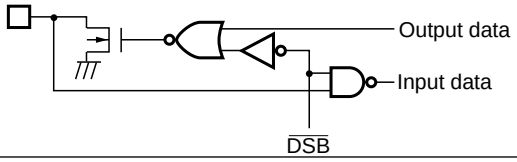
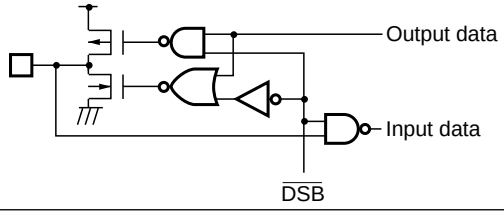
Option	Conditions and notes
1. Output high at reset	The four bits of ports 0, 1, 4, or 5 are set in a group
2. Output low at reset	The four bits of ports 0, 1, 4, or 5 are set in a group

2. Oscillator circuit options
   • Main clock

Option	Circuit	Conditions and notes
1. External clock		The input has Schmitt characteristics
2. Ceramic oscillator		

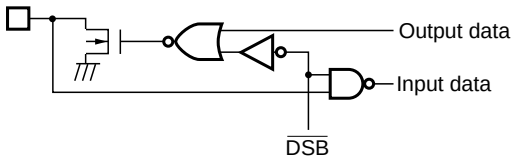
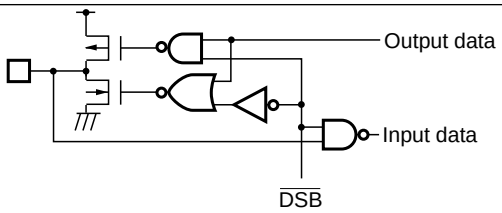
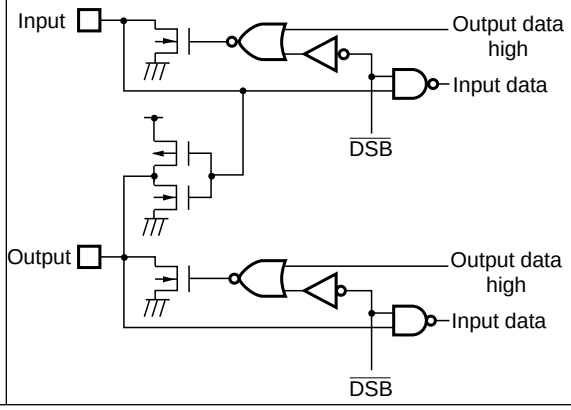
Note: There is no RC oscillator option.

3. Watchdog timer option
   A runaway detection function (watchdog timer) can be selected as an option.
4. Port output type options
   • The output type of each bit (pin) in ports P0, P1, P2, P3 (except for the P33/$\overline{\text{HOLD}}$ pin), P4, P5, P6, and PC can be selected individually from the following two options.

Option	Circuit	Conditions and notes
1. Open-drain output		The port P2, P3, P5, and P6 inputs have Schmitt characteristics.
2. Output with built-in pull-up resistor		The port P2, P3, P5, and P6 inputs have Schmitt characteristics. The CMOS outputs (ports P2, P3, P6, and PC) and the pull-up MOS outputs (P0, P1, P4, and P5) are distinguished by the drive capacity of the p-channel transistor.

5. Inverter array circuit option

One of the following options can be selected for each of the following port sets: P40/P41, P42/P43, PD0/PD1, and PD2/PD3.

Option	Circuit	Conditions and notes
1. Normal port I/O circuit		When the open-drain output type is selected
		When the built-in pull-up resistor output type is selected
2. Inverter I/O circuit		If this option is selected, the I/O circuit is disabled by the $\overline{DSB}$ signal. Also note that the open-drain port output type option and the high level at reset option must be selected.

LC662316 Series Option Data Area and Definitions

ROM area	Bit	Option specified		Option/data relationship
3FF0H	7	P5	Output level at reset	0 = high level, 1 = low level
	6	P4		
	5	Unused		This bit must be set to 0.
	4	Oscillator option		0 = external clock, 1 = ceramic oscillator
	3	Unused		This bit must be set to 0.
	2	P1	Output level at reset	0 = low level, 1 = high level
	1	P0		
	0	Watchdog timer option		0 = none, 1 = yes (present)
3FF1H	7	P13	Output type	0 = OD, 1 = PU
	6	P12		
	5	P11		
	4	P10		
	3	P03	Output type	0 = OD, 1 = PU
	2	P02		
	1	P01		
	0	P00		
3FF2H	7	Unused		This bit must be set to 0.
	6	P32	Output type	0 = OD, 1 = PU
	5	P31		
	4	P30		
	3	P23	Output type	0 = OD, 1 = PU
	2	P22		
	1	P21		
	0	P20		
3FF3H	7	P53	Output type	0 = OD, 1 = PU
	6	P52		
	5	P51		
	4	P50		
	3	P43	Output type	0 = OD, 1 = PU
	2	P42		
	1	P41		
	0	P40		
3FF4H	7	Unused		This bit must be set to 0.
	6			
	5			
	4			
	3	P63	Output type	0 = OD, 1 = PU
	2	P62		
	1	P61		
	0	P60		
3FF5H	7	Unused		This bit must be set to 0.
	6			
	5			
	4			
	3	Unused		This bit must be set to 0.
	2			
	1			
	0			
3FF6H	7	Unused		This bit must be set to 0.
	6			
	5			
	4			
	3	Unused		This bit must be set to 0.
	2			
	1			
	0			

Continued on next page.

Continued from preceding page.

ROM area	Bit	Option specified		Option/data relationship
3FF7H	7	Unused		This bit must be set to 0.
	6			
	5			
	4			
	3	PC3	Output type	0 = OD, 1 = PU
	2	PC2		
	1	Unused		This bit must be set to 0.
	0			
3FF8H	7	ML disabled option		0 = disabled, 1 = enabled
	6	Unused		This bit must be set to 1.
	5	Unused		This bit must be set to 1.
	4	PD3	Inverter output	0 = inverter output, 1 = none
	3	PD1		
	2	Unused		This bit must be set to 1.
	1	P43	Inverter output	0 = inverter output, 1 = none
	0	P41		
3FF9H	7	Unused		This bit must be set to 0.
	6			
	5			
	4			
	3	Unused		This bit must be set to 0.
	2			
	1			
	0			
3FFAH	7	Unused		This bit must be set to 0.
	6			
	5			
	4			
	3	Unused		This bit must be set to 0.
	2			
	1			
	0			
3FFBH	7	Unused		This bit must be set to 0.
	6			
	5			
	4			
	3	Unused		This bit must be set to 0.
	2			
	1			
	0			
3FFCH	7	Unused		This bit must be set to 0.
	6			
	5			
	4			
	3	Unused		This bit must be set to 0.
	2			
	1			
	0			
3FFDH	7	Reserved. Must be set to predefined data values.		This data is generated by the assembler. If the assembler is not used, set this data to '00'.
	6			
	5			
	4			
	3			
	2			
	1			
	0			

Continued on next page.

Continued from preceding page.

ROM area	Bit	Option specified	Option/data relationship
3FFEH	7	Reserved. Must be set to predefined data values.	This data is generated by the assembler. If the assembler is not used, set this data to '00'.
	6		
	5		
	4		
	3		
	2		
	1		
	0		
3FFFH	7	Reserved. Must be set to predefined data values.	This data is generated by the assembler. If the assembler is not used, set this data to '00'.
	6		
	5		
	4		
	3		
	2		
	1		
	0		

Usage Notes

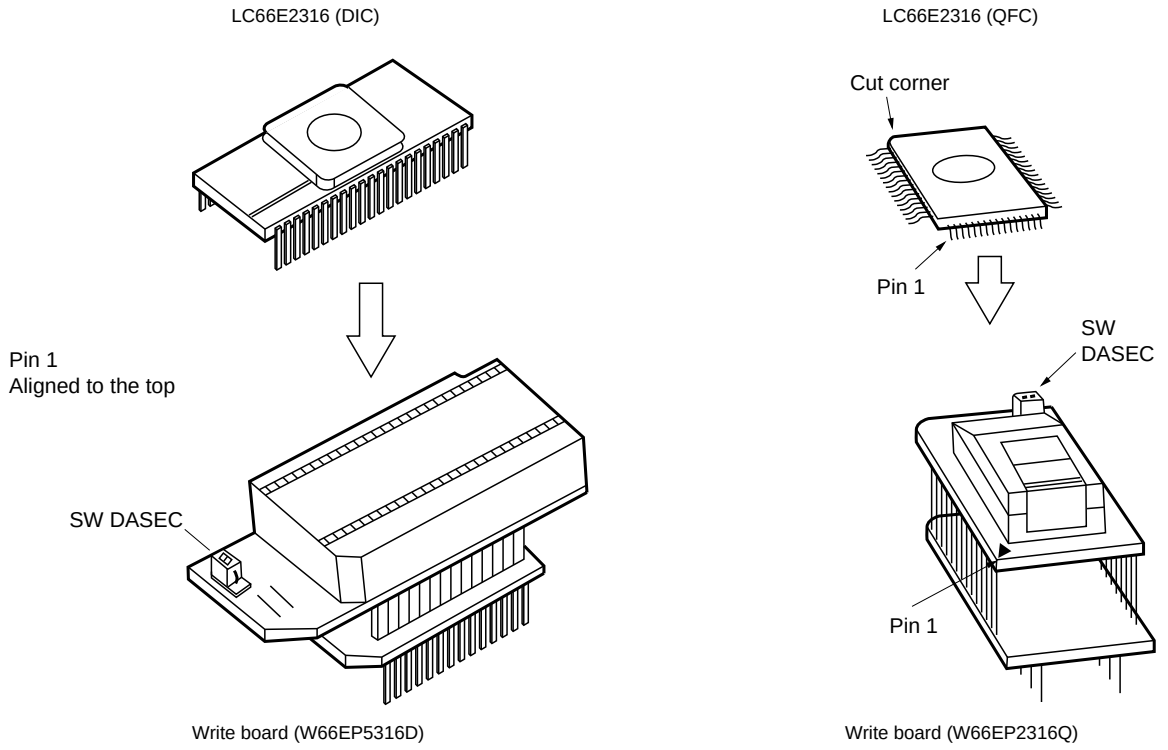
1. Option specification
   When using a Sanyo cross assembler with the LC66E2316, use the version called "LC66S.EXE" and specify the actual microcontroller to be evaluated with the CPU pseudo instruction in the source file. The port options must be specified in the source file. The cross assembler will create an option code list in the option specification area (locations 3FF0H to 3FFFH). It is also possible to directly set up data in the option specification area. If this is done, the options must be specified according to the option code creation table shown on the following page.
2. Writing the EPROM
   Use a special-purpose writing conversion board (the W66EP5316D for the DIP package, and the W66EP2316Q for the QFP package) to allow the EPROM programmers listed below to be used when writing the data created by the cross assembler to the LC66E2316.
   - The EPROM programmers listed below can be used.

Manufacturer	Models that can be used
Advantest	R4945, R4944A, R4943, or equivalent products
Ando	AF9704
AVAL	—
Minato Electronics	MODEL1890A

   - The "27512 (V_{PP} 12.5 V) Intel high-speed write" technique must be used to write the EPROM. Set the address range to location 0 to 3FFFH. The DASEC jumper must be off.
3. Using the data security function
   The data security function sets up the microcontroller in advance so that data that was written to the microcontroller EPROM cannot be read out.
   Use the following procedure to enable the LC66E2316 data security function.
   - Set the write conversion board DASEC jumper to the on position.
   - Write the data to the EPROM once again.

   At this time, since this function will operate, the EPROM programmer will issue an error. However, this error does not indicate that there was a problem in either the programmer or the LSI.
   Notes: 1. If the data at all addresses was "FF" at step 2, the data security function will not be activated.
   2. The data security function will not be activated at step 2 if the "blank → program → verify" operation sequence is used.
   3. Always return the jumper to the off position after the data security function has been activated.

4. Erase procedure
   Use a general-purpose EPROM eraser to erase data written to the EPROM.



Specifications

Absolute Maximum Ratings at Ta = 25°C, V_{SS} = 0 V

Parameter	Symbol	Conditions	Ratings	Unit	Note
Maximum supply voltage	V_{DD} max	V_{DD}	–0.3 to +7.0	V	
Input voltage	$V_{IN}1$	P2, P3 (except for the P33/$\overline{HOLD}$ pin), P61, and P63	–0.3 to +12.0	V	1
	$V_{IN}2$	All other inputs	–0.3 to V_{DD} + 0.3	V	2
Output voltage	$V_{OUT}1$	P2, P3 (except for the P33/$\overline{HOLD}$ pin), P61, and P63	–0.3 to +12.0	V	1
	$V_{OUT}2$	All other inputs	–0.3 to V_{DD} + 0.3	V	2
Output current per pin	$I_{ON}1$	P0, P1, P2, P3 (except for the P33/$\overline{HOLD}$ pin), P4, P5, P6, PC	20	mA	3
	$I_{ON}2$	P41, P43, PC3, PD1, PD3	20	mA	3
	$-I_{OP}1$	P0, P1, P4, P5	2	mA	4
	$-I_{OP}2$	P2, P3 (except for the P33/$\overline{HOLD}$ pin), P6,and PC	4	mA	4
Total pin current	$\Sigma I_{ON}1$	P0, P1, P2, P3 (except for the P33/$\overline{HOLD}$ pin), PD	75	mA	3
	$\Sigma I_{ON}2$	P4, P5, P6, PC	75	mA	3
	$\Sigma I_{OP}1$	P0, P1, P2, P3 (except for the P33/$\overline{HOLD}$ pin), PD	25	mA	4
	$\Sigma I_{OP}2$	P4, P5, P6, PC	25	mA	4
Allowable power dissipation	Pd max	Ta = +10 to +40°C: DIC42S (QFC48)	600 (430)	mW	5
Operating temperature	Topr		–30 to +70	°C	
Storage temperature	Tstg		–55 to +125	°C	

Note: 1. Applies to pins with open-drain output specifications. For pins with other than open-drain output specifications, the ratings in the pin column for that pin apply.
2. For the oscillator input and output pins, levels up to the free-running oscillation level are allowed.
3. Sink current (Applies to PD when the inverter array specifications have been selected.)
4. Source current (Applies to all pins except PD for which the pull-up output specifications, the CMOS output specifications, or the inverter array specifications have been selected. Applies to P8 pins for which the inverter array specifications have been selected.) Contact your Sanyo representative for the electrical characteristics when the inverter array or buffer array options are specified.

Allowable Operating Ranges at Ta = +10 to +40°C, V_{SS} = 0 V, V_{DD} = 4.5 to 5.5 V, unless otherwise specified.

Parameter	Symbol	Conditions	min	typ	max	Unit	Note
Operating supply voltage	V_{DD}	V_{DD}	4.5		5.5	V	
Memory retention supply voltage	$V_{DD}H$	V_{DD}: During hold mode	1.8		5.5	V	
Input high-level voltage	$V_{IH}1$	P2, P3 (except for the P33/$\overline{\text{HOLD}}$ pin), P61, and P63: N-channel output transistor off	0.8 V_{DD}		10.0	V	1
	$V_{IH}2$	P33/$\overline{\text{HOLD}}$, $\overline{\text{RES}}$, OSC1: N-channel output transistor off	0.8 V_{DD}		V_{DD}	V	
	$V_{IH}3$	P0, P1, P4, P5, PC, PD, PE: N-channel output transistor off	0.8 V_{DD}		V_{DD}	V	2
Input low-level voltage	$V_{IL}1$	P2, P3 (except for the P33/$\overline{\text{HOLD}}$ pin), P6, $\overline{\text{RES}}$, and OSC1: N-channel output transistor off	V_{SS}		0.2 V_{DD}	V	1
	$V_{IL}2$	P33/$\overline{\text{HOLD}}$: V_{DD} = 1.8 to 5.5 V	V_{SS}		0.2 V_{DD}	V	
	$V_{IL}3$	P0, P1, P4, P5, PC, PD, PE, TEST: N-channel output transistor off	V_{SS}		0.2 V_{DD}	V	2
Operating frequency (instruction cycle time)	fop (Tcyc)		0.4 (10)		4.20 (0.95)	MHz (µs)	
[External clock input conditions]							
Frequency	f_{ext}	OSC1: Defined by Figure 1. Input the clock signal to OSC1 and leave OSC2 open. (External clock input must be selected as the oscillator circuit option.)	0.4		4.20	MHz	
Pulse width	t_{extH}, t_{extL}	OSC1: Defined by Figure 1. Input the clock signal to OSC1 and leave OSC2 open. (External clock input must be selected as the oscillator circuit option.)	100			ns	
Rise and fall times	t_{extR}, t_{extF}	OSC1: Defined by Figure 1. Input the clock signal to OSC1 and leave OSC2 open. (External clock input must be selected as the oscillator circuit option.)			30	ns	

Note: 1. Applies to pins with open-drain specifications. However, $V_{IH}2$ applies to the P33/$\overline{\text{HOLD}}$ pin.
When ports P2, P3, and P6 have CMOS output specifications they cannot be used as input pins.
2. PC port pins with CMOS output specifications cannot be used as input pins.
Contact your Sanyo representative for the allowable operating ranges for P4 and PD when the inverter array is used.

Electrical Characteristics at Ta = +10 to +40°C, V_{SS} = 0 V, V_{DD} = 4.5 to 5.5 V unless otherwise specified.

Parameter		Symbol	Conditions	min	typ	max	Unit	Note
Input high-level current		$I_{IH}1$	P2, P3 (except for the P33/$\overline{HOLD}$ pin), P61, and P63: V_{IN} = 10.0 V, with the output Nch transistor off			5.0	µA	1
		$I_{IH}2$	P0, P1, P4, P5, PC, OSC1, and P33/$\overline{HOLD}$ (Does not apply to PD, PE, PC2, and PC3): V_{IN} = V_{DD}, with the output Nch transistor off			1.0	µA	1
		$I_{IH}3$	PD, PE, PC2, PC3: V_{IN} = V_{DD}, with the output Nch transistor off			1.0	µA	1
		$I_{IH}4$	$\overline{RES}$: V_{IN} = V_{DD}, operating, halt mode		10		µA	1
		$I_{IH}5$	$\overline{RES}$: V_{IN} = V_{DD}, hold mode			1.0	µA	1
Input low-level current		$I_{IL}1$	Input ports other than PD, PE, PC2, and PC3: V_{IN} = V_{SS}, with the output Nch transistor off	–1.0			µA	2
		$I_{IL}2$	PC2, PC3, PD, PE: V_{IN} = V_{SS}, with the output Nch transistor off	–1.0			µA	2
		$I_{IL}3$	$\overline{RES}$: V_{IN} = V_{SS}		100		µA	1
Output high-level voltage		$V_{OH}1$	P2, P3 (except for the P33/$\overline{HOLD}$ pin), P6, and PC: I_{OH} = –1 mA	V_{DD} – 1.0			V	3
			P2, P3 (except for the P33/$\overline{HOLD}$ pin), P6, and PC: I_{OH} = –0.1 mA	V_{DD} – 0.5				
Value of the output pull-up resistor		R_{PO}	P0, P1, P4, P5	30	100	150	kΩ	4
Output low-level voltage		$V_{OL}1$	P0, P1, P2, P3, P4, P5, P6, and PC (except for the P33/$\overline{HOLD}$ pin): I_{OL} = 1.6 mA			0.4	V	
		$V_{OL}2$	P0, P1, P2, P3, P4, P5, P6, and PC (except for the P33/$\overline{HOLD}$ pin): I_{OL} = 10 mA			1.5	V	
Output off leakage current		$I_{OFF}1$	P2, P3, P61, P63: V_{IN} = V_{DD}			5.0	µA	5
		$I_{OFF}2$	Does not apply to P2, P3, P61, and P63: V_{IN} = V_{DD}			1.0	µA	5
[Schmitt characteristics]								
Hysteresis voltage		V_{HYS}	P2, P3, P5, P6, OSC1 (EXT), $\overline{RES}$		0.1 V_{DD}		V	
High-level threshold voltage		Vt_H		0.5 V_{DD}		0.8 V_{DD}	V	
Low-level threshold voltage		Vt_L		0.2 V_{DD}		0.5 V_{DD}	V	
[Ceramic oscillator]								
Oscillator frequency		f_{CF}	OSC1, OSC2: Figure 2, 4 MHz		4.0		MHz	
Oscillator stabilization time		f_{CFS}	Figure 3, 4 MHz			10.0	ms	
[Serial clock]								
Cycle time	Input	t_{CKCY}	$\overline{SCK0}$: With the timing of Figure 4 and the test load of Figure 5.	0.9			µs	
	Output			2.0			Tcyc	
Low-level and high-level pulse widths	Input	t_{CKL}		0.4			µs	
	Output	t_{CKH}		1.0			Tcyc	
Rise an fall times	Output	t_{CKR}, t_{CKF}				0.1	µs	
[Serial input]								
Data setup time		t_{ICK}	SI0: With the timing of Figure 4. Stipulated with respect to the rising edge (↑) of $\overline{SCK0}$.	0.3			µs	
Data hold time		t_{CKI}		0.3			µs	
[Serial output]								
Output delay time		t_{CKO}	SO0: With the timing of Figure 4 and the test load of Figure 5. Stipulated with respect to the falling edge (↓) of $\overline{SCK0}$.			0.3	µs	

Continued on next page.

Continued from preceding page.

Parameter	Symbol	Conditions	min	typ	max	Unit	Note
[Pulse conditions]							
INT0 high and low-level	t_{IOH}, t_{IOL}	INT0: Figure 6, conditions under which the INT0 interrupt can be accepted, conditions under which the timer 0 event counter or pulse width measurement input can be accepted	2			Tcyc	
High and low-level pulse widths for interrupt inputs other than INT0	t_{IIH}, t_{IIL}	$\overline{INT1}$, INT2: Figure 6, conditions under which the corresponding interrupt can be accepted	2			Tcyc	
PIN1 high and low-level pulse widths	t_{PINH}, t_{PINL}	PIN1: Figure 6, conditions under which the timer 1 event counter input can be accepted	2			Tcyc	
$\overline{RES}$ high and low-level pulse widths	t_{RSH}, t_{RSL}	$\overline{RES}$: Figure 6, conditions under which reset can be applied.	3			Tcyc	
Operating current drain	$I_{DD\ OP}$	V_{DD}: 4-MHz ceramic oscillator		6.0	12	mA	6
		V_{DD}: 4-MHz external clock		6.0	12	mA	
Halt mode current drain	I_{DDHALT}	V_{DD}: 4-MHz ceramic oscillator		4.0	8.0	mA	
		V_{DD}: 4-MHz external clock		4.0	8.0	mA	
Hold mode current drain	I_{DDHOLD}	V_{DD}: V_{DD} = 1.8 to 5.5 V		0.01	10	µA	

Note: 1. With the output Nch transistor off in shared I/O ports with the open-drain output specifications. These pins cannot be used as input pins if the CMOS output specifications are selected.When the port option is selected for PE.
2. With the output Nch transistor off in shared I/O ports with the open-drain output specifications. The rating for the pull-up output specification pins is stipulated in terms of the output pull-up current IPO. These pins cannot be used as input pins if the CMOS output specifications are selected.
3. With the output Nch transistor off for CMOS output specification pins.
4. With the output Nch transistor off for pull-up output specification pins.
5. With the output Nch transistor off for open-drain output specification pins.
6. Reset state

Tone (DTMF) Output Characteristics

DC Characteristics at Ta = +10 to +40°C, V_{SS} = 0 V

Parameter	Symbol	Conditions	min	typ	max	Unit
Tone output voltage (p-p)	V_{T1}	DT: Dual tones, V_{DD} = 4.5 to 5.5 V	0.9	1.3	2.0	V
Row/column tone output voltage ratio	D_{BCR1}	DT: Dual tones, V_{DD} = 4.5 to 5.5 V	1.0	2.0	3.0	dB
Tone distortion	THD1	DT: Single tone, V_{DD} = 4.5 to 5.5 V		2	7	%

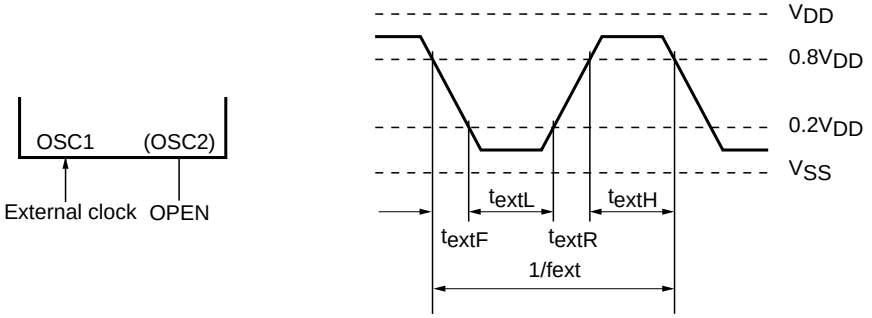


Figure 1 External Clock Input Waveform

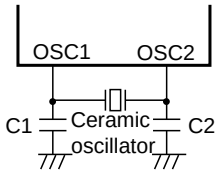


Figure 2 Ceramic Oscillator Circuit

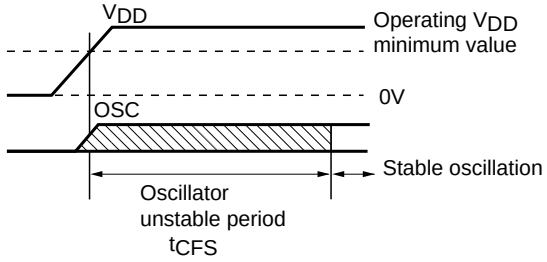


Figure 3 Oscillator Stabilization Period

Table 1 Guaranteed Ceramic Oscillator Constants External capacitor type

External capacitor type		Built-in capacitor type
4 MHz (Murata Mfg. Co., Ltd.) CSA4.00MG	C1 = 33 pF ± 10% C2 = 33 pF ± 10%	4 MHz (Murata Mfg. Co., Ltd.) CST4.00MG
4 MHz (Kyocera Corporation) KBR4.0MS	C1 = 33 pF ± 10% C2 = 33 pF ± 10%	4 MHz (Kyocera Corporation) KBR4.0MES

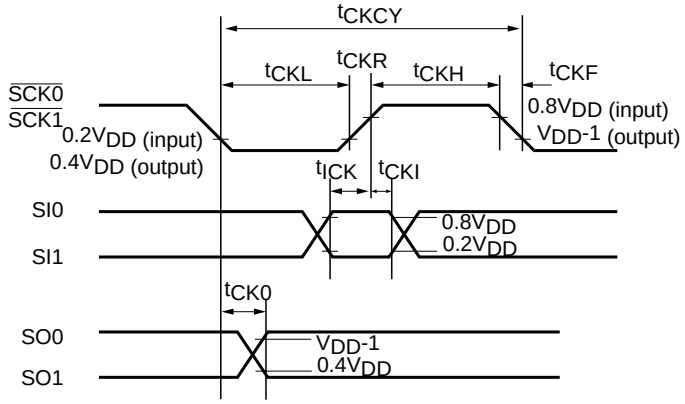


Figure 4 Serial I/O Timing

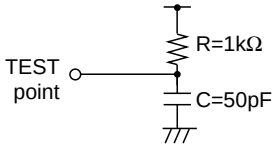


Figure 5 Timing Load

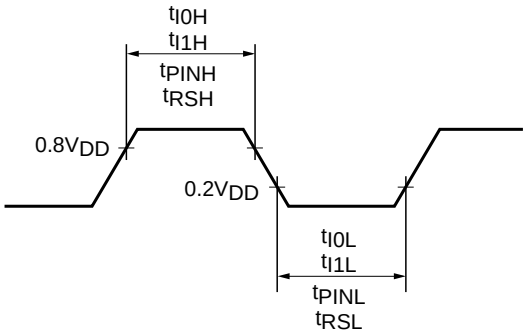


Figure 6 Input Timing for the INT0, $\overline{\text{INT1}}$, INT2, PIN1, and $\overline{\text{RES}}$ pins

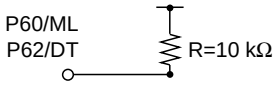


Figure 7 Tone Output Pin Load

LC66XXXX Series Instruction Table (by function)

Abbreviations:

AC: Accumulator
E: E register
CF: Carry flag
ZF: Zero flag
HL: Data pointer DPH, DPL
XY: Data pointer DPX, DPY
M: Data memory
M (HL): Data memory pointed to by the DPH, DPL data pointer
M (XY): Data memory pointed to by the DPX, DPY auxiliary data pointer
M2 (HL): Two words of data memory (starting on an even address) pointed to by the DPH, DPL data pointer
SP: Stack pointer
M2 (SP): Two words of data memory pointed to by the stack pointer
M4 (SP): Four words of data memory pointed to by the stack pointer
in: n bits of immediate data
t2: Bit specification

t2	11	10	01	00
Bit	2^3	2^2	2^1	2^0

PCh: Bits 8 to 11 in the PC
PCm: Bits 4 to 7 in the PC
PCl: Bits 0 to 3 in the PC
Fn: User flag, n = 0 to 15
TIMER0: Timer 0
TIMER1: Timer 1
SIO: Serial register
P: Port
P (i4): Port indicated by 4 bits of immediate data
INT: Interrupt enable flag
( ), [ ]: Indicates the contents of a location
←: Transfer direction, result
∀: Exclusive or
∧: Logical and
∨: Logical or
+: Addition
–: Subtraction
—: Taking the one's complement

Mnemonic		Instruction code		Number of bytes	Number of cycles	Operation	Description	Affected status bits	Note
		D_7 D_6 D_5 D_4	D_3 D_2 D_1 D_0						
[Accumulator manipulation instructions]									
CLA	Clear AC	1 0 0 0	0 0 0 0	1	1	AC ← 0 (Equivalent to LAI 0.)	Clear AC to 0.	ZF	Has a vertical skip function.
DAA	Decimal adjust AC in addition	1 1 0 0 0 0 1 0	1 1 1 1 0 1 1 0	2	2	AC ← (AC) + 6 (Equivalent to ADI 6.)	Add six to AC.	ZF	
DAS	Decimal adjust AC in subtraction	1 1 0 0 0 0 1 0	1 1 1 1 1 0 1 0	2	2	AC ← (AC) + 10 (Equivalent to ADI 0AH.)	Add 10 to AC.	ZF	
CLC	Clear CF	0 0 0 1	1 1 1 0	1	1	CF ← 0	Clear CF to 0.	CF	
STC	Set CF	0 0 0 1	1 1 1 1	1	1	CF ← 1	Set CF to 1.	CF	
CMA	Complement AC	0 0 0 1	1 0 0 0	1	1	AC ← $\overline{(AC)}$	Take the one's complement of AC.	ZF	
IA	Increment AC	0 0 0 1	0 1 0 0	1	1	AC ← (AC) + 1	Increment AC.	ZF, CF	
DA	Decrement AC	0 0 1 0	0 1 0 0	1	1	AC ← (AC) – 1	Decrement AC.	ZF, CF	
RAR	Rotate AC right through CF	0 0 0 1	0 0 0 0	1	1	AC_3 ← (CF), ACn ← (ACn + 1), CF ← (AC_0)	Shift AC (including CF) right.	CF	
RAL	Rotate AC left through CF	0 0 0 0	0 0 0 1	1	1	AC_0 ← (CF), ACn + 1 ← (ACn), CF ← (AC_3)	Shift AC (including CF) left.	CF, ZF	
TAE	Transfer AC to E	0 1 0 0	0 1 0 1	1	1	E ← (AC)	Transfer the contents of AC to E.		
TEA	Transfer E to AC	0 1 0 0	0 1 1 0	1	1	AC ← (E)	Transfer the contents of E to AC.	ZF	
XAE	Exchange AC with E	0 1 0 0	0 1 0 0	1	1	(AC) ↔ (E)	Exchange the contents of AC and E.		
[Memory manipulation instructions]									
IM	Increment M	0 0 0 1	0 0 1 0	1	1	M (HL) ← [M (HL)] + 1	Increment M (HL).	ZF, CF	
DM	Decrement M	0 0 1 0	0 0 1 0	1	1	M (HL) ← [M (HL)] – 1	Decrement M (HL).	ZF, CF	
IMDR i8	Increment M direct	1 1 0 0 I_7 I_6 I_5 I_4	0 1 1 1 I_3 I_2 I_1 I_0	2	2	M (i8) ← [M (i8)] + 1	Increment M (i8).	ZF, CF	
DMDR i8	Decrement M direct	1 1 0 0 I_7 I_6 I_5 I_4	0 0 1 1 I_3 I_2 I_1 I_0	2	2	M (i8) ← [M (i8)] – 1	Decrement M (i8).	ZF, CF	
SMB t2	Set M data bit	0 0 0 0	1 1 t_1 t_0	1	1	[M (HL), t2] ← 1	Set the bit in M (HL) specified by t0 and t1 to 1.		
RMB t2	Reset M data bit	0 0 1 0	1 1 t_1 t_0	1	1	[M (HL), t2] ← 0	Clear the bit in M (HL) specified by t0 and t1 to 0.	ZF	
[Arithmetic, logic and comparison instructions]									
AD	Add M to AC	0 0 0 0	0 1 1 0	1	1	AC ← (AC) + [M (HL)]	Add the contents of AC and M (HL) as two's complement values and store the result in AC.	ZF, CF	
ADDR i8	Add M direct to AC	1 1 0 0 I_7 I_6 I_5 I_4	1 0 0 1 I_3 I_2 I_1 I_0	2	2	AC ← (AC) + [M (i8)]	Add the contents of AC and M (i8) as two's complement values and store the result in AC.	ZF, CF	
ADC	Add M to AC with CF	0 0 0 0	0 0 1 0	1	1	AC ← (AC) + [M (HL)] + (CF)	Add the contents of AC, M (HL) and C as two's complement values and store the result in AC.	ZF, CF	
ADI i4	Add immediate data to AC	1 1 0 0 0 0 1 0	1 1 1 1 I_3 I_2 I_1 I_0	2	2	AC ← (AC) + I_3, I_2, I_1, I_0	Add the contents of AC and the immediate data as two's complement values and store the result in AC.	ZF	
SUBC	Subtract AC from M with CF	0 0 0 1	0 1 1 1	1	1	AC ← [M (HL)] – (AC) – (CF)	Subtract the contents of AC and $\overline{CF}$ from M (HL) as two's complement values and store the result in AC.	ZF, CF	CF will be zero if there was a borrow and one otherwise.
ANDA	And M with AC then store AC	0 0 0 0	0 1 1 1	1	1	AC ← (AC) ∧ [M (HL)]	Take the logical and of AC and M (HL) and store the result in AC.	ZF	
ORA	Or M with AC then store AC	0 0 0 0	0 1 0 1	1	1	AC ← (AC) ∨ [M (HL)]	Take the logical or of AC and M (HL) and store the result in AC.	ZF	

Continued on next page.

Continued from preceding page.

Mnemonic		Instruction code		Number of bytes	Number of cycles	Operation	Description	Affected status bits	Note
		D_7 D_6 D_5 D_4	D_3 D_2 D_1 D_0						
[Arithmetic, logic and comparison instructions]									
EXL	Exclusive or M with AC then store AC	0 0 0 1	0 1 0 1	1	1	AC ← (AC) ∀ [M (HL)]	Take the logical exclusive or of AC and M (HL) and store the result in AC.	ZF	
ANDM	And M with AC then store M	0 0 0 0	0 0 1 1	1	1	M (HL) ← (AC) ∧ [M (HL)]	Take the logical and of AC and M (HL) and store the result in M (HL).	ZF	
ORM	Or M with AC then store M	0 0 0 0	0 1 0 0	1	1	M (HL) ← (AC) ∨ [M (HL)]	Take the logical or of AC and M (HL) and store the result in M (HL).	ZF	
CM	Compare AC with M	0 0 0 1	0 1 1 0	1	1	$\overline{[M\ (HL)]}$ + (AC) + 1	Compare the contents of AC and M (HL) and set or clear CF and ZF according to the result. Magnitude comparison / CF / ZF [M (HL)] > (AC) 0 0 [M (HL)] = (AC) 1 1 [M (HL)] < (AC) 1 0	ZF, CF	
CI i4	Compare AC with immediate data	1 1 0 0 1 0 1 0	1 1 1 1 I_3 I_2 I_1 I_0	2	2	$\overline{I_3\ I_2\ I_1\ I_0}$ + (AC) + 1	Compare the contents of AC and the immediate data I_3 I_2 I_1 I_0 and set or clear CF and ZF according to the result. Magnitude comparison / CF / ZF I_3 I_2 I_1 I_0 > AC 0 0 I_3 I_2 I_1 I_0 = AC 1 1 I_3 I_2 I_1 I_0 < AC 1 0	ZF, CF	
CLI i4	Compare DP_L with immediate data	1 1 0 0 1 0 1 1	1 1 1 1 I_3 I_2 I_1 I_0	2	2	ZF ← 1 if (DP_L) = I_3 I_2 I_1 I_0 ZF ← 0 if (DP_L) ≠ I_3 I_2 I_1 I_0	Compare the contents of DP_L with the immediate data. Set ZF if identical and clear ZF if not.	ZF	
CMB t2	Compare AC bit with M data bit	1 1 0 0 1 1 0 1	1 1 1 1 0 0 t_1 t_0	2	2	ZF ← 1 if (AC, t2) = [M (HL), t2] ZF← 0 if (AC, t2) ≠ [M (HL), t2]	Compare the corresponding bits specified by t0 and t1 in AC and M (HL). Set ZF if identical and clear ZF if not.	ZF	
[Load and store instructions]									
LAE	Load AC and E from M2 (HL)	0 1 0 1	1 1 0 0	1	1	AC ← M (HL), E ← M (HL + 1)	Load the contents of M2 (HL) into AC, E.		
LAI i4	Load AC with immediate data	1 0 0 0	I_3 I_2 I_1 I_0	1	1	AC ← I_3 I_2 I_1 I_0	Load the immediate data into AC.	ZF	Has a vertical skip function
LADR i8	Load AC from M direct	1 1 0 0 I_7 I_6 I_5 I_4	0 0 0 1 I_3 I_2 I_1 I_0	2	2	AC ← [M (i8)]	Load the contents of M (i8) into AC.	ZF	
S	Store AC to M	0 1 0 0	0 1 1 1	1	1	M (HL) ← (AC)	Store the contents of AC into M (HL).		
SAE	Store AC and E to M2 (HL)	0 1 0 1	1 1 1 0	1	1	M (HL) ← (AC) M (HL + 1) ← (E)	Store the contents of AC, E into M2 (HL).		
LA reg	Load AC from M (reg)	0 1 0 0	1 0 t_0 0	1	1	AC ← [M (reg)]	Load the contents of M (reg) into AC. The reg is either HL or XY depending on t_0. reg / T_0 HL 0 XY 1	ZF	

Continued on next page.

Continued from preceding page.

Mnemonic		Instruction code		Number of bytes	Number of cycles	Operation	Description	Affected status bits	Note
		D_7 D_6 D_5 D_4	D_3 D_2 D_1 D_0						
[Load and store instructions]									
LA reg, I	Load AC from M (reg) then increment reg	0 1 0 0	1 0 t_0 1	1	2	AC ← [M (reg)] DP_L ← (DP_L) + 1 or DP_Y ← (DP_Y) + 1	Load the contents of M (reg) into AC. (The reg is either HL or XY.) Then increment the contents of either DP_L or DP_Y. The relationship between t_0 and reg is the same as that for the LA reg instruction.	ZF	ZF is set according to the result of incrementing DP_L or DP_Y.
LA reg, D	Load AC from M (reg) then decrement reg	0 1 0 1	1 0 t_0 1	1	2	AC ← [M (reg)] DP_L ← (DP_L) – 1 or DP_Y ← (DP_Y) – 1	Load the contents of M (reg) into AC. (The reg is either HL or XY.) Then decrement the contents of either DP_L or DP_Y. The relationship between t_0 and reg is the same as that for the LA reg instruction.	ZF	ZF is set according to the result of decrementing DP_L or DP_Y.
XA reg	Exchange AC with M (reg)	0 1 0 0	1 1 t_0 0	1	1	(AC) ↔ [M (reg)]	Exchange the contents of M (reg) and AC. The reg is either HL or XY depending on t_0. reg: HL, XY; T_0: 0, 1		
XA reg, I	Exchange AC with M (reg) then increment reg	0 1 0 0	1 1 t_0 1	1	2	(AC) ↔ [M (reg)] DP_L ← (DP_L) + 1 or DP_Y ← (DP_Y) + 1	Exchange the contents of M (reg) and AC. (The reg is either HL or XY.) Then increment the contents of either DP_L or DP_Y. The relationship between t_0 and reg is the same as that for the XA reg instruction.	ZF	ZF is set according to the result of incrementing DP_L or DP_Y.
XA reg, D	Exchange AC with M (reg) then decrement reg	0 1 0 1	1 1 t_0 1	1	2	(AC) ↔ [M (reg)] DP_L ← (DP_L) – 1 or DP_Y ← (DP_Y) – 1	Exchange the contents of M (reg) and AC. (The reg is either HL or XY.) Then decrement the contents of either DP_L or DP_Y. The relationship between t_0 and reg is the same as that for the XA reg instruction.	ZF	ZF is set according to the result of decrementing DP_L or DP_Y.
XADR i8	Exchange AC with M direct	1 1 0 0 I_7 I_6 I_5 I_4	1 0 0 0 I_3 I_2 I_1 I_0	2	2	(AC) ↔ [M (i8)]	Exchange the contents of AC and M (i8).		
LEAI i8	Load E & AC with immediate data	1 1 0 0 I_7 I_6 I_5 I_4	0 1 1 0 I_3 I_2 I_1 I_0	2	2	E ← I_7 I_6 I_5 I_4 AC ← I_3 I_2 I_1 I_0	Load the immediate data i8 into E, AC.		
RTBL	Read table data from program ROM	0 1 0 1	1 0 1 0	1	2	E, AC ← [ROM (PCh, E, AC)]	Load into E, AC the ROM data at the location determined by replacing the lower 8 bits of the PC with E, AC.		
RTBLP	Read table data from program ROM then output to P4, 5	0 1 0 1	1 0 0 0	1	2	Port 4, 5 ← [ROM (PCh, E, AC)]	Output from ports 4 and 5 the ROM data at the location determined by replacing the lower 8 bits of the PC with E, AC.		
[Data pointer manipulation instructions]									
LDZ i4	Load DP_H with zero and DP_L with immediate data respectively	0 1 1 0	I_3 I_2 I_1 I_0	1	1	DP_H ← 0 DPL ← I_3 I_2 I_1 I_0	Load zero into DP_H and the immediate data i4 into DP_L.		
LHI i4	Load DP_H with immediate data	1 1 0 0 0 0 0 0	1 1 1 1 I_3 I_2 I_1 I_0	2	2	DP_H ← I_3 I_2 I_1 I_0	Load the immediate data i4 into DP_H.		
LLI i4	Load DP_L with immediate data	1 1 0 0 0 0 0 1	1 1 1 1 I_3 I_2 I_1 I_0	2	2	DP_L ← I_3 I_2 I_1 I_0	Load the immediate data i4 into DP_L.		
LHLI i8	Load DP_H, DP_L with immediate data	1 1 0 0 I_7 I_6 I_5 I_4	0 0 0 0 I_3 I_2 I_1 I_0	2	2	DP_H ← I_7 I_6 I_5 I_4 DP_L ← I_3 I_2 I_1 I_0	Load the immediate data into DL_H, DP_L.		
LXYI i8	Load DP_X, DP_Y with immediate data	1 1 0 0 I_7 I_6 I_5 I_4	0 0 0 0 I_3 I_2 I_1 I_0	2	2	DP_X ← I_7 I_6 I_5 I_4 DP_Y ← I_3 I_2 I_1 I_0	Load the immediate data into DL_X, DP_Y.		

Continued on next page.

Continued from preceding page.

Mnemonic		Instruction code		Number of bytes	Number of cycles	Operation	Description	Affected status bits	Note
		D_7 D_6 D_5 D_4	D_3 D_2 D_1 D_0						
[Data pointer manipulation instructions]									
IL	Increment DP_L	0 0 0 1	0 0 0 1	1	1	$DP_L \leftarrow (DP_L) + 1$	Increment the contents of DP_L.	ZF	
DL	Decrement DP_L	0 0 1 0	0 0 0 1	1	1	$DP_L \leftarrow (DP_L) - 1$	Decrement the contents of DP_L.	ZF	
IY	Increment DP_Y	0 0 0 1	0 0 1 1	1	1	$DP_Y \leftarrow (DP_Y) + 1$	Increment the contents of DP_Y.	ZF	
DY	Decrement DP_Y	0 0 1 0	0 0 1 1	1	1	$DP_Y \leftarrow (DP_Y) - 1$	Decrement the contents of DP_Y.	ZF	
TAH	Transfer AC to DP_H	1 1 0 0 1 1 1 1	1 1 1 1 0 0 0 0	2	2	$DP_H \leftarrow (AC)$	Transfer the contents of AC to DP_H.		
THA	Transfer DP_H to AC	1 1 0 0 1 1 1 0	1 1 1 1 0 0 0 0	2	2	$AC \leftarrow (DP_H)$	Transfer the contents of DP_H to AC.	ZF	
XAH	Exchange AC with DP_H	0 1 0 0	0 0 0 0	1	1	$(AC) \leftrightarrow (DP_H)$	Exchange the contents of AC and DP_H.		
TAL	Transfer AC to DP_L	1 1 0 0 1 1 1 1	1 1 1 1 0 0 0 1	2	2	$DP_L \leftarrow (AC)$	Transfer the contents of AC to DP_L.		
TLA	Transfer DP_L to AC	1 1 0 0 1 1 1 0	1 1 1 1 0 0 0 1	2	2	$AC \leftarrow (DP_L)$	Transfer the contents of DP_L to AC.	ZF	
XAL	Exchange AC with DP_L	0 1 0 0	0 0 0 1	1	1	$(AC) \leftrightarrow (DP_L)$	Exchange the contents of AC and DP_L.		
TAX	Transfer AC to DP_X	1 1 0 0 1 1 1 1	1 1 1 1 0 0 1 0	2	2	$DP_X \leftarrow (AC)$	Transfer the contents of AC to DP_X.		
TXA	Transfer DP_X to AC	1 1 0 0 1 1 1 0	1 1 1 1 0 0 1 0	2	2	$AC \leftarrow (DP_X)$	Transfer the contents of DP_X to AC.	ZF	
XAX	Exchange AC with DP_X	0 1 0 0	0 0 1 0	1	1	$(AC) \leftrightarrow (DP_X)$	Exchange the contents of AC and DP_X.		
TAY	Transfer AC to DP_Y	1 1 0 0 1 1 1 1	1 1 1 1 0 0 1 1	2	2	$DP_Y \leftarrow (AC)$	Transfer the contents of AC to DP_Y.		
TYA	Transfer DP_Y to AC	1 1 0 0 1 1 1 0	1 1 1 1 0 0 1 1	2	2	$AC \leftarrow (DP_Y)$	Transfer the contents of DP_Y to AC.	ZF	
XAY	Exchange AC with DP_Y	0 1 0 0	0 0 1 1	1	1	$(AC) \leftrightarrow (DP_Y)$	Exchange the contents of AC and DP_Y.		
[Flag manipulation instructions]									
SFB n4	Set flag bit	0 1 1 1	n_3 n_2 n_1 n_0	1	1	$Fn \leftarrow 1$	Set the flag specified by n4 to 1.		
RFB n4	Reset flag bit	0 0 1 1	n_3 n_2 n_1 n_0	1	1	$Fn \leftarrow 0$	Reset the flag specified by n4 to 0.	ZF	
[Jump and subroutine instructions]									
JMP addr	Jump in the current bank	1 1 1 0 P_7 P_6 P_5 P_4	P_{11} P_{10} P_9 P_8 P_3 P_2 P_1 P_0	2	2	PC13, 12 ← PC13, 12 PC11 to 0 ← P_{11} to P_8	Jump to the location in the same bank specified by the immediate data P12.		This becomes PC12 + $\overline{(PC12)}$ immediately following a BANK instruction.
JPEA	Jump to the address stored at E and AC in the current page	0 0 1 0	0 1 1 1	1	1	PC13 to 8 ← PC13 to 8, PC7 to 4 ← (E), PC3 to 0 ← (AC)	Jump to the location determined by replacing the lower 8 bits of the PC by E, AC.		
CAL addr	Call subroutine	0 1 0 1 P_7 P_6 P_5 P_4	0 P_{10} P_9 P_8 P_3 P_2 P_1 P_0	2	2	PC13 to 11 ← 0, PC10 to 0 ← P_{10} to P_0, M4 (SP) ← (CF, ZF, PC13 to 0), SP ← (SP)-4	Call a subroutine.		
CZP addr	Call subroutine in the zero page	1 0 1 0	P_3 P_2 P_1 P_0	1	2	PC13 to 6, PC10 ← 0, PC5 to 2 ← P_3 to P_0, M4 (SP) ← (CF, ZF, PC12 to 0), SP ← SP-4	Call a subroutine on page 0 in bank 0.		
BANK	Change bank	0 0 0 1	1 0 1 1	1	1		Change the memory bank and register bank.		

Continued on next page.

Continued from preceding page.

Mnemonic		Instruction code		Number of bytes	Number of cycles	Operation	Description	Affected status bits	Note
		D_7 D_6 D_5 D_4	D_3 D_2 D_1 D_0						
[Jump and subroutine instructions]									
PUSH reg	Push reg on M2 (SP)	1 1 0 0 1 1 1 1	1 1 1 1 1 i_1 i_0 0	2	2	M2 (SP) ← (reg) SP ← (SP) – 2	Store the contents of reg in M2 (SP). Subtract 2 from SP after the store. reg: HL (i_1=0, i_0=0); XY (0, 1); AE (1, 0); Illegal value (1, 1)		
POP reg	Pop reg off M2 (SP)	1 1 0 0 1 1 1 0	1 1 1 1 1 i_1 i_0 0	2	2	SP ← (SP) + 2 reg ← [M2 (SP)]	Add 2 to SP and then load the contents of M2(SP) into reg. The relation between i1i0 and reg is the same as that for the PUSH reg instruction.		
RT	Return from subroutine	0 0 0 1	1 1 0 0	1	2	SP ← (SP) + 4 PC ← [M4 (SP)]	Return from a subroutine or interrupt handling routine. ZF and CF are not restored.		
RTI	Return from interrupt routine	0 0 0 1	1 1 0 1	1	2	SP ← (SP) + 4 PC ← [M4 (SP)] CF, ZF ← [M4 (SP)]	Return from a subroutine or interrupt handling routine. ZF and CF are restored.	ZF, CF	
[Branch instructions]									
BAt2 addr	Branch on AC bit	1 1 0 1 P_7 P_6 P_5 P_4	0 0 t_1 t_0 P_3 P_2 P_1 P_0	2	2	PC7 to 0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if (AC, t2) = 1	Branch to the location in the same page specified by P_7 to P_0 if the bit in AC specified by the immediate data t_1 t_0 is one.		
BNAt2 addr	Branch on no AC bit	1 0 0 1 P_7 P_6 P_5 P_4	0 0 t_1 t_0 P_3 P_2 P_1 P_0	2	2	PC7 to 0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if (AC, t2) = 0	Branch to the location in the same page specified by P_7 to P_0 if the bit in AC specified by the immediate data t_1 t_0 is zero.		
BMt2 addr	Branch on M bit	1 1 0 1 P_7 P_6 P_5 P_4	0 1 t_1 t_0 P_3 P_2 P_1 P_0	2	2	PC7 to 0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if [M (HL),t2] = 1	Branch to the location in the same page specified by P_7 to P_0 if the bit in M (HL) specified by the immediate data t_1 t_0 is one.		
BNMt2 addr	Branch on no M bit	1 0 0 1 P_7 P_6 P_5 P_4	0 1 t_1 t_0 P_3 P_2 P_1 P_0	2	2	PC7 to 0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if [M (HL),t2] = 0	Branch to the location in the same page specified by P_7 to P_0 if the bit in M (HL) specified by the immediate data t_1 t_0 is zero.		
BPt2 addr	Branch on Port bit	1 1 0 1 P_7 P_6 P_5 P_4	1 0 t_1 t_0 P_3 P_2 P_1 P_0	2	2	PC7 to 0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if [P (DP_L), t2] = 1	Branch to the location in the same page specified by P_7 to P_0 if the bit in port (DP_L) specified by the immediate data t_1 t_0 is one.		Internal control registers can also be tested by executing this instruction immediately after a BANK instruction. However, this is limited to registers that can be read out.
BNPt2 addr	Branch on no Port bit	1 0 0 1 P_7 P_6 P_5 P_4	1 0 t_1 t_0 P_3 P_2 P_1 P_0	2	2	PC7 to 0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if [P (DP_L), t2] = 0	Branch to the location in the same page specified by P_7 to P_0 if the bit in port (DP_L) specified by the immediate data t_1 t_0 is zero.		Internal control registers can also be tested by executing this instruction immediately after a BANK instruction. However, this is limited to registers that can be read out.

Continued on next page.

Continued from preceding page.

Mnemonic		Instruction code		Number of bytes	Number of cycles	Operation	Description	Affected status bits	Note
		D_7 D_6 D_5 D_4	D_3 D_2 D_1 D_0						
[Branch instructions]									
BC addr	Branch on CF	1 1 0 1 P_7 P_6 P_5 P_4	1 1 0 0 P_3 P_2 P_1 P_0	2	2	PC7 to 0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if (CF) = 1	Branch to the location in the same page specified by P_7 to P_0 if CF is one.		
BNC addr	Branch on no CF	1 0 0 1 P_7 P_6 P_5 P_4	1 1 0 0 P_3 P_2 P_1 P_0	2	2	PC7 to 0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if (CF) = 0	Branch to the location in the same page specified by P_7 to P_0 if CF is zero.		
BZ addr	Branch on ZF	1 1 0 1 P_7 P_6 P_5 P_4	1 1 0 1 P_3 P_2 P_1 P_0	2	2	PC7 to 0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if (ZF) = 1	Branch to the location in the same page specified by P_7 to P_0 if ZF is one.		
BNZ addr	Branch on no ZF	1 0 0 1 P_7 P_6 P_5 P_4	1 1 0 1 P_3 P_2 P_1 P_0	2	2	PC7 to 0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if (ZF) = 0	Branch to the location in the same page specified by P_7 to P_0 if ZF is zero.		
BFn4 addr	Branch on flag bit	1 1 1 1 P_7 P_6 P_5 P_4	n_3 n_2 n_1 n_0 P_3 P_2 P_1 P_0	2	2	PC7 to 0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if (Fn) = 1	Branch to the location in the same page specified by P_0 to P_7 if the flag (of the 16 user flags) specified by n_3 n_2 n_1 n_0 is one.		
BNFn4 addr	Branch on no flag bit	1 0 1 1 P_7 P_6 P_5 P_4	n_3 n_2 n_1 n_0 P_3 P_2 P_1 P_0	2	2	PC7 to 0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if (Fn) = 0	Branch to the location in the same page specified by P_0 to P_7 if the flag (of the 16 user flags) specified by n_3 n_2 n_1 n_0 is zero.		
[I/O instructions]									
IP0	Input port 0 to AC	0 0 1 0	0 0 0 0	1	1	AC ← (P0)	Input the contents of port 0 to AC.	ZF	
IP	Input port to AC	0 0 1 0	0 1 1 0	1	1	AC ← [P (DP_L)]	Input the contents of port P (DP_L) to AC.	ZF	
IPM	Input port to M	0 0 0 1	1 0 0 1	1	1	M (HL) ← [P (DP_L)]	Input the contents of port P (DP_L) to M (HL).		
IPDR i4	Input port to AC direct	1 1 0 0 0 1 1 0	1 1 1 1 I_3 I_2 I_1 I_0	2	2	AC ← [P (i4)]	Input the contents of P (i4) to AC.	ZF	
IP45	Input port 4, 5 to E, AC respectively	1 1 0 0 1 1 0 1	1 1 1 1 0 1 0 0	2	2	E ← [P (4)] AC ← [P (5)]	Input the contents of ports P (4) and P (5) to E and AC respectively.		
OP	Output AC to port	0 0 1 0	0 1 0 1	1	1	P (DP_L) ← (AC)	Output the contents of AC to port P (DP_L).		
OPM	Output M to port	0 0 0 1	1 0 1 0	1	1	P (DP_L) ← [M (HL)]	Output the contents of M (HL) to port P (DP_L).		
OPDR i4	Output AC to port direct	1 1 0 0 0 1 1 1	1 1 1 1 I_3 I_2 I_1 I_0	2	2	P (i4) ← (AC)	Output the contents of AC to P (i4).		
OP45	Output E, AC to port 4, 5 respectively	1 1 0 0 1 1 0 1	1 1 1 1 0 1 0 1	2	2	P (4) ← (E) P (5) ← (AC)	Output the contents of E and AC to ports P (4) and P (5) respectively.		
SPB t2	Set port bit	0 0 0 0	1 0 t_1 t_0	1	1	[P (DP_L), t2] ← 1	Set to one the bit in port P (DP_L) specified by the immediate data t_1 t_0.		
RPB t2	Reset port bit	0 0 1 0	1 0 t_1 t_0	1	1	[P (DP_L), t2] ← 0	Clear to zero the bit in port P (DP_L) specified by the immediate data t_1 t_0.	ZF	
ANDPDR i4, p4	And port with immediate data then output	1 1 0 0 I_3 I_2 I_1 I_0	0 1 0 1 P_3 P_2 P_1 P_0	2	2	P (P_3 to P_0) ← [P (P_3 to P_0)] ∨ I_3 to I_0	Take the logical AND of P (P_3 to P_0) and the immediate data I_3 I_2 I_1 I_0 and output the result to P (P_3 to P_0).	ZF	
ORPDR i4, p4	Or port with immediate data then output	1 1 0 0 I_3 I_2 I_1 I_0	0 1 0 0 P_3 P_2 P_1 P_0	2	2	P (P_3 to P_0) ← [P (P_3 to P_0)] ∨ I_3 to I_0	Take the logical OR of P (P_3 to P_0) and the immediate data I_3 I_2 I_1 I_0 and output the result to P (P_3 to P_0).	ZF	

Continued on next page.

Continued from preceding page.

Mnemonic		Instruction code		Number of bytes	Number of cycles	Operation	Description	Affected status bits	Note
		D_7 D_6 D_5 D_4	D_3 D_2 D_1 D_0						
[Timer control instructions]									
WTTM0	Write timer 0	1 1 0 0	1 0 1 0	1	2	TIMER0 ← [M2 (HL)], (AC)	Write the contents of M2 (HL), AC into the timer 0 reload register.		
WTTM1	Write timer 1	1 1 0 0 1 1 1 1	1 1 1 1 0 1 0 0	2	2	TIMER1 ← (E), (AC)	Write the contents of E, AC into the timer 1 reload register A.		
RTIM0	Read timer 0	1 1 0 0	1 0 1 1	1	2	M2 (HL), AC ← (TIMER0)	Read out the contents of the timer 0 counter into M2 (HL), AC.		
RTIM1	Read timer 1	1 1 0 0 1 1 1 1	1 1 1 1 0 1 0 1	2	2	E, AC ← (TIMER1)	Read out the contents of the timer 1 counter into E, AC.		
START0	Start timer 0	1 1 0 0 1 1 1 0	1 1 1 1 0 1 1 0	2	2	Start timer 0 counter	Start the timer 0 counter.		
START1	Start timer 1	1 1 0 0 1 1 1 0	1 1 1 1 0 1 1 1	2	2	Start timer 1 counter	Start the timer 1 counter.		
STOP0	Stop timer 0	1 1 0 0 1 1 1 1	1 1 1 1 0 1 1 0	2	2	Stop timer 0 counter	Stop the timer 0 counter.		
STOP1	Stop timer 1	1 1 0 0 1 1 1 1	1 1 1 1 0 1 1 1	2	2	Stop timer 1 counter	Stop the timer 1 counter.		
[Interrupt control instructions]									
MSET	Set interrupt master enable flag	1 1 0 0 0 1 0 1	1 1 0 1 0 0 0 0	2	2	MSE ← 1	Set the interrupt master enable flag to one.		
MRESET	Reset interrupt master enable flag	1 1 0 0 1 0 0 1	1 1 0 1 0 0 0 0	2	2	MSE ← 0	Clear the interrupt master enable flag to zero.		
EIH i4	Enable interrupt high	1 1 0 0 0 1 0 1	1 1 0 1 I_3 I_2 I_1 I_0	2	2	EDIH ← (EDIH) ∨ i4	Set the interrupt enable flag to one.		
EIL i4	Enable interrupt low	1 1 0 0 0 1 0 0	1 1 0 1 I_3 I_2 I_1 I_0	2	2	EDIL ← (EDIL) ∨ i4	Set the interrupt enable flag to one.		
DIH i4	Disable interrupt high	1 1 0 0 1 0 0 1	1 1 0 1 I_3 I_2 I_1 I_0	2	2	EDIH ← (EDIH) ∧ $\overline{i4}$	Clear the interrupt enable flag to zero.	ZF	
DIL i4	Disable interrupt low	1 1 0 0 1 0 0 0	1 1 0 1 I_3 I_2 I_1 I_0	2	2	EDIL ← (EDIL) ∧ $\overline{i4}$	Clear the interrupt enable flag to zero.	ZF	
WTSP	Write SP	1 1 0 0 1 1 0 1	1 1 1 1 1 0 1 0	2	2	SP ← (E), (AC)	Transfer the contents of E, AC to SP.		
RSP	Read SP	1 1 0 0 1 1 0 1	1 1 1 1 1 0 1 1	2	2	E, AC ← (SP)	Transfer the contents of SP to E, AC.		
[Standby control instructions]									
HALT	HALT	1 1 0 0 1 1 0 1	1 1 1 1 1 1 1 0	2	2	HALT	Enter halt mode.		
HOLD	HOLD	1 1 0 0 1 1 0 1	1 1 1 1 1 1 1 1	2	2	HOLD	Enter hold mode.		
[Serial I/O control instructions]									
STARTS	Start serial I O	1 1 0 0 1 1 1 0	1 1 1 1 1 1 1 0	2	2	START SI O	Start SIO operation.		
WTSIO	Write serial I O	1 1 0 0 1 1 1 0	1 1 1 1 1 1 1 1	2	2	SIO ← (E), (AC)	Write the contents of E, AC to SIO.		
RSIO	Read serial I O	1 1 0 0 1 1 1 1	1 1 1 1 1 1 1 1	2	2	E, AC ← (SIO)	Read out the contents of SIO into E, AC.		
[Other instructions]									
NOP	No operation	0 0 0 0	0 0 0 0	1	1	No operation	Consume one machine cycle without performing any operation.		
SB i2	Select bank	1 1 0 0 1 1 0 0	1 1 1 1 0 0 I_1 I_0	2	2	PC13, PC12 ← I_1 I_0	Specify the memory bank.		