Ordering number : EN4457

CMOS LSI

SANYO No.4457

LC66562B, 66566B

12 and 16-kbyte on-chip ROM capacities CMOS 4-bit single-chip microcontrollers

Overview

The LC66562B and 66566B are CMOS 4-bit microcontrollers that integrate in a single chip all the functionality required for control applications, including ROM, RAM, I/O ports, serial interfaces, comparator inputs, three-value inputs, timers, and interrupt functions. The LC66562B and 66566B are provided in LC66566A series.

The LC66562B and 66566B differ from the earlier LC66566A series in the power supply voltage range and certain other electrical specifications.

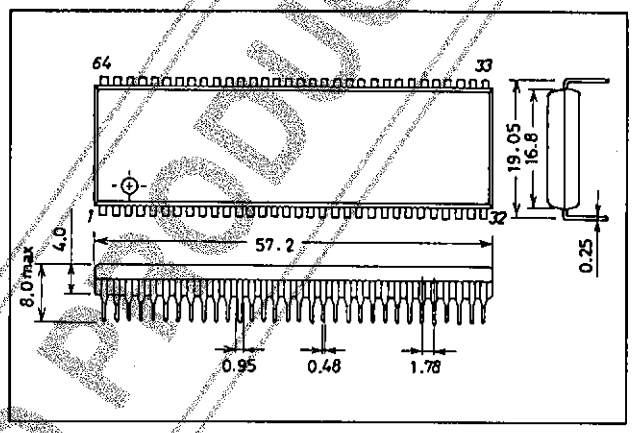
Features and Functions

- A 12- or 16-kbyte on-chip ROM capacity, and an on-chip RAM capacity of 512 by 4 bits
- Support for the full 128 instruction LC66000 series instruction set
- Fifty four I/O ports
- Two 8-bit serial interfaces (that can be connected in a 16-bit cascade format)
- Instruction cycle: 0.92 to 10 µs (3 to 5.5 V)
- Powerful timer and prescaler functions
  - 12-bit timer based time limit timer, event counter, pulse width measurement, and square wave output functions
  - 8-bit timer based time limit timer, event counter, PWM output, and square wave output functions
  - 12-bit prescaler based time base function
- Powerful 11-factor 8-vector interrupt system
  - External interrupts: 6 factors, 3 vectors
  - Internal interrupts: 5 factors, 5 vectors
- Flexible I/O functions
  - Comparator inputs, three-value inputs, 20 mA drive outputs, 15 V breakdown voltage, pull-up/open drain option switching
- Runaway detection function (watchdog timer) option
- 8-bit I/O functions
- Low power functions using halt and hold modes
- Packages: DIP64S, QFP64E
- Test and evaluation ICs: LC66599 (evaluation chip) + EVA850/800-TB665xx
  LC66E516 (built-in EPROM microcontroller)
  LC66P516 (built-in OTPROM microcontroller)

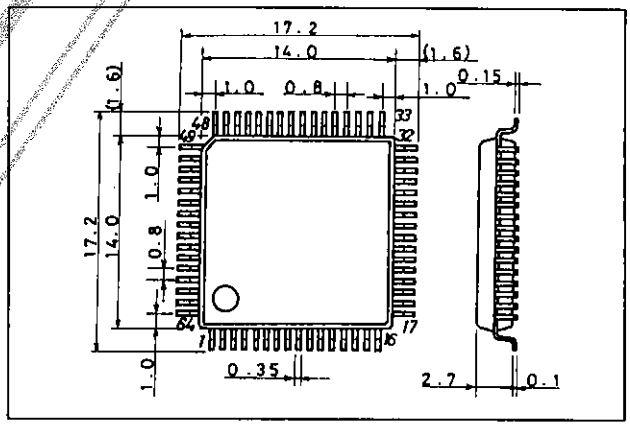
Package Dimensions

unit: mm

3071-DIP64S



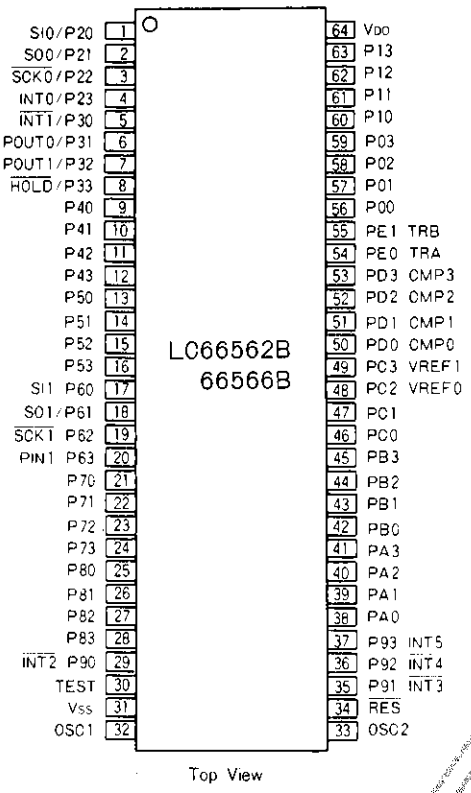
3159-QFP64E



SANYO Electric Co., Ltd. Semiconductor Business Headquarters
TOKYO OFFICE Tokyo Bldg., 1-10, 1 Chome, Ueno, Taito-ku, TOKYO, 110 JAPAN

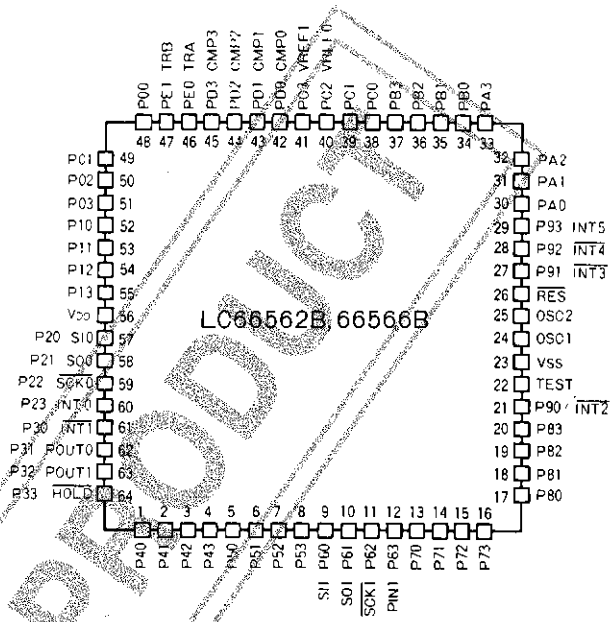
Pin Assignment

DIP64S



Top View

QFP64E

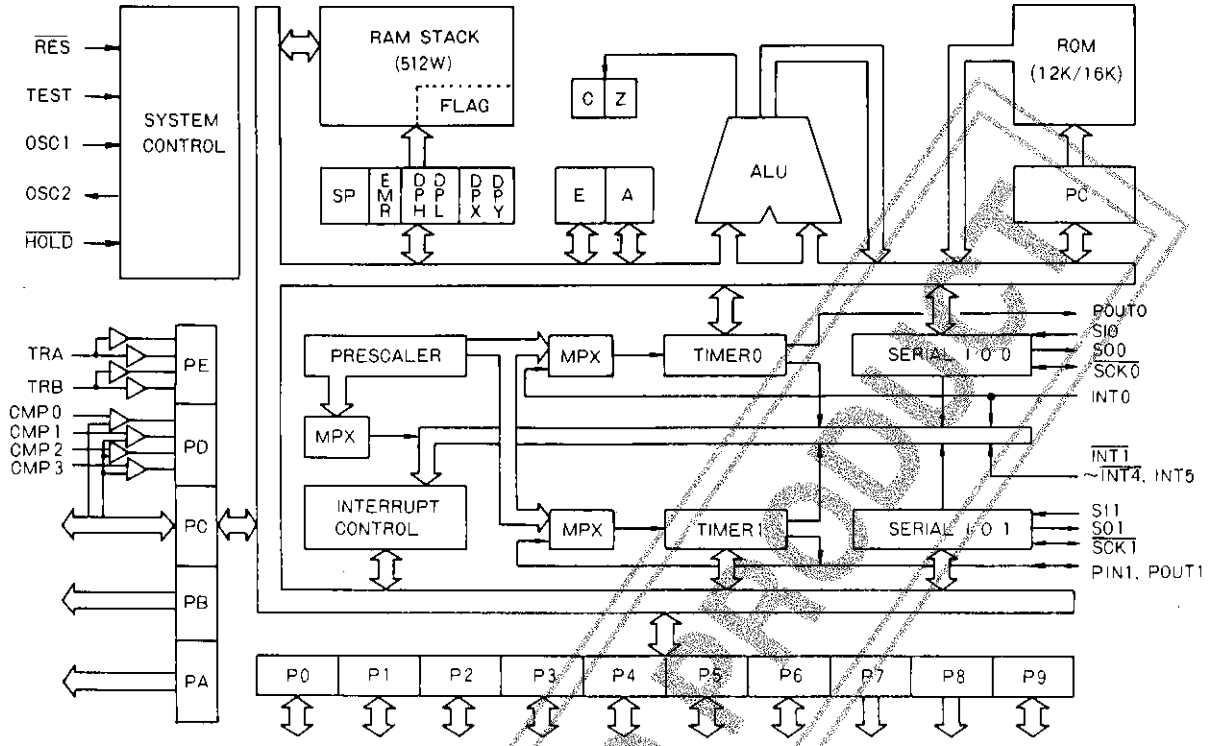


Series Organization

Product no.	Pin count	ROM capacity	RAM capacity	Package		Features
LC66304A/306A/308A	42	4 k/6 k/8 kB	512 W	DIP42S	QFP48E	Normal version
LC66404A/406A/408A	42	4 k/6 k/8 kB	512 W	DIP42S	QFP48E	4.0 to 6.0 V/0.92 µs
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 version
LC66354S/356S/358S*	42	4 k/6 k/8 kB	512 W		QFP44M	2.2 to 5.5 V/3.92 µs
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 version
LC66556B/558B*	64	6 k/8 k	512 W	DIP64S	QFP64E	3.0 to 5.5 V/0.92 µs
LC66562B/566B	64	12 k/16 kB	512 W	DIP64S	QFP64E	
LC66E308	42	EPROM 8 kB	512 W	DIC42S (window)	QFC48 (window)	Window and OTP versions for testing and evaluation
LC66P308	42	OTPROM 8 kB	512 W	DIP42S	QFP48E	
LC66E408	42	EPROM 8 kB	512 W	DIC42S (window)	QFC48 (window)	4.5 to 5.5 V/0.92 µs
LC66P408	42	OTPROM 8 kB	512 W	DIP42S	QFP48E	
LC66E516	64	EPROM 16 kB	512 W	DIC64S (window)	QFC64 (window)	
LC66P516	64	OTPROM 16 kB	512 W	DIP64S	QFP64E	

Note: * Products market with an asterisk are under development.

System Block Diagram



Differences between the LC66562B/LC66566B and the LC66516B Series

Parameter	LC66516B series (including the EVA850/800-TB665XX tools)	LC66562B, 66566B
System level differences • Hardware waiting time for clearing hold mode	65536 cycles: About 64 ms at 4 MHz (Tcyc = 1 µs)	16384 cycles: About 16 ms at 4 MHz (Tcyc = 1 µs)
• The timer 0 value on reset (including the value after clearing hold mode)	The value FF0 is loaded	The value FFC is loaded
Feature differences • Operating power supply voltage/operating speed	LC66512B, 516B 4.0 to 6.0 V/0.92 to 10 µs LC66E516, P516 4.5 to 5.5 V/0.92 to 10 µs	3.0 to 5.5 V/0.92 to 10 µs

- An RC oscillator cannot be used with the LC66562B and LC66566B
- In addition there are differences in output currents and comparator input voltages.
  For details, refer to the LC66516B, LC66E516, and LC66P516 catalogs.
  Special care should be exercised with respect to these differences when using the LC66E516, and LC66P516 evaluation products.

Pin Description

Pin	I/O	Function	Output driver type	Options	Value on reset
P00 P01 P02 P03	I/O	I/O ports P00 to P03 • Input and output in 1 or 4 bit units • P00 to P03 have halt mode control functions	Pch: Pu MOS type Nch: Low sink current type	• Pu MOS attached or Nch OD output • Output level on reset	High or low (option)
P10 P11 P12 P13	I/O	I/O ports P10 to P13 • Input and output in 1 or 4 bit units	Pch: Pu MOS type Nch: Low sink current type	• Pu MOS attached or Nch OD output • Output level on reset	High or low (option)
P20/SI0 P21/SO0 P22/$\overline{SCK0}$ P23/INT0	I/O	I/O ports P20 to P23 • Input and output in 1 or 4 bit units • P20 is also used as the serial input SI0. P21 is also used as the serial output SO0. P22 is also used as the serial clock $\overline{SCK0}$. • P23 is also used as the INT0 interrupt request, the timer 0 event count, and the pulse width measurement inputs	Pch: CMOS type Nch: Low sink current type Breakdown voltage of +15 V on Nch OD	• CMOS or Nch OD output	High
P30/$\overline{INT1}$ P31/POUT0 P32/POUT1	I/O	I/O ports P30 to P32 • Input and output in 1 or 3 bit units • P30 is also used as the $\overline{INT1}$ interrupt request • P31 is also used as the square wave output from timer 0 • P32 is also used as the square wave output and PWM output from timer 1	Pch: CMOS type Nch: Low sink current type Breakdown voltage of +15 V on Nch OD	• CMOS or Nch OD output	High
P33/$\overline{HOLD}$	I	Hold mode control input • Hold mode is set by a HOLD instruction when $\overline{HOLD}$ is low • In hold mode the CPU is restarted by setting the $\overline{HOLD}$ input high • This pin can be used as input port P33 along with P30 to P32 • When the P33/$\overline{HOLD}$ pin is low, the CPU will not be reset by a low level on $\overline{RES}$. Therefore, applications must not hold the P33/$\overline{HOLD}$ pin low when power is first applied.			
P40 P41 P42 P43	I/O	I/O ports P40 to P43 • Input and output in 1 or 4 bit units • Input and output in 8 bit units when used in conjunction with P50 to P53. • ROM data output in 8 bit units when used in conjunction with P50 to P53	Pch: Pu MOS type Nch: Low sink current type	• Pu MOS attached or Nch OD output	High
P50 P51 P52 P53	I/O	I/O ports P50 to P53 • Input and output in 1 or 4 bit units • Input and output in 8 bit units when used in conjunction with P40 to P43. • ROM data output in 8 bit units when used in conjunction with P40 to P43	Pch: Pu MOS type Nch: Low sink current type	• Pu MOS attached or Nch OD output	High
P60/SI1 P61/SO1 P62/$\overline{SCK1}$ P63/PIN1	I/O	I/O ports P60 to P63 • Input and output in 1 or 4 bit units • P60 is also used as the serial input SI1. • P61 is also used as the serial output SO1. • P62 is also used as the serial clock $\overline{SCK1}$ • P63 is also used as the event count input to timer 1	Pch: CMOS type Nch: Low sink current type Breakdown voltage of +15 V on Nch OD	• CMOS or Nch OD output output	High

Notes: PU MOS attached: Output pins with a pull-up MOS transistor
CMOS output: Complementary output
OD output: Open drain output

Continued on next page.

Continued from preceding page.

Pin	I/O	Function	Output driver type	Options	Value on reset
P70 P71 P72 P73	O	Output only ports P70 to P73 • Output in 1 or 4 bit units • The latched output values are read in by input instructions	Pch: Pu MOS type Nch: Low sink current type Breakdown voltage of +15 V on Nch OD	• Pu MOS attached or Nch OD output	High
P80 P81 P82 P83	O	Output only ports P80 to P83 • Output in 1 or 4 bit units • The latched output values are read in by input instructions • Pch OD output option available	Pch: CMOS type Nch: Low sink current type	• CMOS or Pch OD output • Output level on reset	High or low (option)
P90/$\overline{INT2}$ P91/$\overline{INT3}$ P92/$\overline{INT4}$ P93/INT5	I/O	I/O ports P90 to P93 • Input and output in 1 or 4 bit units • P90 is also used as the $\overline{INT2}$ interrupt request • P91 is also used as the $\overline{INT3}$ interrupt request • P92 is also used as the $\overline{INT4}$ interrupt request • P93 is also used as the INT5 interrupt request	Pch: CMOS type Nch: Low sink current type	• CMOS or Nch OD output	High
PA0 PA1 PA2 PA3	O	Output only ports PA0 to PA3 • Output in 1 or 4 bit units • The latched output values are read in by input instructions	Pch: Pu MOS type Nch: Low sink current type Breakdown voltage of +15 V on Nch OD	• Pu MOS attached or Nch OD output	High
PB0 PB1 PB2 PB3	O	Output only ports PB0 to PB3 • Output in 1 or 4 bit units • The latched output values are read in by input instructions	Pch: Pu MOS type Nch: Low sink current type	• Pu MOS attached or Nch OD output	High
PC0 PC1 PC2/VREF0 PC3/VREF1	I/O	I/O ports PC0 to PC3 • Input and output in 1 or 4 bit units • PC2 is also used as the VREF0 comparator comparison voltage pin • PC3 is also used as the VREF1 comparator comparison voltage pin	Pch: CMOS type Nch: Low sink current type	• CMOS or Nch OD output	High
PD0/CMP0 PD1/CMP1 PD2/CMP2 PD3/CMP3	I	Input only ports PD0 to PD3 • Can be switched in software to function as comparator inputs. The PD0 comparison voltage is VREF0. The PD1 to PD3 comparison voltage is VREF1. Comparators are specified in units of PD0, PD1 (PD2, PD3).			Normal input
PE0/TRA PE1/TRB	I	Input only ports • Can be switched in software to function as three value input ports			Normal input
OSC1 OSC2	I O	System clock oscillator pins. When an external clock is used, leave OSC2 open and input the clock signal to OSC1.		• Selection of ceramic oscillator or external clock	
$\overline{RES}$	I	System reset input pin • The CPU is initialized when P33/$\overline{HOLD}$ is high and $\overline{RES}$ is set to low			
TEST	I	CPU test pin. Always connect this pin to V_{SS} during normal operation.			
V_{DD} V_{SS}		Power supply pins			

Notes: PU MOS attached: Output pins with a pull-up MOS transistor
CMOS output: Complementary output
OD output: Open drain output

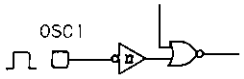
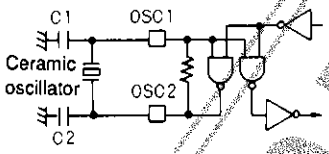
User Options

1. Ports 0, 1, and 8 output level on reset option

The output levels for the I/O ports 0, 1, and 8 in 4-bit sets can selected to be either of the following two option.

Option	Conditions
High level output after reset	The 4 bits of each of ports 0, 1, and 8 as a set
Low level output after reset	The 4 bits of each of ports 0, 1, and 8 as a set

2. Oscillator circuit options

Option	Circuit	Conditions
External clock		Schmidt input
Ceramic oscillator		

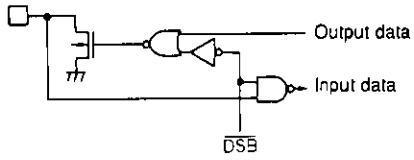
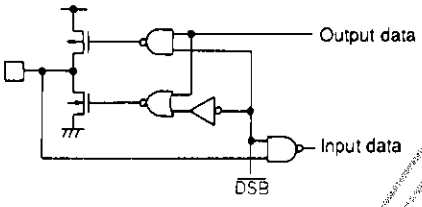
Note: There is no RC oscillator option

3. Watchdog timer option

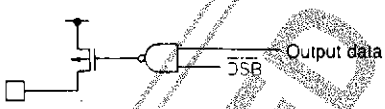
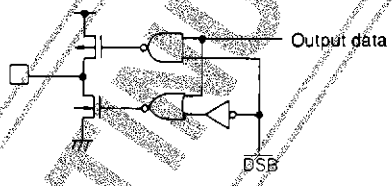
The provision or exclusion of a runaway program detection function can be selected as an option.

4. Port output format option

- The following two options are available in bit units for each of the ports P0, P1, P2, P3 (except for P33/$\overline{\text{HOLD}}$), P4, P5, P6, P7, P9, PA, PB, and PC.

Option	Circuit	Conditions
Open drain output		P7, PA, and PB are output only. P2, P3, P6, and P9 are schmidt inputs.
Pull-up resistor output		P7, PA, and PB are output only. P2, P3, P6, and P9 are schmidt inputs. The CMOS output (P2, P3, P6, P9, and PC) and the Pu-MOS outputs (P0, P1, P4, P5, P7, PA, and PB) are differentiated by the Pch Tr's drive ability.

- The following two options can are available in bit units for P8.

Option	Circuit	Conditions
Open drain output		
Pull-down resistor output		

- The PD comparator inputs and the PE three value inputs are selected in software.

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

Parameter	Symbol	Applicable pins, notes	Conditions	Rated value	Unit	Notes
Maximum power supply voltage	V_{DD} max	V_{DD}		−0.3 to +7.0	V	
Input voltages	V_{IN}(1)	P2, P3 (except for P33/$\overline{HOLD}$), P6		−0.3 to +15.0	V	1
	V_{IN}(2)	Other inputs		−0.3 to V_{DD} + 0.3	V	2
Output voltages	V_{OUT}(1)	P2, P3 (except for P33/$\overline{HOLD}$), P6, P7, PA		−0.3 to +15.0	V	1
	V_{OUT}(2)	Other outputs		−0.3 to V_{DD} + 0.3	V	2
Output current per pin	I_{ON}(1)	P0, P1, P2, P3 (except for P33/$\overline{HOLD}$), P4, P5, P6, P8, P9, PC		4	mA	3
	I_{ON}(2)	P7, PA, PB		20	mA	3
	$-I_{OP}$(1)	P0, P1, P4, P5, P7, PA, PB		2	mA	4
	$-I_{OP}$(2)	P2, P3 (except for P33/$\overline{HOLD}$), P6, P8, P9, PC		4	mA	4
Total pin current	ΣI_{ON}(1)	P2, P3 (except for P33/$\overline{HOLD}$), P4, P5, P6, P7, P8		75	mA	3
	ΣI_{ON}(2)	P0, P1, P9, PA, PB, PC		75	mA	3
	$-\Sigma I_{OP}$(1)	P2, P3 (except for P33/$\overline{HOLD}$), P4, P5, P6, P7, P8		25	mA	4
	$-\Sigma I_{OP}$(2)	P0, P1, P9, PA, PB, PC		25	mA	4
Maximum power dissipation	Pd max	Ta = −30 to +70°C	DIP64S (QFP64E)	600 (430)	mW	5
Operating temperature range	Topr			−30 to +70	°C	
Storage temperature range	T_{stg}			−55 to +125	°C	

Notes: 1. Applies to open drain output specification pins. For output specifications other than open drain, the ratings in other pin items apply.
2. For oscillator input and output, up to the free running oscillation frequency is allowed.
3. Input current (For pin P8, applies to the CMOS output specifications.)
4. Output current (For pins other than P8, applies to the pull-up output and to the CMOS output specifications.)
5. We recommend reflow soldering as the soldering method for QFP packages.
Please contact your Sanyo sales or technical representative for the conditions required for the use of methods in which the package itself is directly immersed in a solder dip bath.

Allowable Operating Ranges, Unless Otherwise Specified at Ta = –30 to +70°C, V_{SS} = 0 V, V_{DD} = 3.0 to 5.5 V

Parameter		Symbol	Applicable pins	Conditions	VDD(V)	Ratings min	Ratings typ	Ratings max	Unit	Notes
Operating power supply voltage		V_{DD}	V_{DD}			3.0		5.5	V	
Memory retention power 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 P33/$\overline{HOLD}$), P6	With the output Nch Tr off		0.8 V_{DD}		13.5	V	1
		V_{IH}(2)	P33/$\overline{HOLD}$, P9 $\overline{RES}$, OSC1	With the output Nch Tr off		0.8 V_{DD}		V_{DD}	V	2
		V_{IH}(3)	P0, P1, P4, P5, PC, PD, PE	With the output Nch Tr off		0.75 V_{DD}		V_{DD}	V	3
		V_{IH}(4)	PE	When three value inputs are used		0.8 V_{DD}		V_{DD}	V	
Intermediate level input voltage		V_{IM}	PE	When three value input used		0.4 V_{DD}		0.6 V_{DD}	V	
Common mode input voltage rage		V_{CMM}(1)	PD0, PC2	When comparator input used		1.5		V_{DD}	V	
		V_{CMM}(2)	PD1, PD2, PD3, PC3			V_{SS}		V_{DD} – 1.5		
Input low level voltage		V_{IL}(1)	P2, P3 (except for P33/$\overline{HOLD}$), P6, P9, $\overline{RES}$, OSC1	Output Nch Tr off		V_{SS}		0.2 V_{DD}	V	2
		V_{IL}(2)	P33/$\overline{HOLD}$		1.8 to 5.5	V_{SS}		0.2 V_{DD}	V	
		V_{IL}(3)	P0, P1, P4, P5, PC, PD, PE, TEST	Output Nch Tr off		V_{SS}		0.25 V_{DD}	V	3
		V_{IL}(4)	PE	When three value input used		V_{SS}		0.2 V_{DD}	V	
Operating frequency		fop (Tcyc)				0.4 (10)		4.35 (0.92)	MHz (µs)	
External clock input conditions	Frequency	f_{ext}	OSC1	According to figure 1. Clock input to OSC1, OSC2 left open. (When the external clock input option is selected as the oscillator circuit.)		0.4		4.35	MHz	
	Pulse width	t_{extH} t_{extL}				100			ns	
	Rise and fall time	t_{extR} t_{extF}						30	ns	

Notes: 1. Applies to the open drain specification pins. However, V_{IH}(2) applies to P33/$\overline{HOLD}$.
2. Applies to the open drain specification pins. The CMOS output specification P9 cannot be used for input.
3. When PE is used for three value input, V_{IH}(4), V_{IM}, and V_{IL}(4) apply. The CMOS output specification PC cannot be used for input.

Electrical characteristics, Unless Otherwise Specified at Ta = –30 to +70°C, V_{SS} = 0 V, V_{DD} = 3.0 to 5.5 V

Parameter		Symbol	Applicable Pins	Conditions		V_{DD}(V)	Ratings min	Ratings typ	Ratings max	Unit	Notes
Input high level current		I_{IH}(1)	P2, P3 (except for P33/HOLD), P6	V_{IN} = 13.5 V, output Nch Tr off					5.0	µA	1
		I_{IH}(2)	P0, P1, P4, P5, P9, PC, OSC1, RES, P33/HOLD (except for PD, PE, PC2, and PC3)	V_{IN} = V_{DD}, output Nch Tr off					1.0	µA	1
		I_{IH}(3)	PD, PE, PC2, PC3	V_{IN} = V_{DD}, output Nch Tr off					1.0	µA	1
Input low level current		I_{IL}(1)	Inputs other than PD, PE, PC2, and PC3	V_{IN} = V_{SS}, output Nch Tr off			–1.0			µA	2
		I_{IL}(2)	PC2, PC3, PD, PE	V_{IN} = V_{SS}, output Nch Tr off			–1.0			µA	2
Output high level current		V_{OH}(1)	P2, P3 (except for P33/HOLD), P6, P8, P9, PC	I_{OH} = –1 mA			V_{DD} – 1.0			V	3
				I_{OH} = –0.1 mA			V_{DD} – 0.5				
		V_{OH}(2)	P0, P1, P4, P5, P7, PA, PB	I_{OH} = –50 µA			V_{DD} – 1.0			V	4
				I_{OH} = –30 µA			V_{DD} – 0.5				
Output pull-up current		I_{PO}	P0, P1, P4, P5, P7, PA, PB	V_{IN} = V_{SS}		5.5	–1.6			mA	4
Output low level current		V_{OL}(1)	P0, P1, P2, P3, P4, P5, P6, P8, P9, PC (except for P33/HOLD)	I_{OL} = 1.6 mA					0.4	V	5
		V_{OL}(2)	P7, PA, PB	I_{OL} = 8 mA					1.5	V	
Output off leakage current		I_{OFF}(1)	P2, P3, P6, P7, PA	V_{IN} = 13.5 V					5.0	µA	6
		I_{OFF}(2)	(Except for P2, P3, P6, P7, P8, and PA)	V_{IN} = V_{DD}					1.0	µA	6
		I_{OFF}(3)	P8	V_{IN} = V_{SS}			–1.0			µA	7
Comparator offset voltage		V_{OFF}(1)	PD1, PD2, PD3	V_{IN} = V_{SS} to V_{DD} – 1.5 V				±50	±300	mV	
		V_{OFF}(2)	PD0	V_{IN} = 1.5 V to V_{DD}				±50	±300	mV	
Schmitt characteristics	Hysteresis voltage	V_{HIS}	P2, P3, RES, P6, P9 OSC1 (RC, EXT)					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	According to figure 2	4 MHz			4.0		MHz	
	Oscillator stabilization time	t_{CFS}		According to figure 3	4 MHz				10	ms	

Notes: 1. For shared I/O ports, open drain output specifications, this apples when the output Nch Tr is off. Cannot be used as input pins under the CMOS specifications.
2. For shared I/O ports with open drain output specifications, this applies when the output Nch Tr is off. Pull-up output specification ratings are stipulated for the output current IPO. Cannot be used as input pins under the CMOS specifications.
3. Under the CMOS output specifications, this applies when the output Nch Tr is off. (Also applies to P8 open drain.)
4. Under the pull-up output specifications, this applies when the output Nch Tr is off.
5. For P8, this applies for the CMOS output specifications.
6. Under the open drain output specifications, this applies when the output Nch Tr is off.
7. Under the open drain output specifications, this applies when the output Pch Tr is off.

Continued on next page.

Continued from preceding page.

Parameter			Symbol	Applicable Pins		Conditions	V_{DD}(V)	Ratings min	Ratings typ	Ratings max	Unit	Notes
Serial clock	Cycle timer	Input	t_{CKCY}	$\overline{SCK0}$, $\overline{SCK1}$		With the timing of figure 4 and the test load of figure 5		0.9			µs	
		Output						2.0			T_{CYC}	
	Low level and high level pulse width	Input	t_{CKL}					0.4			µs	
		Output	t_{CKH}					1.0			T_{CYC}	
	Rise and fall time	Output	t_{CKR} t_{CKF}							0.1	µs	
Serial input	Data setup time		t_{ICK}	SI0, SI1 SI0, SI1		With the timing of figure 4. Stipulated with respect to the rise of $\overline{SCK0}$ and $\overline{SCK1}$.		0.3			µs	
	Data hold time		t_{CKI}					0.3			µs	
Serial output	Output delay time		t_{CKO}	SO0, SO1		With the timing of figure 4 and the test load of figure 5. Stipulated with respect to the fall of $\overline{SCK0}$ and $\overline{SCK1}$.				0.3	µs	
Pulse input conditions	INT0 high and low level pulse width		t_{I0H} t_{I0L}	INT0	Figure 6	• Conditions such that an INT0 interrupt will be accepted • Conditions such that counter a timer 0 event or pulse width measurement input will be accepted		2			T_{CYC}	
	Interrupt input high and low level pulse width other than INT0		t_{I1H} t_{I1L}	$\overline{INT1}$, $\overline{INT2}$, $\overline{INT3}$, $\overline{INT4}$, INT5		• Conditions such that each interrupt will be accepted		2			T_{CYC}	
	PIN1 high and low level pulse width		t_{PINH} t_{PINL}	PIN1		• Conditions such that a timer 1 event counter input will be accepted		2			T_{CYC}	
	$\overline{RES}$ high and low level pulse width		t_{RSH} t_{RSL}	$\overline{RES}$		• Conditions such that a reset occurs		3			T_{CYC}	
Comparator response time			T_{RS}	PD	Figure 7					20	ms	
Operating mode quiescent current			I_{DD} op	V_{DD}		4 MHz ceramic oscillator			3.0	5.0	mA	1
						4 MHz external clock			3.0	5.0	mA	
Halt mode quiescent current			I_{DDHALT}	V_{DD}		4 MHz ceramic oscillator			1.0	2.0	mA	
						4 MHz external clock			1.0	2.0	mA	
Hold mode quiescent current			I_{DDHOLD}	V_{DD}			1.8 to 5.5		0.01	10	µA	

Note: 1. Reset state.

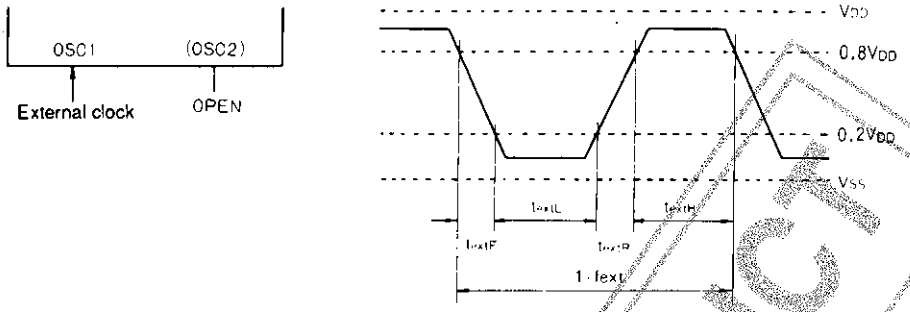


Figure 1: External Clock Input Waveform

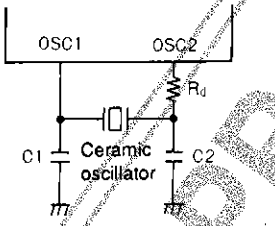


Figure 2: Ceramic Oscillator Circuit

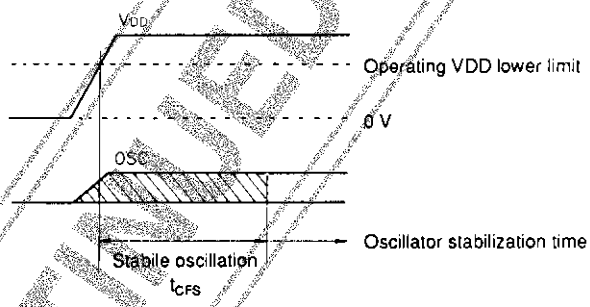


Figure 3: Oscillator Stabilization Time

Table 1: Ceramic Oscillator Guaranteed Constants

Capacitance: External	4 MHz (Murata, Ltd.) CSA4.00MG	C1 = 33 pF ±10%	4 MHz (Kyocera, Ltd.) KBR4.0MS	C1 = 33 pF ±10%
		C1 = 33 pF ±10%		C1 = 33 pF ±10%
		Rd = 0 Ω		Rd = 0 Ω
Capacitance: Internal	4 MHz (Murata, Ltd.) CST4.00MG		4 MHz (Kyocera, Ltd.) KBR4.0MES	

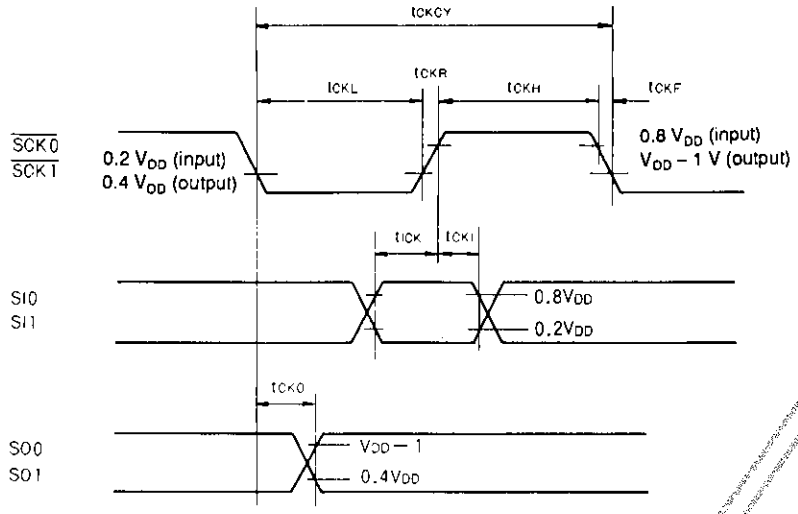


Figure 4: Serial I/O Timing

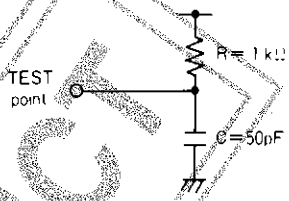


Figure 5: Timing Load

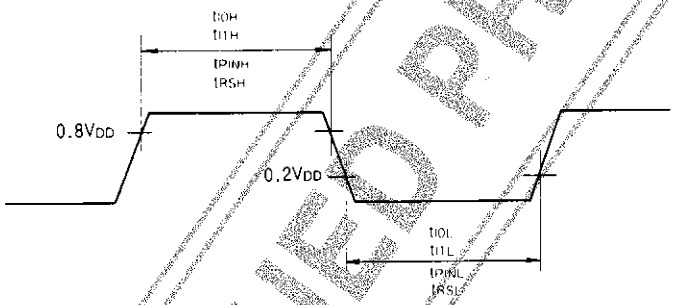


Figure 6: INT0, $\overline{INT1}$, $\overline{INT2}$, $\overline{INT3}$, $\overline{INT4}$, INT5, PIN1, and $\overline{RES}$ Input Timing

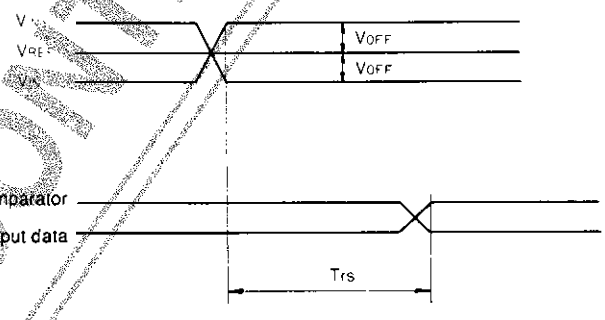
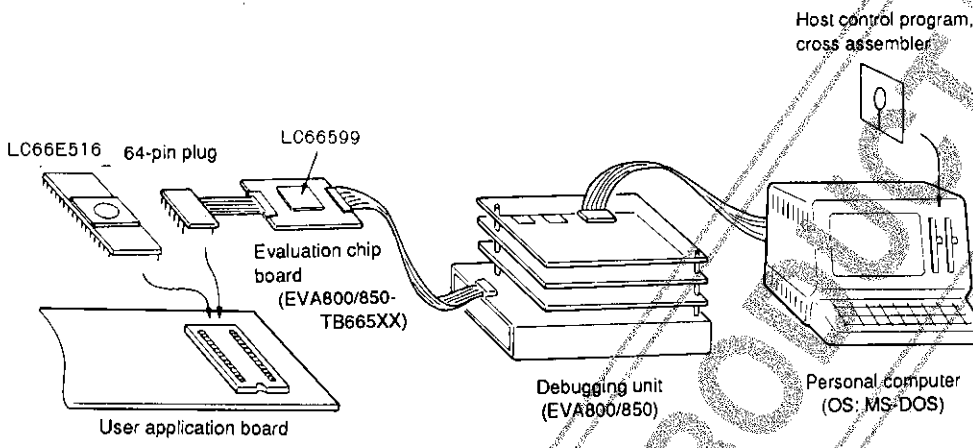


Figure 7: Timing of the Comparator Response Speed Trs

Application Development Tools

Program development for the LC66562B and LC66566B is performed on an IBM compatible PC running MS-DOS. A cross assembler is available for each product. In addition, to assist program development, a program debugging unit (EVA850/800), an evaluation chip board (EVA850/800-TB665XX), and evaluation chip (LC66599) and a built-in EPROM microcontroller (LC66E516) are available.



Structure of the Application Development Tools

1. Program Debugging Unit (EVA850/800)

This is an emulator that provides an EPROM writing function and a function for serial data communication with external equipment, such as the host computer. It supports machine language application development as well as program correction and debugging. It provides program breaking, stepping, and tracing as its main debugging functions. Note that the EVA850/800 monitor ROM uses the MPM665XX.

2. Evaluation chip board (EVA800/850-TB665XX)

The evaluation chip board takes the control signals and ports output by the evaluation chip to a 64-pin connector, and coverts them to a pin layout identical to that of mass produced chips. The evaluation chip board has jumper cables for option and state settings that allow I/O formats and functions identical to the mass produced chip to be setup.

Note however that the hold mode clear timing, and certain electrical characteristics will differ.

Jumpers

Type	OSC		Reset method		User application board power supply technique	
Jumper name	Jumper 1 (J1)		Jumper 2 (J2: $\overline{RES}$)		Jumper 3 (J3: V_{DD})	
Jumper setting and mode	EXT	External oscillator (external clock)	INT (a)	Reset on a RUN instruction from the host computer	ON (a)	Supply V_{DD} to the user application board over the evaluation chip board output
	RC	RC oscillator	EXT (b)	Reset using a reset circuit on the user application board	OFF (b)	Use separate power supplies for the user application board and the evaluation chip power supply
	CF	CF oscillator				

Switches (SW1)

Type	Port 0, 1, and 8 output level on reset						Watchdog timer setting	
Switch name	P0S		P1S		P8S		WDC	
Switch setting and mode	ON	Port 0 high	ON	Port 1 high	ON	Port 8 high	ON	Watchdog timer
	OFF	Port 0 low	OFF	Port 1 low	OFF	Port 8 low	OFF	No watchdog timer

Note: RC0 and RC1 must both be turned on.

Switches SW2 to SW14: Pull-up resistor option setting

- Set these switches to the on position for ports for which the pull-up resistor option is selected, and set them to the open position when the open drain option is specified.
  (SW10 — port 8 is for the pull-down resistor selection.)
- These can be set for individual port pins.

3. Cross Assembler

Cross assembler file name	Corresponding machine type	Programming limitations
LC66S. EXE	LC66562B/566B (LC66E516/P516) (LC66559)	4 SB instruction limitations • LC66562B: SB0, SB1, and SB2 can be used • LC66566B: SB0, SB1, SB2, and SB3 can be used • LC66E516/P516: SB0, SB1, SB2, and SB3 can be used • LC66599: SB0, SB1, SB2, and SB3 can be used

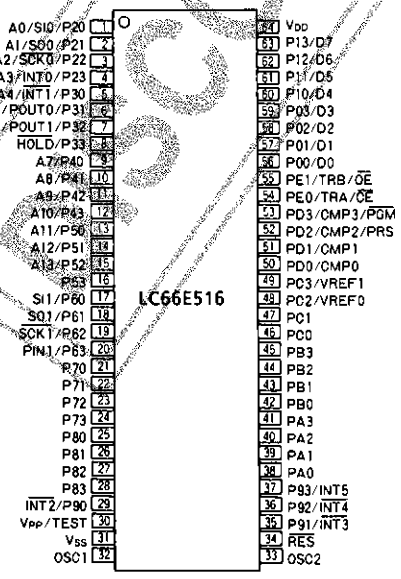
4. Simulation Chip (Refer to the LC66E516 Semiconductor Development News for details.)

The LC66E516 is a built-in EPROM microcontroller. By using a special purpose conversion board (the W66E516DH for DIP, and the W66E516QH for QFP versions) commercial PROM writers can be used to write programs. This allows the operation when mounted in an actual application to be confirmed.

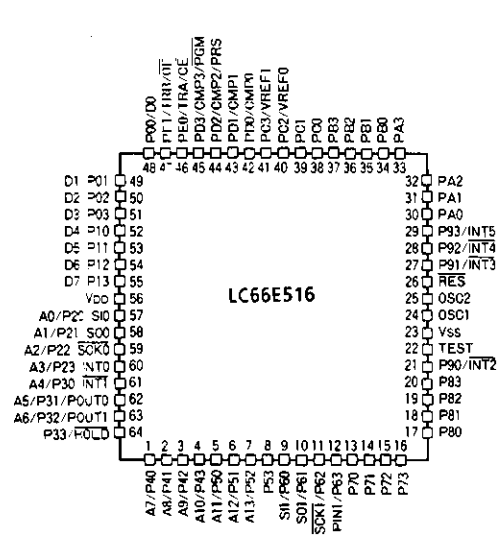
- Form
  The LC66E516 has the same pin layout and functions as the LC66562B and LC66566B.
  Note however that the hold mode clear timing, and certain electrical characteristics will differ.
  The pin assignment is shown in the figures below.
- Options
  The options for the microcontroller to be evaluated (the port 0, 1, and 8 output levels after reset, the watchdog timer, and the port output formats) can be specified by setting internal EPROM data values. This allows evaluation using the same peripheral circuits as the end-product mass production board.

Pin Assignment

Window DIC64S



Window QFC64



LC665XX Series Instruction Set Overview (by function)

Abbreviations used:

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 specified by the DPH, DPL data pointer
M(XY): Data memory specified by the DPX, DPY auxiliary data pointer
M2(HL): Two words of data memory (starting at an even address) specified by the DPH, DPL data pointer
SP: Stack pointer
M2(SP): Two words of data memory specified by the stack pointer
M4(SP): Four words of data memory specified 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 of the PC
PCm: Bits 4 to 7 of the PC
PCl: Bits 0 to 3 of the PC
Fn: User flag, n = 0 to 15
TIMER0: Timer 0
TIMER1: Timer 1
SIO: Serial I/O register
P: Port
P(i4): A port specified by 4 bits of immediate data
INT: Interrupt flag
( ), [ ]: Indicates content
←: Transfer direction, result
∀: Exclusive or
∧: Logical or
∨: Logical and
+: Addition
−: Subtraction
—: Take the one's complement

Instruction group	Mnemonic		Instruction code D_7 D_6 D_5 D_4	D_3 D_2 D_1 D_0	Byte count	Cycle count	Operation	Description of operation	Status changes	Notes
Accumulator manipulation instructions	CLA	Clear AC	1 0 0 0	0 0 0 0	1	1	AC ← 0 (equivalent to LAI 0)	Clears AC	ZF	Also 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)	Adds 6 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 0, 0AH)	Adds 10 to AC	ZF	
	CLC	Clear CF	0 0 0 1	1 1 1 0	1	1	CF ← 0	Clears CF	CF	
	STC	Set CF	0 0 0 1	1 1 1 1	1	1	CF ← 1	Sets CF	CF	
	CMA	Complement AC	0 0 0 1	1 0 0 0	1	1	AC ← $(\overline{AC})$	Takes the one's complement of AC	ZF	
	IA	Increment AC	0 0 0 1	0 1 0 0	1	1	AC ← (AC) + 1	Adds 1 to AC	ZF, CF	
	DA	Decrement AC	0 0 1 0	0 1 0 0	1	1	AC ← (AC) − 1	Subtracts 1 from 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)	Shifts 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)	Shifts AC (including CF) left	CF, ZF	
	TAE	Transfer AC to E	0 1 0 0	0 1 0 1	1	1	E ← (AC)	Transfers the contents of AC to E		
	TEA	Transfer E to AC	0 1 0 0	0 1 1 0	1	1	AC ← (E)	Transfers the contents of E to AC	ZF	
	XAE	Exchange AC with E	0 1 0 0	0 1 0 0	1	1	(AC) ↔ (E)	Exchanges 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	Adds 1 to M(HL)	ZF, CF	
	DM	Decrement M	0 0 1 0	0 0 1 0	1	1	M(HL) ← [M(HL)] − 1	Subtracts 1 from 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	Adds 1 to 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	Subtracts 1 from 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	Sets the bit in M(HL) specified by t_1/t_0		
	RMB t2	Reset M data bit	0 0 1 0	1 1 t_1 t_0	1	1	[M(HL), t2] ← 0	Clears the bit in M(HL) specified by t_1/t_0	ZF	
Arithmetic and comparison instructions	AD	Add M to AC	0 0 0 0	0 1 1 0	1	1	AC ← (AC) + [M(HL)]	Adds (two's complement) AC and the contents of M(HL) and stores 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)]	Adds (two's complement) AC and the contents of M(i8) and stores 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)	Adds (two's complement) the contents of AC, M(HL), and CF and stores 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	Adds (two's complement) AC and the immediate data and stores 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)	Subtracts (two's complement) AC and CF from the contents of M(HL) and stores the result in AC	ZF, CF	Set CF to 0 when there is a borrow, and to 1 when no borrow
	ANDA	And M with AC then store AC	0 0 0 0	0 1 1 1	1	1	AC ← (AC) [M(HL)]	Takes the logical and of AC and the contents of M(HL) and stores 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)]	Takes the logical or of AC and the contents of M(HL) and stores the result in AC	ZF	
	EXL	Exclusive or M with AC then store AC	0 0 0 1	0 1 0 1	1	1	AC ← (AC) ∨ [M(HL)]	Takes the logical exclusive or of AC and the contents of M(HL) and stores 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)]	Takes the logical and of AC and the contents of M(HL) and stores 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)]	Takes the logical or of AC and the contents of M(HL) and stores the result in M(HL)	ZF	

Instruction group	Mnemonic		Instruction code D_7 D_6 D_5 D_4	Instruction code D_3 D_2 D_1 D_0	Byte count	Cycle count	Operation	Description of operation	Status changes	Notes
Arithmetic and comparison instructions	CM	Compare AC with M	0 0 0 1	0 1 1 0	1	1	$\overline{[M(HL)]} + (AC) + 1$	Compares the contents of AC and M(HL), and sets or resets the CF and ZF flags accordingly Magnitude relationship / 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	$\overline{I_3\ I_2\ I_1\ I_0} + (AC) + 1$	Compares the contents of AC with the immediate data I_3 to I_0 and sets or resets the CF and ZF flags accordingly Magnitude relationship / 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) ≠ $I_3\ I_2\ I_1\ I_0$
								Compares DP_L with the immediate data, and either sets ZF if they were equal or clears ZF if different	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]
								Compares the bit specified by the two bits t_1/t_0 in the contents of AC and M(HL), and either sets ZF if they were equal or clears ZF if different	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)	Loads the contents of M2(HL) into AC and E		
	LAI i4	Load AC with immediate data	1 0 0 0	I_3 I_2 I_1 I_0	1	1	AC ← $I_3\ I_2\ I_1\ I_0$	Loads immediate data into AC	ZF	Also 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)]	Loads 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)	Stores 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)	Stores the contents of AC and 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)]	Loads the contents of M(reg) into AC. reg is either HL or XY. reg / t_0 HL / 0 XY / 1	ZF	
	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	Loads the contents of M(reg) into AC. reg is either HL or XY. Then, it increments either DP_L or DP_Y. The relationship between reg and t_0 is the same as that for LA reg.	ZF	ZF is set according to the result of the increment of 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	Loads the contents of M(reg) into AC. reg is either HL or XY. Then, it decrements either DP_L or DP_Y. The relationship between reg and t_0 is the same as that for XA reg.	ZF	ZF is set according to the result of the increment of 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)]	Exchanges the contents of M(reg) and AC. reg is either HL or XY. reg / t_0 HL / 0 XY / 1		

Instruction group	Mnemonic		Instruction code D_7 D_6 D_5 D_4	D_3 D_2 D_1 D_0	Byte count	Cycle count	Operation	Description of operation	Status changes	Notes
Load and store instructions	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)] $DP_L \leftarrow (DP_L) + 1$ Or: $DP_Y \leftarrow (DP_Y) + 1$	Exchanges the contents of M(reg) and AC. reg is either HL or XY. Then, it increments either DP_L or DP_Y. The relationship between reg and t_0 is the same as that for XA reg	ZF	ZF is set according to the result of the increment of 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)] $DP_L \leftarrow (DP_L) - 1$ Or: $DP_Y \leftarrow (DP_Y) - 1$	Exchanges the contents of M(reg) and AC. reg is either HL or XY. Then, it decrements either DP_L or DP_Y. The relationship between reg and t_0 is the same as that for XA reg	ZF	ZF is set according to the result of the increment of 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)]	Exchanges the contents of AC and M(i8)		
	LEAI i8	Load E and 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	$E \leftarrow I_7\ I_6\ I_5\ I_4$ $AC \leftarrow I_3\ I_2\ I_1\ I_0$	Loads the immediate data i8 into E and AC		
	RTBL	Read table data from program ROM	0 1 0 1	1 0 1 0	1	2	E, AC ← [ROM(PCh, E, AC)]	Loads into E and AC the ROM data at the location specified by the PC with the lower 8 bits replaced by the contents of E and 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)]	Outputs to ports 4 and 5 the ROM data at the location specified by the PC with the lower 8 bits replaced by the contents of E and AC		
Data and 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	$DP_H \leftarrow 0$ $DP_L \leftarrow I_3\ I_2\ I_1\ I_0$	Loads 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	$DP_H \leftarrow I_3\ I_2\ I_1\ I_0$	Loads 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	$DP_L \leftarrow I_3\ I_2\ I_1\ I_0$	Loads 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	$DP_H \leftarrow I_7\ I_6\ I_5\ I_4$ $DP_L \leftarrow I_3\ I_2\ I_1\ I_0$	Loads immediate data into DP_H and 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 1 0 I_3 I_2 I_1 I_0	2	2	$DP_X \leftarrow I_7\ I_6\ I_5\ I_4$ $DP_Y \leftarrow I_3\ I_2\ I_1\ I_0$	Loads immediate data into DP_X and DP_Y		
	IL	Increment DP_L	0 0 0 1	0 0 0 1	1	1	$DP_L \leftarrow (DP_L) + 1$	Increments the contents of DP_L	ZF	
	DL	Decrement DP_L	0 0 1 0	0 0 0 1	1	1	$DP_L \leftarrow (DP_L) - 1$	Decrements the contents of DP_L	ZF	
	IY	Increment DP_Y	0 0 0 1	0 0 1 1	1	1	$DP_Y \leftarrow (DP_Y) + 1$	Increments the contents of DP_Y	ZF	
	DY	Decrement DP_Y	0 0 1 0	0 0 1 1	1	1	$DP_Y \leftarrow (DP_Y) - 1$	Decrements 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	$DP_H \leftarrow (AC)$	Transfers the contents of AC into 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	$AC \leftarrow (DP_H)$	Transfers the contents of DP_H into AC	ZF	
	XAH	Exchange AC with DP_H	0 1 0 0	0 0 0 0	1	1	(AC) ↔ (DP_H)	Exchanges 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	$DP_L \leftarrow (AC)$	Transfers the contents of AC into 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	$AC \leftarrow (DP_L)$	Transfers the contents of DP_L into AC	ZF	
	XAL	Exchange AC with DP_L	0 1 0 0	0 0 0 1	1	1	(AC) ↔ (DP_L)	Exchanges 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	$DP_X \leftarrow (AC)$	Transfers the contents of AC into 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	$AC \leftarrow (DP_X)$	Transfers the contents of DP_X into AC	ZF	
	XAX	Exchange AC with DP_X	0 1 0 0	0 0 1 0	1	1	(AC) ↔ (DP_X)	Exchanges 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	$DP_Y \leftarrow (AC)$	Transfers the contents of AC into 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	$AC \leftarrow (DP_Y)$	Transfers the contents of DP_Y into AC	ZF	
	XAY	Exchange AC with DP_Y	0 1 0 0	0 0 1 1	1	1	(AC) ↔ (DP_Y)	Exchanges the contents of AC and DP_Y		

Instruction group	Mnemonic		Instruction code D_7 D_6 D_5 D_4	D_3 D_2 D_1 D_0	Byte count	Cycle count	Operation	Description of operation	Status changes	Notes
Flag manipulation instructions	SFB n4	Set flag bit	0 1 1 1	n_3 n_2 n_1 n_0	1	1	Fn ← 1	Sets the flag bit specified by n4		
	RFB n4	Reset flag bit	0 0 1 1	n_3 n_2 n_1 n_0	1	1	Fn ← 0	Clears the flag bit specified by n4	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, PC12 ← PC13, PC12 PC11 to PC0 ← P_{11} to P_0	Jumps to the location in the current bank specified by the immediate data P12		PC12 becomes ($\overline{PC12}$) when this instruction is executed 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 PC8 ← PC13 to PC8 PC7 to PC4 ← (E) PC3 to PC0 ← (AC)	Jumps to the location specified by the PC with the lower 8 bits replaced by E and 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 PC11 ← 0 PC10 to PC0 ← P10 to P0 M4(SP) ← (CF, ZF, PC13 to PC0) SP ← (SP) − 4	Calls 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 PC6, PC10 ← PC0 PC5 to PC2 ← P3 to P0 M4(SP) ← (CF, ZF, PC12 to PC0) SP ← SP − 4	Calls a subroutine in page 0 of bank 0		
	BANK	Change bank	0 0 0 1	1 0 1 1	1	1		Changes the memory bank and register bank		
	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	Saves the contents of reg in M2(SP) and then decrements SP by 2		
	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)]	Adds 2 to SP and loads the contents of M(SP) into reg. The relationship between i_1/i_0 and reg is identical to 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)]	Returns from a subroutine or interrupt processing routine. The ZF and CF flags 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)]	Returns from a subroutine or interrupt processing routine. The ZF and CF flags 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 PC0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if(AC, t2) = 1	Branches to the location specified by P_7 to P_0 on the current page if the AC bit specified by the immediate data t_1/t_0 is 1		
	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 PC0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if(AC, t2) = 0	Branches to the location specified by P_7 to P_0 on the current page if the AC bit specified by the immediate data t_1/t_0 is 0		
	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 PC0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if[M(HL), t2] = 1	Branches to the location specified by P_7 to P_0 on the current page if the M(HL) bit specified by the immediate data t_1/t_0 is 1		
	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 PC0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if[M(HL), t2] = 0	Branches to the location specified by P_7 to P_0 on the current page if the M(HL) bit specified by the immediate data t_1/t_0 is 0		

reg	i_1	i_0
HL	0	0
XY	0	1
AE	1	0
Illegal	1	1

Instruction group	Mnemonic		Instruction code D_7 D_6 D_5 D_4	Instruction code D_3 D_2 D_1 D_0	Byte count	Cycle count	Operation	Description of operation	Status changes	Notes
Branch instructions	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 PC0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if[P(DP_L), t2] = 1	Branches to the location specified by P_7 to P_0 on the current page if the port (DP_L) bit specified by the immediate data t_1/t_0 is 1		Also valid for internal control registers by use directly following a bank instruction. However, this is limited to registers that allow reads.
	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 PC0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if[P(DP_L), t2] = 0	Branches to the location specified by P_7 to P_0 on the current page if the port (DP_L) bit specified by the immediate data t_1/t_0 is 1		Also valid for internal control registers by use directly following a bank instruction. However, this is limited to registers that allow reads.
	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 PC0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if(CF) = 1	Branches to the location specified by P_7 to P_0 on the current page if CF is 1		
	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 PC0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if(CF) = 0	Branches to the location specified by P_7 to P_0 on the current page if CF is 0		
	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 PC0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if(ZF) = 1	Branches to the location specified by P_7 to P_0 on the current page if ZF is 1		
	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 PC0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if(ZF) = 0	Branches to the location specified by P_7 to P_0 on the current page if ZF is 0		
	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 PC0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if(Fn) = 1	Branches to the location specified by P_7 to P_0 on the current page if, of the 16 flags, the flag specified by n_3 to n_0 is 1		
	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 PC0 ← P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 if(Fn) = 1	Branches to the location specified by P_7 to P_0 on the current page if, of the 16 flags, the flag specified by n_3 to n_0 is 0		
I/O instructions	IP0	Input port 0 to AC	0 0 1 0	0 0 0 0	1	1	AC ← (P0)	Inputs 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)]	Inputs 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)]	Inputs 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)]	Inputs the contents of port 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)]	Inputs 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)	Outputs 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)]	Outputs 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)	Outputs the contents of AC to port 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)	Outputs 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	Sets the port P(DP_L) bit 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	Clears the port P(DP_L) bit specified by the immediate data t_1/t_0	ZF	

Instruction group	Mnemonic		Instruction code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	Byte count	Cycle count	Operation	Description of operation	Status changes	Notes
I/O instructions	ANDPDR i4, p4	And port with immediate data then output	1 1 0 0 0 1 0 1 I_3 I_2 I_1 I_0 P_3 P_2 P_1 P_0	2	2	$P(P_3$ to $P_0) \leftarrow$ $[P(P_3$ to $P_0)] \vee I_3$ to I_0	Outputs to port $P(P_3$ to $P_0)$ the result of taking the logical and of port $P(P_3$ to $P_0)$ and the immediate data I_3 to I_0	ZF	
	ORPDR i4, p4	Or port with immediate data then output	1 1 0 0 0 1 0 0 I_3 I_2 I_1 I_0 P_3 P_2 P_1 P_0	2	2	$P(P_3$ to $P_0) \leftarrow$ $[P(P_3$ to $P_0)] \vee I_3$ to I_0	Outputs to port $P(P_3$ to $P_0)$ the result of taking the logical or of port $P(P_3$ to $P_0)$ and the immediate data I_3 to I_0	ZF	
Timer control instructions	WTTM0	Write timer 0	1 1 0 0 1 0 1 0	1	2	TIMER0 ← [M2(HL)], (AC)	Writes the contents of M2(HL) and 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)	Writes the contents of E and 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)	Reads out the contents of the timer 0 counter into M2(HL) and 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)	Reads out the contents of the timer a counter into E and AC		
	START0	Start timer 0	1 1 0 0 1 1 1 1 1 1 1 0 0 1 1 0	2	2	Start timer 0 counter	Starts the timer 0 counter		
	START1	Start timer 1	1 1 0 0 1 1 1 1 1 1 1 0 0 1 1 1	2	2	Start timer 1 counter	Starts 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	Stops 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	Stops the timer 1 counter		
Interrupt control instructions	MSET	Set interrupt master enable flag	1 1 0 0 1 1 0 1 0 1 0 1 0 0 0 0	2	2	MSE ← 1	Sets the interrupt master enable flag		
	MRESET	Reset interrupt master enable flag	1 1 0 0 1 1 0 1 1 0 0 1 0 0 0 0	2	2	MSE ← 0	Clears the interrupt master enable flag		
	EIH i4	Enable interrupt high	1 1 0 0 1 1 0 1 0 1 0 1 I_3 I_2 I_1 I_0	2	2	EDIH ← (EDIH) ∨ i4	Sets the interrupt enable flags		
	EIL i4	Enable interrupt low	1 1 0 0 1 1 0 1 0 1 0 0 I_3 I_2 I_1 I_0	2	2	EDIL ← (EDIL) ∨ i4	Sets the interrupt enable flags		
	DIH i4	Disable interrupt high	1 1 0 0 1 1 0 1 1 0 0 1 I_3 I_2 I_1 I_0	2	2	EDIH ← (EDIH) ∧ $\overline{i4}$	Clears the interrupt enable flags	ZF	
	DIL i4	Disable interrupt low	1 1 0 0 1 1 0 1 1 0 0 0 I_3 I_2 I_1 I_0	2	2	EDIL ← (EDIL) ∧ $\overline{i4}$	Clears the interrupt enable flags	ZF	
	WTSP	Write SP	1 1 0 0 1 1 1 1 1 1 0 1 1 0 1 0	2	2	SP ← (E), (AC)	Transfer the contents of E and AC into SP		
	RSP	Read SP	1 1 0 0 1 1 1 1 1 1 0 1 1 0 0 1	2	2	E, AC ← (SP)	Transfers the contents of SP into E and AC		
Standby control instructions	HALT	HALT	1 1 0 0 1 1 1 1 1 1 0 1 1 1 1 0	2	2	HALT	Sets halt mode		
	HOLD	HOLD	1 1 0 0 1 1 1 1 1 1 0 1 1 1 1 1	2	2	HOLD	Sets hold mode		
Serial I/O control instructions	STARTS	Start serial I/O	1 1 0 0 1 1 1 1 1 1 1 0 1 1 1 0	2	2	START SIO	Starts SIO operation		
	WTSIO	Write serial I/O	1 1 0 0 1 1 1 1 1 1 1 0 1 1 1 1	2	2	SIO ← (E), (AC)	Writes the contents of E and AC into the SIO register		
	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)	Reads the contents of the SIO register into E and AC		
Other instructions	NOP	No operation	0 0 0 0 0 0 0 0	1	1	No operation	Takes up a single machine cycle without performing an operation		
	SB i2	Select bank	1 1 0 0 1 1 1 1 1 1 0 0 0 0 I_1 I_0	2	2	PC13, 12 ← I_1 I_0	Specifies the memory bank		