HT44500/445A0/445R0 SPECIFICATION

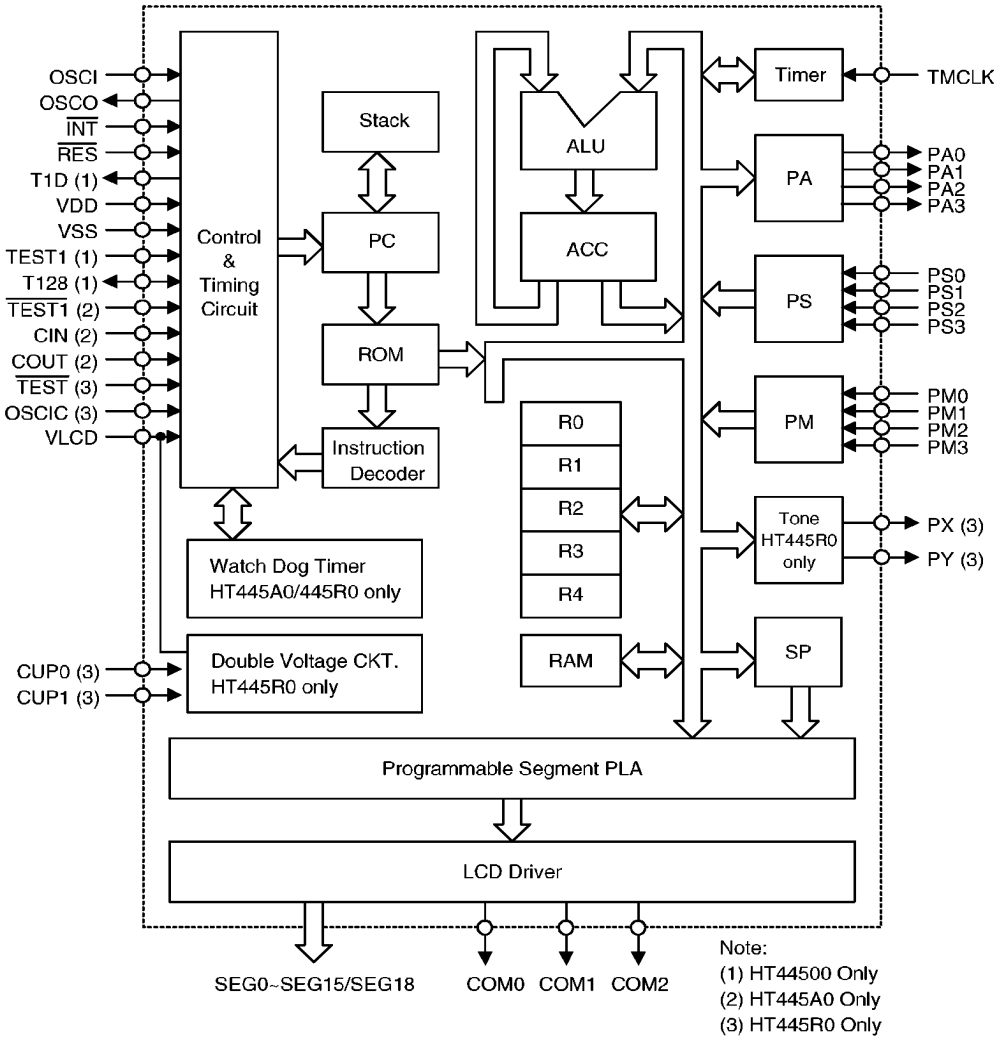
Features

- Operating voltage: 3.0V~5.0V
  1.5V for HT445R0
- Two 4-bit input ports
- 4-bit output port
- 16 × 3 or 19 × 3 segment LCD driver
- Timer input
- 2K × 8 program memory ROM
- Data memory RAM
  64 × 4 size for the HT44500
  128 × 4 size for the HT445A0/445R0
- 5 working registers
- LCD segment PLA
- RC or crystal oscillator system clock
- Internal 128Hz circuit for LCD driver clock
- 8-bit programmable timer with external clock or built-in frequency sources
- External interrupt input
- Internal timer overflow interrupt
- 3.0V LCD application
- 1/3 duty, 1/2 bias LCD application
- Programmable LCD segment PLA, compatible with any LCD panel
- Halt function to reduce power consumption and wake-up feature
- All instructions in 1 or 2 machine cycles
- One-level subroutine nesting
- 8-bit table read instruction

General Description

The HT44500/445A0/445R0 are three processors from Holtek's 4-bit stand alone single chip microcontroller range specifically designed for LCD product applications. The three devices are similar in most ways apart from LCD driver capability, RAM size and some special features. Both the HT445A0 and the HT445R0 possess watch dog timers while the HT445R0 has the additional feature of a voltage doubler circuit giving the advantage of low voltage operation. All devices are ideally suited for multiple LCD low power applications among which are calculators, scales, and hand held LCD products.

Block Diagram



Note:

ACC: Accumulator

PC: Program counter

R0~R4: Working registers

PA: Output port

PS,PM: Input ports

SP: Strobe pointer

SEG0~SEG15 for HT44500

SEG0~SEG18 for HT445A0/445R0

Pad Description HT44500

Pad No.	Pad Name	I/O	Mask Option	Function
1 42 41 40	PS3 PS2 PS1 PS0	I	Pull-high or none	4-bit input port
2	TMCLK	I	10 internal frequency sources and pull-high or none	Input for TIMER clock TIMER can be clocked by external or ten internal frequency sources.
3~6	PM0~PM3	I	Pull-high or none	4-bit input port
7 34 35	TEST1 T128 T1D	I I/O O	—	For test mode only TEST1 is left open in normal operations. T1D provides a 1/4 fosc output.
8~23	SEG15~SEG0	O	—	LCD driver outputs for LCD segments
24	VSS	I	—	Negative power supply, GND
25	VLCD	I	—	LCD bias power supply
26~28	COM0~COM2	O	2 or 3 connections	Outputs for LCD commons
29 30	OSCI OSCO	I O	Crystal or RC	OSCI,OSCO are connected to a resistor (RC) or crystal for an internal system clock
31	$\overline{INT}$	I	—	External interrupt input, pull high, high to low edge triggered
32	VDD	I	—	Positive power supply
33	$\overline{RES}$	I	—	Input to reset an internal LSI Reset is active on low level. With an internal pull high resistor
36~39	PA0~PA3	O	CMOS or NMOS open drain	4-bit output port

Note:

In the output condition T128 provides a 128Hz frequency output

In the input condition a 128Hz frequency is applied on pin T128 for the LCD clock

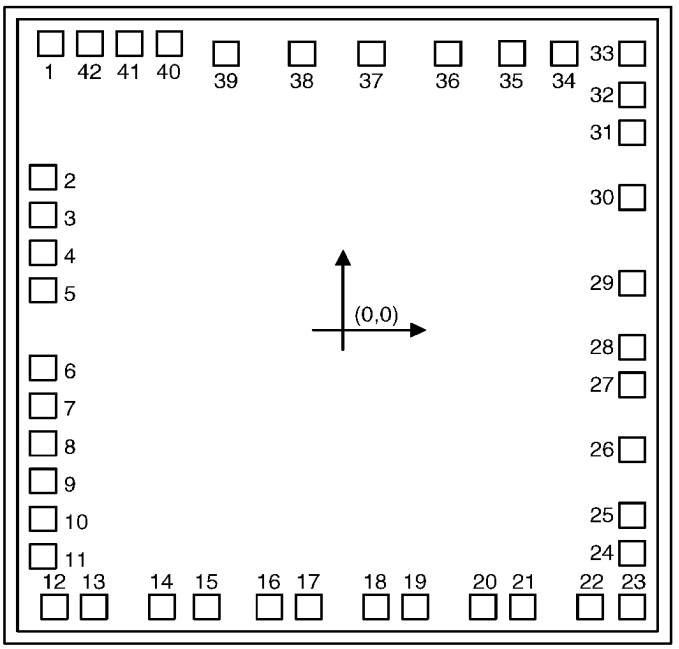
Pad Description HT445A0

Pad No.	Pad Name	I/O	Mask Option	Function
45,1~3	PM0~PM3	I	Pull-high latch or none	4-bit input port
4	$\overline{\text{TEST1}}$	I	—	For test mode only $\overline{\text{TEST1}}$ should be left open for normal operations.
5~23	SEG18~SEG0	O	—	LCD driver outputs for LCD segments
24 27	OSCI OSCO	I O	Crystal or RC	OSCI,OSCO are connected to a resistor (RC) or crystal for an internal system clock.
25 26	CIN COUT	I	Metal bonding	Oscillator built in capacitor
28~30	COM2~COM0	O	2 or 3 connections	Outputs for LCD commons
31	VLCD	I	—	LCD bias power supply
32	VSS	I	—	Negative power supply, GND
33	VDD	I	—	Positive power supply
34	$\overline{\text{RES}}$	I	—	Input to reset an internal LSI Reset is active on low level. A pull-high resistor is built-in.
35	$\overline{\text{INT}}$	I	—	Input (with a pull high resistor) for an external interrupt Activated on a high to low edge trigger transition
36~39	PA0~PA3	O	CMOS or NMOS open drain	4-bit output port
40	TMCLK	I	Internal frequency source and pull-high or none	Input for TIMER clock TIMER can be clocked by external clock or internal frequency sources (IFS). $\text{IFS} = \frac{\text{system clock}}{2^{n+m}}$ where n=0~14, m=0 or 2
41~44	PS0~PS3	I	Pull-high latch or none	4-bit input port

Pad Description HT445R0

Pad No.	Pad Name	I/O	Mask Option	Function
2~5	PM0~PM3	I	Pull-high latch or none	4-bit input port
6	$\overline{\text{TEST}}$	I	—	For test mode only $\overline{\text{TEST}}$ pin should be left open for normal operation.
7~25	SEG18~SEG0	O	—	LCD driver outputs for LCD segments
26 28	OSCI OSCO	I O	Crystal or RC	OSCI,OSCO are connected to a resistor (RC) or crystal for an internal system clock.
27	OSCIC	I	Metal bonding	Oscillator built in capacitor
29~31	COM2~COM0	O	2 or 3 connections	Outputs for LCD commons
32	VLCD	I	—	LCD bias power supply
33~34	CUP0,CUP1	I	—	Voltage doubler capacitor
35~36	PY,PX	O	14 internal frequency sources	Single tone outputs
37	VSS	I	—	Negative power supply, GND
38	VDD	I	—	Positive power supply
39	$\overline{\text{RES}}$	I	—	Input to reset an internal LSI Reset is active on low level. With an internal pull-high resistor.
40	$\overline{\text{INT}}$	I	—	Input (with a pull high resistor) for an external interrupt Activated on a high to low edge trigger transition
41~44	PA0~PA3	O	CMOS or NMOS open drain	4 bit output port
45	TMCLK	I	Internal frequency source and pull-high or none	Input for TIMER clock TIMER can be clocked by external clock or internal frequency sources. $\text{IFS} = \frac{\text{system clock}}{2^{n+m}}$ where n=0~14, m=0 or 2
46~48,1	PS0~PS3	I	Pull-high latch or none	4 bit input port

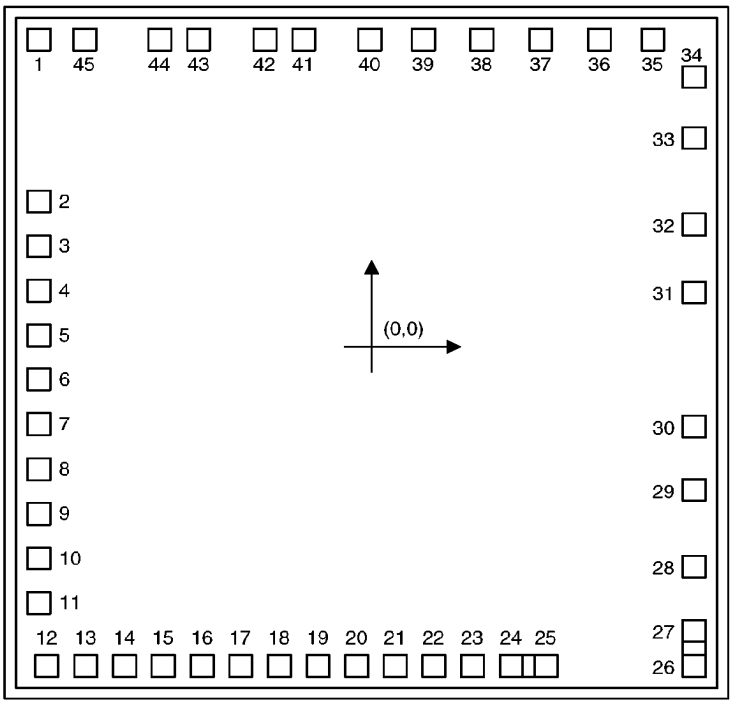
Pad Position HT44500



Chip size: 3070 × 3190 μm

* The IC substrate should be connected to VSS in the PCB layout artwork.

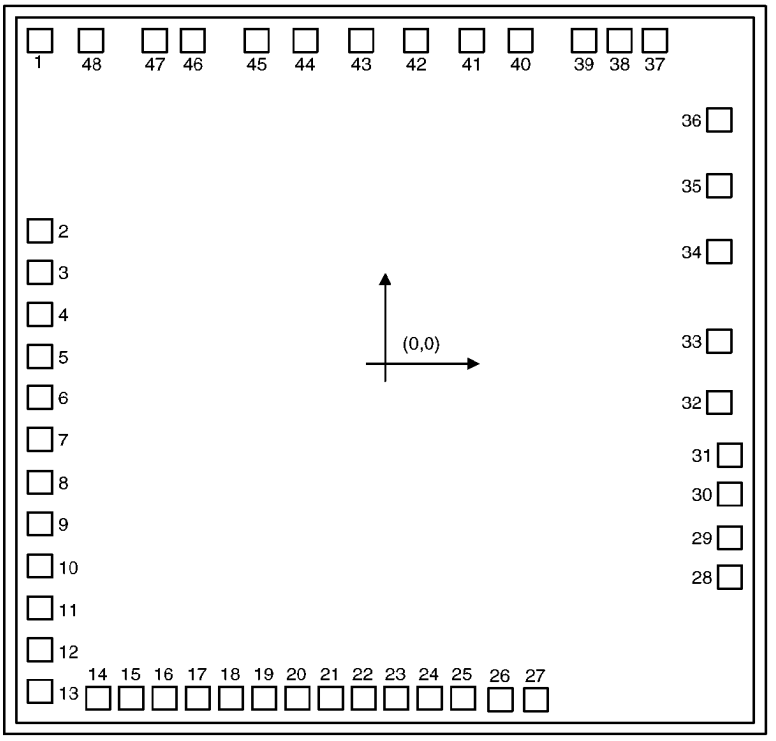
Pad Position HT445A0



Chip size: 3450 × 3030 μm

* The IC substrate should be connected to VSS in the PCB layout artwork.

Pad Position HT445R0



Chip size: 3490 × 3030 μm

* The IC substrate should be connected to VSS in the PCB layout artwork.

Pad Coordinates HT44500

Unit: μm

Pad No.	Pad Name	X	Y	Pad No.	Pad Name	X	Y
1	PS3	–1369.75	1409	22	SEG1	1159.25	–1359
2@	TMCLK	–1404.75	751	23	SEG0	1354.75	–1359
3*	PM0	–1404.75	566	24*	VSS	1354.75	–1096
4*	PM1	–1404.75	381	25*	VLCD	1354.75	–911
5	PM2	–1404.75	196	26*	COM0	1354.75	–587
6	PM3	–1404.75	–187	27*	COM1	1354.75	–270
7*	TEST1	–1404.75	–372	28*	COM2	1354.75	–85
8*	SEG15	–1404.75	–557	29*	OSCI	1354.75	232
9*	SEG14	–1404.75	–742	30*	OSCO	1354.75	654
10*	SEG13	–1404.75	–927	31*	$\overline{\text{INT}}$	1354.75	971
11*	SEG12	–1404.75	–1112	32*	VDD	1354.75	1156
12*	SEG11	–1350.75	–1359	33*	$\overline{\text{RES}}$	1354.75	1359
13*	SEG10	–1165.75	–1359	34*	T128	1037.75	1359
14*	SEG9	–848.75	–1359	35*	T1D	792.25	1360
15*	SEG8	–637.75	–1359	36*	PA0	491.75	1359
16*	SEG7	–346.75	–1359	37*	PA1	135.25	1359
17*	SEG6	–161.75	–1359	38*	PA2	–191.25	1359
18*	SEG5	155.25	–1359	39*	PA3	–547.75	1359
19*	SEG4	340.25	–1359	40	PS0	–814.75	1409
20*	SEG3	657.25	–1359	41	PS1	–999.75	1409
21	SEG2	842.25	–1359	42	PS2	–1184.75	1409

* These pins must be bonded out for functional testing.

@ If the internal source is configured, TMCLK is not bonded out.

Pad Coordinates HT445A0

Unit: μm

Pad No.	Pad Name	X	Y	Pad No.	Pad Name	X	Y
1*	PM1	–1592.5	1274.75	24*	OSCI	669	–1324.75
2	PM2	–1592.5	602.75	25	CIN	835.5	–1324.75
3	PM3	–1592.5	417.75	26	COUT	1542.5	–1324.75
4*	$\overline{\text{TEST1}}$	–1592.5	232.75	27*	OSCO	1542.5	–1178.75
5*	SEG18	–1592.5	47.75	28*	COM2	1542.5	–912.75
6*	SEG17	–1592.5	–137.25	29*	COM1	1542.5	–595.75
7*	SEG16	–1592.5	–322.75	30*	COM0	1542.5	–332.75
8*	SEG15	–1592.5	–509.75	31*	VLCD	1542.5	226.75
9*	SEG14	–1592.5	–694.75	32*	VSS	1542.5	508.75
10*	SEG13	–1592.5	–879.75	33*	VDD	1542.5	865.75
11*	SEG12	–1592.5	–1064.75	34*	$\overline{\text{RES}}$	1542.5	1119.25
12*	SEG11	–1552	–1324.75	35*	$\overline{\text{INT}}$	1344.5	1274.75
13*	SEG10	–1367	–1324.75	36*	PA0	1090	1274.75
14*	SEG9	–1182	–1324.75	37*	PA1	809	1274.75
15*	SEG8	–997.05	–1324.75	38*	PA2	528	1274.75
16*	SEG7	–812	–1324.75	39*	PA3	247	1274.75
17*	SEG6	–627	–1324.75	40@	TMCLK	–9.5	1274.75
18*	SEG5	–442	–1324.75	41	PS0	–326.5	1274.75
19*	SEG4	–257	–1324.75	42	PS1	–511.5	1274.75
20	SEG3	–72	–1324.75	43	PS2	–828.5	1274.75
21	SEG2	113	–1324.75	44	PS3	–1013.5	1274.75
22	SEG1	298	–1324.75	45*	PM0	–1330.5	1274.75
23	SEG0	483	–1324.75				

* These pins must be bonded out for functional testing.

@ If the internal source is configured, TMCLK is not bonded out.

Pad Coordinates HT445R0

Unit: μm

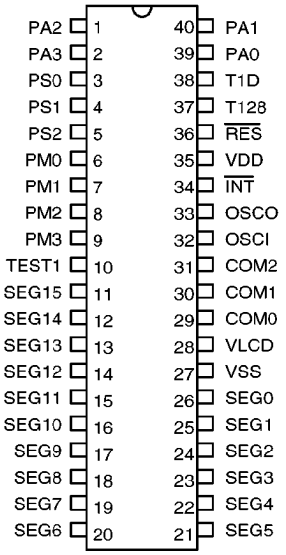
Pad No.	Pad Name	X	Y	Pad No.	Pad Name	X	Y
1	PS3	–1610.75	1274.25	25	SEG0	364.25	–1320.25
2*	PM0	–1610.75	525.25	26*	OSCI	540.25	–1325.25
3*	PM1	–1610.75	360.25	27	OSCIC	705.25	–1325.25
4	PM2	–1610.75	195.25	28*	OSCO	1610.75	–842.75
5	PM3	–1610.75	30.25	29	COM2	1610.75	–687.75
6*	$\overline{\text{TEST}}$	–1610.75	–134.75	30	COM1	1610.75	–515.75
7*	SEG18	–1610.75	–299.75	31	COM0	1610.75	–360.75
8*	SEG17	–1610.75	–468.75	32*	VLCD	1560.75	–153.25
9*	SEG16	–1610.75	–633.75	33*	CUP0	1560.75	89.75
10*	SEG15	–1610.75	–798.75	34*	CUP1	1560.75	441.25
11*	SEG14	–1610.75	–963.75	35	PY	1560.75	701.25
12*	SEG13	–1610.75	–1128.75	36	PX	1560.75	961.25
13*	SEG12	–1610.75	–1293.75	37*	VSS	1260.25	1274.25
14*	SEG11	–1340.75	–1320.25	38*	VDD	1095.25	1274.25
15*	SEG10	–1185.75	–1320.25	39*	$\overline{\text{RES}}$	930.25	1274.25
16*	SEG9	–1030.75	–1320.25	40	$\overline{\text{INT}}$	633.25	1274.25
17*	SEG8	–875.75	–1320.25	41	PA0	407.75	1274.25
18*	SEG7	–720.75	–1320.25	42	PA1	146.75	1274.25
19*	SEG6	–565.75	–1320.25	43	PA2	–113.25	1274.25
20*	SEG5	–410.75	–1320.25	44	PA3	–374.25	1274.25
21*	SEG4	–255.75	–1320.25	45@	TMCLK	–600.25	1274.25
22	SEG3	–100.75	–1320.25	46	PS0	–897.25	1274.25
23	SEG2	54.25	–1320.25	47	PS1	–1075.25	1274.25
24	SEG1	209.25	–1320.25	48	PS2	–1372.25	1274.25

*: These pins must be bonded out for function testing.
@: If the internal source is configured, TMCLK is not bonded out in the package.

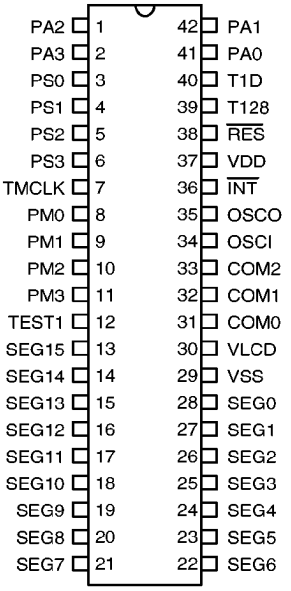
Package & Pin Assignment

HT44500

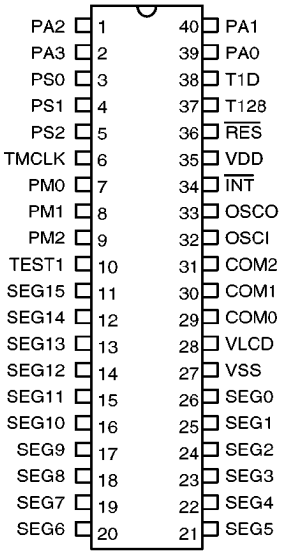
40 Pin DIP-A package



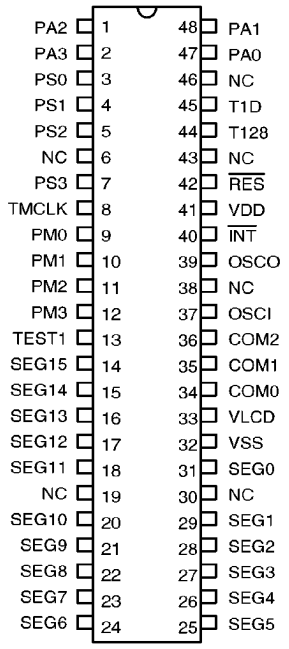
42 Pin DIP package



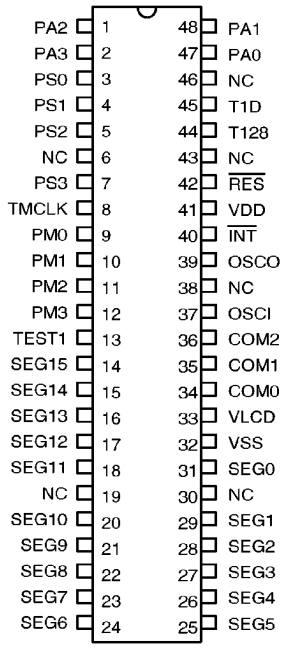
40 Pin DIP-B package



48 Pin DIP package

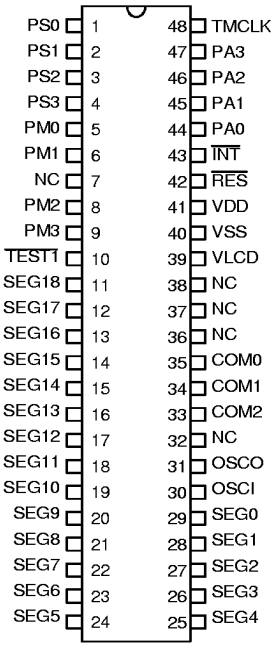


48 Pin SSOP-A package

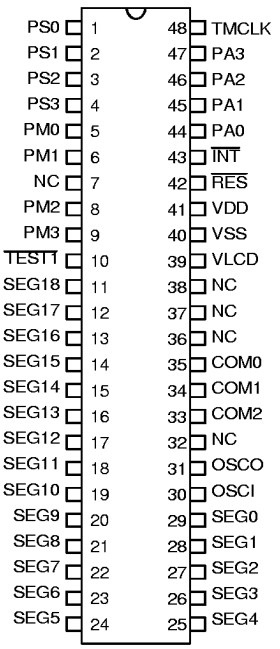


HT445A0

48 Pin DIP package

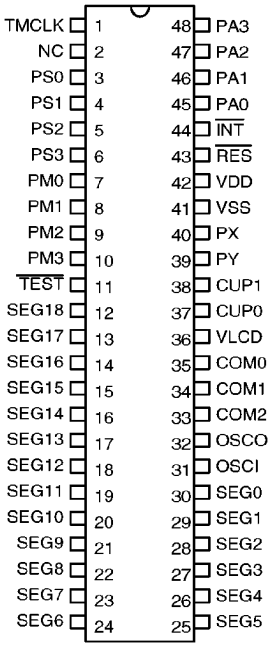


48 Pin SSOP-A package

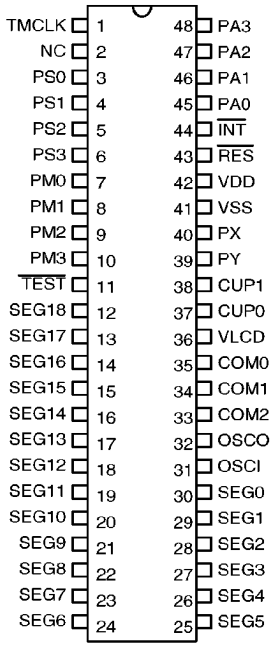


HT445R0

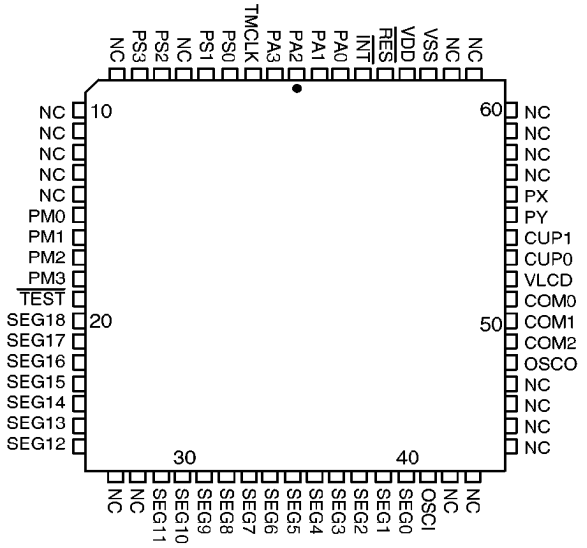
48 Pin DIP package



48 Pin SSOP-A package



68 Pin PLCC package



Absolute Maximum Ratings

Parameter	Symbol	Minimum	Maximum	Unit
Supply Voltage	V_{DD}	–0.3	6	V
Input Voltage	V_I	V_{SS}–0.3	V_{DD}+0.3	V
Storage Temperature	T_{STG}	–50	125	°C
Operating Temperature	T_{OP}	0	70	°C

A.C. Characteristics HT44500/445A0

(Ta=25°C)

Symbol	Parameter	Test Condition		Min.	Typ.	Max.	Unit
		V_{DD}	Condition				
f_{SYS}	System clock	3V	RC oscillator	32	—	800	KHz
		5V		32	—	2000	KHz
		3V	Crystal oscillator	32	—	2000	KHz
		5V		32	—	4000	KHz
f_{LCD}	LCD clock	3V	—	—	128*	—	Hz
		5V		—	128*	—	Hz
t_{COM}	LCD common period	—	—	—	$(1/f_{LCD})\times 3$	—	s
t_{CY}	Cycle time	—	—	—	$(1/f_{SYS})\times 4$	—	ms
f_{TIMER}	Timer I/P frequency	3V	—	0	—	400	KHz
		5V		0	—	800	KHz
t_{RES}	Reset pulse width	—	—	5	—	—	ms
t_{INT}	Interrupt pulse width	—	—	1	—	—	μs

* In general, f_{LCD} is selected and optimized by Holtek according to f_{SYS} and the operating voltage.

A.C. Characteristics HT445R0

(Ta=25°C)

Symbol	Parameter	Test Condition		Min.	Typ.	Max.	Unit
		V_{DD}	Condition				
f_{SYS}	System clock	1.5V	RC oscillator	—	32	50	KHz
			Crystal oscillator	—	32	50	KHz
f_{LCD}	LCD clock	1.5V	—	—	128*	—	Hz
t_{COM}	LCD common period	1.5V	—	—	$(1/f_{LCD})\times3$	—	s
t_{CY}	Cycle time	1.5V	—	—	$(1/f_{SYS})\times4$	—	ms
f_{TIMER}	Timer I/P frequency	1.5V	—	0	—	50	KHz
t_{RES}	Reset pulse width	1.5V	—	5	—	—	ms
t_{INT}	Interrupt pulse width	1.5V	—	1	—	—	μs

* In general, f_{LCD} is selected and optimized by Holtek according to f_{SYS} and the operating voltage.

D.C. Characteristics HT44500

(Ta=25°C)

Symbol	Parameter	Test Condition		Min.	Typ.	Max.	Unit
		V_{DD}	Condition				
V_{DD}	Operating voltage	—	—	2.4	—	5.5	V
I_{DD1}	Operating current	3V	No load, f_{SYS}=1MHz	—	360	800	μA
		5V		—	700	1500	μA
I_{DD2}	Operating current	3V	No load, crystal f_{SYS}=32KHz	—	10	50	μA
		5V		—	30	100	μA
I_{STB}	Stand-by current	3V	No load, HALT mode	—	—	1	μA
		5V		—	—	2	μA
V_{IL}	Input low voltage	3V	—	0	—	$0.2V_{DD}$	V
		5V	—	0	—	$0.2V_{DD}$	V
V_{IH}	Input high voltage	3V	—	$0.8V_{DD}$	—	3	V
		5V	—	$0.8V_{DD}$	—	5	V
I_{OL1}	Port A output sink current	3V	V_{DD}=3V, V_{OL}=0.3V	1.5	3.4	—	mA
		5V	V_{DD}=5V, V_{OL}=0.5V	4	8	—	mA
I_{OH1}	Port A output source current	3V	V_{DD}=3V, V_{OH}=2.7V	–0.7	–1.5	—	mA
		5V	V_{DD}=5V, V_{OH}=4.5V	–2	–3.5	—	mA
I_{OL2}	Segment 0~2 output sink current	3V	V_{LCD}=3V, V_{OL}=0.3V	150	300	—	μA
		5V	V_{LCD}=5V, V_{OL}=0.5V	400	700	—	μA
I_{OH2}	Segment 0~2 output source current	3V	V_{LCD}=3V, V_{OH}=2.7V	–80	–150	—	μA
		5V	V_{LCD}=5V, V_{OH}=4.5V	–150	–400	—	μA
I_{OL3}	Segment 3~15 output sink current	3V	V_{LCD}=3V, V_{OL}=0.3V	80	140	—	μA
		5V	V_{LCD}=5V, V_{OL}=0.5V	150	300	—	μA
I_{OH3}	Segment 3~15 output source current	3V	V_{LCD}=3V, V_{OH}=2.7V	–40	–80	—	μA
		5V	V_{LCD}=5V, V_{OH}=4.5V	–80	–180	—	μA
R_{PH}	Pull-high resistance	3V	PS, PM, $\overline{INT}$, $\overline{RES}$, TMCLK	30	—	300	KΩ
		5V					
V_{LCD}	LCD supply voltage	—	—	2.5	3	3.5	V
I_{LCD}	LCD supply current	3V	V_{LCD}=3V, all segments on	—	—	120	μA

D.C. Characteristics HT445A0

(Ta=25°C)

Symbol	Parameter	Test Condition		Min.	Typ.	Max.	Unit
		V_{DD}	Condition				
V_{DD}	Operating voltage	—	—	2.4	—	5.5	V
I_{DD1}	Operating current	3V	No load, f_{SYS}=1MHz	—	300	1000	μA
		5V		—	500	2000	μA
I_{DD2}	Operating current	3V	No load, crystal f_{SYS}=32KHz	—	10	50	μA
		5V		—	30	100	μA
I_{STB}	Stand-by current	3V	No load, HALT mode	—	—	1	μA
		5V		—	—	2	μA
V_{IL}	Input low voltage	3V	—	0	—	$0.2V_{DD}$	V
		5V	—	0	—	$0.2V_{DD}$	V
V_{IH}	Input high voltage	3V	—	$0.8V_{DD}$	—	3	V
		5V	—	$0.8V_{DD}$	—	5	V
I_{OL1}	Port A output sink current	3V	V_{DD}=3V, V_{OL}=0.3V	1.2	2.6	—	mA
		5V	V_{DD}=5V, V_{OL}=0.5V	3.6	6.5	—	mA
I_{OH1}	Port A output source current	3V	V_{DD}=3V, V_{OH}=2.7V	–0.4	–1	—	mA
		5V	V_{DD}=5V, V_{OH}=4.5V	–0.9	–2	—	mA
I_{OL2}	Segment 0~15 output sink current	3V	V_{LCD}=3V, V_{OL}=0.3V	50	120	—	μA
		5V	V_{LCD}=5V, V_{OL}=0.5V	130	260	—	μA
I_{OH2}	Segment 0~15 output source current	3V	V_{LCD}=3V, V_{OH}=2.7V	–30	–60	—	μA
		5V	V_{LCD}=5V, V_{OH}=4.5V	–60	–160	—	μA
R_{PH1}	Pull-high resistance	3V	$\overline{RES}$	150	—	650	KΩ
		5V		50	—	500	KΩ
		3V	PS, PM	30	—	300	KΩ
		5V					
R_{PH2}	Pull-high resistance	3V	$\overline{INT}$, TMCLK	40	90	200	KΩ
		5V		20	50	100	KΩ
V_{LCD}	LCD supply voltage	—	—	2.5	3	3.5	V
I_{LCD}	LCD supply current	3V	V_{LCD}=3V, all segments on	—	—	120	μA

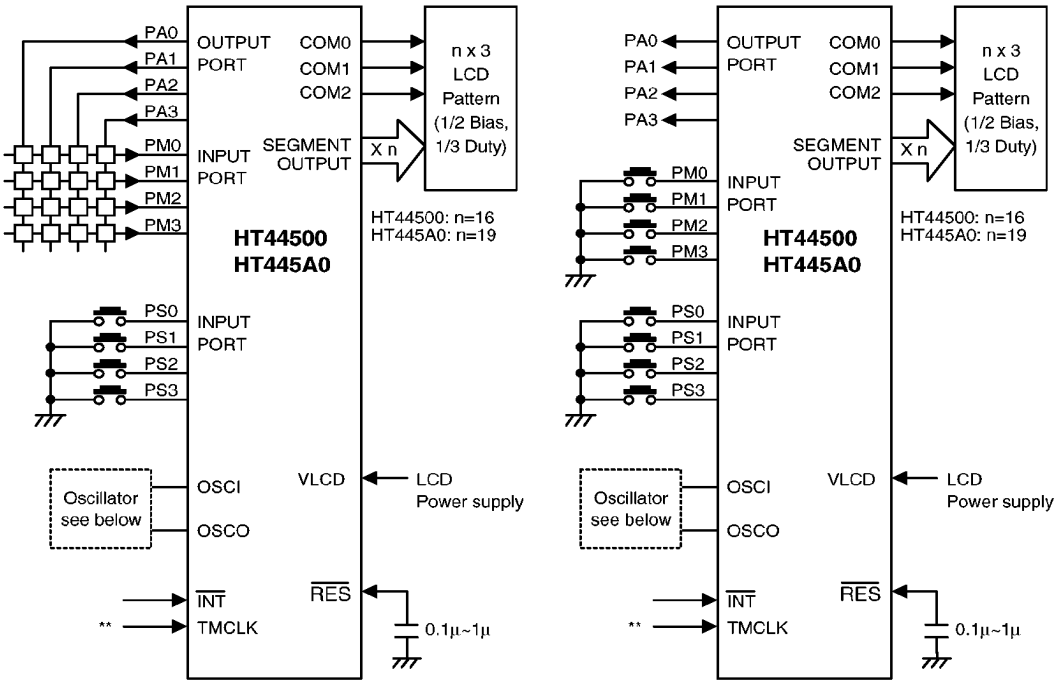
D.C. Characteristics HT445R0

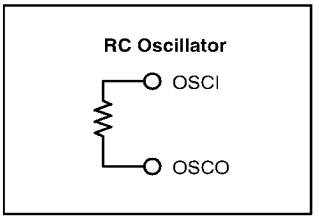
(Ta=25°C)

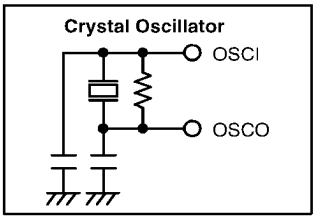
Symbol	Parameter	Test Condition		Min.	Typ.	Max.	Unit
		V_{DD}	Condition				
V_{DD}	Operating voltage	1.5V	—	1.2	1.5	1.7	V
I_{DD}	Operating current	1.5V	No load, f_{SYS}=32KHz	—	5	10	μA
I_{STB}	Stand-by current	1.5V	No load, HALT mode	—	—	0.5	μA
V_{IL}	Input low voltage	1.5V	—	0	—	$0.2V_{DD}$	V
V_{IH}	Input high voltage	1.5V	—	$0.8V_{DD}$	—	V_{DD}	V
I_{OL1}	Port A output sink current	1.5V	V_{DD}=1.5V, V_{OL}=0.15V	500	700	—	μA
I_{OH1}	Port A output source current	1.5V	V_{DD}=1.5V, V_{OH}=1.35V	–160	–200	—	μA
I_{OL2}	Segment 0~18 output sink current	1.5V	V_{LCD}=3V, V_{OL}=0.3V	80	130	—	μA
I_{OH2}	Segment 0~18 output source current	1.5V	V_{LCD}=3V, V_{OH}=2.7V	–10	–65	—	μA
I_{OL3}	PX, PY output sink current	1.5V	V_{OL}=0.15V	500	800	—	μA
I_{OH3}	PX, PY output source current	1.5V	V_{OH}=1.35V	–160	300	—	μA
R_{PH1}	Pull-high resistance	1.5V	PS, PM, $\overline{INT}$	100	—	500	KΩ
R_{PH2}	Pull-high resistance	1.5V	$\overline{RES}$	30	—	300	KΩ
R_{PH3}	Pull-high resistance	1.5V	TMCLK	200	—	600	KΩ
V_{LCD}	LCD supply voltage	1.5V	VLCD	2.4	3	3.4	V

Application Diagram

HT44500/445A0







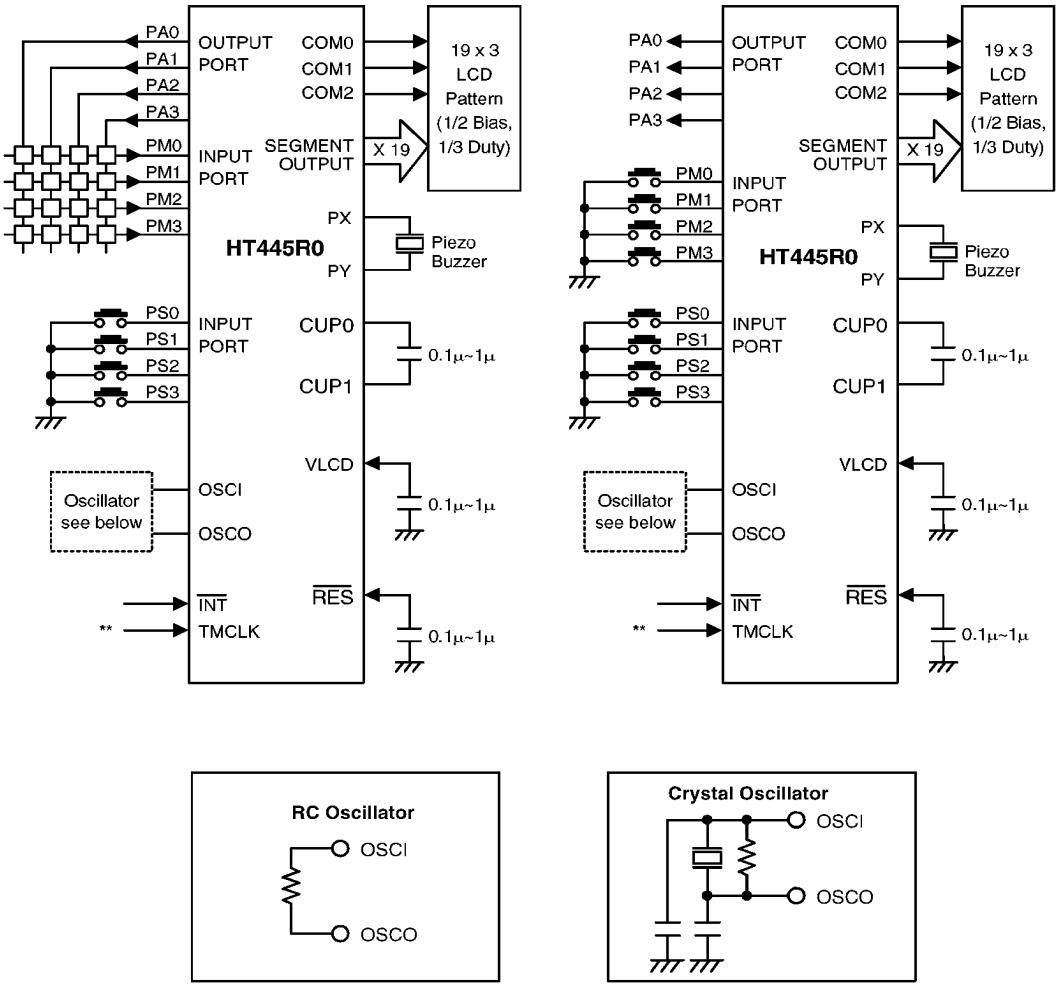
Note:

The RC oscillator resistor value depends upon the required oscillator frequency. The range is from 5K to 700K

The timer clock input TMCLK may come from an external clock or from an internal frequency source

Application Diagram

HT445R0



Note:

The RC oscillator resistor value depends upon the required oscillator frequency. The range is from 5K to 700K

The timer clock input TMCLK may come from an external clock or from an internal frequency source

SYSTEM ARCHITECTURE

Program Counter - PC

This counter addresses the program ROM and is arranged as an 11-bit binary counter from PC0 to PC10 whose contents specify a maximum of 2048 addresses. The program counter increments by 1 or 2 with each execution of an instruction.

When executing the jump instruction (JMP, JNZ, JC, JTMR, etc.), a subroutine call, an initial reset, an internal interrupt, an external interrupt or returning from a subroutine, the program counter (PC) is loaded with the data corresponding to each instruction.

For the JMP instruction, branch instructions and subroutine call instructions, the address space is capable of directly specifying 2048 addresses.

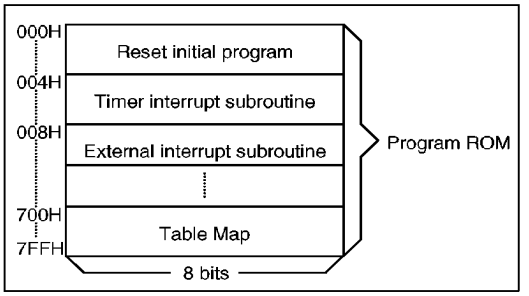
Note: P10~P0: Instruction code bits
S10~S0: Stack register bits

Program Memory - ROM

The program memory is used to store executable program instructions. It is organized into 2048×8 bits and is addressed by the program counter (PC).

There are four special locations in the program memory.

Location 0



Program Memory

Activating the $\overline{RES}$ pin of the processor causes the first instruction to be fetched from location 0.

Location 4
Contains the timer interrupt resulting from a TIMER overflow. If the interrupts are enabled, it causes the program to jump to this subroutine.

Location 8
Activating the $\overline{INT}$ input pin of the processor with the interrupts enabled causes the program to jump to this location.

Location 700H to 7FFH
The last 256 bytes of program memory addressed from 700H to 7FFH can also be used as a look–up table. Instructions such as READ R4A and READ [SP] are available for reading the table and transferring the table data to the ACC and R4 or to the LCD latch.

Therefore the first instruction to be executed after initialization is stored in location 0. The instruction word of a timer service routine is

Mode	Program Counter										
	PC10	PC9	PC8	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
Initial reset	0	0	0	0	0	0	0	0	0	0	0
Internal interrupt	0	0	0	0	0	0	0	0	1	0	0
External interrupt	0	0	0	0	0	0	0	1	0	0	0
Jump, Branch instruction	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0
Subroutine call	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0
Return from subroutine	S10	S9	S8	S7	S6	S5	S4	S3	S2	S1	S0

Program Counter

stored in location 4, and the first instruction of an external interrupt service subroutine is stored in location 8 and the lookup table is in location 700H to 7FFH.

Stack Register - STACK

The stack register is a group of registers used to save the contents of the program counter (PC) and is arranged in 11 bits×1 level. Each subroutine call or interrupt causes the contents of the PC to be stored onto the stack register. At the end of a subroutine or an interrupt (indicated by a return instruction RET), the contents of the stack register are returned to the PC.

Working Registers - R0,R1,R2,R3,R4

There are 5 working registers (R0,R1,R2,R3, R4) usually used to store the frequently accessed intermediate results. Using the instructions INC Rn and DEC Rn the working registers can increment (+1) or decrement (–1). The JNZ Rn (n=0,1,4) instruction makes efficient use of the working registers as a program loop counter. Also the register pairs R0,R1 and R2,R3 are used as a data memory pointer when the memory transfer instruction is executed.

Data Memory - RAM

The static data memory is arranged in a 64×4 bit format for the HT44500 and in a 128×4 bit format for the HT445A0/445R0. It is used to store data. All of the data memory locations are indirectly addressable through the register pair R1,R0 or R3,R2; for example MOV A,[R3R2] or MOV [R3R2],A. The relationship between the data pointer RAM locations is shown in the table.

Accumulator - ACC

The accumulator is the most important data register in the processor. It is one of the sources of input to the ALU and the destination of the results of the operations performed in the ALU. Data to and from the I/O ports and memory also passes through the accumulator.

Arithmetic and Logic Unit - ALU

This circuit performs the following arithmetic and logical operations ...

HT44500 Data Pointer		RAM Location
R1 (R3)	R0 (R2)	
0000	0000	00H
0000	0001	01H
:	:	:
:	:	:
:	:	:
0001	0000	10H
:	:	:
0011	1111	3FH

HT44500 RAM Pointer

HT445A0/445R0 Data Pointer		RAM Location
R1 (R3)	R0 (R2)	
0000	0000	00H
0000	0001	01H
:	:	:
:	:	:
:	:	:
0001	0000	10H
:	:	:
0111	1111	7FH

HT445A0/445R0 RAM Pointer

- Add with or without carry
- Subtract with or without carry
- AND, OR, Exclusive-OR
- Rotate right, left through carry
- BCD decimal adjust for addition
- Increment, decrement
- Data transfers
- Branch decisions

The ALU not only outputs the results of data operations, but also sets the status of the carry flag (CF) in some instructions.

Timer Counter

The HT44500/445A0/445R0 all contain a programmable 8-bit count-up counter which can be

used to count external events or as a clock to generate an accurate time base.

If the 8-bit timer clock is supplied by an external source from pin TMCLK then synchronization problems may occur when reading the data from the timer. It is therefore suggested that the timer is stopped before retrieving the data.

The Timer/Counter may be set and read with software instructions and stopped by a hardware reset or a TIMER OFF instruction. To restart the timer, load the counter with the value XXH and then issue a TIMER ON instruction. Note that XX is the desired start count immediate value of the 8 bits. Once the Timer/Counter is started it increments to a maximum count of FFH and then overflows to zero (00H). It then continues to count until stopped by a TIMER OFF instruction or a reset. The 8-bit counter will increment on the rising edge of the clock whether it is internally or externally generated.

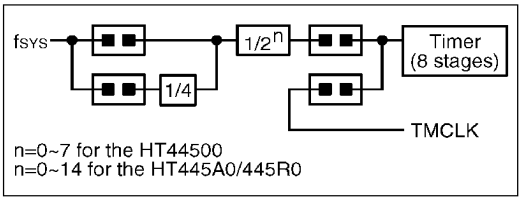
The increment from the maximum count of FFH to a zero (00H) triggers a timer flag TMFG and an internal interrupt request. The interrupt may be enabled or disabled by executing the EI and DI instructions. If the interrupt is enabled the timer overflow will cause a subroutine call to location 4. The state of the timer flag is also testable with the conditional jump instruction JTMR. The flag is cleared after the interrupt or the JTMR instruction is executed.

If the timer is configured with an external source, the internal timer overflow interrupt is also used to wake up the processor from a halt state. It should however be noted that the timer clock overflow interrupt and wake up function are determined by mask option.

If an internal source is used the frequency is determined by the system clock and the parameter n as defined in the equation.

$$\text{Frequency of TIMER clock} = \frac{\text{system clock}}{2^{n+m}}$$

where m=0 or 2 selectable by mask option. The parameter n ranges from 0 to 7 for the HT44500 or 0 to 14 for the HT445A0/445R0.



Timer Configurations

Interrupt

The HT44500/445A0/445R0 provide both internal and external interrupt modes. The DI and EI instructions are used to disable and enable the interrupts. When the $\overline{INT}$ pin is triggered on a high to low transition in the enable interrupt mode and the program is not within a CALL subroutine, the external interrupt is activated. This causes a subroutine call to location 8 and resets the interrupt latch.

Likewise when the timer flag is set in the enable interrupt mode and the program is not within a CALL subroutine the internal interrupt is activated. This causes a subroutine call to location 4 and resets the timer flag. If both external and internal interrupts arrive at the same time, the external one will be serviced first.

When running under a CALL subroutine or DI the interrupt acknowledge is on hold until the RET or EI instruction is invoked. The CALL instruction should not be used within an interrupt routine as unpredictable behaviour may occur. If within a CALL subroutine both internal and external interrupts occur, no matter what order they arrive in the external interrupt, will be serviced first after leaving the CALL subroutine. This also applies if the two interrupts arrive at the same time.

The interrupts are disabled by a hardware reset or a DI instruction. They remain disabled until the EI instruction is executed.

Whether the internal timer overflow interrupt can be activated or not is decided by mask option.

Initial Reset

The HT44500/445A0/445R0 provide an $\overline{RES}$ pin for system initialization. This pin is equipped with an internal pull high resistor and in combination with an external 0.1μ~1μF capacitor, provides an internal reset pulse of sufficient length to guarantee a reset to all internal circuits. If the reset pulse is generated externally, the $\overline{RES}$ pin must be held low for at least 5ms. Normal circuit operation will not commence until the $\overline{RES}$ pin returns high.

The reset performs the following functions:

- Sets the program counter PC to 000H
- Disables the interrupt mode
- Stops the timer
- Resets the timer and carry flag
- Turns off all LCD segments
- Sets Port A high or floating
- Clears the watch dog timer
  (for the HT445A0/445R0 only)

Halt

This special feature interrupts the chip's normal operation to reduce power consumption. When a HALT is executed the following happens ...

- The system clock will be stopped.
- The contents of the on-chip RAM and registers remain unchanged.
- The VLCD voltage remains on all of the LCD segments and commons, so the differential voltage is zero and the LCDs are blank.
- The watch dog timer resets to zero and stops counting - HT445A0/445R0 only.

The halt status can be terminated by an external interrupt, a hardware reset or timer overflow.

For a timer overflow to wake up the processor from a HALT state, it has to be driven by an external clock. In addition a mask option has to be chosen to allow a timer overflow to wake-up the processor.

For the HT445A0 and HT445R0 series the watch dog timer will begin counting when the halt status is terminated.

When the halt status is terminated by an external interrupt, the following procedures take place ...

Case 1: If the system is in an interrupt-disable state before entering the halt state:

- The instruction HALT is executed and the system enters a halt state.
- A falling edge transition on $\overline{INT}$ will awaken the system and return to the main program instruction following the HALT command.
- The interrupt signal will be held until the system receives an enable interrupt command at which point the held interrupt will be serviced.

Case 2: If the system is in an interrupt enable state:

- The instruction HALT is executed and the system enters a halt state.
- A falling edge transition on $\overline{INT}$ will awaken the system and execute the external interrupt subroutine.

Tone Generator

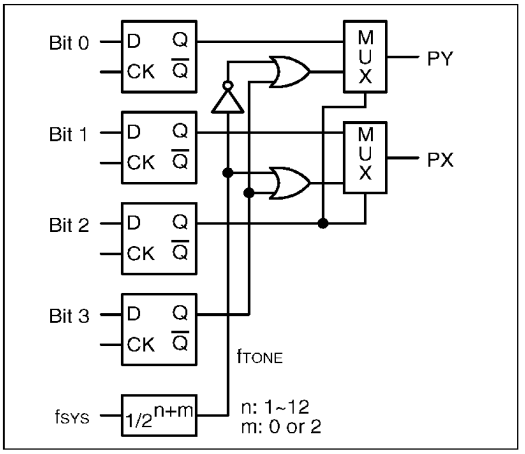
The HT445R0 provides a tone generator to generate differential driving single tone signals. A control register located at the address A0H of data memory is used to control the tone outputs. The tone output pair, PX and PY, can be used as level output lines or tone output pair by controlling the control register.

The bit 2 of control register is an S/W switch used to select the level output or tone output function. If the bit 2 of control register is set to "1", the level output function is selected; otherwise the tone output function is selected instead. Once the level function is selected, the bit 0 and bit 1 states of control register will respond to the tone output pair, PY and PX.

The tone output can be turned off or on by setting the bit 3 of control register to "1" or "0"

respectively. If the tone output function is selected and the bit 3 of the control register is set to "0", the PX and PY will output the differential driving signals. Note that the bit 3 of control register is used to control the tone signals only. The function of control register and the tone generator configuration are described in the following.

Tone generator control register (A0H)	
Bit 0	This bit stores the level output values of PY.
Bit 1	This bit stores the level output values of PX.
Bit 2	The level output function is selected if this bit is set to "1"; otherwise the tone output function is selected.
Bit 3	The tone frequency is enabled or disabled by setting this bit to "0" or "1" repectively.

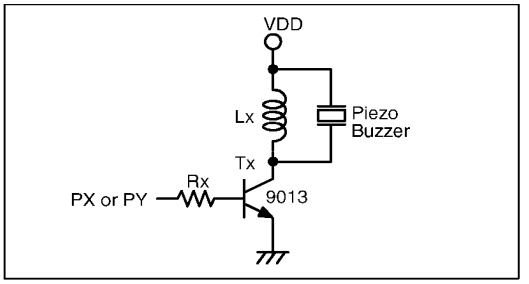


The tone frequency is derived from the internal frequency sources. A choice of 14 tone frequencies are available. The following equation shows the tone frequency options.

$$f_{TONE} = \frac{f_{SYS}}{2^{n+m}}$$

... where n ranges from 1 to 12 and m=0 or 2.

The following diagram shows the application of pulling the voltage of PX or PY output.



Strobe Pointer - SP

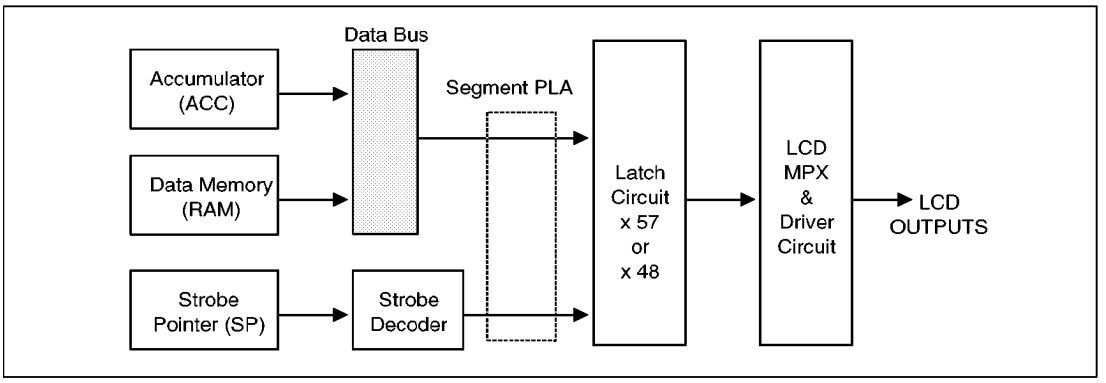
When executing the WRITE [SP], READ [SP] instructions to transfer display data, the LCD latch is addressed by the strobe pointer. The contents of the strobe pointer can be transferred to the accumulator and vice versa. The strobe pointer can be affected by the following instructions:

Instruction	Description
MOV SP,A	Move ACC to strobe pointer
MOV A,SP	Load ACC from strobe pointer
INC SP	Increment strobe pointer
DEC SP	Decrement strobe pointer

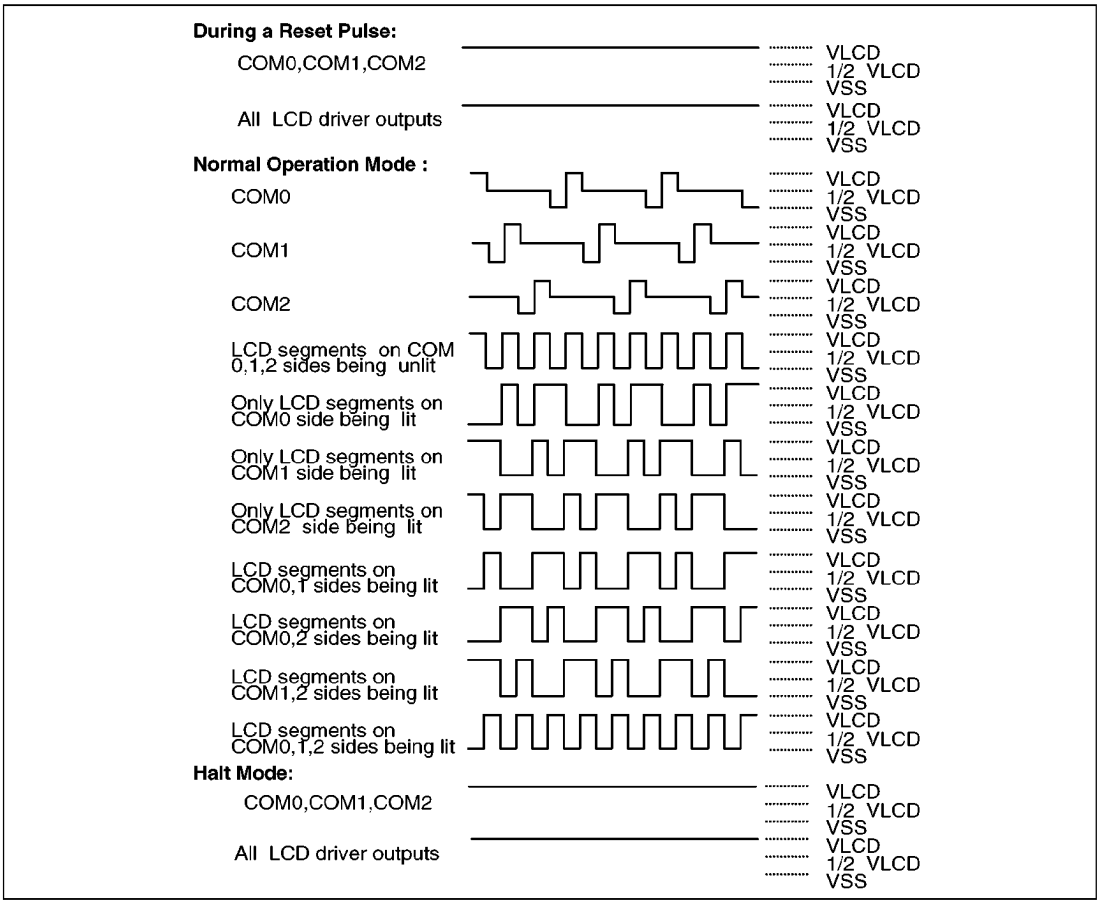
Note that the strobe pointer defines only the lower 3 bits with the MSB of the pointer always set to zero. For example if the SP is set to 8 by instruction MOV SP,A and the SP value read again using MOV A,SP, the value in the accumulator will be 0 instead of 8.

LCD Driver Output

The number of LCD driver outputs is 16×3 for the HT44500 and 19×3 for the HT445A0 and HT445R0. The output waveforms of the common outputs and segment outputs are designed to drive a 1/2 bias, 1/3 duty LCD device. Either 2 or 3 Common connections can be chosen by mask option. All the LCD segments are blank during a reset and a halt. The waveforms of the common and segment outputs are shown in the diagram.



Segment PLA



LCD Driver Output

SP2~SP0	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
000								
001								
010								
011							10-2	12-1
100								
101			9-2		8-0			
110								
111				2-2		4-2		

PLA Mask Option Setup Table

Segment PLA Circuit

The block diagram shows the operation of the LCD driver when the WRITE [SP] instruction is executed.

The contents of the accumulator and data memory are stored in the latch circuits specified by the strobe pointer. For the HT44500 with its 16 segment drive capability, there are 48 latches i.e. 3×16 or 3 latches per segment. For the HT445A0/445R0 with its 19 segment drive capability there are 57 latches i.e. 3×19.

The data bus has eight outputs from DB0 to DB7, and six decoder outputs from PST0 to PST5 generated from SP0, SP1 and SP2 giving a total of 8 combinations. The input data to the latch is provided by DB0 to DB7 while the clock signals of the latch are provided by PST0 to PST5. The segment PLA allows any combination of these lines to be selected by mask option.

Of the 64 signals obtained by combining DB0 to DB7 and PST0 to PST5, 48 signals can be selected for the HT44500 and 57 for the HT445A0/445R0 by programming the above mentioned segment PLA. The diagram shows the hardware configuration of the PLA. The READ [SP] instruction allows the ROM data whose address is specified by page 7, the accumulator and the RAM, to be transferred to the LCD latch.

Since the latch input and the latch clock are selected with the segment PLA, combination of the segments in the LCD driver output is flexible.

The PLA configuration table gives an example of how the PLA is setup. This table has to be filled in before IC manufacture, to enable Holtek to configure the PLA correctly.

In the table each single code of the three SP lines are used as a pointer to specify up to 8 LCD latch registers. The value in these latches is controlled by the value on the data bus of the corresponding bit. The accumulator controls the data in the upper nibble of this data bus value, and the data in the memory location set by R1R0 controls the lower nibble. The WRITE [SP] instruction is used to write this data to the LCD latch and to illuminate the correct segments. Note that the values in the table are supplied in a two number format SEG–COM where SEG represents the segment number and COM the common number.

For example in the table if segments 9–2 and 8–0 are to be illuminated, the corresponding latch register, SP, will be 5 or 101. The high nibble of the data bus, which is controlled by the accumulator, should be given the value 2, and the low nibble, controlled by the M(R1,R0) register, should be given the value 8. So ACC=2 and M(R1,R0) = 8. A WRITE [SP] instruction then illuminates these two segments. Subsequently to turn off these two segments, the SP is again equal to 5 and an other WRITE [SP] with ACC=0 and M(R1,R0) = 0 should be issued.

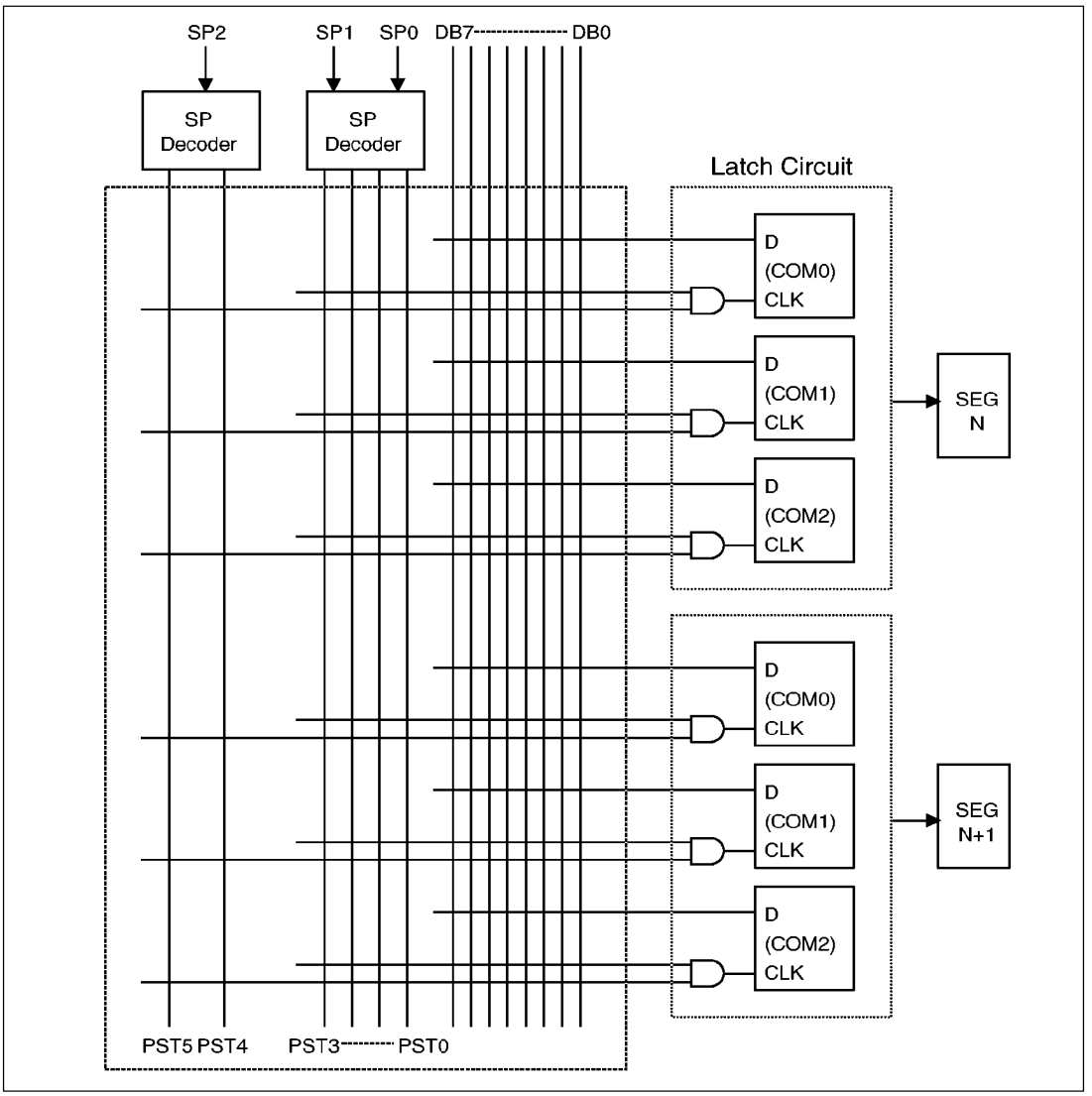
HT445A0/445R0 Watch Dog Timer

Both the HT445A0 and HT445R0 possess an

internal watch dog timer which is used to reset the controller at a defined interval. Note that the HT44500 has no watch dog timer. It can be used to examine the application program to see if it has jumped to unknown locations or not. There are three RAM ports to control the timer operation.

RAM port	Address	Operation
1	F0H~FFH	Clear timer
2	E0H~EFH	Start timer
3	D0H~DFH	Stop timer

Whenever the specified RAM port is accessed the corresponding action is implemented. For



Segment PLA

example if the application program needs to clear the watch dog timer, it can execute the following instructions:

MOV [R1R0],A where R1R0 = F0H~FFH
or MOV A,[R1R0]

MOV [R3R2],A where R3R2 = F0H~FFH
or MOV A,[R3R2]

where A can be any value.

The following three watch dog configurations are selectable by mask option:

- No watch dog timer
- Watch dog timer with clear function only
- Watch dog timer with all three functions

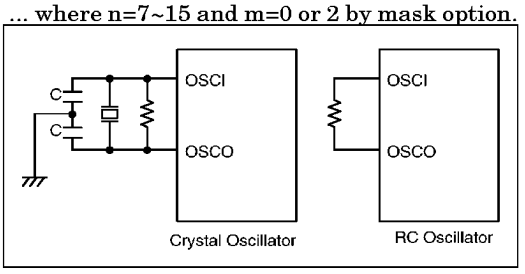
For the last two options the watch dog timer is cleared and starts counting during a power on reset. If the HALT instruction is executed, the watch dog timer resets to zero and stops counting. It starts re-counting when the system has been waken up by an initial reset or an external interrupt.

During normal operation, the application program will reset the timer before overflow occurs. If timer overflow occurs then the program is not under control. The controller will perform a reset to initialize the system.

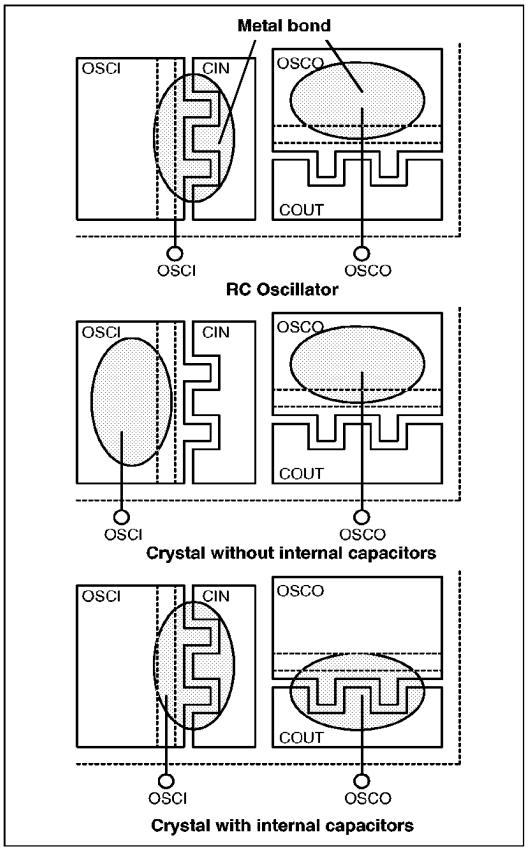
The watch dog timer is made from an 8-stage T-FF counter which can be scaled by the value 2^n where n has a range from 7 to 17 by mask option. The duration of the watch dog timer can be calculated by the following equation:

$$\text{Timer duration} = \frac{2^{(n+m+8)}}{\text{system clock}}$$

... where n=7~15 and m=0 or 2 by mask option.



Oscillator Configurations



HT445A0 Oscillator Metal Bonding

HT445R0 Voltage Doubler

The HT445R0 posseses an internal voltage doubler circuit which takes the low power supply voltage of the device and doubles it to provide a 3V voltage supply to allow operation with LCD's. For the voltage doubler to operate effectively a capacitor must be connected between CUP0 and CUP1 and between VLCD and VSS.

Oscillator Circuit

The system clock oscillation circuit can be selected to be a crystal or RC oscillator by mask option.

A crystal or ceramic resonator connected between OSCI and OSCO provides the feedback and phase shift required for oscillation. Using a

ceramic resonator will give reduced accuracy and will require different capacitor values. For the HT44500/445A0 the value of C will be 0~10p for the crystal oscillator and from 30p~180p for the ceramic resonator depending upon the frequency. The HT445R0 is different due to its low voltage operation. For a crystal frequency of 32KHz the capacitor values should be a nominal 13pF and the "crystal without capacitor" option selected.

For the RC oscillator a resistor is connected between OSCI and OSCO. The value of the resistor ranges from 5K to 700K.

A machine cycle consists of a sequence of 4 states numbered T1 to T4. Each state lasts for one oscillator period. For the HT44500/ 445A0 the system oscillator frequency ranges from 32KHz to 4MHz and for the HT445R0 the oscillator frequency can be up to 50KHz.

Metal bonding option for the HT445A0 oscillator

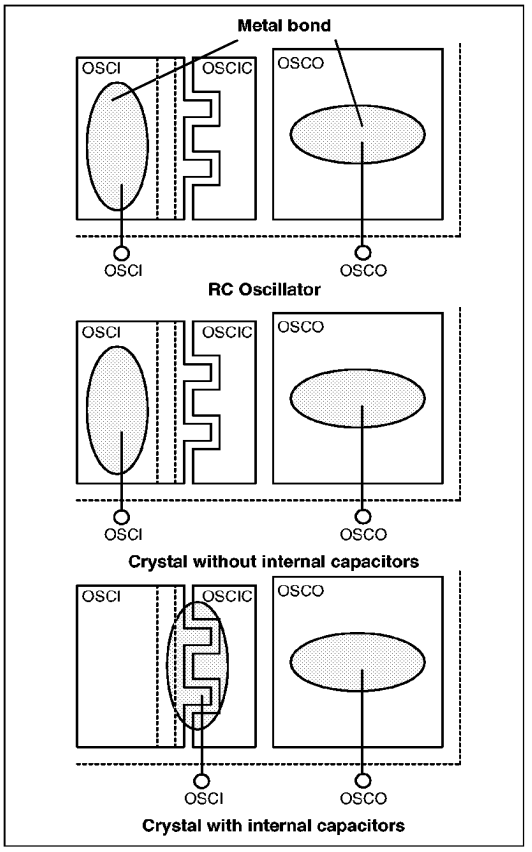
The following table shows the oscillator bonding options.

Oscillator type	Bonding method
RC oscillator	Bond OSCI to CIN only. OSCO not bonded to COUT.
Crystal without capacitors	OSCI and OSCO not bonded to CIN and COUT
Crystal with capacitors	Bond OSCI to CIN Bond OSCO to COUT

Using two internal capacitors the HT445A0 has 3 ways to provide the oscillator circuit: an RC oscillator, a crystal oscillator without the built-in capacitor and a crystal oscillator with the built-in capacitor. The built-in capacitors are connected to pads CIN and COUT. The following diagram shows the pad connections for OSCI/OSCO and CIN/COUT. Placing a metal bond across a pair of pads connects the corresponding pad to the internal capacitor.

Metal bonding option for the HT445R0 oscillator

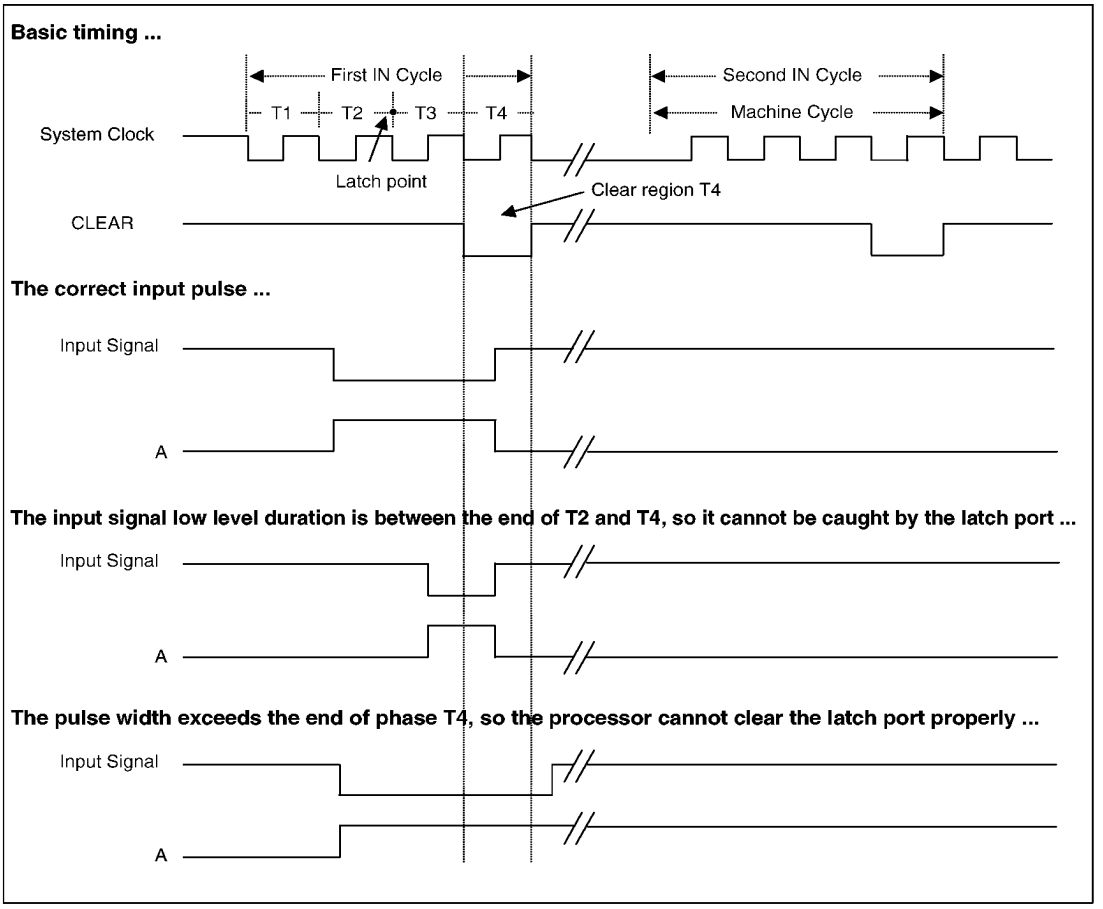
The following table shows the oscillator bonding options.



HT445R0 Oscillator Metal Bonding

Oscillator type	Bonding method
RC oscillator	OSCI not bonded to OSCIC
Crystal without capacitors	OSCI not bonded to OSCIC
Crystal with capacitors	Bond OSCI to OSCIC

Using an internal capacitor and the HT445R0 has 3 ways to provide the oscillator circuit: an RC oscillator, a crystal oscillator without the built-in capacitor and a crystal oscillator with the built-in



HT445A0/445R0 Input Port Timing

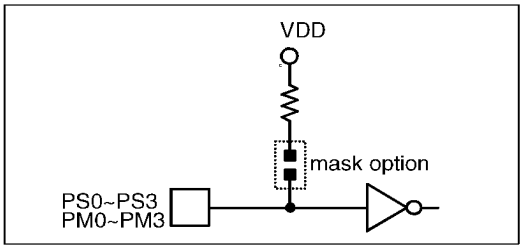
capacitor. The built-in capacitor is connected to pad OSCIC. The following diagram shows the pad connections for OSCI and OSCIC. Placing a metal bond across a pair of pads connects the corresponding pad to the internal capacitor.

For the HT445R0 it is recommended that a crystal oscillator of 32KHz is used for the system clock. If this is the case then external capacitor values of 13p should be chosen and the "crystal without capacitor" option must be selected.

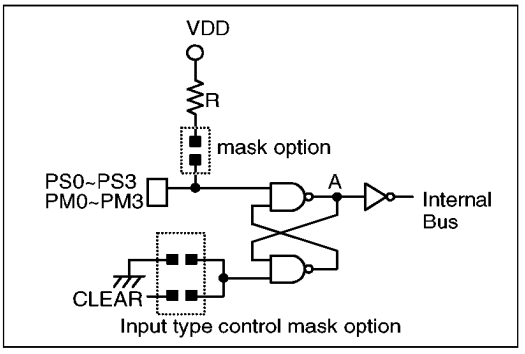
HT44500 Input Ports - PS,PM

The input ports PS,PM of the HT44500 are configured differently from the other two devices in that no latch option is offered.

For both ports a mask option is available to select the input to be of a pull-high type or non pull-high type.



HT44500 Input Ports - PS,PM



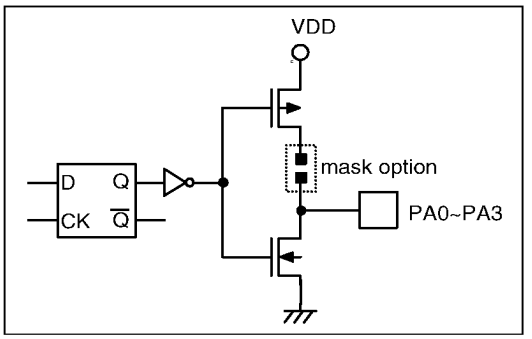
HT445A0/445R0 Input Ports PS, PM

Port-M also can be used as the conditional jump test input when the JPMn instruction is executed.

HT445A0/445R0 Input Ports - PS,PM

The input ports of the HT445A0/445R0 differ from the HT44500 in that a latch option is offered on each pin in addition to pull high resistors. For normal functioning non-latch type the mask option should select VSS, but for latch type operation the CLEAR option is selected. The normal type input is the same as the HT44500 where the processor polls the state of the input line to determine its logic level. In the latch type an input pulse on the input line will be latched which can then be read at any time by the processor. This option allows input pulses to be detected and reduces the processor loading time.

There are 4 phases in a machine cycle. If the port is configured as a latch type, the processor will read the data in the latch port at the second phase T2 by executing the instruction "IN A,Pi", where Pi = PS,PM. During the 4th phase of the same cycle, the processor will clear the latch port by setting the CLEAR to 0. Attention must be piad to two improper operations which may occur with latch type configurations. The first is: if a low going input occurs after the T2 phase and ends before the end of the T4 phase, the pulse will not be seen by the processor. The other is: if the low level duration extends to the end input instruction's machine cycle, the processor will not clear the latch and the data will still be held in the latch giving a false reading the next time the input latch is read. The diagram gives more details.



Output Port PA

Output Ports - PA

The output port, PA, is configured in the following way.

The mask option to selects the output configuration to be of a standard CMOS output type or an open drain NMOS output type. At the initial clear mode the output ports are at the VDD level if they are of CMOS types. However, the output ports are in a floating state if they are of NMOS types.

Mask Options

The following either/or options are available by mask option which the user must select prior to manufacture.

- 4-bit input ports PM and PS with or without a pull high resistors
- 4-bit input ports PM and PS with or without latch option for the HT445A0, HT445R0 only
- Output port PA to be CMOS or open drain NMOS
- 8-bit programmable timer with an external clock or internal frequency source
- Timer clock overflow interrupt enable
- Processer can be waken-up from a halt state by a timer overflow if an external timer clock is used.

- Watch dog timer options for the HT445A0, HT445R0 only
- PLA configuration
- RC or crystal oscillator
- Crystal oscillator with or without an internal capacitor
- 2 or 3 common connections
- Tone frequency options for the HT445R0 only

Software Tools

To ease the programming task and reduce development time, Holtek supplies a development system for the HT44500/445A0/445R0. The system runs under an IBM PC-XT/AT environment and consists of both a hardware emulation board and a suite of programs including powerful debug functions. The user can download the code from the PC to the emulation board for verification. The main features of the system are as follows.

- Can incorporate the user's text editor or word processor with Holtek's cross assembler to form an integrated development system
- Supports mouse functions with its window based human interface
- Performs stand-alone operation for demonstration purposes
- Auto-executes self test function at every power on reset
- Provides symbolic debugging capabilities
- User defined mask options
- RC with variable resistor or crystal system clock provided
- Displays and modifies registers, carry flag, timer, port output level and internal RAM
- Single instruction stepping
- Jumps unconditionally to any address and halts any time during execution
- Provides up to 8 breakpoint settings
- Real time 255 forward step or 256 backward step trace

After program verification on the emulation board, the customer supplies Holtek with the verified code prior to manufacture.

INSTRUCTION SET

Instruction Set Summary

Mnemonic	Description	Byte	Cycle	CF
Arithmetic				
ADD A,[R1R0]	Add data memory to ACC	1	1	√
ADC A,[R1R0]	Add data memory with carry to ACC	1	1	√
SUB A,[R1R0]	Subtract data memory from ACC	1	1	√
SBC A,[R1R0]	Subtract data memory from ACC with borrow	1	1	√
ADD A,XH	Add immediate data to ACC	2	2	√
SUB A,XH	Subtract immediate data from ACC	2	2	√
DAA	Decimal adjust ACC for addition	1	1	√
Logic operation				
AND A,[R1R0]	AND data memory to ACC	1	1	—
OR A,[R1R0]	OR data memory to ACC	1	1	—
XOR A,[R1R0]	Exclusive-OR data memory to ACC	1	1	—
AND A,XH	AND immediate data to ACC	2	2	—
OR A,XH	OR immediate data to ACC	2	2	—
XOR A,XH	Exclusive-OR immediate data to ACC	2	2	—
Increment & Decrement				
INC A	Increment ACC	1	1	—
INC Rn	Increment register, n=0~4	1	1	—
INC [R1R0]	Increment data memory	1	1	—
DEC A	Decrement ACC	1	1	—
DEC Rn	Decrement register, n=0~4	1	1	—
DEC [R1R0]	Decrement data memory	1	1	—
Rotate				
RLC A	Rotate ACC left through the carry	1	1	√
RRC A	Rotate ACC right through the carry	1	1	√
Input & Output				
IN A,Pi	Input port-i to ACC, port-i=PM,PS	1	1	—
OUT PA,A	Output ACC to port-A	1	1	—

Mnemonic	Description	Byte	Cycle	CF
Data Move				
MOV A,Rn	Move register to ACC, n=0~4	1	1	—
MOV Rn,A	Move ACC to register, n=0~4	1	1	—
MOV A,[R1R0]	Move data memory to ACC	1	1	—
MOV A,[R3R2]	Move data memory to ACC	1	1	—
MOV [R1R0],A	Move ACC to data memory	1	1	—
MOV [R3R2],A	Move ACC to data memory	1	1	—
MOV A,XH	Move immediate data to ACC	1	1	—
MOV R1R0,XXH	Move immediate data to R1 and R0	2	2	—
MOV R3R2,XXH	Move immediate data to R3 and R2	2	2	—
MOV R4,XH	Move immediate data to R4	1	1	—
Branch				
JMP addr	Jump unconditionally	2	2	—
JC addr	Jump on carry=1	2	2	—
JNC addr	Jump on carry=0	2	2	—
JTMR addr	Jump on timer out	2	2	—
JPMn addr	Jump on Port–M bit n=1, n=0~3	2	2	—
JZ A,addr	Jump on ACC is zero	2	2	—
JNZ A,addr	Jump on ACC is not zero	2	2	—
JNZ Rn,addr	Jump on register Rn not zero, n=0,1,4	2	2	—
Subroutine				
CALL addr	Subroutine call	2	2	—
RET	Return from subroutine and interrupt	1	1	—
Flag				
CLC	Clear carry flag	1	1	0
STC	Set carry flag	1	1	1
EI	Enable interrupt	1	1	—
DI	Disable interrupt	1	1	—
NOP	No operation	1	1	—
Timer				
TIMER XXH	Set 8 bits immediate data to TIMER	2	2	—
TIMER ON	Set TIMER start count	1	1	—
TIMER OFF	Set TIMER stop count	1	1	—
MOV A,TMRL	Move low nibble of TIMER to ACC	1	1	—
MOV A,TMRH	Move high nibble of TIMER to ACC	1	1	—

Mnemonic	Description	Byte	Cycle	CF
Table Read				
READ R4A	Read page 7 of ROM code to R4 & ACC	1	2	—
READ [SP]	Read page 7 of ROM code to LCD latch	1	2	—
WRITE [SP]	Write ACC & data memory to LCD latch	1	1	—
MOV A,SP	Move strobe pointer to ACC	1	1	—
MOV SP,A	Move ACC to strobe pointer	1	1	—
INC SP	Increment strobe pointer	1	1	—
DEC SP	Decrement strobe pointer	1	1	—
Miscellaneous				
HALT	Enter power down mode	2	2	—

The following table is provided for the HT445A0/HT445R0 devices only.

Mnemonic	Description	Byte	Cycle	CF
Watch Dog				

MOV [R1R0],A
MOV A,[R1R0]
MOV [R3R2],A
MOV A,[R3R2]
	Clear watch dog timer, where R1R0,R3R2=F0H~FFH. A can be any value.	1
		1
		1
		1
			1
			1
			1
			1
				—
				—
				—
				—

MOV [R1R0],A
MOV A,[R1R0]
MOV [R3R2],A
MOV A,[R3R2]
	Start watch dog timer, where R1R0,R3R2=E0H~EFH. A can be any value.	1
		1
		1
		1
			1
			1
			1
			1
				—
				—
				—
				—

MOV [R1R0],A
MOV A,[R1R0]
MOV [R3R2],A
MOV A,[R3R2]
	Stop watch dog timer, where R1R0,R3R2=D0H~DFH. A can be any value.	1
		1
		1
		1
			1
			1
			1
			1
				—
				—
				—
				—

Instruction Definitions

ADC A,[R1R0] Add data memory contents and carry to accumulator

Machine code 0 0 0 0 1 0 0 0

Description The contents of the data memory location addressed by the register pair "R1,R0" and the carry are added to the accumulator. Carry is affected.

Operation ACC ← ACC+M(R1,R0)+CF

ADD A,XH Add immediate data to accumulator

Machine code 1 1 1 1 0 0 0 1 0 0 0 0 d d d d

Description The specified data is added to the accumulator. Carry is affected.

Operation ACC ← ACC+XH

ADD A,[R1R0] Add data memory content to accumulator

Machine code 0 0 0 0 1 0 0 1

Description The contents of the data memory location addressed by the register pair "R1,R0" are added to the accumulator. Carry is affected.

Operation ACC ← ACC+M(R1,R0)

AND A,XH Logical AND immediate data to accumulator

Machine code 1 1 1 1 0 0 1 1 0 0 0 0 d d d d

Description Data in the accumulator is logically ANDed with the immediate data specified by the code.

Operation ACC ← ACC "AND" XH

AND A,[R1R0] Logical AND accumulator with data memory

Machine code 0 0 0 0 1 1 0 0

Description Data in the accumulator is logically ANDed with the data memory location addressed by the register pair "R1,R0".

Operation ACC ← ACC "AND" M(R1,R0)

CALL address Subroutine call

Machine code 1 1 1 0 1 a a a a a a a a a a a a

Description The program counter bits 0~10 are saved in the stack. The program counter is then loaded from the directly-specified address.

Operation Stack ← PC+2
PC ← address

CLC Clear carry flag

Machine code 0 0 0 0 0 0 0 0

Description The carry flag is reset to zero.

Operation CF ← 0

DAA Decimal-Adjust accumulator

Machine code 0 0 1 1 0 1 1 0

Description The accumulator value is adjusted to BCD (Binary Code Decimal) code, if the contents of the accumulator is greater than 9 or CF (Carry flag) is one.

Operation If ACC>9 or CF=1 then
ACC ← ACC+6, CF ← 1
esle
ACC ← ACC, CF ← CF

DEC A Decrement accumulator

Machine code 0 1 1 1 1 1 1 1

Description Data in the accumulator is decremented by one. Carry flag is not affected.

Operation ACC ← ACC–1

DEC Rn Decrement register

Machine code 0 0 0 1 n n n 1

Description Data in the working register "Rn" is decremented by one. Carry flag is not affected.

Operation Rn ← Rn–1; Rn=R0,R1,R2,R3,R4, for nnn=0,1,2,3,4

DEC SP Decrement strobe pointer

Machine code 0 0 1 1 1 1 0 1

Description The contents of the strobe pointer is decremented by one.

Operation SP ← SP–1

DEC [R1R0] Decrement data memory

Machine code 0 0 0 1 1 0 1 1

Description Data in the data memory specified by the register pair "R1,R0" is decremented by one. Carry flag is not affected.

Operation M(R1,R0) ← M(R1,R0)–1

DI Disable interrupt

Machine code 0 0 0 1 1 1 1 1

Description Internal time-out interrupt and external interrupt are disabled.

EI Enable interrupt

Machine code 0 0 0 1 1 1 1 0

Description Internal time-out interrupt and external interrupt are enabled.

HALT Halt system clock

Machine code 0 0 1 1 0 0 1 0 x x x x x x x x

Description Turn off system clock, and enter power down mode.
x represerts a don't case condition.

Operation PC ← PC+2

IN A,Pi Input port to accumulator

Machine code PM 0 0 1 0 1 1 0 0

PS 0 0 1 1 0 0 0 0

Description The data on port "Pi" is transferred to the accumulator.

Operation ACC ← Pi; Pi=PM or PS

INC A Increment accumulator

Machine code 0 1 0 0 0 0 0 1

Description Data in the accumulator is incremented by one. Carry flag is not affected.

Operation ACC ← ACC+1

INC Rn Increment register

Machine code 0 0 0 1 n n n 0

Description Data in the working register "Rn" is incremented by one. Carry flag is not affected.

Operation Rn ← Rn+1; Rn=R0,R1,R2,R3,R4 for nnn=0,1,2,3,4

INC SP Increment strobe pointer

Machine code 0 0 1 1 1 1 0 0

Description The contents of the strobe pointer are incremented by one.

Operation SP ← SP+1

INC [R1R0] Increment data memory

Machine code 0 0 0 1 1 0 1 0

Description Data in the data memory specified by the register pair "R1,R0" is incremented by one. Carry flag is not affected.

Operation M(R1,R0) ← M(R1,R0)+1

JC address Jump if carry is set

Machine code 1 1 0 0 0 a a a a a a a a a a a

Description Bits 0~10 of the program counter are replaced with the directly-specified address, if the CF (Carry flag) is set to one.

Operation PC ← address, if CF=1
PC ← PC+2, if CF=0

JMP address Direct jump

Machine code 1 1 1 0 0 a a a a a a a a a a a

Description Bits 0~10 of the program counter are replaced with the directly-specified address.

Operation PC ← address

JNC address Jump if carry is not set

Machine code 1 1 0 0 1 a a a a a a a a a a a

Description Bits 0~10 of the program counter are replaced with the directly-specified address, if the CF (Carry flag) is set to zero.

Operation PC ← address, if CF=0
PC ← PC+2, if CF=1

JNZ A,address Jump if accumulator is not zero

Machine code 1 0 1 1 1 a a a a a a a a a a a

Description Bits 0~10 of the program counter are replaced with the directly-specified address, if the accumulator is not zero.

Operation PC ← address, if ACC≠0
PC ← PC+2, if ACC=0

JNZ Rn,address Jump if register is not zero

Machine code R0 1 0 1 0 0 a a a a a a a a a a a

R1 1 0 1 0 1 a a a a a a a a a a a

R4 1 1 0 1 1 a a a a a a a a a a a

Description Bits 0~10 of the program counter are replaced with the directly-specified address, if the register is not zero.

Operation PC ← address, if Rn≠0; Rn=R0,R1,R4
PC ← PC+2, if Rn=0

JPMn address Jump if port PM bit n is set

Machine code 1 0 0 n n a a a a a a a a a a a

Description Bits 0~10 of the program counter are replaced with the directly-specified address, if the port PM bit n is set.

Operation PC ← address, if PM bit n=1
PC ← PC+2, if PM bit n=0

JTMR address Jump if time-out

Machine code 1 1 0 1 0 a a a a a a a a a a a

Description Bits 0~10 of the program counter are replaced with the directly-specified address, if the TMFG (Timer flag) is set to one.

Operation PC ← address, if TMFG=1
PC ← PC+2, if TMFG=0

JZ A,address Jump if accumulator is zero

Machine code 1 0 1 1 0 a a a a a a a a a a a

Description Bits 0~10 of the program counter are replaced with the directly-specified address, if the accumulator is zero.

Operation PC ← address, if ACC=0
PC ← PC+2, if ACC≠0

MOV A,Rn Move register to accumulator

Machine code 0 0 1 0 n n n 1

Description Data in the working register "Rn" is moved to the accumulator.

Operation ACC ← Rn; Rn=R0,R1,R2,R3,R4, for nnn=0,1,2,3,4

MOV A,SP Move strobe pointer to accumulator

Machine code 0 0 1 1 1 1 1 0

Description The contents of the strobe pointer are loaded to the accumulator.

Operation ACC ← SP

MOV A,TMRH Move timer to high nibble accumulator

Machine code 0 0 1 0 1 0 1 1

Description The high nibble data of the timer counter is loaded to the accumulator.

Operation ACC ← TIMER (high nibble)

MOV A,TMRL Move timer to low nibble accumulator

Machine code 0 0 1 0 1 0 1 0

Description The low nibble data of the timer counter is loaded to the accumulator.

Operation ACC ← TIMER (low nibble)

MOV A,XH Move immediate data to accumulator

Machine code 0 1 1 1 d d d d

Description The 4-bit data specified by the code is loaded to the accumulator.

Operation ACC ← XH

MOV A,[R1R0] Move data memory to accumulator

Machine code 0 0 0 0 0 1 0 0

Description Data in the data memory specified by the register pair "R1,R0" is moved to the accumulator. The watch dog timer can also be accessed by this instruction with the specified RAM port. This applies to the HT445A0/445R0 only.

Operation ACC ← M(R1,R0)

MOV A,[R3R2] Move data memory to accumulator

Machine code 0 0 0 0 0 1 1 0

Description Data in the data memory specified by the register pair "R3,R2" is moved to the accumulator. The watch dog timer can also be accessed by this instruction with the specified RAM port. This applies to the HT445A0/445R0 only.

Operation ACC ← M(R3,R2)

MOV R1R0,XXH Move immediate data to R1 and R0

Machine code 0 1 0 1 d d d d 0 0 0 0 d d d d

Description The 8-bit data specified by the code is loaded to the working register R1 and R0, the high nibble of the data is loaded to R1, and the low nibble of the data is loaded to R0.

Operation R1 ← XH (high nibble)
R0 ← XH (low nibble)

MOV R3R2,XXH Move immediate data to R3 and R2

Machine code 0 1 1 0 d d d d 0 0 0 0 d d d d

Description The 8-bit data specified by the code is loaded to the working register R3 and R2, the high nibble of the data is loaded to R3, and the low nibble of the data is loaded to R2.

Operation R3 ← XH (high nibble)
R2 ← XH (low nibble)

MOV R4,XH Move immediate data to R4

Machine code 0 1 0 0 d d d d

Description The 4-bit data specified by the code is loaded to the working register R4.

Operation R4 ← XH

MOV Rn,A Move accumulator to register

Machine code 0 0 1 0 n n n 0

Description Data in the accumulator is moved to the working register "Rn".

Operation Rn ← ACC; Rn=R0,R1,R2,R3,R4, for nnn=0,1,2,3,4

MOV SP,A Move accumulator to strobe pointer

Machine code 0 0 1 1 1 0 1 0

Description The contents of the accumulator is moved to the strobe pointer.

Operation SP ← ACC

MOV [R1R0], A Move accumulator to data memory

Machine code 0 0 0 0 0 1 0 1

Description Data in the accumulator is moved to the data memory specified by the register pair "R1,R0". The watch dog timer can also be accessed by this instruction with the specified RAM port. This applies to the HT445A0/445R0 only.

Operation M(R1,R0) ← ACC

MOV [R3R2], A Move accumulator to data memory

Machine code 0 0 0 0 0 1 1 1

Description Data in the accumulator is moved to the data memory specified by the register pair "R3,R2". The watch dog timer can also be accessed by this instruction with the specified RAM port. This applies to the HT445A0/445R0 only.

Operation M(R3,R2) ← ACC

NOP No operation

Machine code 0 0 0 0 1 1 1 1

Description Do nothing, but one instruction cycle is delayed.

OR A,XH Logical OR immediate data to accumulator

Machine code 1 1 1 1 0 1 0 1 0 0 0 0 d d d d

Description Data in the accumulator is logically ORed with the immediate data specified by the code.

Operation ACC ← ACC "OR" XH

OR A,[R1R0] Logical OR accumulator with data memory

Machine code 0 0 0 0 1 1 1 0

Description Data in the accumulator is logically ORed with the data memory location addressed by the register pair "R1,R0".

Operation ACC ← ACC "OR" M(R1,R0)

OUT PA,A Output accumulator data to port PA

Machine code PA 0 0 1 0 1 1 0 1

Description The data in the accumulator is transferred to port PA and latched.

Operation PA ← ACC

READ R4A Read ROM code to R4 and accumulator

Machine code 0 0 1 1 1 0 0 0

Description The 8-bits of ROM code addressed by page 7, ACC and M(R1,R0) are moved to the working register R4 and the accumulator. The high nibble of the ROM code is loaded to R4 and the low nibble of the ROM code is loaded to the accumulator. The address of the ROM code is specified by the following description:
Page 7 → ROM code address bit 10~8 are "111"
ACC → ROM code address bit 7~4
M(R1,R0) → ROM code addres bits 3~0

Operation R4 ← ROM code (high nibble)
ACC ← ROM code (low nibble)

READ [SP] Read ROM code to LCD latch

Machine code 0 0 1 1 1 0 0 1

Description The 8-bits of ROM code addressed by page 7, ACC and M(R1,R0) are moved to the LCD latch. The LCD latch is specified by the SP (strobe pointer). The address of the ROM code is specified by the following description :
Page 7 → ROM code address bit 10~8 are "111"
ACC → ROM code address bit 7~4
M(R1,R0) → ROM code address bit 3~0

Operation LCD latch [SP] ← ROM code [Page 7, ACC, M(R1,R0)]

RET Return from subroutine and interrupt

Machine code 0 0 1 0 1 1 1 1

Description The program counter bits 0~10 are restored from the stack.

Operation PC ← Stack

RLC A Rotate accumulator left through carry

Machine code 0 0 0 0 0 0 1 1

Description The contents of the accumulator are rotated one bit left. Bit 3 replaces the carry bit; the carry bit is rotated into the bit 0 position.

Operation An+1 ← An; An: Accumulator bit n (n=0,1,2)
A0 ← CF
CF ← A3

RRC A Rotate accumulator right through carry

Machine code 0 0 0 0 0 0 1 0

Description The contents of the accumulator are rotated one bit right. Bit 0 replaces the carry bit; the carry bit is rotated into the bit 3 position.

Operation An ← An+1; An: Accumulator bit n (n=0,1,2)
A3 ← CF
CF ← A0

SBC A,[R1R0] Subtract data memory content and carry from ACC

Machine code 0 0 0 0 1 0 1 0

Description The contents of the data memory location addressed by the register pair "R1,R0" and the complement of the carry are subtracted from the accumulator. Carry is set, if a borrow does not take place in subtraction; otherwise carry is cleared.

Operation ACC ← ACC+$\overline{M(R1,R0)}$+CF

STC Set carry flag

Machine code 0 0 0 0 0 0 0 1

Description The carry flag is set to one.

Operation CF ← 1

SUB A,XH Subtract immediate data from accumulator

Machine code 1 1 1 1 0 0 1 0 0 0 0 0 d d d d

Description The specified data is subtracted from the accumulator. Carry is set, if a borrow does not take place in subtraction; otherwise carry is cleared.

Operation ACC ← ACC+$\overline{XH}$+1

SUB A,[R1R0] Subtract data memory contents from accumulator

Machine code 0 0 0 0 1 0 1 1

Description The contents of the data memory location addressed by the register pair "R1,R0" is subtracted from the accumulator. Carry is set, if a borrow does not take place in subtraction; otherwise carry is cleared.

Operation ACC ← ACC+$\overline{M(R1,R0)}$+1

TIMER OFF Set timer stop counting

Machine code 0 0 0 1 1 1 0 1

Description The timer counter stops counting, when the "TIMER OFF" instruction is executed.

TIMER ON Set timer start counting

Machine code 0 0 0 1 1 1 0 0

Description The timer counter starts counting, when the "TIMER ON" instruction is executed.

TIMER XXH Set immediate data to timer counter

Machine code 0 0 1 1 1 0 1 1 d d d d d d d d

Description The 8-bit data specified by the code is loaded to the timer counter.

Operation TIMER ← XXH

WRITE [SP] Write accumulator and data memory to LCD latch

Machine code 0 0 1 0 1 1 1 0

Description The accumulator and data memory M(R1,R0) are moved to the LCD latch. The LCD latch is specified by the SP (strobe pointer). The contents of the accumulator are loaded to the high nibble of the LCD latch, and the contents of the data memory M(R1,R0) is loaded to the low nibble of the LCD latch.

Operation LCD latch [SP] ← ACC (high nibble)
LCD latch [SP] ← M(R1,R0)

XOR A,XH Logical XOR immediate data to accumulator

Machine code 1 1 1 1 0 1 0 0 0 0 0 0 d d d d

Description Data in the accumulator is Exclusive-ORed with the immediate data specified by the code.

Operation ACC ← ACC "XOR" XH

XOR A,[R1R0] Logical XOR accumulator with data memory

Machine code 0 0 0 0 1 1 0 1

Description Data in the accumulator is Exclusive-ORed with the data memory location addressed by the register pair "R1,R0".

Operation ACC ← ACC "XOR" M(R1,R0)