十速

TM8763

with LCD Driver

User Manual

Rev 1.0

tenx reserves the right to change or discontinue the manual and online documentation to this product herein to improve reliability, function or design without further notice. **tenx** does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others. **tenx** products are not designed, intended, or authorized for use in life support appliances, devices, or systems. If Buyer purchases or uses tenx products for any such unintended or unauthorized application, Buyer shall indemnify and hold tenx and its officers, employees, subsidiaries, affiliates and distributors harmless against all claims, cost, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use even if such claim alleges that tenx was negligent regarding the design or manufacture of the part.

AMENDMENT HISTORY

Version	Date	Description
V1.0	Dec, 2016	New release.

CONTENTS

1. General Description

1.1 GENERAL DESCRIPTION

The TM8763 is an embedded high-performance 4-bit microcomputer with LCD/LED driver. It contains all the necessary functions, such as 4-bit parallel processing ALU, ROM, RAM, I/O ports, timer, clock generator, dual clock operation, EL panel driver, LCD driver, look-up table, watchdog timer and key matrix scanning circuitry in a signal chip.

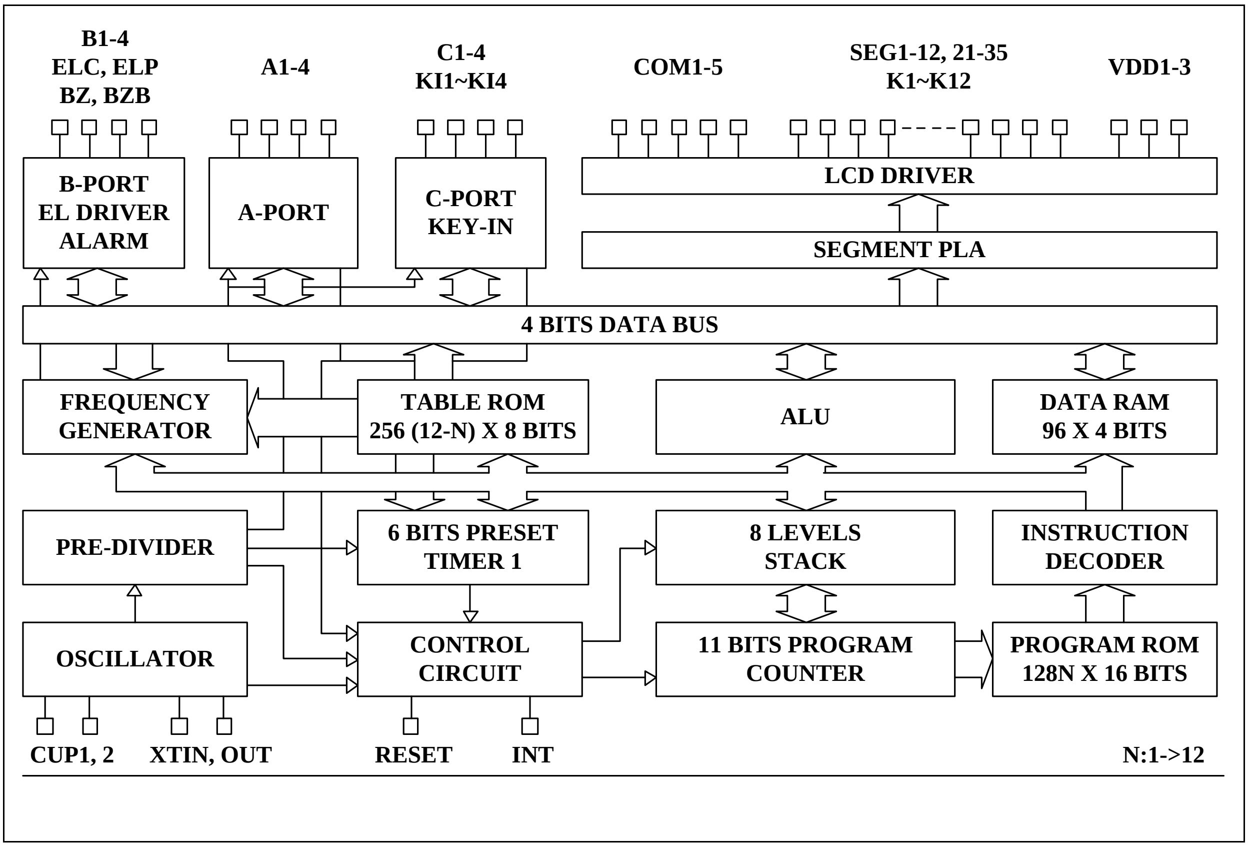
1.2 FEATURES

1. Low power dissipation.
2. Powerful instruction set (135 instructions).
   - Binary addition, subtraction, BCD adjust, logical operation in direct and index addressing mode.
   - Single bit manipulation (set, reset, decision for branch).
   - Various conditional branch.
   - 16 working registers and manipulation.
   - Table look-up.
   - LCD driver data transfer.
3. LCD/LED driver output.
   - 5 common outputs and 27 segment outputs (up to drive 135 LCD/LED segments).
   - 1/2 Duty, 1/3 Duty, 1/4 Duty or 1/5 Duty for both LCD/LED drivers is selected by MASK option.
   - 1/2 Bias or 1/3 Bias for LCD driver is selected by MASK option.
   - Single instruction to turn off all segments.
   - Segment outputs(SEG1~12,21~35) could be defined as CMOS or P_open drain type output by mask option.
4. Memory capacity.
   - ROM capacity 1536 x 16 bits.
   - RAM capacity 96 x 4 bits.
5. Input/output ports.
   - Port IOA 4 pins (with internal pull-low), muxed with SEG24~27.
   - Port IOB 4 pins (with internal pull-low), muxed with SEG28~31.
   - Port IOC 4 pins (with internal pull-low/low-level-hold), muxed with SEG32~SEG35.

   IOC port had built-in input signal chattering prevention circuitry.
6. 8 level subroutine nesting.
7. Interrupt function.
   - External factors 3 (INT pin, Port IOC & KI input).
   - Internal factors 2 (Pre-Divider, Timer1).

8. Built-in EL panel driver.
   - ELC, ELP (Muxed with SEG28, SEG29).
9. Built in Alarm, clock or single tone melody generator.
   - BZB, BZ (Muxed with SEG30, SEG31).
10. Built in key matrix scanning function.
    - K1~K12 (Shared with SEG1~SEG12).
    - KI1~KI4 (Muxed with SEG32~SEG35).
11. One 6-bit programmable timer with programmable clock source.
12. Watch dog timer.
13. Built-in Voltage doubler, halver, tripler charge pump circuit.
14. Dual clock operation.
    - slow clock oscillation can be defined as X'tal or external RC type oscillator by mask option.
    - fast clock oscillation can be defined as internal R or external R type oscillator by mask option.
15. HALT function.
16. STOP function.

1.3 BLOCK DIAGRAM



1.4 PAD DIAGRAM

The substrate of chip should be connected to GND.

1.5 PAD COORDINATE

No	Name	No	Name
1	BAK	23	SEG9(K9)
2	XIN	24	SEG10(K10)
3	XOUT	25	SEG11(K11)
4	GND	26	SEG12(K12)
5	VDD1	27	SEG21
6	VDD2	28	SEG22
7	VDD3	29	SEG23
8	CUP1	30	SEG24/IOA1
9	CUP2	31	SEG25/IOA2
10	COM1	32	SEG26/IOA3
11	COM2	33	SEG27/IOA4
12	COM3	34	SEG28/IOB1/ELC
13	COM4	35	SEG29/IOB2/ELP
14	COM5	36	SEG30/IOB3/BZB
15	SEG1(K1)	37	SEG31/IOB4/BZ
16	SEG2(K2)	38	SEG32/IOC1/KI1
17	SEG3(K3)	39	SEG33/IOC2/KI2
18	SEG4(K4)	40	SEG34/IOC3/KI3
19	SEG5(K5)	41	SEG35/IOC4/KI4
20	SEG6(K6)	42	RESET
21	SEG7(K7)	43	INT
22	SEG8(K8)	44	TEST

1.6 PIN DESCRIPTION

Name	I/O	Description
BAK	P	Power Back-up pin (+). • At Li Mode, connect a 0.1u capacitor to GND.
VDD1,2,3	P	LCD supply voltage and positive supply pins. • In Ag power mode, connect positive power to VDD1. • In Li or ExtV power mode, connect positive power to VDD2.
RESET	I	Input pin for external reset request signal, built-in internal pull-down resistor. • Reset cycle time can be defined as “PH15/2” or “PH12/2” by mask option. • Reset Type can be defined as “Level reset” or “Pulse reset” by mask option.
INT	I	Input pin for external interrupt request signal. • Falling edge or rising edge triggered is defined by mask option. • Internal pull-down or pull-up resistor is defined by mask option.
TESTA	I	Test signal input pin.
CUP1,2	O	Switching pins for supply the LCD driving voltage to the VDD1,2,3 pins. • Connect the CUP1 and CUP2 pins with non-polarized electrolytic capacitor when chip operated in 1/2 or 1/3 bias mode. • In no BIAS mode application, leave these pins opened
XIN XOUT	I O	Time base counter frequency (clock specified. LCD alternating frequency. Alarm signal frequency) or system clock oscillation. • 32KHz Crystal oscillator. • In FAST ONLY mode option, connect an external resistor could compose a RC oscillator.
COM1~5	O	Output pins for driving the common pins of the LCD or LED panel.
SEG1-12, SEG21-35	O	Output pins for driving the LCD or LED panel segment.
IOA1-4	I/O	Input / Output port A.
IOB1-4	I/O	Input / Output port B.
IOC1-4	I/O	Input / Output port C.
ELC/ELP	O	Output port for EL driver.
BZB/BZ	O	Output port for alarm, clock or single tone melody generator
K1~12	O	Output port for key matrix scanning. (Shared with SEG1~SEG12)
KI1~4	I	Input port for key matrix.
GND	P	Negative supply voltage.

1.7 CHARACTERISTICS

ABSOLOUTE MAXIMUM RATINGS

(GND= 0V)

Name	Symbol	Range	Unit
Maximum Supply Voltage	VDD1	-0.3 to 5.5	V
	VDD2	-0.3 to 5.5	
	VDD3	-0.3 to 8.5	
Maximum Input Voltage	Vin	-0.3 to VDD1/2+0.3	
Maximum output Voltage	Vout1	-0.3 to VDD1/2+0.3	
	Vout2	-0.3 to VDD3+0.3	
Maximum Operating Temperature	Topg	-20 to +70	°C
Maximum Storage Temperature	Tstg	-25 to +125	

POWER CONSUMPTION

at Ta= -20°C to 70°C, GND=0V

Name	Sym.	Condition	Min.	Typ.	Max.	Unit
HALT mode	IHALT1	Only 32.768KHz Crystal oscillator operating, without loading. Ag mode, VDD1=1.5V, BCF = 0		2		uA
	IHALT2	Only 32.768KHz Crystal oscillator operating, without loading. Li mode, VDD2=3.0V, BCF = 0		2		
STOP mode	ISTOP				1	

Note: When RC oscillator function is operating, the current consumption will depend on the frequency of oscillation.

INTERNAL RC FREQUENCY RANGE

Option Mode	BAK	Min.	Typ.	Max.
250 KHz	1.5V	200 KHz	300 KHz	400 KHz
	3.0V	200 KHz	250 KHz	300 KHz
500 KHz	1.5V	450 KHz	600 KHz	750 KHz
	3.0V	400 KHz	500 KHz	600 KHz

ALLOWABLE OPERATING FREQUENCY

at Ta= -20℃ to 70℃, GND=0V

Condition	**Max, Operating Frequency**
BAK=1.5V (VDD1)	800 KHz
BAK=3V (VDD2)	4 MHz

ALLOWABLE OPERATING CONDITIONS

at Ta= -20℃ to 70℃, GND=0V

Name	Symb.	Condition	Min.	Max.	Unit
Supply Voltage	VDD1		1.2	5.25	V
	VDD2		2.4	5.25	
	VDD3		2.4	8.0	
Oscillator Start-Up Voltage	VDDB	Crystal Mode	1.3		
Oscillator Sustain Voltage	VDDB		1.2		
Supply Voltage	VDD1	Ag Mode	1.2	1.65	
	VDD2	EXT-V, Li Mode	2.4	5.25	
Input “H” Voltage	Vih1	Ag Battery Mode	VDD1-0.7	VDD1+0.7	
Input “L” Voltage	Vil1		-0.7	0.7	
Input “H” Voltage	Vih2	Li Battery Mode	VDD2-0.7	VDD2+0.7	
Input “L” Voltage	Vil2		-0.7	0.7	
Input “H” Voltage	Vih3	OSCIN at Ag Battery Mode	0.8xVDD1	VDD1	
Input “L” Voltage	Vil3		0	0.2xVDD1	
Input “H” Voltage	Vih4	OSCIN at Li Battery Mode	0.8xVDD2	VDD2	
Input “L” Voltage	Vil4		0	0.2xVDD2	
Input “H” Voltage	Vih5	CFIN at Li Battery or EXT-V Mode	0.8xVDD2	VDD2	
Input “L” Voltage	Vil5		0	0.2xVDD2	
Input “H” Voltage	Vih6	RC Mode	0.8xVDDO	VDDO	
Input “L” Voltage	Vil6		0	0.2xVDDO	
Operating Freq.	Fopg1	Crystal Mode	32		KHz
	Fopg2	RC Mode	10	1000	

ELECTRICAL CHARACTERISTICS

Input Resistance

at #1: VDD1=1.5 (Ag);

at #2: VDD2=3.0V (Li);

at #3: VDD2=5.0V (Ext-V);

Name	Symb.	Condition	Min.	Typ.	Max.	Unit
"L" Level Hold Tr. (IOC)	Rllh1	Vi=0.2VDD1, #1	10	40	100	KΩ
	Rllh2	Vi=0.2VDD2, #2				
	Rllh3	Vi=0.2VDD2, #3	5	20	50	
IOC Pull-Down Tr.	Rmad1	Vi=VDD1, #1	200	500	1000	
	Rmad2	Vi=VDD2, #2				
	Rmad3	Vi=VDD3, #3	100	250	500	
INT Pull-up Tr.	Rintu1	Vi=VDD1, #1	200	500	1000	
	Rintu2	Vi=VDD2, #2				
	Rintu3	Vi=VDD3, #3	100	250	500	
INT Pull-Down Tr.	Rintd1	Vi=GND, #1	200	500	1000	
	Rintd2	Vi=GND, #2				
	Rintd3	Vi=GND, #3	100	250	500	
RES Pull-Down R	Rres1	Vi=GND or VDD1, #1	50	80	100	
	Rres2	Vi=GND or VDD2, #2				
	Rres3	Vi=GND or VDD2, #3				

DC Output Characteristics

at #4: VDD1=1.2V (Ag);

at #5: VDD2=2.4V (Li);

at #6: VDD2=4V (Ext-V);

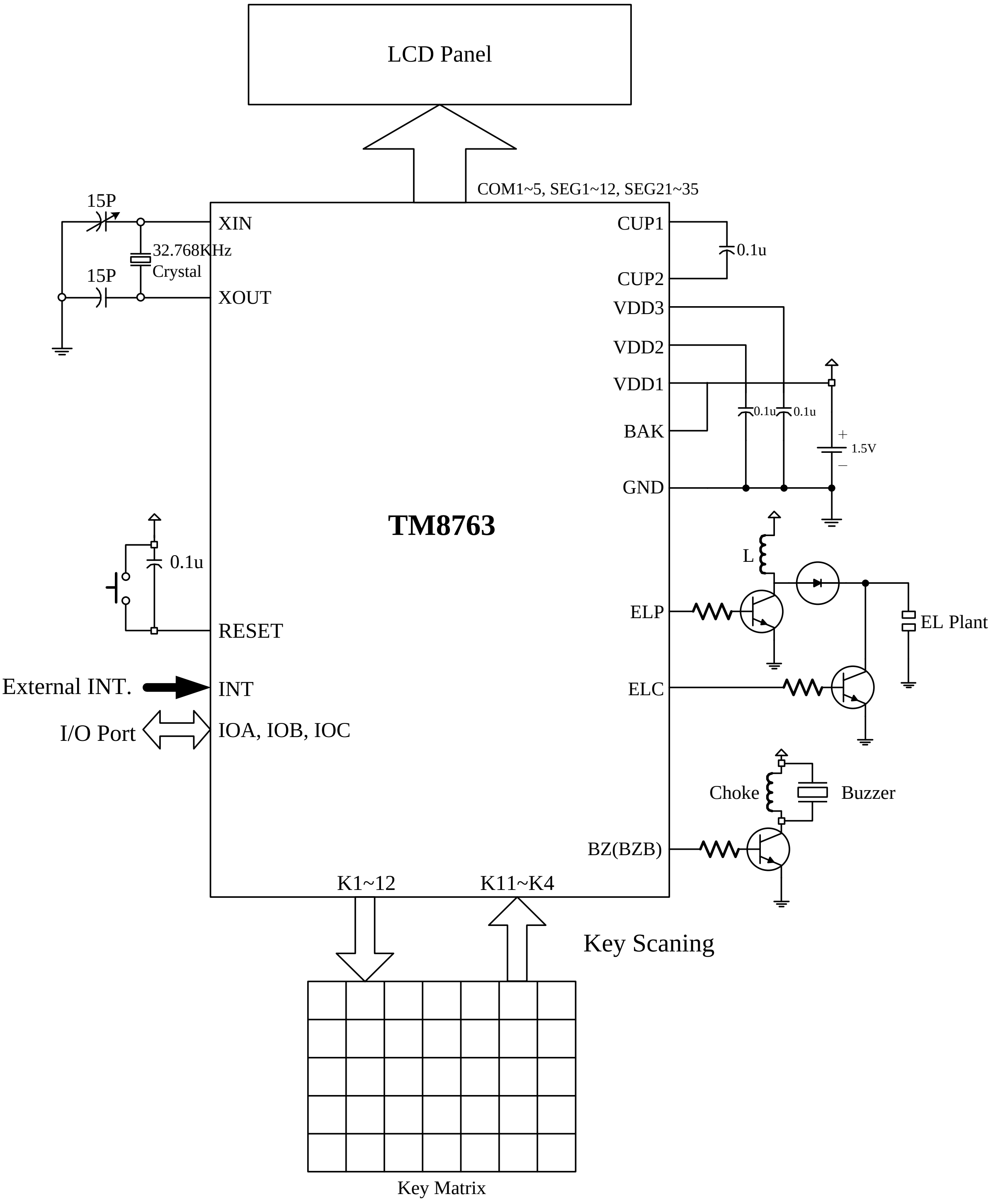
Name	Symb.	Condition	Port	Min.	Typ.	Max.	Unit
Output "H" Voltage	Voh1c	Ioh= -100uA, #1	SEG1~12 SEG21~35	1.0			V
	Voh2c	Ioh= -1mA, #2		2.0			
	Voh3c	Ioh= -3mA, #3		3.2			
Output "L" Voltage	Vol1c	Iol=200uA, #1				0.2	
	Vol2c	Iol=2mA, #2				0.4	
	Vol3c	Iol=6mA, #3				0.8	

Segment Driver Output Characteristics

Name	Symb.	Condition	For	Min.	Typ.	Max.	Unit
Static Display Mode							
Output ”H” Voltage	Voh1d	Ioh= -1uA, #1	SEG-n	2.2			V
	Voh2d	Ioh= -1uA, #2		2.2			
	Voh3d	Ioh= -1uA, #3		3.8			
Output ”L” Voltage	Vol1d	Iol=1uA, #1				0.2	
	Vol2d	Iol=1uA, #2				0.2	
	Vol3d	Iol=1uA, #3				0.2	
Output ”H” Voltage	Voh1e	Ioh= -10uA, #1	COM-n	2.2			
	Voh2e	Ioh= -10uA, #2		2.2			
	Voh3e	Ioh= -10uA, #3		3.8			
Output ”L” Voltage	Vol1e	Iol=10uA, #1				0.2	
	Vol2e	Iol=10uA, #2				0.2	
	Vol3e	Iol=10uA, #3				0.2	
1/2 Bias Display Mode							
Output ”H” Voltage	Voh12f	Ioh= -1uA, #1, #2	SEG-n	2.2			V
	Voh3f	Ioh= -1uA, #3		3.8			
Output ”L” Voltage	Vol12f	Iol=1uA, #1, #2				0.2	
	Vol3f	Iol=1uA, #3				0.2	
Output ”H” Voltage	Voh12g	Ioh= -10uA, #1, #2	COM-n	2.2			
	Voh3g	Ioh= -10uA, #3		3.8			
Output ”M” Voltage	Vom12g	Iol/h= +/-10uA, #1, #2		1.0		1.4	
	Vom3g	Iol/h= +/-10uA, #3		1.8		2.2	
Output ”L” Voltage	Vol12g	Iol=10uA,#1, #2				0.2	
	Vol3g	Iol=10uA, #3				0.2	
1/3 Bias display Mode							
Output ”H” Voltage	Voh12i	Ioh= -1uA, #1, #2	SEG-n	3.4			V
	Voh3i	Ioh= -1uA, #3		5.8			
Output ”M1” Voltage	Vom12i	Iol/h= +/-10uA, #1, #2		1.0		1.4	
	Vom13i	Iol/h= +/-10uA, #3		1.8		2.2	
Output ”M2” Voltage	Vom22i	Iol/h= +/-10uA, #1, #2		2.2		2.6	
	Vom23i	Iol/h= +/-10uA, #3		3.8		4.2	
Output ”L” Voltage	Vol12i	Iol=1uA, #1, #2				0.2	
	Vol3i	Iol=1uA, #3				0.2	
Output ”H” Voltage	Voh12j	Ioh= -10uA, #1, #2	COM-n	3.4			
	Voh3j	Ioh= -10uA, #3		5.8			
Output ”M1” Voltage	Vom12j	Iol/h= +/-10uA, #1, #2		1.0		1.4	
	Vom13j	Iol/h= +/-10uA, #3		1.8		2.2	
Output ”M2” Voltage	Vom22j	Iol/h= +/-10uA, #1, #2		2.2		2.6	
	Vom23j	Iol/h= +/-10uA, #3		3.8		4.2	
Output ”L” Voltage	Vol12j	Iol=10uA, #1, #2				0.2	
	Vol3j	Iol=10uA, #3				0.2	

1.8 TYPICAL APPLICATION CIRCUIT

This application circuit is simply an example, and is not guaranteed to work.



Ag power mode, 1/3Bias, 1/5Duty

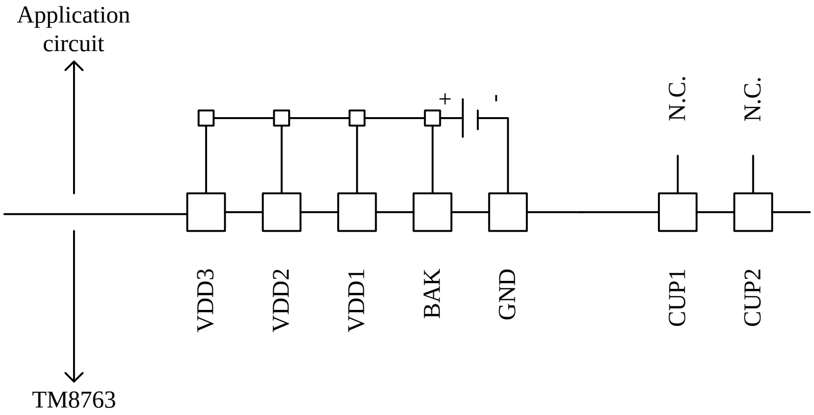
2. TM8763 Internal System Architecture

2.1 Power Supply

TM8763 could operate at Ag, Li, and EXTV 3 types supply voltage, all of these operating types are defined by mask option. The power supply circuitry also generated the necessary voltage level to drive the LCD panel with different bias. Shown below are the connection diagrams for 1/2 bias,1/3 bias and no bias application.

2.1.1 Ag BATTERY POWER SUPPLY

Operating voltage range : 1.2V ~ 1.8V.



For different LCD bias application, the connection diagrams are shown below:

2.1.1.1 NO LCD BIAS NEED AT Ag BATTERY POWER SUPPLY

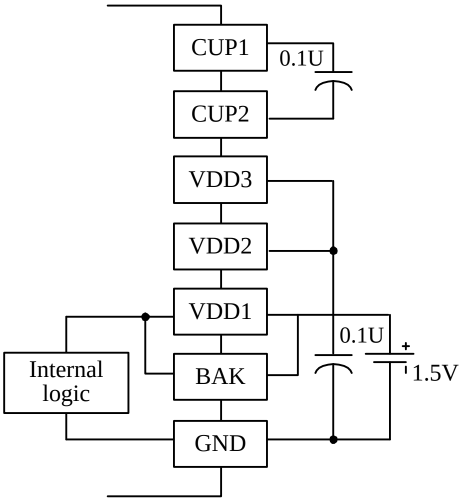
MASK OPTION table:

Mask Option name	Selected item
POWER SOURCE	(3) 1.5V BATTERY
LCD BIAS	(3) NO BIAS

Note 1: The input/output ports operate between GND and VDD1.

Note 2: At the initial clear mode the backup flag (BCF) is set. When the backup flag is set, the oscillator circuit becomes large in inverter size and the oscillation conditions are improved, but the operating current is also increased. Therefore, the backup flag must be reset unless otherwise required. For the backup flag, refer to 3-5.

2.1.1.2 1/2 BIAS AT AG BATTERY POWER SUPPLY



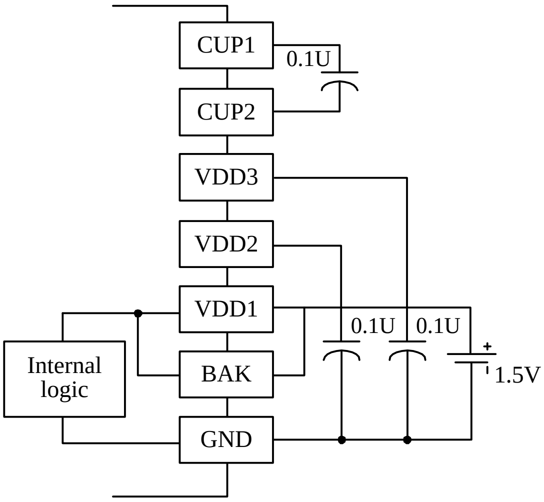
MASK OPTION table:

Mask Option name	Selected item
POWER SOURCE	(3) 1.5V BATTERY
LCD BIAS	(2) 1/2 BIAS

Note 1: The input/output ports operate between GND and VDD1.

Note 2: At the initial clear mode the backup flag (BCF) is set. When the backup flag is set, the oscillator circuit becomes large in inverter size and the oscillation conditions are improved, but the operating current is also increased. Therefore, the backup flag must be reset unless otherwise required. For the backup flag, refer to 3-5.

2.1.1.3 1/3 BIAS AT AG BATTERY POWER SUPPLY



MASK OPTION table:

Mask Option name	Selected item
POWER SOURCE	(3) 1.5V BATTERY
LCD BIAS	(1) 1/3 BIAS

Note 1: The input/output ports operate between GND and VDD1.

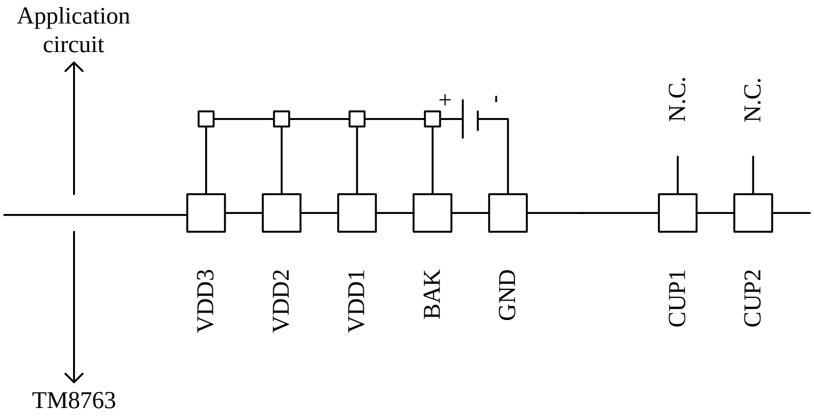
Note 2: At the initial clear mode the backup flag (BCF) is set. When the backup flag is set, the oscillator circuit becomes large in inverter size and the oscillation conditions are improved, but the operating current is also increased. Therefore, the backup flag must be reset unless otherwise required. For the backup flag, refer to 3-5.

2.1.2 LI BATTERY POWER SUPPLY

Operating voltage range: 2.4V ~ 3.6V.

For different LCD bias application, the connection diagrams are shown below:

2.1.2.1 NO BIAS AT LI BATTERY POWER SUPPLY



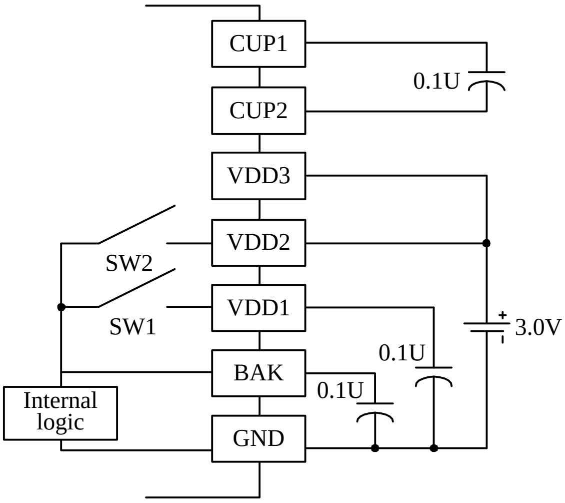
MASK OPTION table:

Mask Option name	Selected item
POWER SOURCE	(2) 3V BATTERY OR HIGHER
LCD BIAS	(3) NO BIAS

Note 1: The input/output ports operate between GND and VDD2.

2.1.2.2 1/2 BIAS AT LI BATTERY POWER SUPPLY

The backup flag (BCF) must be reset after the operation of the halver circuit is fully. stabilized and a voltage of approximately 1/2 * VDD2 appears on the VDD1 pin.



Backup flag (BCF)	SW1	SW2
BCF=0	ON	OFF
BCF=1	OFF	ON

MASK OPTION table:

Mask Option name	Selected item
POWER SOURCE	(2) 3V BATTERY OR HIGHER
LCD BIAS	(2) 1/2 BIAS

Note 1: The input/output ports operate between GND and VDD2.

Note 2: At the initial clear mode, the backup flag (BCF) is set. When the backup flag is set, the internal logic operated on VDD2 and the oscillator circuit becomes large in driver size.

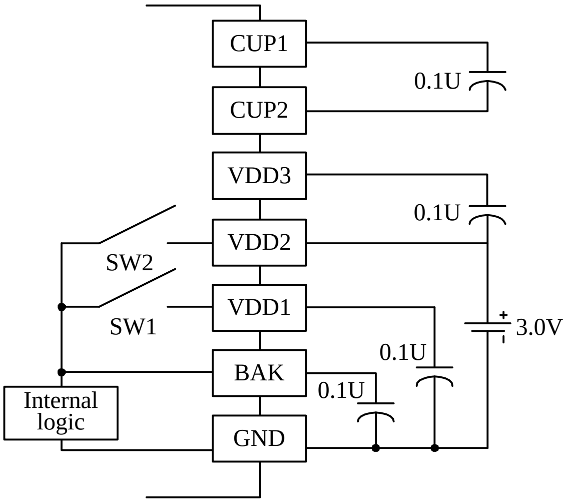
At the backup flag set mode, the operating current is increased. Therefore, the backup flag must be reset unless otherwise required. For the backup flag, refer to 3-5.

Note 3: The VDD1 level (≈1/2 * VDD2) at the off-state of SW1 is used as an intermediate voltage level for the LCD driver.

2.1.2.3 1/3 BIAS AT LI BATTERY POWER SUPPLY

The backup flag (BCF) must be reset after the operation of the halver circuit is fully stabilized and a voltage of approximately 1/2 * VDD2 appears on the VDD1 pin.

Backup flag (BCF)	SW1	SW2
BCF=0	ON	OFF
BCF=1	OFF	ON



MASK OPTION table:

Mask Option name	Selected item
POWER SOURCE	(2) 3V BATTERY OR HIGHER
LCD BIAS	(1) 1/3 BIAS

Note 1: The input/output ports operate between GND and VDD2.

Note 2: At the initial clear mode the backup flag (BCF) is set. When the backup flag is set, the internal logic operated on VDD2 and the oscillator circuit becomes large in inverter size.
At the backup flag set mode the operating current is increased. Therefore, the backup flag must be reset unless otherwise required. For the backup flag, refer to 3-5.

Note 3: The VDD1 level (≈1/2*VDD) at the off-state of SW1 is used as an intermediate voltage level for LCD driver.

2.1.3 EXTV POWER SUPPLY

Operating voltage range: 3.6V ~ 5.4V.

For different LCD bias application, the connection diagrams are shown below:

2.1.3.1 NO BIAS AT EXT-V BATTERY POWER SUPPLY

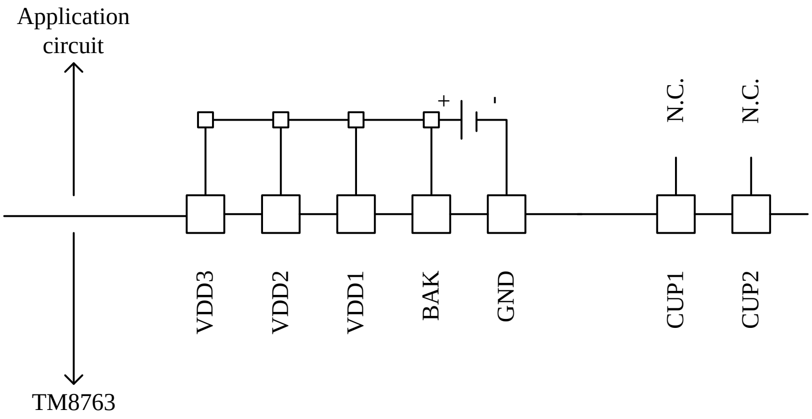
MASK OPTION table:

Mask Option name	Selected item
POWER SOURCE	(1) EXT-V
LCD BIAS	(3) NO BIAS

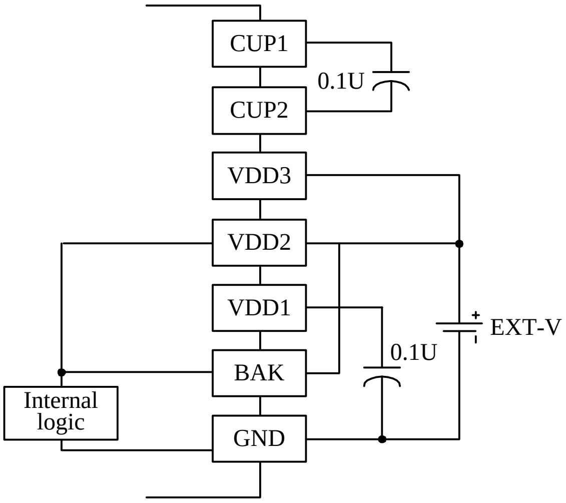
Note 1: The input/output ports operate between GND and VDD2.

Note 2: At the initial clear mode the backup flag (BCF) is reset.

Note 3: At the backup flag set mode the operating current is increased.



2.1.3.2 1/2 BIAS AT EXT-V POWER SUPPLY



MASK OPTION table:

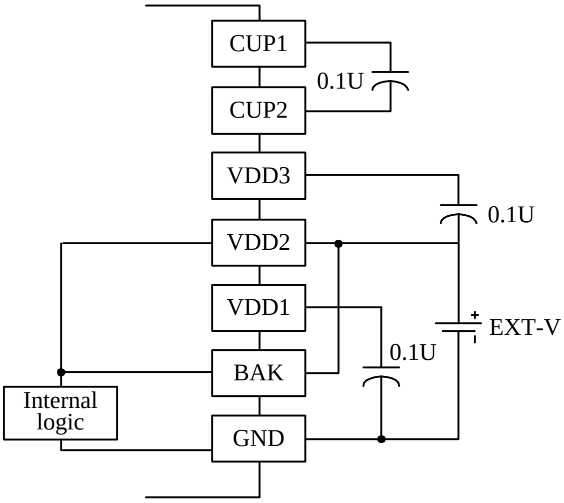
Mask Option name	Selected item
POWER SOURCE	(1) EXT-V
LCD BIAS	(2) 1/2 BIAS

Note 1: The input/output ports operate between GND and VDD2.

Note 2: At the initial clear mode the backup flag (BCF) is reset.

Note 3: At the backup flag set mode the operating current is increased. Therefore, the backup flag must be reset unless otherwise required.

2.1.3.3 1/3 BIAS AT EXT-V POWER SUPPLY



MASK OPTION table:

Mask Option name	Selected item
POWER SOURCE	(1) EXT-V
LCD BIAS	(1) 1/3 BIAS

Note 1: The input/output ports operate between GND and VDD2.

Note 2: At the initial clear mode the backup flag (BCF) is reset.

Note 3: At the backup flag set mode the operating current is increased. Therefore, the backup flag must be reset unless otherwise required.

2.2 SYSTEM CLOCK

XT clock (slow clock oscillator) and CF clock (fast clock oscillator) compose the clock oscillation circuitry and the block diagram is shown below.

The system clock generator provided the necessary clocks for execution of instruction. The pre-divider generated several clocks with different frequencies for the usage of LCD driver, frequency generator … etc.

The following table shows the clock sources of system clock generator and pre-divider in different conditions.

	PH0	BCLK
Slow clock only option	XT clock	XT clock
fast clock only option	CF clock	CF clock
Initial state(dual clock option)	XT clock	XT clock
Halt mode(dual clock option)	XT clock	XT clock
Slow mode(dual clock option)	XT clock	XT clock
Fast mode(dual clock option)	XT clock	CF clock

2.2.1 CONNECTION DIAGRAM OF SLOW CLOCK OSCILLATOR (XT CLOCK)

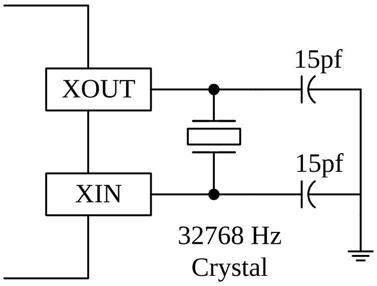
This clock oscillation circuitry provides the lower speed clock to the system clock generator, pre-divider, timer, chattering prevention of IO port and LCD circuitry. This oscillator will be disabled when the fast clock only option is selected by mask option, or it will be active all the time after the initial reset. In stop mode, this oscillator will be stopped.

There are 2 type oscillators can be used in slow clock oscillator, selected by mask option:

2.2.1.1 External 32.768KHz Crystal oscillator (XT CLOCK)

MASK OPTION table:

Mask Option name	Selected item
SLOW CLOCK TYPE FOR SLOW ONLY OR DUAL	(1) X’tal



(1) X'tal

When backup flag (BCF) is set to 1, the oscillator operates with an extra buffer in parallel in order to shorten the oscillator start-up time but this will increase the power consumption. Therefore, the backup flag should be reset unless required otherwise.

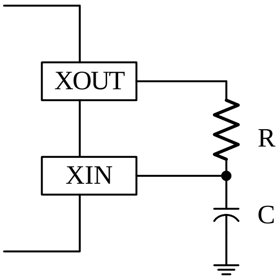
The following table shows the power consumption of Crystal oscillator in different conditions:

	Ag power option	Li power option	EXT-V option
BCF=1	Increased	Increased	Increased
BCF=0	Normal	Normal	Increased
Initial reset	Increased	Increased	Increased
After reset	Increased	Increased	Increased

2.2.1.2 External RC oscillator (XT CLOCK)

MASK OPTION table:

Mask Option name	Selected item
SLOW CLOCK TYPE FOR SLOW ONLY OR DUAL	(2) RC



(2) RC

2.2.2 CONNECTION DIAGRAM OF FAST CLOCK OSCILLATOR (CF CLOCK)

The CF clock is a multiple type oscillator (mask option) which provide a faster clock source to system. In single clock operation (fast only), this oscillator will provide the clock to the system clock generator, pre-divider, timer, I/O port chattering prevention clock and LCD circuitry. In dual clock operation, CF clock provides the clock to system clock generator only.

When the dual clock option is selected by mask option, this oscillator will be inactive most of the time except when the FAST instruction is executed. After the FAST instruction is executed, the clock source (BCLK) of the system clock generator will be switched to CF clock and the clock source for other functions will still come from XT clock. Halt mode, stop mode or SLOW instruction execution will stop this oscillator and the system clock (BCLK) will be switched to XT clock.

There are 3 type oscillators can be used in slow clock oscillator, selected by mask option:

2.2.2.1 RC OSCILLATOR WITH EXTERNAL RESISTOR (CF CLOCK)

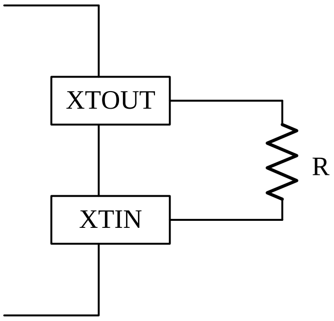
This kind of oscillator could only be used in "FAST only" option, the fast clock source of dual clock mode can't use this oscillator. When this oscillator is used, the frequency option of the RC oscillator with internal RC is not cared.

MASK OPTION table:

Mask Option name	Selected item
CLOCK SOURCE	(2) FAST ONLY & USE EXTERNAL RESISTOR

MASK OPTION table:

Mask Option name	Selected item
FAST CLOCK OSC TYPE FOR FAST ONLY OR DUAL	(1) or (2), don't care



External Resistor

2.2.2.2 RC OSCILLATOR WITH INTERNAL RESISTOR (CF CLOCK)

Two kinds of the frequencies could be selected in this mode of oscillator, the one is 250 KHz and the other is 500KHz. When this oscillator is used, leave CFOUT and CFIN two pins opened.

This kind of oscillator could be used in "FAST only" or "DUAL clock" options.

MASK OPTION table:

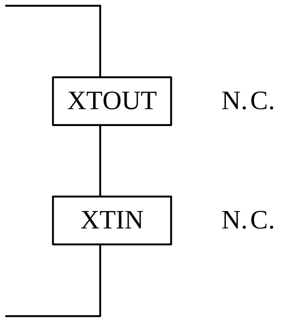
Mask Option name	Selected item
CLOCK SOURCE	(1) FAST ONLY & USE INTERNAL RESISTOR or (4) DUAL

For 250KHz output frequency:

Mask Option name	Selected item
FAST CLOCK OSC TYPE FOR FAST ONLY OR DUAL	(1) INTERNAL RESISTOR FOR 250 KHz

For 500KHz output frequency:

Mask Option name	Selected item
FAST CLOCK OSC TYPE FOR FAST ONLY OR DUAL	(2) INTERNAL RESISTOR FOR 500 KHz



Internal RC

FREQUENCY RANGE OF INTERNAL RC OSCILLATOR

Option Mode	BAK	Min.	Typ.	Max.
250 KHz	1.5V	200 KHz	300 KHz	400 KHz
	3.0V	200 KHz	250 KHz	300 KHz
500 KHz	1.5V	450 KHz	600 KHz	750 KHz
	3.0V	400 KHz	500 KHz	600 KHz

2.2.3 COMBINATION OF THE CLOCK SOURCES

There are three types of combination of the clock sources that can be selected by mask option:

2.2.3.1 DUAL CLOCK

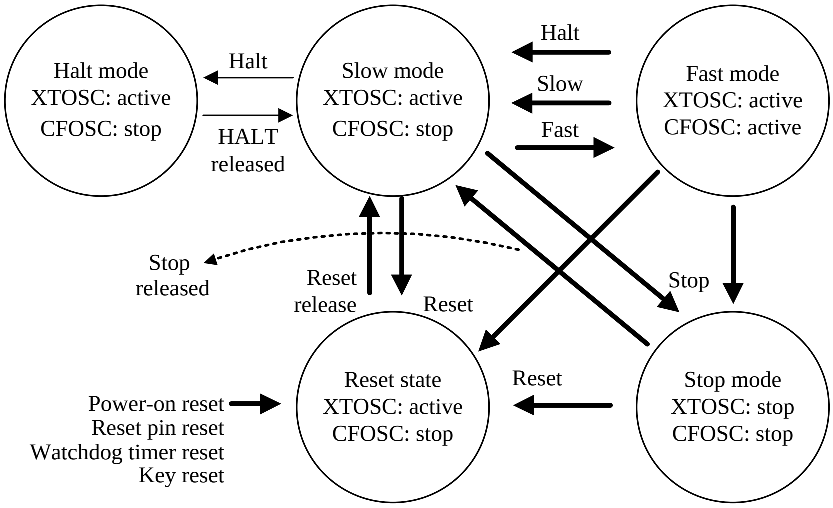
MASK OPTION table:

Mask Option name	Selected item
CLOCK SOURCE	(4) DUAL

The operation of the dual clock option is shown in the following figure.

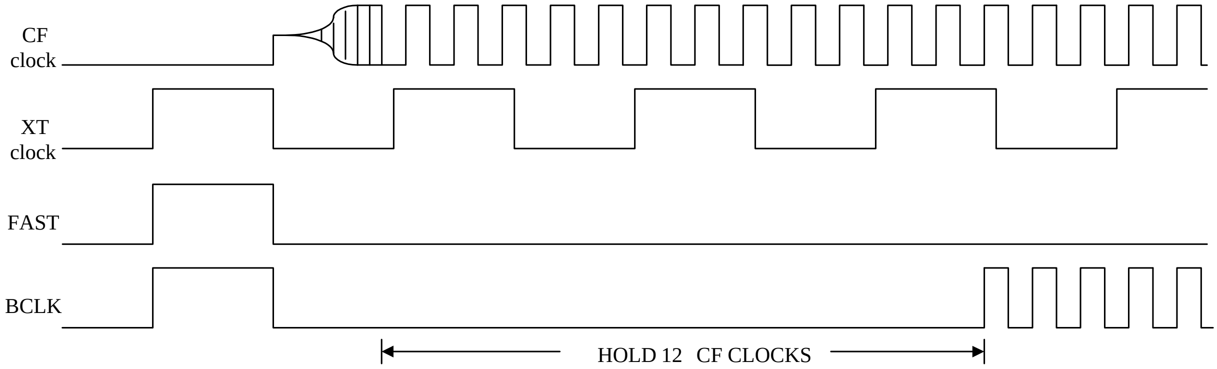
When this option is selected by mask option, the clock source (BCLK) of system clock generator will switch between XT clock and CF clock according to the user’s program. When the halt and stop instructions are executed, the clock source (BCLK) will switch to XT clock automatically.

The XT clock provides the clock to the pre-divider, timer, I/O port chattering prevention and LCD circuitry in this option.



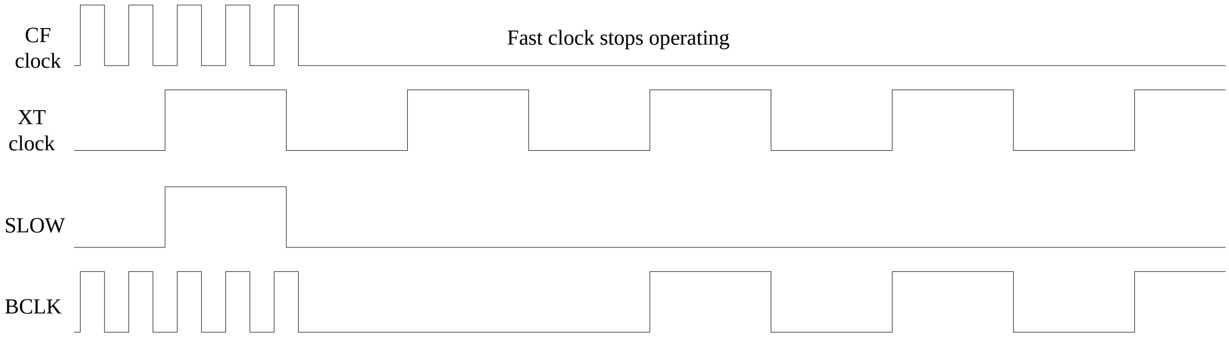
State Diagram of Dual Clock Option was shown on above figure.

After executing FAST instruction, the system clock generator will hold 12 CF clocks after the CF clock oscillator starts up and then switches CF clock to BCLK. This will prevent the incorrect clock from delivering to the system clock in the start-up duration of the fast clock oscillator.



This figure shows the System Clock Switches from Slow to Fast

After executing SLOW instruction, the system clock generator will hold 2 XT clocks and then switches XT clock to BCLK.



This figure shows the System Clock Switches from Fast to Slow

2.2.3.2 SINGLE CLOCK

MASK OPTION table:

For Fast clock oscillator only

Mask Option name	Selected item
CLOCK SOURCE	(1) FAST ONLY & USE INTERANL RESISTOR or (2) FAST ONLY & USE EXTERANL RESISTOR

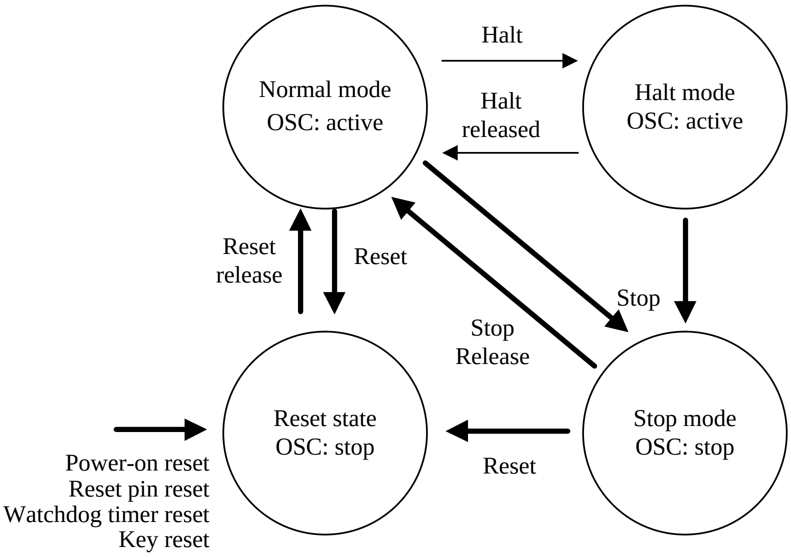
For slow clock oscillator only

Mask Option name	Selected item
CLOCK SOURCE	(3) SLOW ONLY

The operation of the single clock option is shown in the following figure.

Either XT or CF clock may be selected by mask option in this mode. The FAST and SLOW instructions will perform as the NOP instruction in this option.

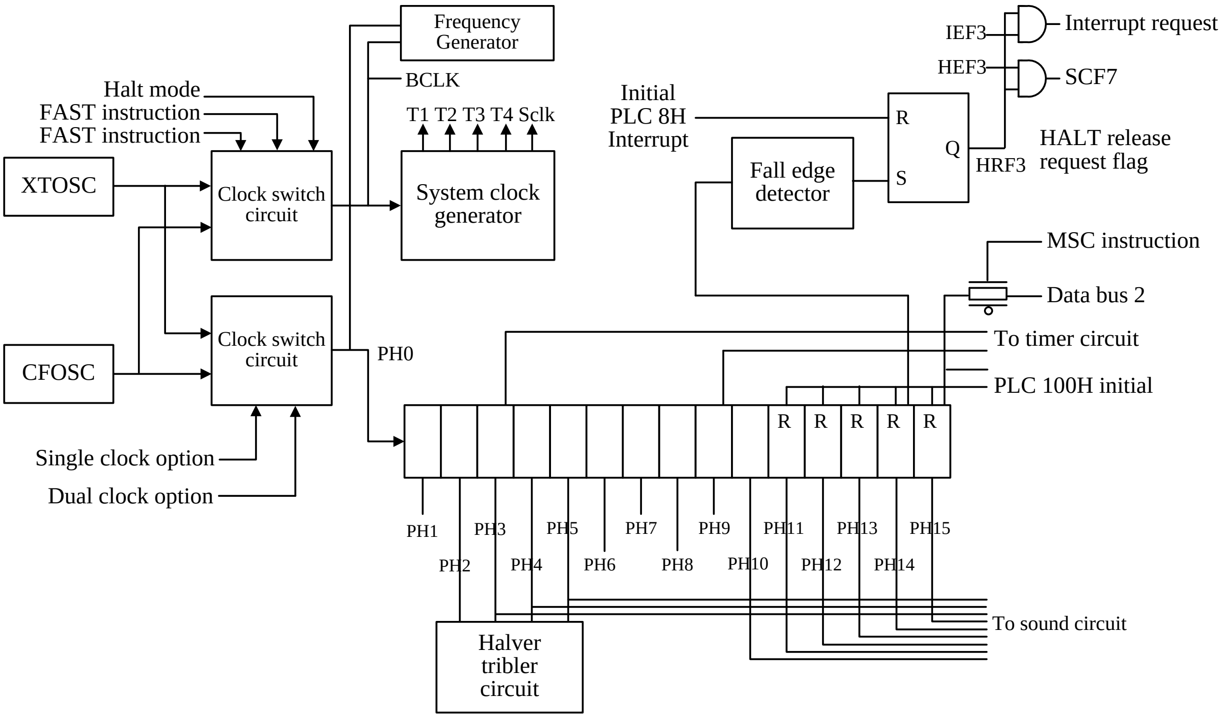
The backup flag (BCF) will be set to 1 automatically before the program enters the stop mode. This could ensure the Crystal oscillator would start up in a better condition.



This figure shows the State Diagram of Single Clock Option

2.2.4 PREDIVIDER

The pre-divider is a 15-stage counter that receives the clock from the output of clock switch circuitry (PH0) as input. When PH0 is changed from "H" level to "L" level, the content of this counter changes. The PH11 to PH15 of the pre-divider are reset to "0" when the PLC 100H instruction is executed or at the initial reset mode. The pre-divider delivers the signal to the halver / tripler circuit, alternating frequency for LCD display, system clock, sound generator and halt release request signal (I/O port chattering prevention clock).



This figure shows the Pre-divider and its Peripherals

The PH14 delivers the halt mode release request signal, setting the halt mode release request flag (HRF3). In this case, if the pre-divider interrupt enable mode (IEF3) is provided, the interrupt is accepted; and if the halt release enable mode (HEF3) is provided, the halt release request signal is delivered, setting the start condition flag 7 (SCF7) in status register 3 (STS3).

The clock source of pre-divider is PH0, and 4 kinds of frequency of PH0 could be selected by mask option:

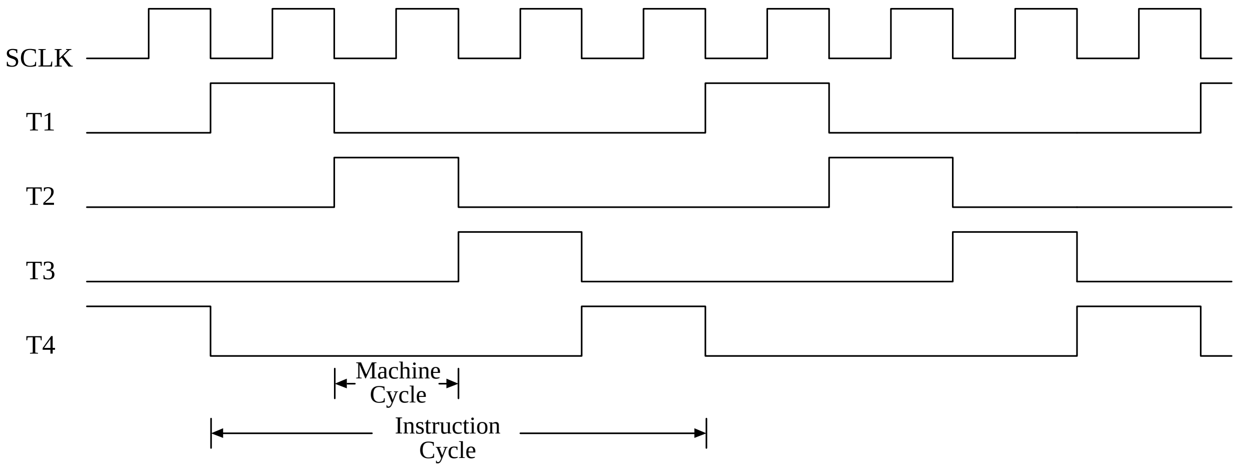
MASK OPTION table:

Mask Option name	Selected item
PH0 <-> BCLK FOR FAST ONLY	(1) PH0=BCLK
PH0 <-> BCLK FOR FAST ONLY	(2) PH0=BCLK/4
PH0 <-> BCLK FOR FAST ONLY	(3) PH0=BCLK/8
PH0 <-> BCLK FOR FAST ONLY	(4) PH0=BCLK/16

2.2.5 SYSTEM CLOCK GENERATOR

For the system clock, the clock switch circuit permits the different clock input from XTOSC and CFOSC to be selected. The FAST and SLOW instructions can switch the clock input of the system clock generator (SGC).

The basic system clock is shown below:



2.3 PROGRAM COUNTER (PC)

This is an 11-bit counter, which addresses the program memory (ROM) up to 1536 addresses.

- The program counter (PC) is normally increased by one (+1) with every instruction execution.

  PC 🡐 PC + 1

- When executing JMP instruction, subroutine call instruction (CALL), interrupt service routine or reset occurs, the program counter (PC) loads the specified address corresponding to table 2-1.

  PC 🡐 specified address shows in Table 2- 1

- When executing a jump instruction except JMP and CALL, the program counter (PC) loads the specified address in the operand of instruction.

  PC 🡐 specified address in operand

- Return instruction (RTS)

  PC 🡐 content of stack specified by the stack pointer

  Stack pointer 🡐 stack pointer – 1

Table 2- 1

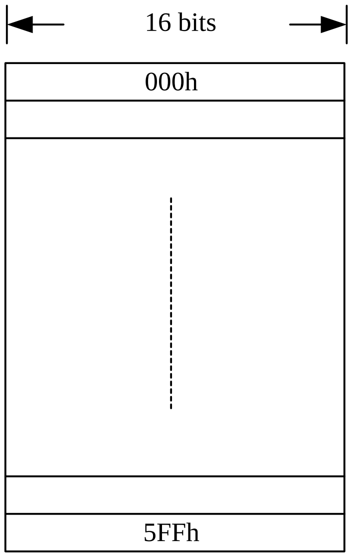
	PC10	PC9	PC8	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
Initial reset	0	0	0	0	0	0	0	0	0	0	0
Interrupt 2 (INT pin)	0	0	0	0	0	0	1	0	0	0	0
Interrupt 0 (input port C)	0	0	0	0	0	0	1	0	1	0	0
Interrupt 1 (timer 1 interrupt)	0	0	0	0	0	0	1	1	0	0	0
Interrupt 3 (pre-divider interrupt)	0	0	0	0	0	0	1	1	1	0	0
Interrupt 5 (Key Scanning interrupt)	0	0	0	0	0	1	0	0	1	0	0
Jump 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

P10 to P0 : Low-order 11 bits of instruction operand.

When executing the subroutine call instruction or interrupt service routine, the contents of the program counter (PC) are automatically saved to the stack register (STACK).

2.4 PROGRAM/TABLE MEMORY

The built-in mask ROM is organized with 1536 x 16 bits.



Both instruction ROM (PROM) and table ROM (TROM) shares this memory space together. The partition formula for PROM and TROM is shown below:

Instruction ROM memory space= (128 * N) words,

Table ROM memory space=256 (16 - N) bytes (N=1~12).

Note: The data width of table ROM is 8-bit

The partition of memory space is defined by mask option, the table is shown below:

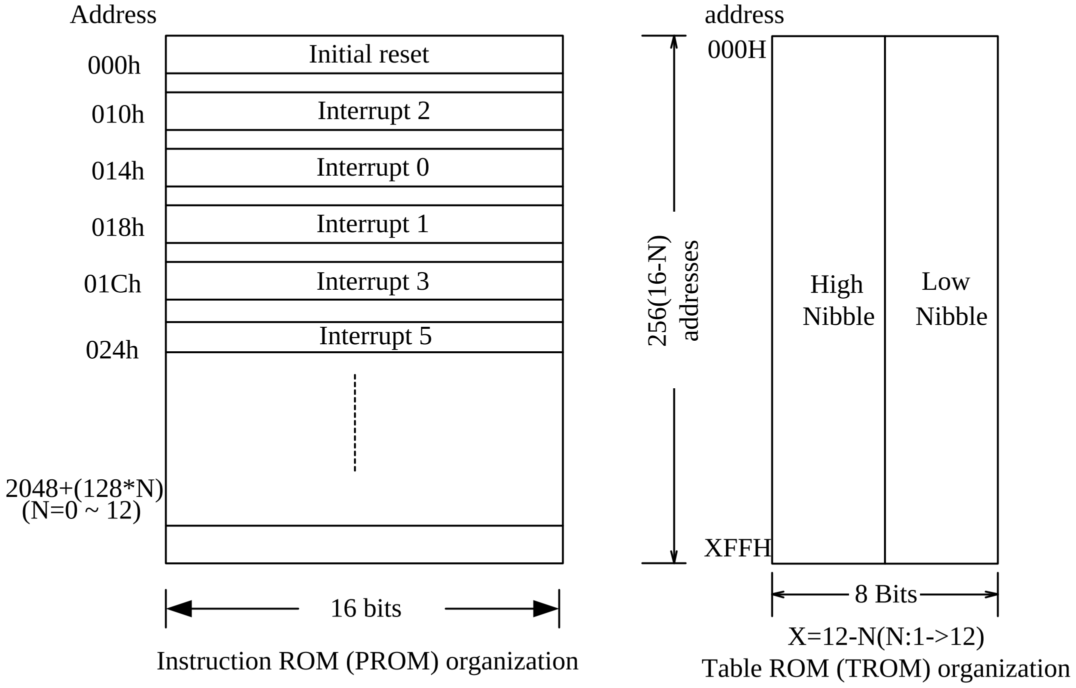
MASK OPTION table:

Mask Option name	Selected item	Instruction ROM memory space (Words)	Table ROM memory space (Bytes)
INSTRUCTION ROM <-> TABLE ROM	1 (N=1)	128	2816
	2 (N=2)	256	2560
	3 (N=3)	384	2304
	4 (N=4)	512	2048
	5 (N=5)	640	1792
	6 (N=6)	768	1536
	7 (N=7)	896	1280
	8 (N=8)	1024	1024
	9 (N=9)	1152	768
	A (N=10)	1280	512
	B (N=11)	1408	256
	C (N=12)	1536	0

2.4.1 INSTRUCTION ROM (PROM)

There are some special locations that serve as the interrupt service routines, such as reset address (000H), interrupt 0 address (014H), interrupt 1 address (018H), interrupt 2 address (010H), interrupt 3 address (01CH), interrupt 5 address (024H), and interrupt 6 address (028H) in the program memory.

This figure shows the Organization of ROM



This figure shows the Organization of ROM

2.4.2 TABLE ROM (TROM)

The table ROM is organized with 256(12-N) x 8 bits that shared the memory space with instruction ROM, as shown in the figure above. This memory space stores the constant data or look up table for the usage of main program. All of the table ROM addresses are specified by the index address register (@HL). The data width could be 8 bits (256 (12-N) x 8 bits) or 4 bits (512 (12-N) x 4 bits) which depends on the different usage. Refer to the explanation of instruction chapter.

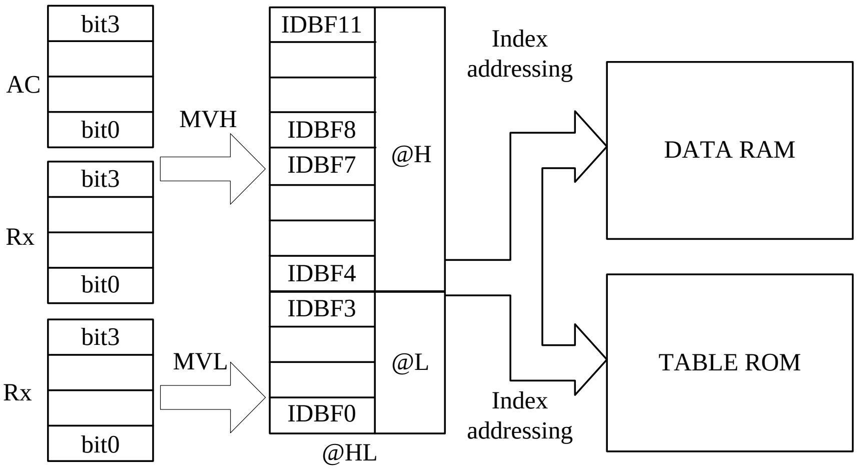
2.5 INDEX ADDRESS REGISTER (@HL)

This is a versatile address pointer for the data memory (RAM) and table ROM (TROM). The index address register (@HL) is a 12-bit register, and the contents of the register can be modified by executing MVH and MVL instructions. Executed MVL instruction will load the content of specified data memory to the lower nibble of the index register (@L). In the same manner, executed MVH instructions may load the contents of the data RAM (Rx) and AC into the higher nibble of the register @H.

@L is a 4-bit register and @H is an 8-bit register.

@H register				@H register				@L register			
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Bit3	Bit2	Bit1	Bit0
IDBF11	IDBF10	IDBF9	IDBF8	IDBF7	IDBF6	IDBF5	IDBF4	IDBF3	IDBF2	IDBF1	IDBF0

The index address register can specify the full range addresses of the table ROM and data memory.



This figure shows the diagram of the index address register

2.6 STACK REGISTER (STACK)

Stack is a special design register following the first-in-last-out rule. It is used to save the contents of the program counter sequentially during subroutine call or execution of the interrupt service routine.

The contents of stack register are returned sequentially to the program counter (PC) while executing return instructions (RTS).

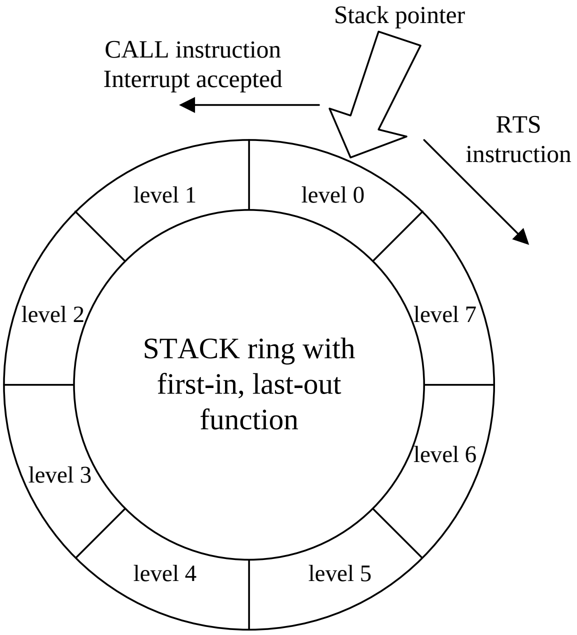
The stack register is organized using 11 bits by 8 levels but with no overflow flag; hence only 8 levels of subroutine call or interrupt are allowed (If the stacks are full, and either interrupt occurs or subroutine call executes, the first level will be overwritten).

Once the subroutine call or interrupt causes the stack register (STACK) overflow, the stack pointer will return to 0 and the content of the level 0 stack will be overwritten by the PC value.

The contents of the stack register (STACK) are returned sequentially to the program counter (PC) during execution of the RTS instruction.

Once the RTS instruction causes the stack register (STACK) underflow, the stack pointer will return to level 7 and the content of the level 7 stack will be restored to the program counter.

The following figure shows the diagram of the stack.



2.7 DATA MEMORY (RAM)

The static RAM is organized with 96 addresses x 4 bits and is used to store data. The address range of data memory is from 00h to 7Fh, but addresses between 50h to 6Fh are not reachable.

The data memory may be accessed using two methods:

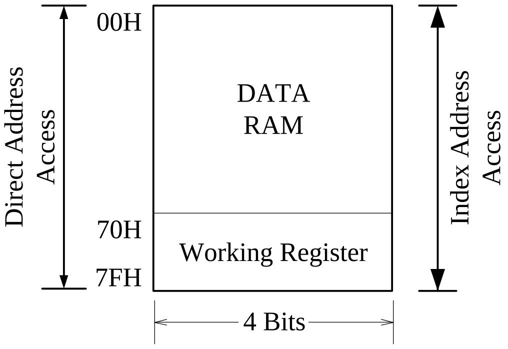
1. **Direct addressing mode**

   The address of the data memory is specified by the instruction and the addressing range is from 00H to 7FH. (Addresses between 50h to 6Fh are not reachable)

2. **Index addressing mode**

   The index address register (@HL) specifies the address of the data memory and all address space from 00H to 1FFH can be accessed. (Addresses between 50h to 6Fh are not reachable)

   The 16 specified addresses (70H to 7FH) in the direct addressing memory are also used as 16 working registers. The function of working register will be described in detail in section 2-6.



This figure shows the Data Memory (RAM) and Working Register Organization

2.8 WORKING REGISTER (WR)

The locations 70H to 7FH of the data memory (RAM) are not only used as general-purpose data memory but also as the working register (WR). The following will introduce the general usage of working registers:

1. Be used to perform operations on the contents of the working register and immediate data. Such as : ADCI, ADCI*, SBCI, SBCI*, ADDI, ADDI*, SUBI, SUBI*, ADNI, ADNI*, ANDI, ANDI*, EORI, EORI*, ORI, ORI*

2. Be transferred the data between the working register and any address in the direct addressing data memory (RAM). Such as:

   MWR Rx, Ry; MRW Ry, Rx

3. Decode (or directly transfer) the contents of the working register and output to the LCD PLA circuit. Such as:

   LCT, LCB, LCP

2.9 ACCUMULATOR (AC)

The accumulator (AC) is a register that plays the most important role in operations and controls. By using it in conjunction with the ALU (Arithmetic and Logic Unit), data transfer between the accumulator and other registers or data memory can be performed.

2.10 ALU (Arithmetic and Logic Unit)

This is a circuitry that performs arithmetic and logic operation. The ALU provides the following functions:

Binary addition/subtraction	(INC, DEC, ADC, SBC, ADD, SUB, ADN, ADCI, SBUI, ADNI)
Logic operation	(AND, EOR, OR, ANDI, EORI, ORI)
Shift	(SR0, SR1, SL0, SL1)
Decision	(JB0, JB1, JB2, JB3, JC, JNC, JZ, and JNZ)
BCD operation	(DAA, DAS)

2.11 HEXADECIMAL CONVERT TO DECIMAL (HCD)

Decimal format is another number format for TM8763. When the content of the data memory has been assigned as decimal format, it is necessary to convert the results to decimal format after the execution of ALU instructions. When the decimal converting operation is processing, all of the operand data (including the contents of the data memory (RAM), accumulator (AC), immediate data, and look-up table) should be in the decimal format, or the results of conversion will be incorrect.

Instructions DAA, DAA*, DAA @HL can convert the data from hexadecimal to decimal format after any addition operation. The conversion rules are shown in the following table and illustrated in example 1.

AC data before DAA execution	CF data before DAA execution	AC data after DAA execution	CF data after DAA execution
0 ≤ AC ≤ 9	CF=0	no change	no change
A ≤ AC ≤ F	CF=0	AC=AC+6	CF=1
0 ≤ AC ≤ 3	CF=1	AC=AC+6	no change

Example 1:

```
LDS     10h, 9          ; Load immediate data"9"to data memory address 10H.
LDS     11h, 1          ; Load immediate data"1"to data memory address 11H
                        ; and AC.
RF      1h              ; Reset CF to 0.
ADD*    10h             ; Contents of the data memory address 10H and AC are
                        ; binary-added; the result loads to AC & data memory address
                        ; 10H. (R10=AC=AH, CF=0)
DAA*    10h             ; Convert the content of AC to
                        ; decimal format.
                        ; The result in the data memory address 10H is"0"and in
                        ; the CF is "1". This represents the decimal number"10".
```

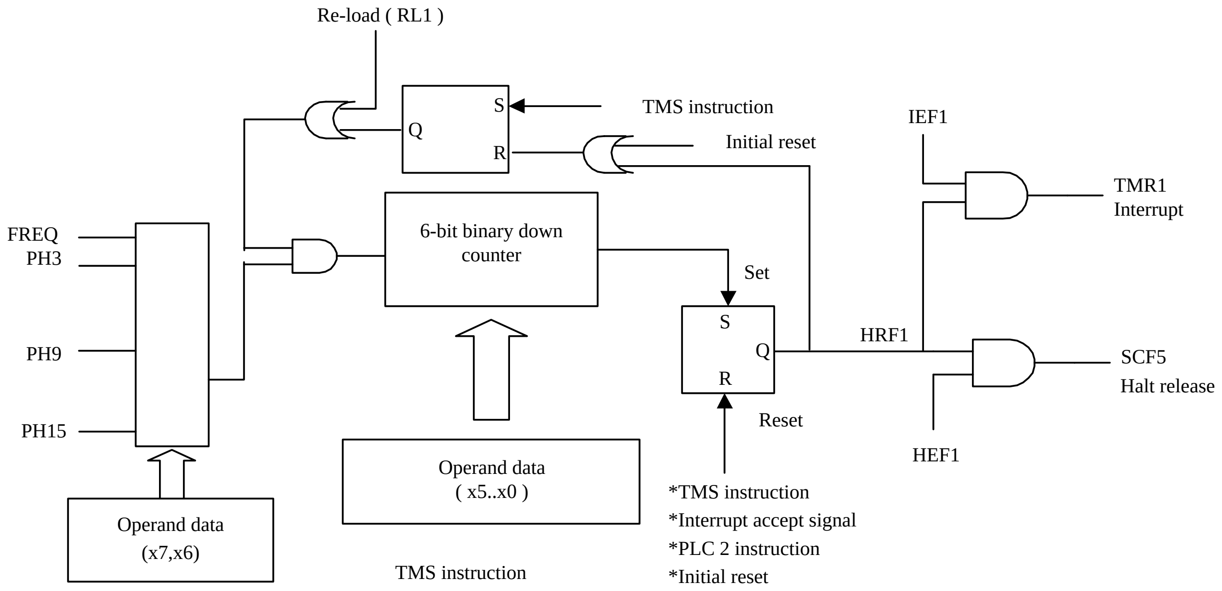
Instructions DAS, DAS*, DAS @HL can convert the data from hexadecimal format to decimal format after any subtraction operation. The conversion rules are shown in the following table and illustrated in Example 2.

AC data before DAS execution	CF data before DAS execution	AC data after DAS execution	CF data after DAS execution
0 ≤ AC ≤ 9	CF=1	No change	no change
6 ≤ AC ≤ F	CF=0	AC=AC+A	no change

Example 2:

```
LDS     10h, 1          ; Load immediate data”1”to the data memory address 10H.
LDS     11h, 2          ; Load immediate data”2”to the data memory address 11H and AC.
SF      1h              ; Set CF to 1, which means no borrowing has occurred.
SUB*    10h             ; Content of data memory address 10H is binary-subtracted;
                        ; the result loads to data memory address
                        ; 10H. (R10=AC=FH, CF=0)
DAS*    10h             ; Convert the content of the data memory address 10H to decimal format.
                        ; The result in the data memory address 10H is”9”and in
                        ; the CF is “0”. This represents the decimal number”–1”.
```

2.12 TIMER 1 (TMR1)



This figure shows the TMR1 organization.

2.12.1 NORMAL OPERATION

TMR1 consists of a programmable 6-bit binary down counter, which is loaded and enabled by executing TMS or TMSX instruction.

Once the TMR1 counts down to 3Fh, it generates an underflow signal to set the halt release request flag1 (HRF1) to 1 and then stop to count down.

When HRF1=1, and the TMR1 interrupt enable flag (IEF1) =1, the interrupt is generated.

When HRF1=1, if the IEF1=0 and the TMR1 halt release enable (HEF1) =1, program will escapes from halt mode (if CPU is in halt mode) and then set the start condition flag 5 (SCF5) to 1 in the status register 3 (STS3).

After power on reset, the default clock source of TMR1 is PH3.

If watchdog reset occurred, the clock source of TMR1 will still keep the previous selection.

The following table shows the definition of each bit in TMR1 instructions

OPCODE	Select clock		Initiate value of timer					
TMSX X	X7	X6	X5	X4	X3	X2	X1	X0
TMS Rx	AC3	AC2	AC1	AC0	Rx3	Rx2	Rx1	Rx0
TMS @HL	bit7	bit6	bit5	Bit4	bit3	bit2	bit1	bit0

The following table shows the clock source setting for TMR1

X7	X6	clock source
0	0	PH9
0	1	PH3
1	0	PH15
1	1	FREQ

Notes:

1. When the TMR1 clock is PH3
   TMR1 set time= (Set value+error) * 8 * 1/fosc (KHz) (ms)
2. When the TMR1 clock is PH9
   TMR1 set time= (Set value+error) * 512 * 1/fosc (KHz) (ms)
3. When the TMR1 clock is PH15
   TMR1 set time= (Set value + error) * 32768 * 1/fosc (KHz) (ms)
   Set value: Decimal number of timer set value
   error: the tolerance of set value, 0 < error <1.
   fosc: Input of the predivider
   PH3: The 3rd stage output of the predivider
   PH9: The 9th stage output of the predivider
   PH15: The 15th stage output of the predivider
4. When the TMR1 clock is FREQ
   TMR1 set time= (Set value+error) * 1/FREQ (KHz) (ms).
   FREQ: refer to section 3-3-4.

2.12.2 RE-LOAD OPERATION

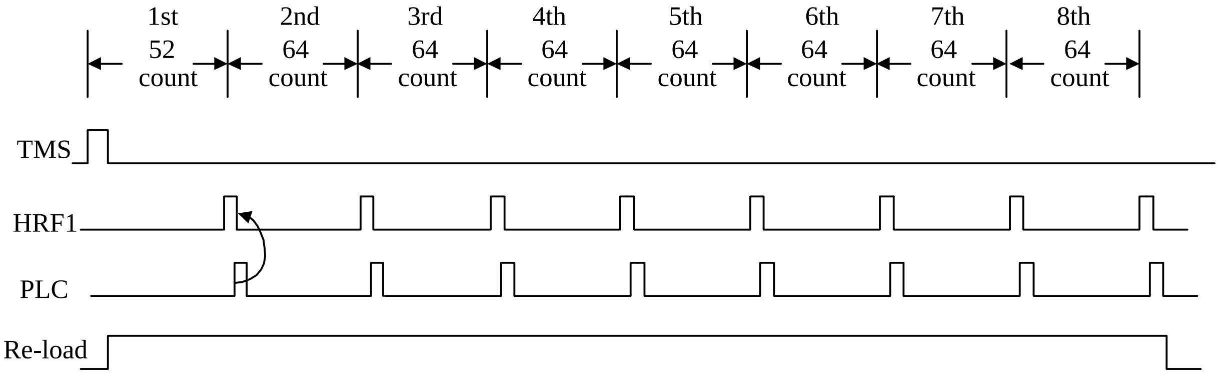
TMR1 provides the re-load function which can extend any time interval greater than 3Fh. The SF 80h instruction enables the re-load function and RF 80h instruction disables it.

When the re-load function is enabled, the TMR1 will not stop counting until the re-load function is disabled and TMR1 underflows again. During this operation, the program must use the halt release request flag or interrupt to check the wanted counting value.

- It is necessary to execute the TMS or TMSX instruction to set the down count value before the re-load function is enabled, because TMR1 will automatically count down with an unknown value once the re-load function is enabled.

- Never disable the re-load function before the last expected halt release or interrupt occurs. If TMS related instructions are not executed after each halt release or interrupt occurs, the TMR1 will stop operating immediately after the re-load function is disabled.

For example, if the expected count down value is 500, it may be divided as 52 + 7 * 64. First, set the initiate count down value of TMR1 to 52 and start counting, then enable the TMR1 halt release or interrupt function. Before the first time underflow occurs, enable the re-load function. The TMR1 will continue operating even though TMR1 underflow occurs. When halt release or interrupt occurs, clear the HRF1 flag by PLC instruction. After halt release or interrupt occurs 8 times, disable the re-load function and the counting is completed.

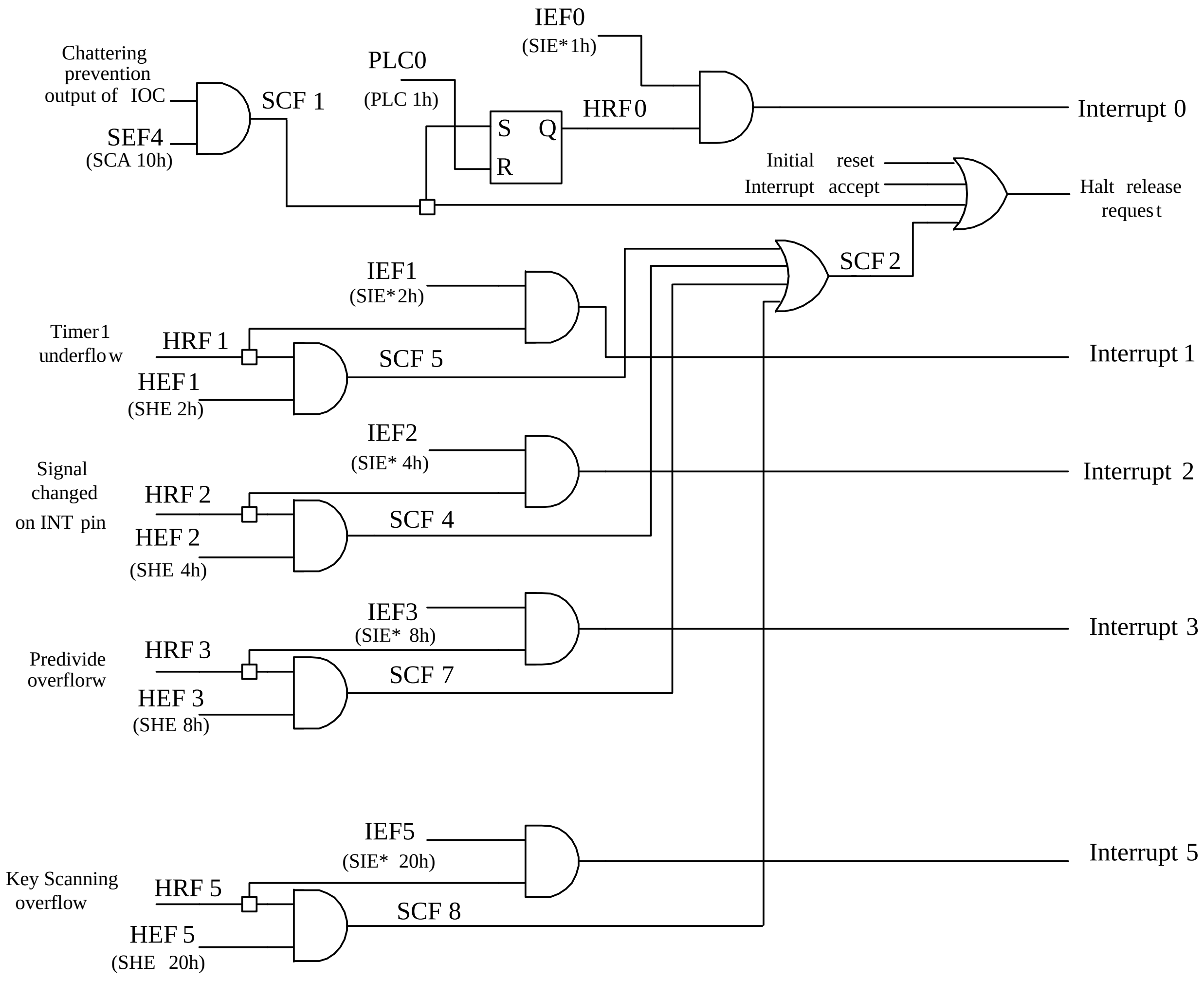


In the following example, S/W enters the halt mode to wait for the underflow of TMR1.

```
        LDS         0, 0                ; initiate the underflow counting register
        PLC         2
        SHE         2                   ; enable the HALT release caused by TMR1
        TMSX        34h                 ; initiate the TMR1 value (52) and clock source is ϕ9
        SF 80h                          ; enable the re-load function
RE_LOAD:
        HALT
        INC*        0                   ; increase the underflow counter
        PLC         2                   ; clear HRF1
        JB3         END_TM1             ; if the TMR1 underflow counter is equal to 8, exit subroutine
        JMP         RE_LOAD
END_TM1:
        RF80h                           ; disable the re-load function
```

2.13 STATUS REGISTER (STS)

The status register (STS) is organized with 4 bits and comes in 4 types: status register 1 (STS1) to status register 4 (STS4). The following figure shows the configuration of the start condition flags for TM8763.



2.13.1 STATUS REGISTER 1 (STS1)

Status register 1 (STS1) consists of 2 flags:

1. Carry flag (CF)

   The carry flag is used to save the result of the carry or borrow during the arithmetic operation.

2. Zero flag (Z)

   Indicates the accumulator (AC) status. When the content of the accumulator is 0, the Zero flag is set to 1. If the content of the accumulator is not 0, the zero flag is reset to 0.

3. The MAF instruction can be used to transfer data in status register 1 (STS1) to the accumulator (AC) and the data memory (RAM).

4. The MRA instruction can be used to transfer data of the data memory (RAM) to the status register 1 (STS1).

The bit pattern of status register 1 (STS1) is shown below.

Bit 3	Bit 2	Bit 1	Bit 0
Carry flag (AC)	Zero flag (Z)	NA	NA
Read / write	Read only	Read only	Read only

2.13.2 STATUS REGISTER 2 (STS2)

Status register 2 (STS2) consists of start condition flag 1, 2 (SCF1, SCF2) and the backup flag.

The MSB instruction can be used to transfer data of status register 2 (STS2) to the accumulator (AC) and the data memory (RAM), but it is impossible to transfer data of the data memory (RAM) to status register 2 (STS2).

The following table shows the bit pattern of each flag in status register 2 (STS2).

Bit 3	Bit 2	Bit 1	Bit 0
NA	Start condition flag 2 (SCF2)	Start condition flag 1 (SCF1)	Backup flag (BCF)
NA	Halt release caused by SCF4,5,7,8	Halt release caused by the IOC port	The back up mode status
NA	Read only	Read only	Read only

Start condition flag 1 (SCF1)

When the SCA instruction specified signal change occurs at port IOC to release the halt mode, SCF1 will be set. Executing the SCA instruction will cause SCF1 to be reset to 0

Start condition flag 2 (SCF2)

When a factor other than port IOC causes the halt mode to be released, SCF2 will be set to1. In this case, if one or more start condition flags in SCF4, 5, 7, 8 are set to 1; SCF2 will also be set to 1 simultaneously. When all of the flags in SCF4, 5, 7, 8 are clear, start condition flag 2 (SCF2) is reset to 0.

Note: If start condition flag is set to 1, the program will not be able to enter halt mode.

Backup flag (BCF)

This flag could be set/reset by executing the SF 2h/RF 2h instruction.

2.13.3 STATUS REGISTER 3 (STS3)

When the halt mode is released by start condition flag 2 (SCF2), status register 3 (STS3) will store the status of the factor in the release of the halt mode.

Status register 3 (STS3) consists of 4 flags:

1. Start condition flag 4 (SCF4)

Start condition flag 4 (SCF4) is set to 1 when the signal change at the INT pin causes the halt release request flag 2 (HRF2) to be outputted and the halt release enable flag 2 (HEF2) is set beforehand. To reset start condition flag 4 (SCF4), the PLC instruction must be used to reset the halt release request flag 2 (HRF2) or the SHE instruction must be used to reset the halt release enable flag 2 (HEF2).

2. Start condition flag 5 (SCF5)

   Start condition flag 5 (SCF5) is set when an underflow signal from Timer 1 (TMR1) causes the halt release request flag 1 (HRF1) to be outputted and the halt release enable flag 1 (HEF1) is set beforehand. To reset start condition flag 5 (SCF5), the PLC instruction must be used to reset the halt release request flag 1 (HRF1) or the SHE instruction must be used to reset the halt release enable flag 1 (HEF1).

3. Start condition flag 7 (SCF7)

   Start condition flag 7 (SCF7) is set when an overflow signal from the pre-divider causes the halt release request flag 3 (HRF3) to be outputted and the halt release enable flag 3 (HEF3) is set beforehand. To reset start condition flag 7 (SCF7), the PLC instruction must be used to reset the halt release request flag 3 (HRF3) or the SHE instruction must be used to reset the halt release enable flag 3 (HEF3).

4. The 15th stage's content of the pre-divider.

5. The MSC instruction is used to transfer the contents of status register 3 (STS3) to the accumulator (AC) and the data memory (RAM).

The following table shows the Bit Pattern of Status Register 3 (STS3)

Bit 3	Bit 2	Bit 1	Bit 0
Start condition flag 7 (SCF7)	15th stage of the pre-divider	Start condition flag 5 (SCF5)	Start condition flag 4 (SCF4)
Halt release caused by pre-divider overflow		Halt release caused by TMR1 underflow	Halt release caused by INT pin
Read only	Read only	Read only	Read only

2.13.4 STATUS REGISTER 3X (STS3X)

When the halt mode is released with start condition flag 2 (SCF2), status register 3X (STS3X) will store the status of the factor in the release of the halt mode.

Status register 3X (STS3X) consists of 3 flags:

1. Start condition flag 8 (SCF8)

   SCF8 is set to 1 when any one of KI1~4=1/0 (KI1~4=1 in LED mode KI1~4=0 in LCD mode) causes the halt release request flag 5 (HRF5) to be outputted and the halt release enable flag 5 (HEF5) is set beforehand. To reset the start condition flag 8 (SCF8), the PLC instruction must be used to reset the halt release request flag 5 (HRF5) or the SHE instruction must be used to reset the halt release enable flag 5 (HEF5).

The MCX instruction can be used to transfer the contents of status register 3X (STS3X) to the accumulator (AC) and the data memory (RAM).

The following table shows the Bit Pattern of Status Register 3X (STS3X)

Bit 3	Bit 2	Bit 1	Bit 0
NA	NA	NA	Start condition flag 8 (SCF8)
			Halt release caused by SKI underflow
NA	NA	NA	Read only

2.13.5 STATUS REGISTER 4 (STS4)

Status register 4 (STS4) consists of 3 flags:

1. System clock selection flag (CSF)

   The system clock selection flag (CSF) indicates which clock source of the system clock generator (SCG) is used. Executing SLOW instruction will change the clock source (BCLK) of the system clock generator (SCG) to the slow speed oscillator (XT clock), and the system clock selection flag (CSF) is reset to 0. Executing FAST instruction will change the clock source (BCLK) of the system clock generator (SCG) to the fast speed oscillator (CF clock), and the system clock selection flag (CSF) is set to 1. For the operation of the system clock generator, refer to 3-3.

2. Watchdog timer enable flag (WTEF)

   The watchdog timer enable flag (WDF) indicates the operating status of the watchdog timer.

   The MSD instruction can be used to transfer the contents of status register 4 (STS4) to the accumulator (AC) and the data memory (RAM).

The following table shows the Bit Pattern of Status Register 4 (STS4)

Bit 3	Bit 2	Bit 1	Bit 0
NA	NA	Watchdog timer Enable flag (WDF)	System clock selection flag (CSF)
NA	NA	Read only	Read only

2.13.6 START CONDITION FLAG 11 (SCF11)

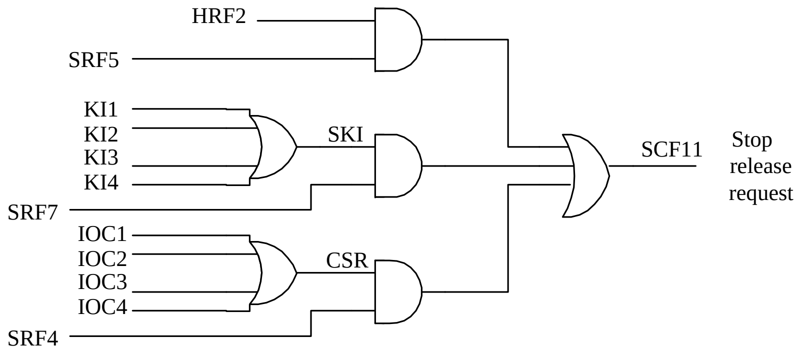
Start condition flag 11 (SCF11) will be set to 1 in STOP mode when the following conditions are met :

- A high level signal comes from the OR-ed output of the pins defined as input mode in IOC port, which causes the stop release flag of IOC port (CSR) to output, and stop release enable flag 4 (SRF4) is set beforehand.

- A high level signal comes from the OR-ed output of the signals latch for KI1~4, which causes the stop release flag of Key Scanning (SKI) to output, and stop release enable flag 4 (SRF7) is set beforehand.

- The signal change from the INT pin causes the halt release flag 2 (HRF2) to output and the stop release enable flag 5 (SRF5) is set beforehand.

The following figure shows the organization of start condition flag 11 (SCF 11).



The stop release flags (SKI, CSR, HRF2) were specified by the stop release enable flags (SRFx) and these flags should be clear before the chip enters the stop mode. All of the pins in IOC port had to be defined as the input mode and keep in 0 state before the chip enters the STOP mode, or the program can not enter the STOP mode.

Instruction SRE is used to set or reset the stop release enable flags(SRF4, 5, 7).

The following table shows the stop release request flags

	The OR-ed latched signals for KI1~4	The OR-ed input mode pins of IOC port	The rising or falling edge on INT pin
Stop release request flag	SKI	CSR	HRF2
Stop release enable flag	SRF7	SRF4	SRF5

2.14 ONTROL REGISTER (CTL)

The control register (CTL) comes in 4 types: control register 1 (CTL1) to control register 4 (CTL4).

2.14.1 CONTROL REGISTER 1 (CTL1)

The control register 1 (CTL1), being a 1-bit register:

1. Switch enable flag 4 (SEF4)

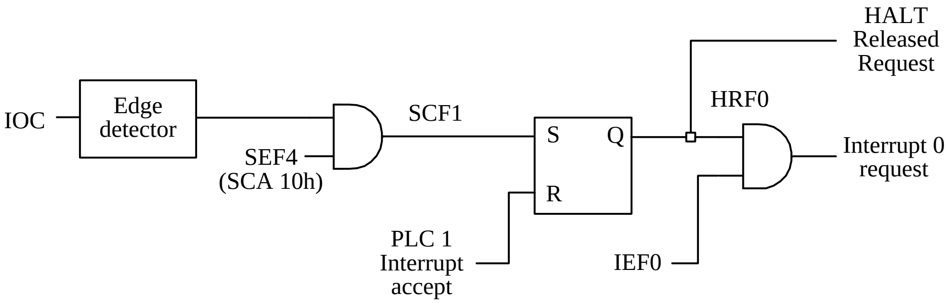
   Stores the status of the input signal change at pins of IOC defined as input mode that causes the halt mode or stop mode to be released.

Executed SCA instruction may set or reset these flags.

The following table shows Bit Pattern of Control Register 1 (CTL1)

Bit 4
Switch enable flag 4 (SEF4)
Enables the halt release caused by the signal change on IOC port
Write only

The following figure shows the organization of control register 1 (CTL1).



2.14.1.1 The Setting for Halt Mode

If the SEF4 is set to 1, the signal changed on IOC port will cause the halt mode to be released, and set SCF1 to 1. Because the input signal of IOC port were ORed, so it is necessary to keep the unchanged input signals at “ 0 ” state and only one of the input signal could change state.

2.14.1.2 The Setting for Stop Mode

If SRF4 and SEF4 are set, the stop mode will be released to set the SCF1 when a high level signal is applied to one of the input mode pins of IOC port and the other pins stay in ”0” state.

After the stop mode is released, TM8763 enters the halt condition.

The high level signal must hold for a while to cause the chattering prevention circuitry of IOC port to detect this signal and then set SCF1 to release the halt mode, or the chip will return to the stop mode again.

2.14.1.3 Interrupt for CTL1

The control register 1 (CTL1) performs the following function in the execution of the SIE instruction to enable the interrupt function.

The input signal changes at the input pins in IOC port will deliver the SCF1 when SEF4 has been set to 1 by executing SCA instruction. Once the SCF1 is delivered, the halt release request flag (HRF0) will be set to 1. In this case, if the interrupt enable flag 0 (IEF0) is set to 1 by executing SIE instruction, the interrupt request flag 0 (interrupt 0) will be delivered to interrupt the program.

If the interrupt 0 is accepted by SEF4 and IEF0, the interrupt 0 request to the next signal change at IOC will be inhibited. To release this mode, SCA instruction must be executed again. Refer to 2-16-1-1.

2.14.2 CONTROL REGISTER 2 (CTL2)

Control register 2 (CTL2) consists of halt release enable flags 1, 2, 3, 5 (HEF1, 2, 3, 5) and is set by SHE instruction. The bit pattern of the control register (CTL2) is shown below.

Halt release enable flag		HEF5	
Halt release condition		Enable the halt release caused by Key Scanning (HRF5)	
Halt release enable flag	HEF3	HEF2	HEF1
Halt release condition	Enable the halt release caused by pre-divider overflow (HRF3)	Enable the halt release caused by INT pin (HRF2)	Enable the halt release caused by TM1 underflow (HRF1)

When the halt release enable flag 1 (HEF1) is set, an underflow signal from TMR1 causes the halt mode to be released. In the same manner, the following conditions will cause the halt mode to be released respectively when HEF2, 3, 5 are set to 1: the signal change at the INT pin, an overflow signal from the pre-divider and a ‘H’ signal from OR-ed output of KI1~4 latch signals.

When the stop release enable flag 5 (SRF5) and the HEF2 are set, the signal change at the INT pin can cause the stop mode to be released.

When the stop release enable flag 7 (SRF7) and the HEF5 are set, the ‘H’ signal from OR-ed output of K1~4 latch signals can cause the stop mode to be released.

2.14.3 CONTROL REGISTER 3 (CTL3)

Control register 3 (CTL3) is organized with 7 bits of interrupt enable flags (IEF) to enable/disable interrupts.

The interrupt enable flag (IEF) is set/reset by SIE* instruction. The bit pattern of control register 3 (CTL3) is shown below.

Interrupt enable flag	IEF5	IEF3	IEF2
Interrupt request flag	Enable the interrupt request caused by Key Scanning (HRF5)	Enable the interrupt request caused by pre-divider overflow (HRF3)	Enable the interrupt request caused by INT pin (HRF2)
Interrupt flag	Interrupt 4	Interrupt 3	Interrupt 2
Interrupt enable flag	IEF1	IEF0	
Interrupt request flag	Enable the interrupt request caused by TM1 underflow (HRF1)	Enable the interrupt request caused by IOC port signal to be changed (HRF0)	
Interrupt flag	Interrupt 1	Interrupt 0	

When any of the interrupts are accepted, the corresponding HRFx and the interrupt enable flag (IEF) will be reset to 0 automatically. Therefore, the desirable interrupt enable flag (IEFx) must be set again before exiting from the interrupt routine.

2.14.4 CONTROL REGISTER 4 (CTL4)

Control register 4 (CTL4), being a 3-bit register, is set/reset by SRE instruction.

The following table shows the Bit Pattern of Control Register 4 (CTL4)

Stop release enable flag	SRF7	SRF5	SRF4 (SRF3)
Stop release request flag	Enable the stop release request caused by signal change on KI1~4 (SKI)	Enable the stop release request caused by signal change on INT pin (HRF2)	Enable the stop release request caused by signal change on IOC

When the stop release enable flag 7 (SRF7) is set to 1, the input signal change at the KI1~4 pins causes the stop mode to be released. In the same manner, when SRF4 (SRF3) and SRF5 are set to 1, the input signal change at the input mode pins of IOC port and the signal changed on INT pin causes the stop mode to be released respectively.

Example:

This example illustrates the stop mode released by port IOC, KI1~4 and INT pin. Assume all of the pins in IOD and IOC have been defined as input mode.

```
PLC    25h          ; Reset the HRF0, HRF2 and HRF5.
SHE    24h          ; HEF2 and HEF5 is set so that the signal change at INT or KI1~4 pin
                    ; causes start condition flag 4 or 8 to be set.
SCA    10h          ; SEF4 is set so that the signal changes at port IOC
                    ; cause the start conditions SCF1 to be set.
SRE    0b0h         ; SRF7,5,4 are set so that the signal changes at KI1~4 pins, port
                    ; IOC and INT pin cause the stop mode to be released.
```

```
STOP                    ; Enter the stop mode.

……………                ;STOP release

MSC     10h             ; Check the signal change at INT pin that causes the stop mode to be
                        ; released.
MSB     11h             ; Check the signal change at port IOC that causes the stop mode to be
                        ; released.
MCX     12h             ; Check the signal change at KI1~4 pins that causes the stop mode to be
                        ; released.
```

2.15 HALT FUNCTION

The halt function is provided to minimize the current dissipation of the TM8763 when LCD is operating. During the halt mode, the program memory (ROM) is not in operation and only the oscillator circuit, pre-divider circuit, sound circuit, I/O port chattering prevention circuit, and LCD driver output circuit are in operation. (If the timer has started operating, the timer counter still operates in the halt mode).

After the HALT instruction is executed and no halt release signal (SCF1, SCF3, HRF1 ~ 6) is delivered, the CPU enters the halt mode.

The following 3 conditions are available to release the halt mode.

(1) An interrupt is accepted.

When an interrupt is accepted, the halt mode is released automatically, and the program will enter halt mode again by executing the RTS instruction after completion of the interrupt service.

When the halt mode is released and an interrupt is accepted, the halt release signal is reset automatically.

(2) The signal change specified by the SCA instruction is applied to port IOC(SCF1).

(3) The halt release condition specified by the SHE instruction is met (HRF1 ~ HRF6).

When the halt mode is released in either (2) or (3), it is necessary that the MSB, MSC, or MCX instruction is executed in order to test the halt release signal and that the PLC instruction is then executed to reset the halt release signal (HRF).

Even when the halt instruction is executed in the state where the halt release signal is delivered, the CPU does not enter the halt mode.

2.16 HEAVY LOAD FUNCTION

When heavy loading (lamp light-up, motor start, etc.) causes a temporary voltage drop on supply voltage, the heavy loading function (set BCF=1) prevents TM8763 from malfunctioning, especially where a battery with high internal impedance, such as Li battery or alkali battery, is used.

During back up mode, the 32.768KHz Crystal oscillator will add an extra buffer in parallel and switch the internal power (BAK) from VDD1 to VDD2 (Li power option only). In this condition, all of the functions in TM8763 will work under VDD voltage range; this will cause TM8763 to get better noise immunity.

For shorten the start-up time of 32.768KHz Crystal oscillator, TM8763 will set the BCF to 1 during reset cycle and reset BCF to 0 after reset cycle automatically in Ag and Li power mode option. In EXT-V power mode option, however, BCF is set to 1 by default setting and can not be reset to 0, and BCF will be reset to 0 by default setting during normal operation.

Table 3- 1 The back-up flag status in different conditions

	Ag option	Li option	EXT-V option	Remark
Reset cycle	BCF=1	BCF=1	BCF=0	large current
After reset cycle	BCF=1	BCF=1	BCF=0	large current
SF 2 executed	BCF=1	BCF=1	BCF=1	large current
RF 2 executed	BCF=0	BCF=0	BCF=0	

For low power consumption application, reset BCF to 0 is necessary; the 32.768KHz Crystal oscillator operates with a normal buffer only, so switch the internal power (BAK) to VDD1 (Li power option only). In this condition, only peripheral circuitry operates under VDD voltage range; the other functions will operate under 1/2 VDD voltage range. In Ag and EXT-V power options, the internal power (BAK) will not be affected by the setting of BCF. With Li power option, it is necessary to connect a 0.1uf capacitor from BAK power pin to GND for the backup mode application.

When the heavy load function is performed, the current dissipation will increase.

Table 3- 2 Ag power option:

	Initial reset	After reset	STOP mode	SF 2	RF 2
BCF	1	1	1*	1	0
Internal logic	VDD	VDD	VDD	VDD	VDD
Peripheral logic	VDD	VDD	VDD	VDD	VDD

Table 3- 3 Li power option:

	Initial reset	After reset	Stop mode	SF 2	RF 2
BCF	1	1	1*	1	0
Internal logic	VDD	VDD	VDD	VDD	1/2 VDD
Peripheral logic	VDD	VDD	VDD	VDD	VDD

Table 3- 4 EXT-V power option:

	Initial reset	After reset	Stop mode	SF 2	RF 2
BCF	0	0	1*	1	1
Internal logic	VDD	VDD	VDD	VDD	VDD
Peripheral logic	VDD	VDD	VDD	VDD	VDD

Note: When the program enters the stop mode, the BCF will set to 1 automatically to insure that the low speed oscillator will start up in a proper condition while stop release occurs.

2.17 STOP FUNCTION (STOP)

The stop function is another solution to minimize the current dissipation for TM8763. In stop mode, all of functions in TM8763 are held including oscillators. All of the LCD corresponding signals (COM and Segment) will output "L" level. In this mode, TM8763 does not dissipate any power in the stop mode. Because the stop mode will set the BCF flag to 1 automatically, it is recommended to reset the BCF flag after releasing the stop mode in order to reduce power consumption.

Before the stop instruction is executed, all of the signals on the pins defined as input mode of IOC port must be in the "L" state, and no stop release signal (SRFn) should be delivered. The CPU will then enter the stop mode.

The following conditions cause the stop mode to be released.

- One of the signals on the input mode pin of IOC port is in "H" state and holds long enough to cause the CPU to be released from halt mode.
- A signal change in the INT pin.
- The stop release condition specified by the SRE instruction is met. (INT pin is exclusive)

When the TM8763 is released from the stop mode, the TM8763 enters the halt mode immediately and will process the halt release procedure. If the "H" signal on the IOC port does not hold long enough to set the SCF1, once the signal on the IOC port returns to "L", the TM8702 will enter the stop mode immediately. The backup flag (BCF) will be set to 1 automatically after the program enters the stop mode.

The following diagram shows the stop release procedure:

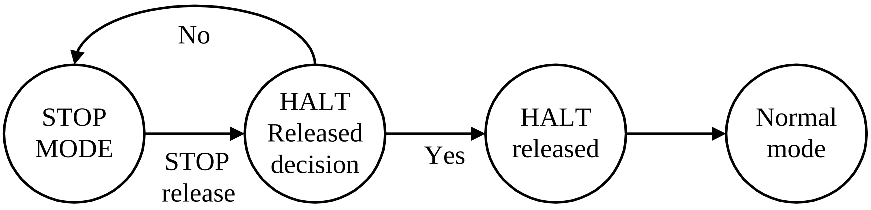


Figure 3- 16 The stop release state machine

Before the stop instruction is executed, the following operations must be completed:

- Specify the stop release conditions by the SRE instruction.
- Specify the halt release conditions corresponding to the stop release conditions if needed.
- Specify the interrupt conditions corresponding to the stop release conditions if needed.

When the stop mode is released by an interrupt request, the TM8763 will enter the halt mode immediately. While the interrupt is accepted, the halt mode will be released by the interrupt request. The stop mode returns by executing the RTS instruction after completion of interrupt service.

After the stop release, it is necessary that the MSB, MSC or MCX instruction be executed to test the halt release signal and that the PLC instruction then be executed to reset the halt release signal. Even when the stop instruction is executed in the state where the stop release signal (SRF) is delivered, the CPU does not enter the stop mode but the halt mode. When the stop mode is released and an interrupt is accepted, the halt release signal (HRF) is reset automatically.

2.18 BACK UP FUNCTION

TM8763 provide a back up mode to avoid system malfunction when heavy loading occurred, such as buzzer is active, LED is lighting… etc. Since the heavy loading will cause a large voltage drop on the supply voltage, and the system will be malfunction in this condition.

Once the program enter back up mode (BCF=1), 32.768 KHz Crystal oscillator will operate in a large driver condition and internal logic function operates with higher supply voltage. TM8763 will get more power supply noise margin while back up mode is active but also increases more power consumption.

The back up flag (BCF) indicated the status of back up function. BCF flag could be set or reset by executing SF or RF instruction respectively.

The back up function has different performance corresponding to different power mode option, shown in the following table.

1.5V battery mode:

TM8763 status	BCF flag status
Initial reset cycle	BCF=1 (hardware controlled)
After initial reset cycle	BCF=1 (hardware controlled)
Executing SF 2h instruction	BCF=1
Executing RF 2h instruction	BCF=0
HALT mode	Previous state
STOP mode	BCF=1 (hardware controlled)

TM8763 status	BCF=0	BCF=1
32.768KHz Crystal Oscillator	Small driver	Large driver
Voltage on BAK pin	VDD1	VDD1
Internal operating voltage	VDD1	VDD1

3V battery or higher mode:

TM8763 status	BCF flag status
Initial reset cycle	BCF=1 (hardware controlled)
After initial reset cycle	BCF=1 (hardware controlled)
Executing SF 2h instruction	BCF=1
Executing RF 2h instruction	BCF=0
HALT mode	Previous state
STOP mode	BCF=1 (hardware controlled)

	BCF=0	BCF=1
32.768KHz Crystal Oscillator	Small driver	Large driver
Voltage on BAK pin	VDD1	VDD2
Internal operating voltage	VDD1	VDD2

Ext-V power mode:

TM8763 status	BCF flag status
Initial reset cycle	BCF=0 (hardware controlled)
After initial reset cycle	BCF=0 (hardware controlled)
Executing SF 2h instruction	BCF=1
Executing RF 2h instruction	BCF=0
HALT mode	Previous state
STOP mode	BCF=1 (hardware controlled)

	BCF=0	BCF=1
32.768 KHz Crystal Oscillator	Large driver	Large driver
Voltage on BAK pin	VDD2	VDD2
Internal operating voltage	VDD2	VDD2

Note : For power saving reason, it is recommend to reset BCF flag to 0 when back up mode is not used.

3. CONTROL FUNCTION

3.1 INTERRUPT FUNCTION

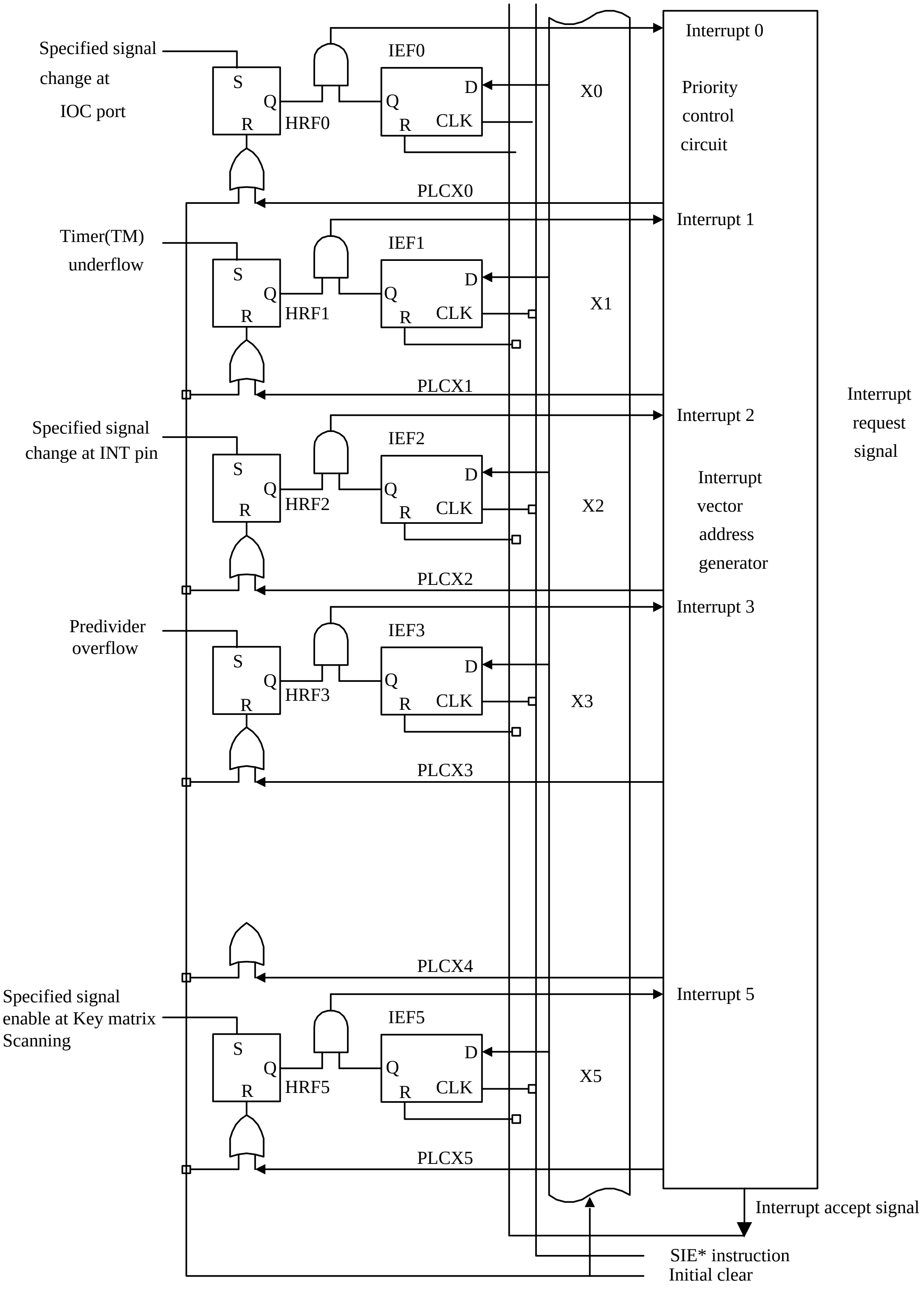
There are 5 interrupt resources: 3 external interrupt factors and 2 internal interrupt factors. When an interrupt is accepted, the program in execution is suspended temporarily and the corresponding interrupt service routine specified by a fix address in the program memory (ROM) is called.

The following table shows the flag and service of each interrupt:

Table 3-5 Interrupt information

Interrupt source	INT pin	IOC port	TMR1 underflow	Pre-divider overflow	Key matrix Scanning
Interrupt vector	010H	014H	018H	01CH	024H
Interrupt enable flag	IEF2	IEF0	IEF1	IEF3	IEF5
Interrupt priority	4th	3rd	2nd	1st	5th
Interrupt request flag	Interrupt 2	Interrupt 0	Interrupt 1	Interrupt 3	Interrupt 5

The following figure shows the Interrupt Control Circuit



3.1.1 INTERRUPT REQUEST AND SERVICE ADDRESS

3.1.1.1 External interrupt factor

The external interrupt factor involves the use of the INT pin, IOC ports, or Key matrix Scanning.

1. External INT pin interrupt request

   By using mask option, either a rise or fall of the signal at the INT pin can be selected for applying an interrupt. If the interrupt enable flag 2 (IEF2) is set and the signal on the INT pin change that matches the mask option will issue the HRF2, interrupt 2 is accepted and the instruction at address10H is executed automatically. It is necessary to apply level "L" before the signal rises and level "H" after the signal rises to the INT pin for at least 1 machine cycle.

2. I/O port IOC interrupt request.

3. An interrupt request signal (HRF0) is delivered when the input signal changes at I/O port IOC specified by the SCA instruction. In this case, if the interrupt enabled by flag 0 (IEF0) is set to 1, interrupt 0 is accepted and the instruction at address 14H is executed automatically.

4. Key matrix Scanning interrupt request.

   An interrupt request signal (HRF5) is delivered when the input signal generated in scanning interval. If the interrupt enable flag 5 (IEF5) is set to 1 and interrupt 5 is accepted, the instruction at address 24H will be executed automatically.

3.1.1.2 Internal interrupt factor

The internal interrupt factor involves the use of timer 1 (TMR1) and the pre-divider.

1. Timer1 (TMR1) interrupt request

   An interrupt request signal (HRF1) is delivered when timer1 (TMR1) underflows. In this case, if the interrupt enable flag 1 (IEF1) is set, interrupt 1 is accepted and the instruction at address 18H is executed automatically.

2. Pre-divider interrupt request

   An interrupt request signal (HRF3) is delivered when the pre-divider overflows. In this case, if the interrupt enable flag3 (IEF3) is set, interrupt 3 is accepted and the instruction at address 1CH is executed automatically.

3.1.2 INTERRUPT PRIORITY

If all interrupts are requested simultaneously during a state when all interrupts are enabled, the pre-divider interrupt is given the first priority and other interrupts are held. When the interrupt service routine is initiated, all of the interrupt enable flags (IEF0~IEF6) are cleared and should be set with the next execution of the SIE instruction. Refer to Table 3-1.

<u>Example:</u>

```
; Assume all interrupts are requested simultaneously when all interrupts are enabled, and all of the
; the pins of IOC have been defined as input mode.
```

```
PLC     7Fh             ; Clear all of the HRF flags
SCA     10h             ; enable the interrupt request of IOC
SIE*    3Fh             ; enable all interrupt requests

;……………………...…     ; all interrupts are requested simultaneously.

;Interrupt caused by the predivider overflow occurs, and interrupt  service is concluded.

SIE*    27h             ; Enable the interrupt request (except  the predivider).

;Interrupt caused by the TM1 underflow occurs, and interrupt service is concluded.

SIE*    25h             ; Enable the interrupt request (except  the predivider and TMR1).

;Interrupt caused by the IOC port, and interrupt service is concluded.

SIE*    24h             ; Enable the interrupt request (except  the predivider, TMR1,
                        ; and IOC port)

;Interrupt caused by the INT pin, and interrupt service is concluded.

SIE*    20h             ; Enable the interrupt request (except  the predivider, TMR1,
                        ; IOC port, and INT)
;Interrupt caused by the Key matrix Scanning, and interrupt service is concluded.
;All interrupt requests have been processed.
```

3.1.3 INTERRUPT SERVICING

When an interrupt is enabled, the program in execution is suspended and the instruction at the interrupt service address is executed automatically(Refer to Table 3-1). In this case, the CPU performs the following services automatically.

(1) As for the return address of the interrupt service routine, the addresses of the program counter (PC) installed before interrupt servicing began are saved in the stack register (STACK).

(2) The corresponding interrupt service routine address is loaded in the program counter (PC).

The interrupt request flag corresponding to the interrupt accepted is reset and the interrupt enable flags are all reset.

When the interrupt occurs, the TM8763 will follow the procedure below:

```
Instruction 1           ; In this instruction, interrupt is accepted.
NOP                     ; TM8763 stores the program counter data into the STACK. At this time,
;no instruction will be executed, as with NOP instruction.
Instruction A           ; The program jumps to the interrupt service routine.
Instruction B
Instruction C
.............
```

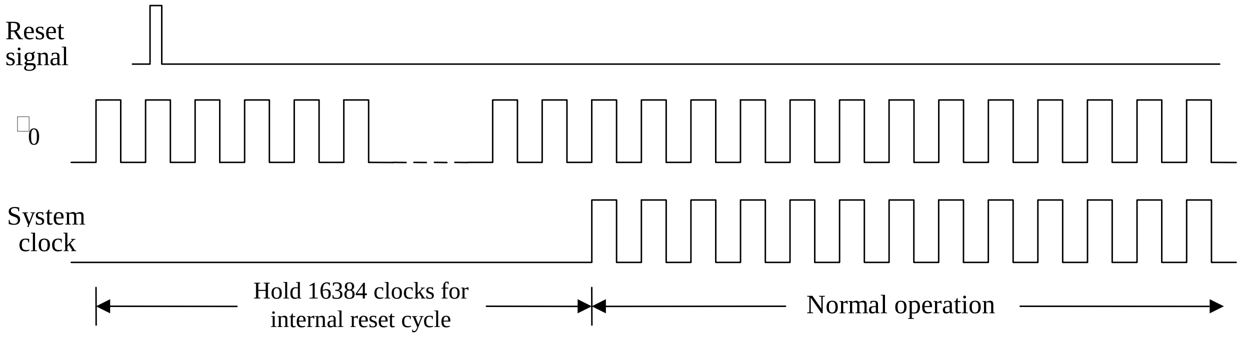
```
RTS                      ;Finishes the interrupt service routine
Instruction 1*           ;re-executes the instruction which was interrupted.
Instruction 2
```

Note: If instruction 1 is “halt” instruction, the CPU will return to “halt” after interrupt.
When an interrupt is accepted, all interrupt enable flags are reset to 0 and the corresponding HRF flag will be cleared; the interrupt enable flags(IEF) must be set again in the interrupt service routine as required.

3.2 RESET FUNCTION

TM8763 contains four reset sources: power-on reset, RESET pin reset, IOC port reset and watchdog timer reset.

When reset signal is accepted, TM8763 will generate a time period for internal reset cycle and there are two types of internal reset cycle time could be selected by mask option, the one is PH15/2 and the other is PH12/2.



Internal reset cycle time is PH15/2

MASK OPTION table:

Mask Option name	Selected item
RESET TIME	(1) PH15/2

In this option, the reset cycle time will be extended 16384 clocks (clock source comes form pre-divider) long at least.

Internal reset cycle time is PH12/2

MASK OPTION table:

Mask Option name	Selected item
RESET TIME	(2) PH12/2

In this option, the reset cycle time will be extended 2048 clocks (clock source comes form pre-divider) long at least.

3.2.1 POWER ON RESET

TM8763 provides a power on reset function. If the power (VDD) is turned on or power supply drops below 0.6V, it will generate a power-on reset signal.

Power-on reset function can be disabled by mask option.

MASK OPTION table:

Mask Option name	Selected item
POWER ON RESET	(1) USE
POWER ON RESET	(2) NO USE

Note: When the power on reset option is selected, connected a capacitor between VDD and GND is necessary.

3.2.2 RESET PIN RESET

When "H" level is applied to the reset pin, the reset signal will issue. Built in a pull down resistor on this pin.

Two types of reset method for RESET pin and the type could be mask option, the one is level reset and other is pulse reset.

It is recommended to connect a capacitor (0.1uf) between RESET pin and VDD. This connection will prevent the bounce signal on RESET pin.

3.2.2.1 Level Reset

Once a “1” signal applied on the RESET pin, TM8763 will not release the reset cycle until the signal on RESET pin returned to “0”. After the signal on reset pin is cleared to 0, TM8763 begins the internal reset cycle and then release the reset status automatically.

MASK OPTION table:

Mask Option name	Selected item
RESET PIN TYPE	(1) LEVEL

3.2.2.2 Pulse Reset

Once a “1” signal applied on the RESET pin, TM8763 will escape from reset state and begin the normal operation after internal reset cycle automatically no matter what the signal on RESET pin returned to “0” or not.

MASK OPTION table:

Mask Option name	Selected item
RESET PIN TYPE	(2) PULSE

The following table shows the initial condition of TM8763 in reset cycle.

Program counter	(PC)	Address 000H
Start condition flags 1 to 7	(SCF1-7)	0
Backup flag	(BCF)	1 (Ag, Li version) 0 (EXTV version)
Stop release enable flags 4,5,7	(SRF3,4,5,7)	0
Switch enable flags 4	(SEF3,4)	0
Halt release request flag	(HRF 0~6)	0
Halt release enable flags 1 to 3	(HEF1-6)	0
Interrupt enable flags 0 to 3	(IEF0-6)	0
Alarm output	(ALARM)	DC 0
Pull-down flags in I/OC port		1 (with pull-down resistor)
Input/output ports I/OA, I/OB, I/OC	(PORT I/OA, I/OB, I/OC)	Input mode
I/OC port chattering clock	Cch	PH10*
EL panel driver pumping clock source and duty cycle	Celp	PH0, duty cycle is 3/4
EL panel driver clearing clock source and duty cycle	Celc	PH8, duty cycle is 1/4
Resistor frequency converter	(RFC)	Inactive, RR/RT/RH output 0
LCD driver output		All lighted (mask option)*
Timer 1		Inactive
Watchdog timer	(WDT)	Reset mode, WDF=0
Clock source	(BCLK)	XT clock (slow speed clock in dual clock option)

Notes: PH3: the 3rd output of predivider
PH10: the 10th output of predivider
Mask option can unlighted all of the LCD output

3.2.3 IOC Port/Key Matrix RESET

Key reset function is selected by mask option. When IOC port or key matrix scanning input (KI1~4) is in used, the ‘0’ signal applied to all these pins that had be set as input mode in the same time (KI1~4 pins need to wait scanning time), reset signal is delivered.

MASK OPTION table:

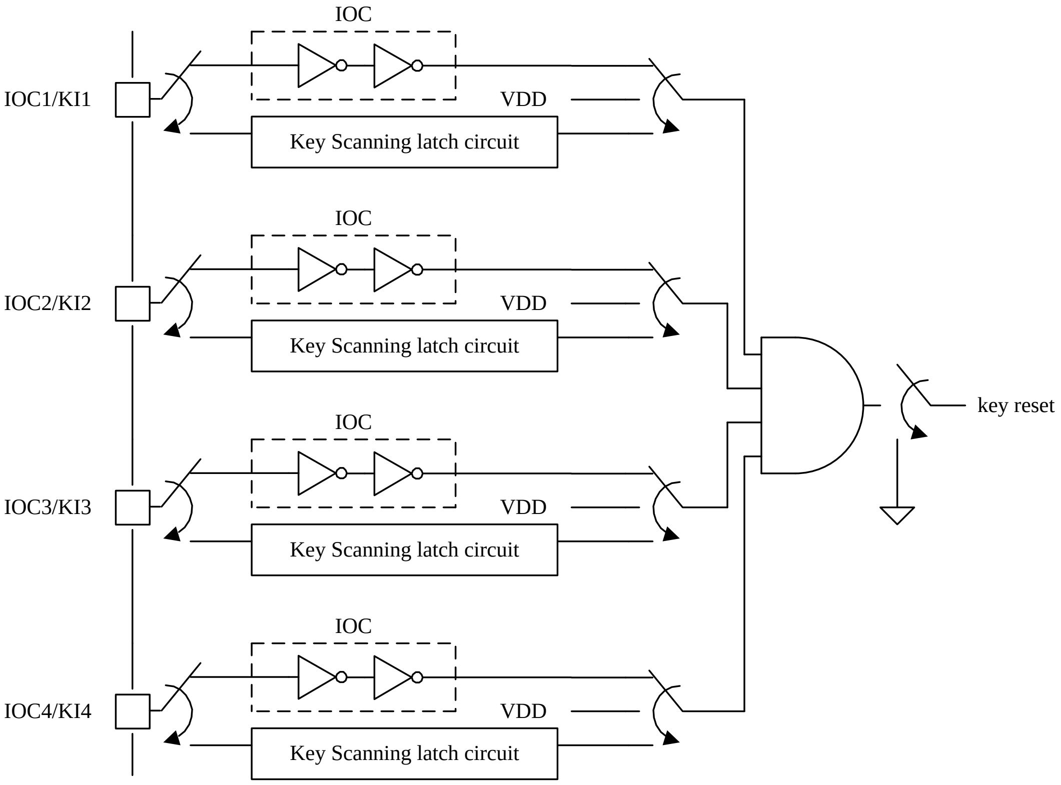
IOC or KI pins are used as key reset:

Mask Option name	Selected item
IOC1/KI1 FOR KEY RESET	(1) USE
IOC2/KI2 FOR KEY RESET	(1) USE
IOC3/KI3 FOR KEY RESET	(1) USE
IOC4/KI4 FOR KEY RESET	(1) USE

IOC or KI pins aren’t used as key reset:

Mask Option name	Selected item
IOC1/KI1 FOR KEY RESET	(2) NO USE
IOC2/KI2 FOR KEY RESET	(2) NO USE
IOC3/KI3 FOR KEY RESET	(2) NO USE
IOC4/KI4 FOR KEY RESET	(2) NO USE

The following figure shows the key reset organization.

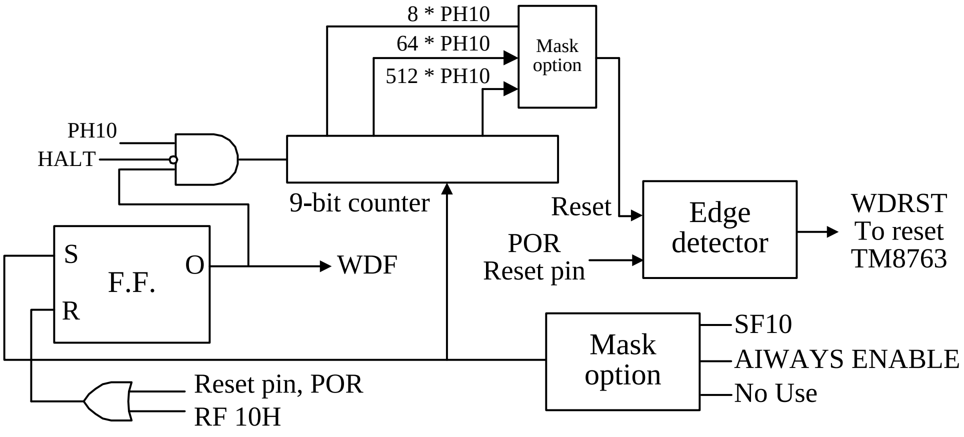

IOC
IOC1/KI1
VDD
Key Scanning latch circuit
IOC
IOC2/KI2
VDD
Key Scanning latch circuit
IOC
IOC3/KI3
VDD
Key Scanning latch circuit
IOC
IOC4/KI4
VDD
Key Scanning latch circuit
key reset

3.2.4 WATCHDOG RESET

The timer is used to detect unexpected execution sequence caused by software run-away. The watchdog timer consists of a 9-bit binary counter. The timer input (PH10) is the 10th stage output of the pre-divider.

When the watchdog timer overflows, it generates a reset signal to reset TM8763 and most of the functions in TM8763 will be initiated except for the watchdog timer (which is still active), WDF flag will not be affected and PH0~PH10 of the pre-divider will not be reset.

The following figure shows the watchdog timer organization.



During initial reset (power on reset [POR] or reset pin), the timer is inactive and the watchdog flag (WDF) is reset. Instruction SF 10h will enable the watchdog timer and set the watchdog flag (WDF) to 1. At the same time, the content of the timer will be cleared. Once the watchdog timer is enabled, the timer will be paused when the program enters the halt mode or stop mode. When the TM8763 wakes up from the halt or stop mode, the timer operates continuously. It is recommended to execute SF 10h instruction before the program enters the halt or stop mode in order to initialize the watchdog timer.

Once the watchdog timer is enabled, the program must execute SF 10h instruction periodically to prevent the timer overflowed.

The overflow time interval of watchdog timer is selected by mask option:

MASK OPTION table:

Mask Option name	Selected item
WATCHDOG TIMER OVERFLOW TIME INTERVAL	(1) 8 x PH10
	(2) 64 x PH10
	(3) 512 x PH10

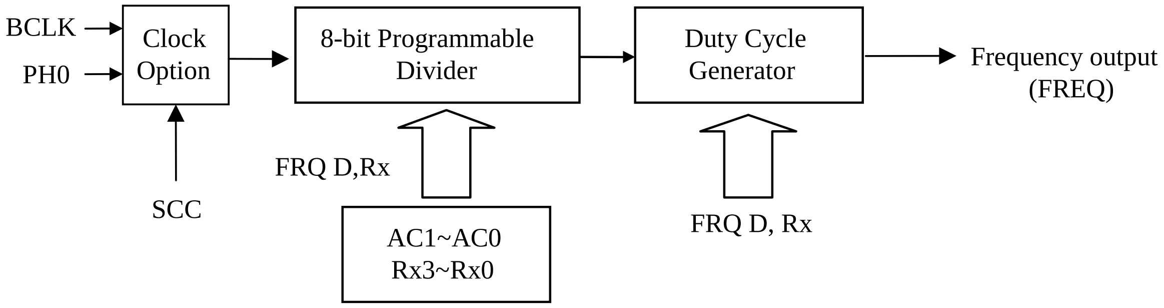
Note: timer overflow time interval is about 16 seconds when PH0=32.768 KHz

3.3 CLOCK GENERATOR

3.3.1 FREQUENCY GENERATOR

The Frequency Generator is a versatile programmable divider that is capable of delivering a clock with wide frequency range and different duty cycles. The output of the frequency generator may be the clock source for the alarm function, timer1, timer2 and RFC counter.

The following shows the organization of the frequency generator.



SCC instruction may specify the clock source selection for the frequency generator. The frequency generator outputs the clock with different frequencies and duty cycles corresponding to the presetting data of FRQ related instructions. The FRQ related instructions preset a letter N into the programming divider and letter D into the duty cycle generator. The frequency generator will then output the clock using the following formula:

FREQ= (clock source) / ((N+1) * X) Hz. (X=1, 2, 3, 4 for 1/1, 1/2, 1/3, 1/4 duty)

This letter N is a combination of data memory and accumulator (AC), or the table ROM data or operand data specified in the FRQX instruction. The following table shows the bit pattern of the combination.

The following table shows the bit pattern of the preset letter N

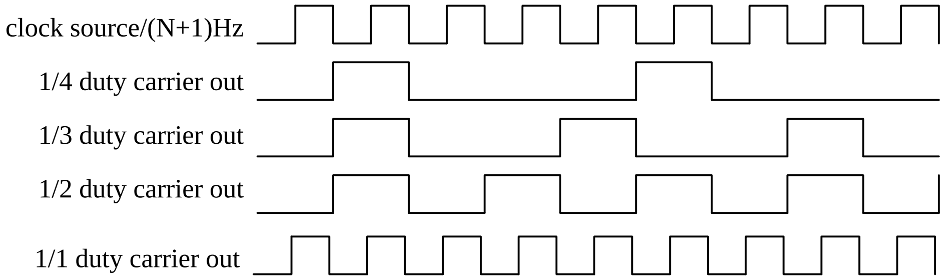
	The bit pattern of preset letter N							
Programming divider	bit7	Bit6	bit 5	bit 4	bit 3	Bit 2	bit 1	bit 0
FRQ D,Rx	AC3	C2	AC1	AC0	Rx3	Rx2	Rx1	Rx0
FRQ D,@HL	T7	T6	T5	T4	T3	T2	T1	T0
FRQX D,X	X7	X6	X5	X4	X3	X2	X1	X0

Notes: 1. T0 ~ T7 represents the data of table ROM.
2. X0 ~ X7 represents the data specified in operand X.

The following table shows the bit pattern of the preset letter D

Preset Letter D		Duty Cycle
D1	D0	
0	0	1/4 duty
0	1	1/3 duty
1	0	1/2 duty
1	1	1/1 duty

The following diagram shows the output waveform for different duty cycles.



3.3.2 3Melody Output

The frequency generator may generate the frequency for melody usage. When the frequency generator is used to generate the melody output, the tone table is shown below:

1. The clock source is PH0, i.e. 32,768 Hz
2. The duty cycle is 1/2 Duty (D=2)
3. “FREQ” is the output frequency
4. “ideal” is the ideal tone frequency
5. “%” is the frequency deviation

The following table shows the note table for melody application

Tone	N	FREQ	Ideal	%	Tone	N	FREQ	Ideal	%
C2	249	65.5360	65.4064	0.19	C4	62	260.063	261.626	-0.60
#C2	235	69.4237	69.2957	0.18	#C4	58	277.695	277.183	0.18
D2	222	73.4709	73.4162	0.07	D4	55	292.571	293.665	-0.37
#D2	210	77.6493	77.7817	-0.17	#D4	52	309.132	311.127	-0.64
E2	198	82.3317	82.4069	-0.09	E4	49	327.680	329.628	-0.59
F2	187	87.1489	87.3071	-0.18	F4	46	348.596	349.228	-0.18
#F2	176	92.5650	92.4986	0.07	#F4	43	372.364	369.994	0.64
G2	166	98.1078	97.9989	0.11	G4	41	390.095	391.995	-0.48
#G2	157	103.696	103.826	-0.13	#G4	38	420.103	415.305	1.16
A2	148	109.960	110.000	-0.04	A4	36	442.811	440.000	0.64
#A2	140	116.199	116.541	-0.29	#A4	34	468.114	466.164	0.42
B2	132	123.188	123.471	-0.23	B4	32	496.485	493.883	0.53
C3	124	131.072	130.813	0.20	C5	30	528.516	523.251	1.01
#C3	117	138.847	138.591	0.19	#C5	29	546.133	554.365	-1.48
D3	111	146.286	146.832	-0.37	D5	27	585.143	587.330	-0.37
#D3	104	156.038	155.563	0.31	#D5	25	630.154	622.254	1.27
E3	98	165.495	164.814	0.41	E5	24	655.360	659.255	-0.59
F3	93	174.298	174.614	-0.18	F5	22	712.348	698.456	1.99
#F3	88	184.090	184.997	-0.49	#F5	21	744.727	739.989	0.64
G3	83	195.048	195.998	-0.48	G5	20	780.190	783.991	-0.48
#G3	78	207.392	207.652	-0.13	#G5	19	819.200	830.609	-1.37
A3	73	221.405	220.000	0.64	A5	18	862.316	880.000	-2.01
#A3	69	234.057	233.082	0.42	#A5	17	910.222	932.328	-2.37
B3	65	248.242	246.942	0.53	B5	16	963.765	987.767	-2.43

Note:

1. Above variation does not include X'tal variation.
2. If PH0=65536 Hz, C3 - B5 may have more accurate frequency.

During the application of melody output, sound effect output or carrier output of remote control, the frequency generator needs to combine with the alarm function (BZB, BZ). For detailed information about this application, refer to section 3-4.

3.3.3 Halver/Doubler/Tripler

The halver/doubler/tripler circuits are used to generate the bias voltage for LCD and are composed of a combination of PH2, PH3, PH4, PH5. When the Li battery application is used, the 1/2 VDD voltage generated by the halver operation is supplied to the circuits which are not related to input/output operation.

3.3.4 Alternating Frequency for LCD

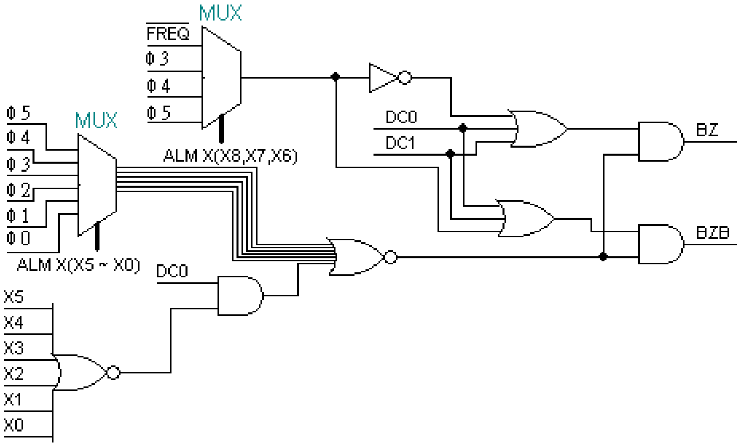
The alternating frequency for LCD is a frequency used to make the LCD waveform.

3.4 BUZZER OUTPUT PINS

There are two output pins, BZB and BZ. Each are MUXed with IOB3 and IOB4 by mask option, respectively. BZB and BZ pins are versatile output pins with complementary output polarity. When buzzer output function combined with the clock source comes from the frequency generator, this output function may generate melody, sound effect or carrier output of remote control.

MASK OPTION table:

Mask Option name	Selected item
SEG30/IOB3/BZB	(3) BZB
SEG31/IOB4/BZ	(3) BZ



This figure shows the organization of the buzzer output.

3.4.1 BASIC BUZZER OUTPUT

The buzzer output (BZ, BZB) is suitable for driving a transistor for the buzzer with one output pin or driving a buzzer with BZ and BZB pins directly. It is capable of delivering a modulation output in any combination of one signal of FREQ, PH3 (4096 Hz), PH4 (2048 Hz), PH5 (1024 Hz) and multiple signals of PH10 (32 Hz), PH11 (16 Hz), PH12(8Hz), PH13 (4 Hz), PH14 (2 Hz), PH15 (1 Hz). The ALM instruction is used to specify the combination. The higher frequency clock is the carrier of modulation output and the lower frequency clock is the envelope of the modulation output.

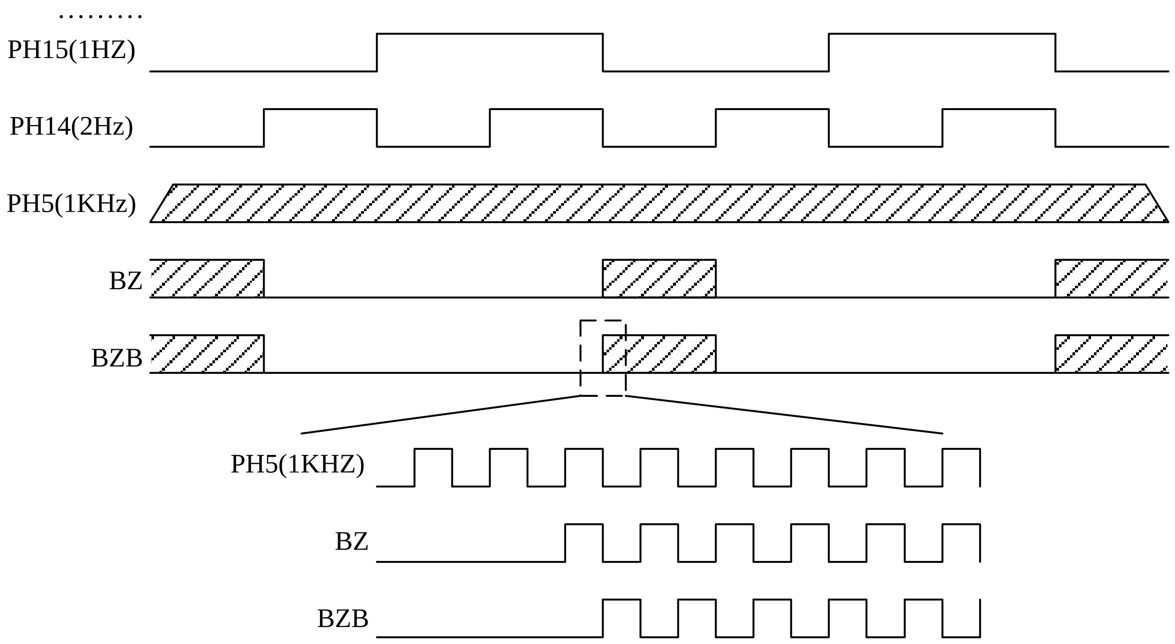
Note:

1. The high frequency clock source should only be one of PH3, PH4, PH5 or FREQ, and the lower frequency may be any/all of the combinations from PH10 ~ PH15.
2. The frequencies in () corresponding to the input clock of the pre-divider (PH0) is 32768Hz.
3. The BZ and BZB pins will output DC0 after the initial reset.

Example:

Buzzer output generates a waveform with 1KHz carrier and (PH15+PH14) envelope.

```
LDS     20h, 0Ah
……….
ALM     70h             ; Output the waveform.
```



In this example, the BZ and BZB pins will generate the waveform as shown in the following figure :

3.4.2 THE CARRIER FOR REMOTE CONTROL

If buzzer output combines with the timer and frequency generator, the output of the BZ pin may deliver the waveform for the IR remote controller. For remote control usage, the setting value of the frequency generator must be greater than or equal to 3, and the ALM instruction must be executed immediately after the FRQ related instructions in order to deliver the FREQ signal to the BZ pin as the carrier for IR remote controller.

Example:

```
SHE         2           ; Enable timer 1 halt release enable flag.
    TMSX    3Fh         ; Set value for timer 1 is 3Fh and the clock source is PH9.
    SCC     40h         ; Set the clock source of  the frequency generator as BCLK.
    FRQX    2, 3        ; FREQ=BCLK/ (4*2), setting value for the frequency generator
                        ; is 3 and duty cycle is 1/2.
    ALM     1C0h        ; FREQ signal is outputted. This instruction must be executed
                        ; after the FRQ related instructions.
    HALT                ; Wait for the halt release caused by timer 1.
    …………………….        ; Halt released.
    ALM     0           ; Stop the buzzer output.
```

3.5 INPUT / OUTPUT PORTS

Three I/O ports are available in TM8763: IOA, IOB and IOC. Each I/O port is composed of 4 bits and has the same basic function.

When the I/O pins are defined as non-IO function by mask option, the input / output function of the pins will be disabled.

3.5.1 IOA PORT

IOA1~IOA4 pins are MUX with CX/SEG24, RR/SEG25, RT/SEG26 and RH/SEG27 pins respectively by mask option.

MASK OPTION table:

Mask Option name	Selected item
SEG24/IOA1/CX	(2) IOA1
SEG25/IOA2/RR	(2) IOA2
SEG26/IOA3/RT	(2) IOA3
SEG27/IOA4/RH	(2) IOA4

In initial reset cycle, the IOA port is set as input mode and each bit of port can be defined as input mode or output mode individually by executing SPA instructions. Executing OPA instructions may output the content of specified data memory to the pins defined as output mode; the pins defined as the input mode will still remain the input mode.

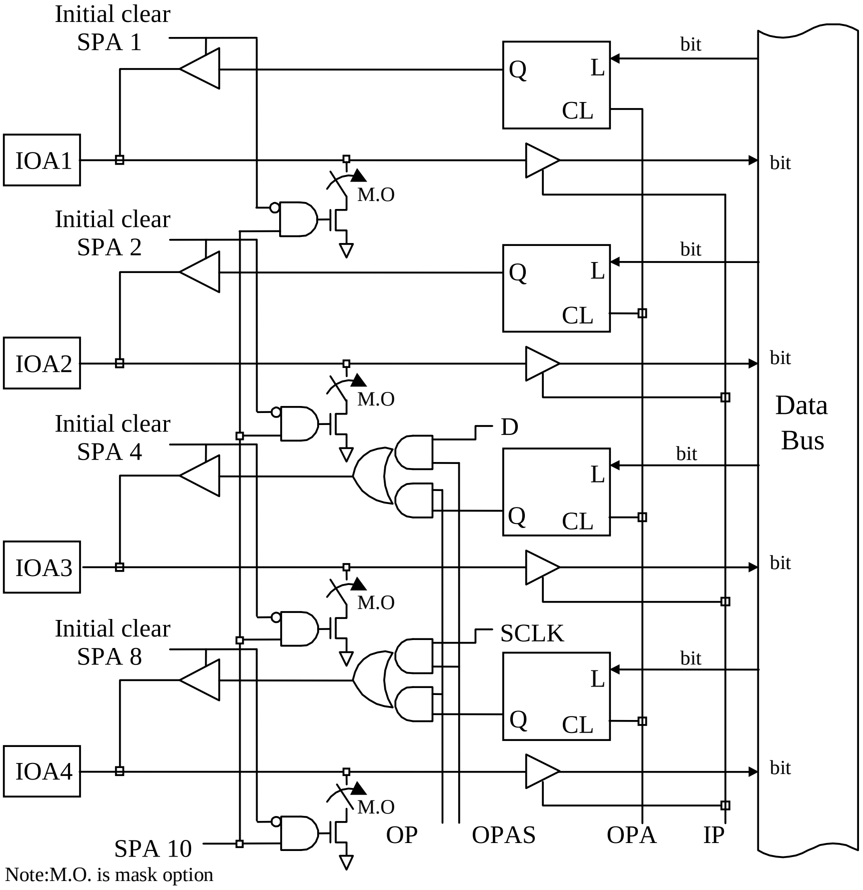
Executing IPA instructions may store the signals applied to the IO pins into the specified data memory. When the IO pins are defined as the output mode, executing IPA instruction will store the content that stored in the latch of the output pin into the specified data memory.

Before executing SPA instruction to define the I/O pins as the output mode, the OPA instruction must be executed to output the data to those output latches beforehand. This will prevent the chattering signal on the I/O pin when the I/O mode changed.

IOA port had built-in pull-down resistor by mask option and executing SPA instruction to enable / disable this device.

MASK OPTION table:

Mask Option name	Selected item
IOA PULL LOW RESISTOR	(1) USE
IOA PULL LOW RESISTOR	(2) NO USE



This figure shows the organization of IOA port.

Note: If the input level is in the floating state, a large current (straight-through current) flows to the input buffer. The input level must not be in the floating state.

3.5.1.1 Pseudo Serial Output

IOA port may operate as a pseudo serial output port by executing OPAS instruction. IOA port must be defined as the output mode before executing OPAS instruction.

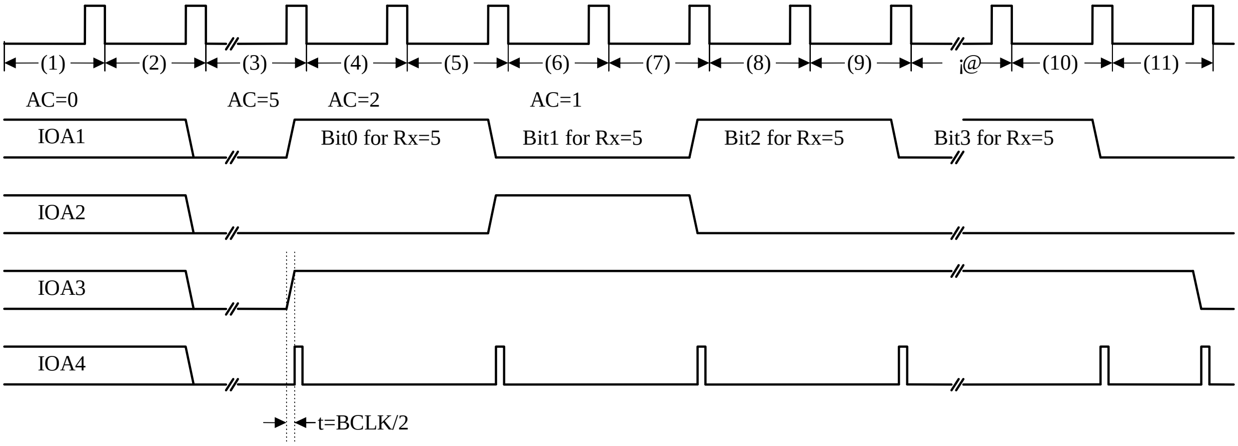
1. BIT0 and BIT1 of the port deliver RAM data.
2. BIT2 of the port delivers the constant value of the OPAS.
3. BIT3 of the port delivers pulses.

Shown below is a sample program using the OPAS instruction.

```
(1)   LDS      0AH, 0
(2)   OPA      0AH
      SPA      0FH
      :
      :
      LDS      1,5
```

```
(3)  OPAS    1,1    ; Bit 0 output, shift gate open
(4)  SR0     1      ; Shifts bit 1 to bit 0
(5)  OPAS    1,1    ; Bit 1 output
(6)  SR0     1      ; Shifts bit2 to bit 0
(7)  OPAS    1,1    ; Bit 2 output
(8)  SR0     1      ; Shifts bit 3 to bit 0
(9)  OPAS    1,1    ; Bit 3 output
     :
     :
(10) OPAS    1,1    ; Last data
(11) OPAS    1,0    ; Shift gate closes
```

The timing chart below illustrates the above program.



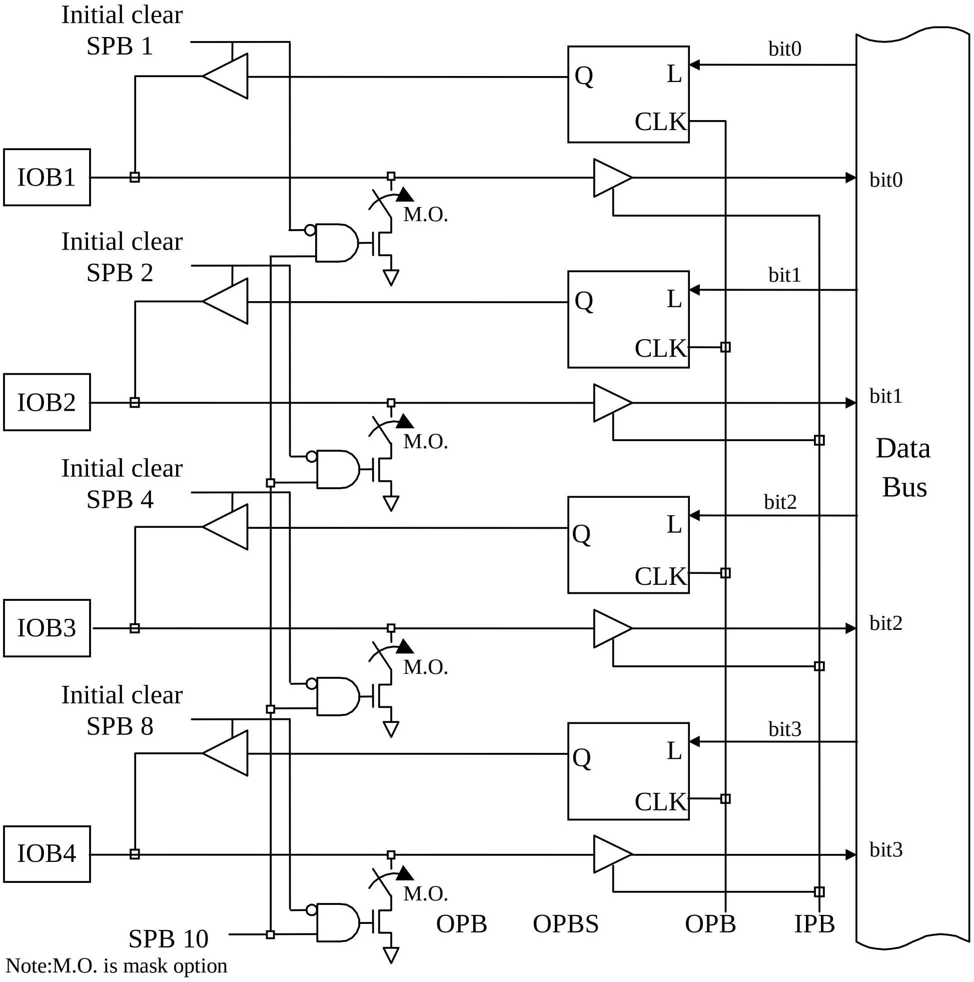
If IOA1 pin is used as the CX pin for RFC function and the other pins (IOA2 ~ IOA3) are used for normal IO pins, IOA1 pin must always be defined as the output mode to avoid the influence from the CX when the input chattering prevention function is active. On the other hand, the RFC counter can receive the signal changes on IOA1 when the RFC counter is enabled.

3.5.2 IOB PORT

IOB1~IOB4 pins are MUXed with ELC/SEG28, ELP/SEG29, BZB/SEG30 and BZ/SEG31 pins respectively by mask option.

MASK OPTION table:

Mask Option name	Selected item
SEG28/IOB1/ELC	(2) IOB1
SEG29/IOB2/ELP	(2) IOB2
SEG30/IOB3/BZB	(2) IOB3
SEG31/IOB4/BZ	(2) IOB4



The following figure shows the organization of IOB port.

Note: If the input level is in the floating state, a large current (straight-through current) flows to the input buffer. The input level must not be in the floating state.

After the reset cycle, the IOB port is set as input and each bit of port can be defined as input or output individually by executing SPB instructions. Executing OPB instructions may output the content of specified data memory to the pins defined as output mode; the other pins which are defined as the input will still be input.

Executed IPB instructions may store the signals applied on the IOB pins into the specified data memory. When the IOB pins are defined as the output, executing IPB instruction will save the data stored in the output latch into the specified data memory.

Before executing SPB instruction to define the I/O pins as output, the OPB instruction must be executed to output the data to the output latches. This will prevent the chattering signal on the I/O pin when the I/O mode changed.

IOB port had built-in pull-down resistor by mask option and executing SPB instruction to enable / disable this device.

MASK OPTION table:

Mask Option name	Selected item
IOB PULL LOW RESISTOR	(1) USE
IOB PULL LOW RESISTOR	(2) NO USE

3.5.3 IOC PORT

IOC1~IOC4 pins are MUXed with KI1/SEG32, KI2/SEG33, KI3/SEG34 and KI4/SEG35 pins respectively by mask option.

MASK OPTION table:

Mask Option name	Selected item
SEG32/IOC1/KI1	(2) IOC1
SEG33/IOC2/KI2	(2) IOC2
SEG34/IOC3/KI3	(2) IOC3
SEG35/IOC4/KI4	(2) IOC4

After the reset cycle, the IOC port is set as input mode and each bit of port can be defined as input mode or output mode individually by executing SPC instruction. Executed OPC instruction may output the content of specified data memory to the pins defined as output; the other pins which are defined as the input will still remain the input mode.

Executed IPC instructions may store the signals applied to the IOC pins in the specified data memory. When the IOC pins are defined as the output, executing IPC instruction will save the data stored in the output latches in the specified data memory.

Before executing SPC instruction to define the IOC pins as output, the OPC instruction must be executed to output the data to those output latches. This will prevent the chattering signal when the IOC pins change to output mode.

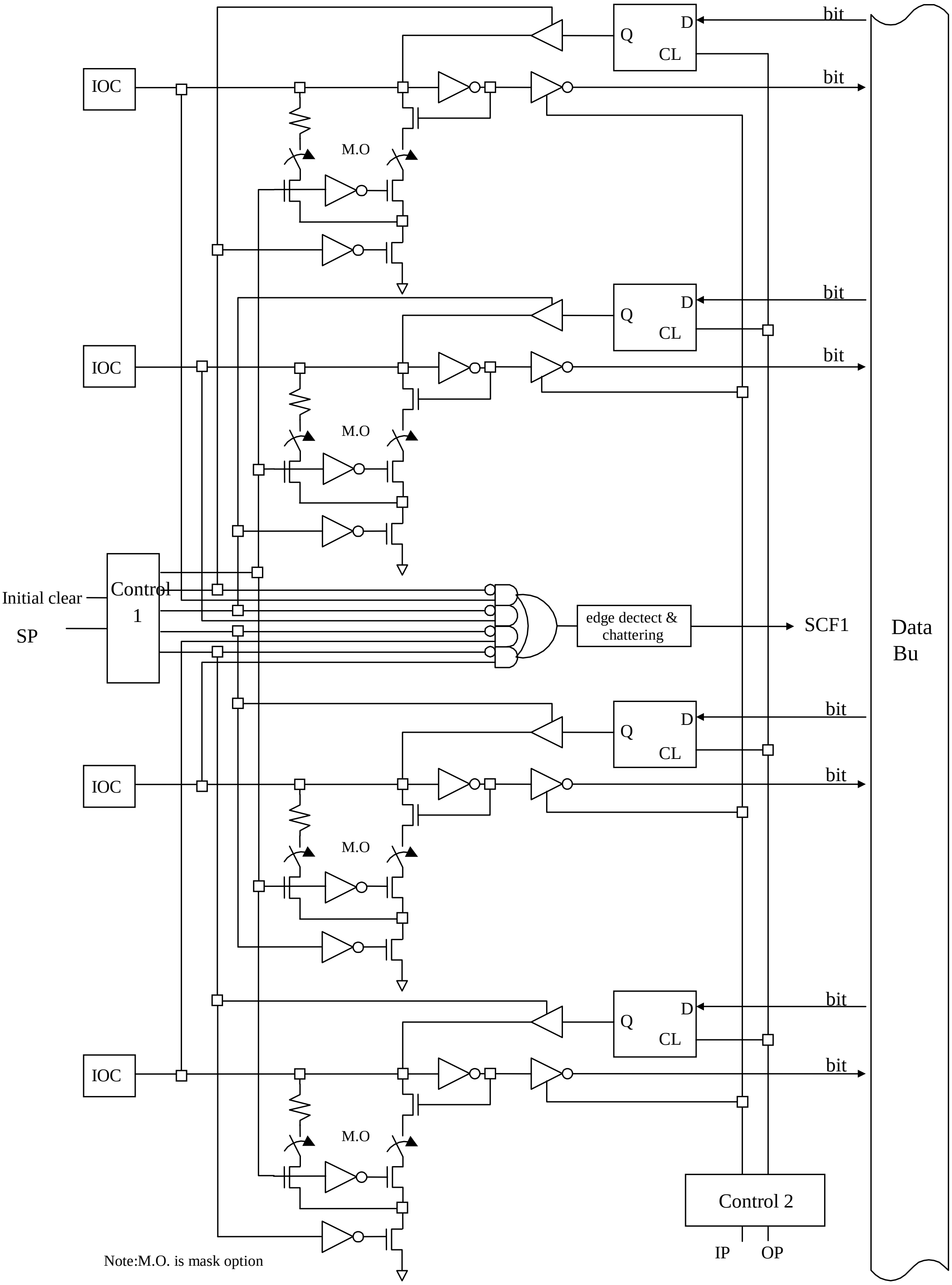
IOC port had built-in pull-down resistor by mask option and executing SPC instruction to enable / disable this device.

MASK OPTION table:

Mask Option name	Selected item
IOC PULL LOW RESISTOR	(1) USE
IOC PULL LOW RESISTOR	(2) NO USE

The pull-down resistor and low-level-hold device in each IOC pin can’t exist in the same time. When the pull-down resistor is enabled, the low-level-hold device will be disable, vise versa. Executing SPC 10h instruction to enable the pull-low device and disable the low-level hold device, executing SPC 0h to disable the pull-low device and enable the low-level hold device.

When the low-level hold device is enabled by mask option, the initial reset will enable the pull-low device and disable the low-level hold device.



This figure shows the organization of IOC port.

Note: If the input level is in the floating state, a large current (straight-through current) flows to the input buffer when both the pull low and L-level hold devices are disabled. The input level must not be in the floating state

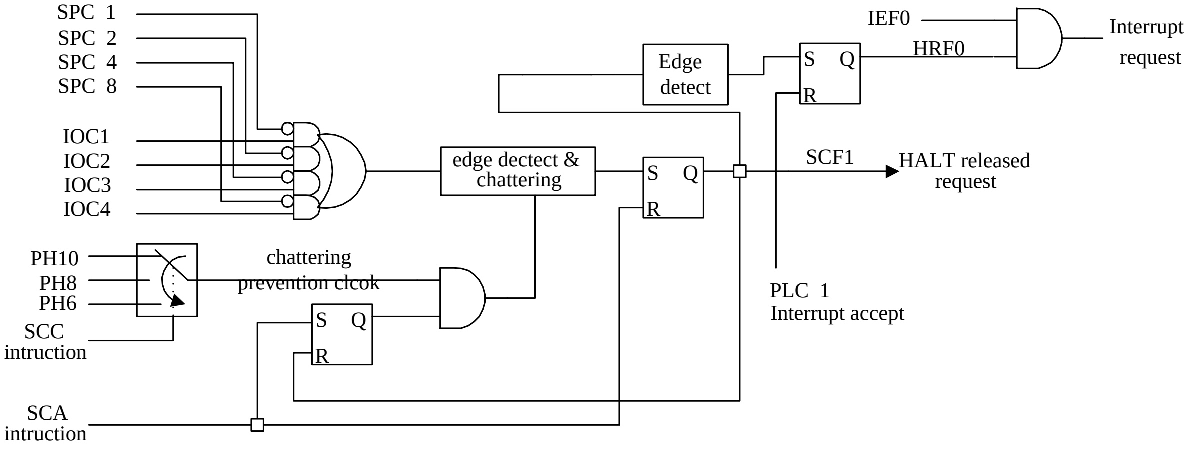
When the IOC pin has been defined as the output mode, both the pull-low and low-level hold devices will be disabled.

Low-level-hold function option

Mask Option name	Selected item
C PORT LOW LEVEL HOLD	(1) USE
C PORT LOW LEVEL HOLD	(2) NO USE

3.5.3.1 Chattering Prevention Function and Halt Release

The port IOC is capable of preventing high / low chattering of the switch signal applied on IOC1 to IOC4 pins. The chattering prevention time can be selected as PH10 (32ms), PH8 (8ms) or PH6 (2ms) by executing SCC instruction, and the default selection is PH10 after the reset cycle. When the pins of the IOC port are defined as output, the signals applied to the output pins will be inhibited for the chattering prevention function. The following figure shows the organization of chattering prevention circuitry.



Note: The default prevention clock is PH10

This chattering prevention function works when the signal at the applicable pin (ex. IOC1) is changed from ”L” level to ”H” level or from ”H” level to ”L” level, and the remaining pins (ex, IOC2 to IOC4) are held at ”L” level.

When the signal changes at the input pins of IOC port specified by the SCA instruction occur and keep the state for at least two chattering clock (PH6, PH8, PH10) cycles, the control circuit at the input pins will deliver the halt release request signal (SCF1). At that time, the chattering prevention clock will stop due to the delivery of SCF1. The SCF1 will be reset to 0 by executing SCA instruction and the chattering prevention clock will be enabled at the same time. If the SCF1 has been set to 1, the halt release request flag 0 (HRF0) will be delivered. In this case, if the port IOC interrupt enable mode (IEF0) is provided, the interrupt is accepted.

Since no flip-flop is available to hold the information of the signal at the input pins IOC1 to IOC4, the input data at the port IOC must be read into the RAM immediately after the halt mode is released.

3.6 EL PANEL DRIVER

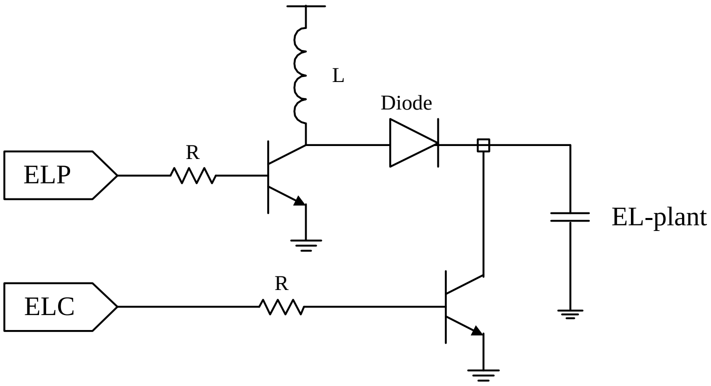
TM8763 provides an EL panel driver for the backlight of the LCD panel. The user can choose different voltage pumping frequencies, duty cycle and ON/OFF frequency to operate, with few external components. This circuitry could generate output voltage up to AC 150V or above for driving the EL-plant; the ELC and ELP output is MUXed with IOB1/SEG28 and IOB2/SEG29, and is selected by mask option.

MASK OPTION table:

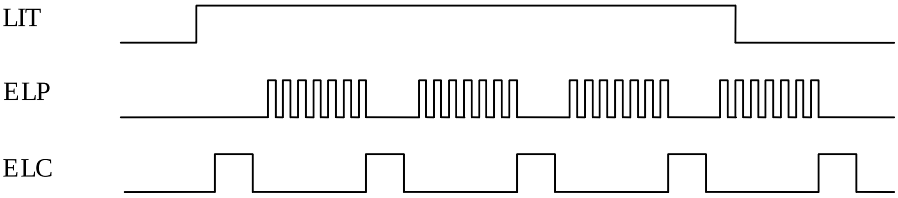
Mask Option name	Selected item
SEG28/IOB1/ELC	(3) ELC
SEG29/IOB2/ELP	(3) ELP

The ELP pin will output clocks to pump voltage to the EL-plant, the ELC pin will output the pulse to discharge the EL-plant. The EL-plant driver will not operate until the light control signal (LIT) is enabled. Once the light control signal (LIT) is enabled, the ELC pin will output a pulse to discharge the capacitor before the pumping clocks output to ELP pin. This will insure that there is no residual voltage that may cause damage while the first pumping clock is applied.

When the light control signal (LIT) is disabled, the ELC pin will output a pulse to discharge the EL-plant after the last pumping clock.



This figure shows the application circuit of EL-plant.



This figure shows the output waveform of EL-plant driver

The ELP/ELC pulse frequency and duty cycle could be defined by executing ELC instruction. When ELC pin outputs the discharge pulse, the clock on ELP pin will be inhibited.

For ELP setting:

(X8, X7, X6)	Pumping clock frequency	(X5, X4)	Duty cycle
000	PH0	00	3/4 duty
100	BCLK	01	2/3 duty
101	BCLK/2	10	1/2 duty
110	BCLK/4	11	1/1 duty
111	BCLK/8		

For ELC setting:

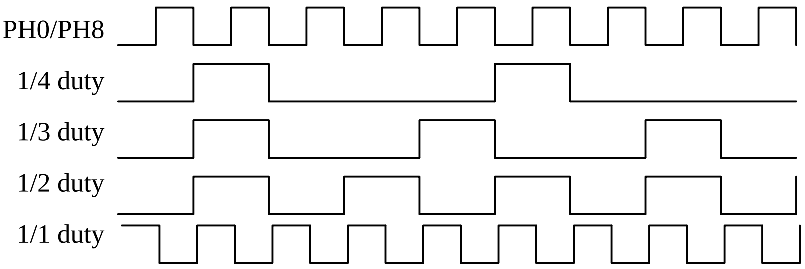
(X3, X2)	Discharge pulse frequency	(X1, X0)	Duty cycle
00	PH8	00	1/4 duty
01	PH7	01	1/3 duty
10	PH6	10	1/2 duty
11	PH5	11	1/1 duty

The default setting after the initial reset is:

ELP: PH0 clock of pre-divider and 3/4 duty cycle

ELC: PH8 clock of pre-divider and 1/4 duty cycle

The timing of the duty cycle is shown below:



Example:

```
ELC     110h    ; ELP outputs BCLK clock with 1/3 duty cycle and ELC outputs ϕ8 clock
                ; with 1/4 duty cycle.
SF      4h      ; Enables the light control signal (LIT) and turns on the EL-light driver.
……………….
RF      4h      ; Disables the light control signal and turns off the EL-light driver.
```

3.7 EXTERNAL INT PIN

The INT pin can be selected as pull-up or pull-down or open type by mask option. The signal change (either rising edge or falling edge by mask option) sets the interrupt flag, delivering the halt release request flag 2 (HRF2). In this case, if the halt release enable flag (HEF2) is provided, the start condition flag 2 is delivered. If the INT pin interrupt enable mode (IEF2) is provided, the interrupt is accepted.

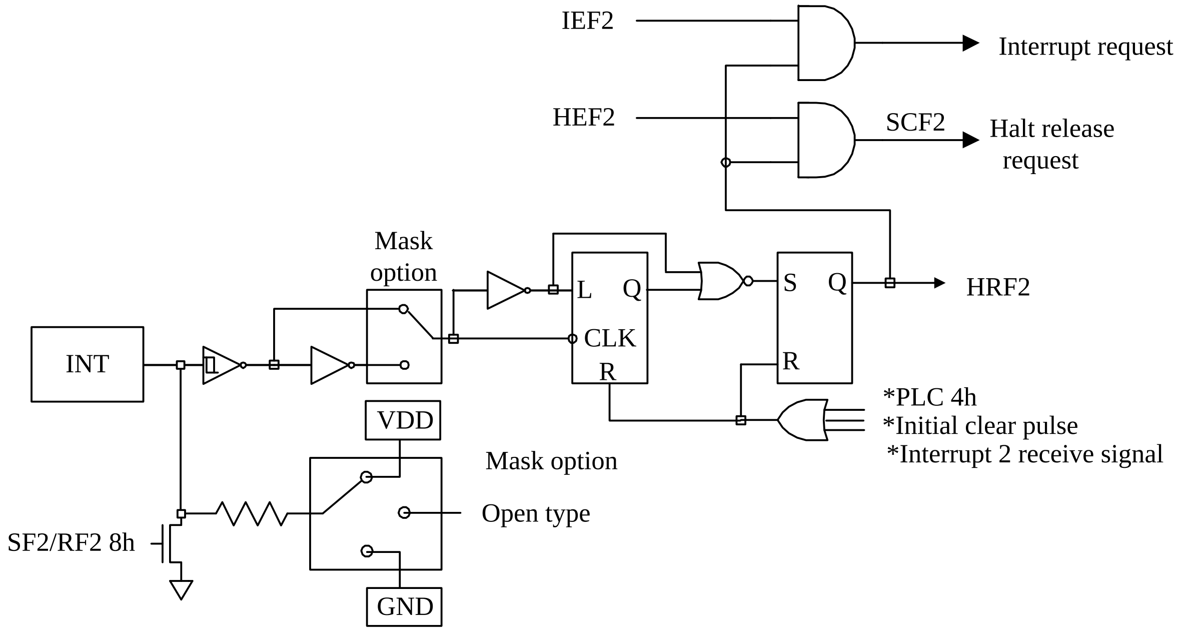
MASK OPTION table:

For internal resistor type:

Mask Option name	Selected item
INT PIN INTERNAL RESISTOR	(1) PULL HIGH
	(2) PULL LOW
	(3) OPEN TYPE

For input triggered type:

Mask Option name	Selected item
INT PIN TRIGGER MODE	(1) RISING EDGE
	(2) FALLING EDGE



Note: For Ag battery power supply, positive power is connected to VDD1; for anything other than Ag battery power supply, it is connected to VDD2.

This figure shows the INT Pin Configuration

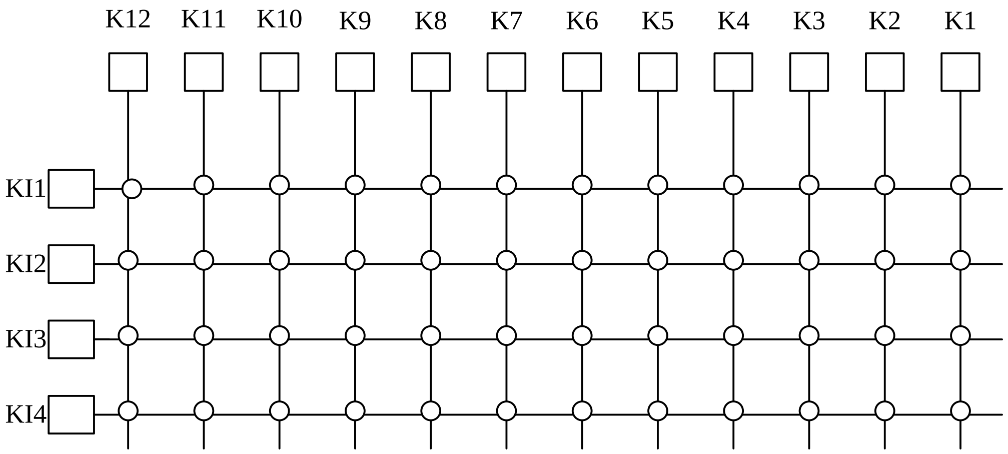
3.8 Key Matrix Scanning

TM8763 shared the timing of LCD/LED waveform to scan the key matrix circuitry and these scanning output pins are SEG1~12(for easy to understand, named these pins as K1 ~ K12). The time sharing of LCD/LED waveform will not affect the display of LCD/LED panel. The input port of key matrix circuitry is composed by KI1 ~ KI4 pins (these pins are muxed with SEG32 ~ SEG35 pins and selected by mask option).

MASK OPTION table:

Mask Option name	Selected item
SEG32/IOC1/KI1	(3) KI1
SEG33/IOC2/KI2	(3) KI2
SEG34/IOC3/KI3	(3) KI3
SEG35/IOC4/KI4	(3) KI4

The typical application circuit of key matrix scanning is shown below:



Executing SPK X instruction could set the scanning type of key matrix. The bit pattern of this instruction is shown below :

Instruction	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SPK X	X7	X6	X5	X4	X3	X2	X1	X0

Bit Patten	Setting : Halt Release by
X6=0	Normal Key Scanning
X6=1	Scanning Cycle

The bit pattern of X (for Key Matrix scanning output to SEG1~12)

X7	X5	X4	X3	X2	X1	X0	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	K11	K12
0	0	0	0	0	0	0	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
0	0	0	0	0	0	1	Hi-z	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
0	0	0	0	0	1	0	Hi-z	Hi-z	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
0	0	0	0	0	1	1	Hi-z	Hi-z	Hi-z	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
0	0	0	0	1	0	0	Hi-z	Hi-z	Hi-z	Hi-z	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
0	0	0	0	1	0	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
0	0	0	0	1	1	0	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
0	0	0	0	1	1	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	Hi-z	Hi-z	Hi-z	Hi-z
0	0	0	1	0	0	0	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	Hi-z	Hi-z	Hi-z
0	0	0	1	0	0	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	Hi-z	Hi-z
0	0	0	1	0	1	0	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	Hi-z
0	0	0	1	0	1	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1
0	0	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	1	0	0	0	0	0	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
1	0	X	0	X	X	X	1	1	1	1	1	1	1	1	Hi-z	Hi-z	Hi-z	Hi-z
1	0	X	1	X	X	X	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	1	1	1
1	1	0	0	0	X	X	1	1	1	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
1	1	0	0	1	X	X	Hi-z	Hi-z	Hi-z	Hi-z	1	1	1	1	Hi-z	Hi-z	Hi-z	Hi-z
1	1	0	1	0	X	X	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	1	1	1
1	1	0	1	1	X	X	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
1	1	1	0	0	0	X	1	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
1	1	1	0	0	1	X	Hi-z	Hi-z	1	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
1	1	1	0	1	0	X	Hi-z	Hi-z	Hi-z	Hi-z	1	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
1	1	1	0	1	1	X	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	1	Hi-z	Hi-z	Hi-z	Hi-z
1	1	1	1	0	0	X	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	1	Hi-z	Hi-z
1	1	1	1	0	1	X	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	1
1	1	1	1	1	X	X	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z

Notes: 1. 1=H/L (LED/LCD)
2. K1~12=SEG1~12 output in scanning interval

When KI1~4 is defined for Key matrix scanning input by mask option, it is necessary to execute SPC instruction to set the internal unused IOC port as output mode before the key matrix scanning function is active. The organization of Key matrix scanning input port is shown in next page.

Once one of KI1~4 pin detected the signal change from "Hi-z" to "1", TM8763 will set HRF5 to 1. If HEF5 had been set to 1 beforehand, this will cause SCF7 to be set and release the HALT mode. After the key scanning cycle finished, the states of KI1 ~ 4 will be latched into the IOC port. Executing the IPC instruction could store these states into data RAM.

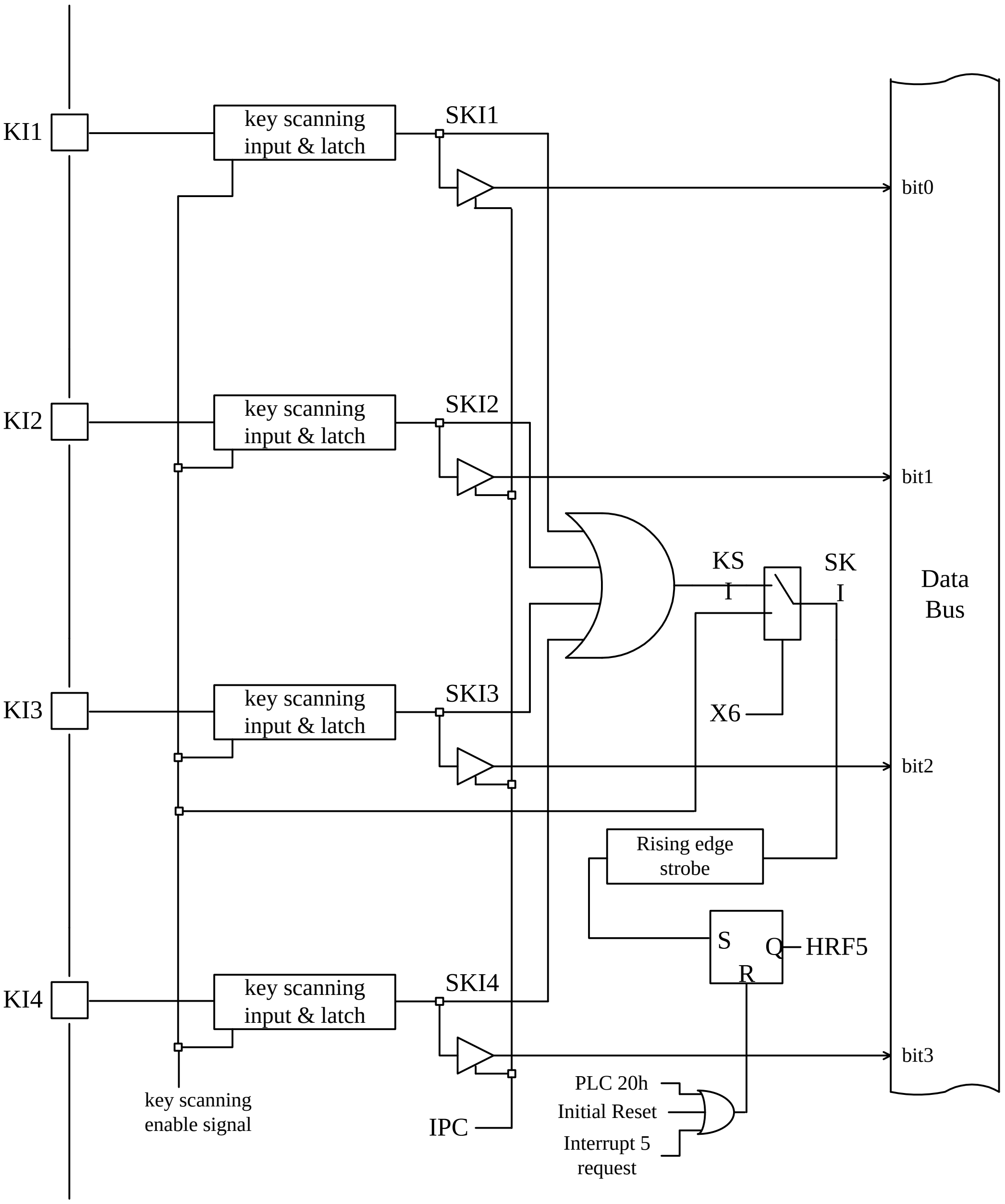
Executing PLC 20h instruction could clear HRF5 flag.

Since the key matrix scanning function shared the timing of LCD/LED waveform, so the scanning frequency is corresponding to LCD frame frequency and LCD duty cycle. The formula for key matrix scanning frequency is shown below:

key matrix scanning frequency (Hz) = (LCD frame frequency) x (LCD duty cycle) x2

Note: "2" is a factor

For example, if the LCD frame frequency is 32Hz, and duty cycle is 1/5 duty, the scanning frequency for key matrix is: 320 Hz (32 x 5 x 2).


KI1
KI2
KI3
KI4
key scanning input & latch
key scanning input & latch
key scanning input & latch
key scanning input & latch
SKI1
SKI2
SKI3
SKI4
bit0
bit1
bit2
bit3
Data Bus
KS I
SK I
X6
Rising edge strobe
S
Q
R
HRF5
PLC 20h
Initial Reset
Interrupt 5 request
key scanning enable signal
IPC

This figure shows the organization of Key matrix scanning input

Example:

```
        SPC     0fh                     ; Disable all the pull-down device on internal IOC port.
                                        ; Set all of the IOC pins as output mode.
        SPK     10h                     ; Generate HALT released request when key depressed
                                        ; Scanning all columns simultaneous in each cycle.
        PLC     20h                     ; Clear HRF5
        SHE     20h                     ; Set HEF5.
        HALT                            ; wait for the halt release caused by key matrix.
        MCX     10h                     ; Check SCF8 (SKI).
         B0     ski_release
        ………….
        …………
ski_release:
        IPC     10h                     ; read KI1~4 input latch state.
        JB0     ki1_release
        JB1     ki2_release
        JB2     ki3_release
        JB3     ki4_release
        .
        .
ki1_release:
        SPK     40h                     ; Check key depressed on K1 column.
        PLC     20h                     ; Clear HRF5 to avoid the false HALT released
        CALL    wait_scan_again         ; Waiting for the next key matrix scanning cycle.
                                        ; The waiting period must longer than key matrix scanning
                                        ; cycle.
        IPC     10h                     ; Read KI1 input latch state.
        JB0     ki1_seg1
        ………….
        ………….
        SPK     4fh                     ; Only enable SEG16 scanning output.
        PLC     20h                     ; Clear HRF5 to avoid the false HALT released
        CALL    wait_scan_again         ; Wait for time over halt LCD clock cycle to sure scan again.
        IPC     10h                     ; Read KI1 input latch state.
        JB0     kil_seg16
        ………….
        ………….
wait_scan_again:
      HALT
      PLC       20h
      RTS
```

4. LCD/LED DRIVER OUTPUT

4.1 LCD DRIVER OUTPUT

When the LED mode option is not selected in the mask option, the entire segment pins and common pins will be used for LCD driver or DC output port.

The number of the LCD driver outputs in TM8763 is 27 segment pins with 5 common pins. All of these output pins could also be used as DC output ports (mask option). If more than one of LCD driver output pin was defined as DC output, the following mask option must be selected.

MASK OPTION table:

When all of SEG and COM pins have been used to drive LCD panel

Mask Option name	Selected item
LCD/LED ACTIVE TYPE	(1) LCD

When more than one of SEG or COM pins had been used for DC output port :

Mask Option name	Selected item
LCD ACTIVE TYPE	(4) O/P

During the initial reset cycle, all of LCD's lighting system may be lighted or unlighted by mask option. All of the LCD output will keep the initial setting until the LCD relative instructions are executed to change the output data.

MASK OPTION table:

Mask Option name	Selected item
LCD DISPLAY IN RESET CYCLE	(1) ON
	(2) OFF

4.1.1 LCD LIGHTING SYSTEM IN TM8763

There are several LCD lighting systems could be selected by mask option in TM8763, they are:

- 1/2 bias 1/2 duty, 1/2 bias 1/3 duty, 1/2 bias 1/4 duty, 1/2bias 1/5duty,
- 1/3 bias 1/3 duty, 1/3 bias 1/4 duty, 1/3 bias 1/5duty,

All of these lighting systems are combined with 2 kinds of mask options, the one is “LCD DUTY CYCLE” and the other is “ LCD BIAS”.

MASK OPTION table:

LCD duty cycle option

Mask Option Name	Selected Item
LCD/LED DUTY CYCLE	(1) O/P
	(2) DUPLEX (1/2 duty)
	(3) 1/3 DUTY
	(4) 1/4 DUTY
	(5) 1/5 DUTY

LCD bias option

Mask Option name	Selected item
LCD BIAS	(3) NO BIAS
	(2) 1/2 BIAS
	(1) 1/3 BIAS

The frame frequency for each lighting system is shown below; these frequencies could be selected by mask option. (All of the LCD frame frequencies in the following tables based on the clock source frequency of the pre-divider (PH0) is 32768 Hz).

The LCD alternating frequency in **duplex (1/2 duty)** type

Mask Option name	Selected item	Remark (alternating frequency)
LCD frame frequency	(1) SLOW	16 Hz
	(2) TYPICAL	32 Hz
	(2) FAST	64 Hz
	(2) O/P	0Hz (LCD not used)

The LCD alternating frequency in **1/3 duty** type

Mask Option name	Selected item	Remark (alternating frequency)
LCD frame frequency	(1) SLOW	21 Hz
	(2) TYPICAL	42 Hz
	(2) FAST	85 Hz
	(2) O/P	0Hz (LCD not used)

The LCD alternating frequency in **1/4 duty** type

Mask Option name	Selected item	Remark (alternating frequency)
LCD frame frequency	(1) SLOW	16 Hz
	(2) TYPICAL	32 Hz
	(2) FAST	64 Hz
	(2) O/P	0Hz (LCD not used)

The LCD alternating frequency in **1/5 duty** type

Mask Option name	Selected item	Remark (alternating frequency)
LCD frame frequency	(1) SLOW	25 Hz
	(2) TYPICAL	51 Hz
	(2) FAST	102 Hz
	(2) O/P	0Hz (LCD not used)

The following table shows the relationship between the LCD lighting system and the maximum number of driving LCD segments.

LCD Lighting System	Maximum Number of Driving LCD Segments	Remarks
Static	27	Connect VDD3 to VDD2
Duplex	54	Connect VDD3 to VDD2
1/2bias 1/3duty	81	Connect VDD3 to VDD2
1/2bias 1/4duty	108	Connect VDD3 to VDD2
1/2bias 1/5duty	135	Connect VDD3 to VDD2
1/3 bias 1/3 duty	81	
1/3 bias 1/4 duty	108	
1/3 bias 1/5 duty	135	

When choosing the LCD frame frequency, it is recommended to choice the frequency that higher than 24Hz. If the frame frequency is lower than 24Hz, the pattern on the LCD panel will start to flash.

4.1.2 DC OUTPUT

TM8763 permits LCD driver output pins (SEG1 ~ SEG27) to be defined as CMOS type DC output or P open-drain DC output ports by mask option. In this case, it is possible to use some LCD driver output pins for DC output and the rest LCD driver output pins for LCD driver. Refer to 4-3-4.

The configurations of CMOS output type and P open-drain type are shown below.

When the LCD driver output pins (SEG) are defined as DC output, the output data on this port will not be affected while the program entered stop mode or LCD turn-off mode.

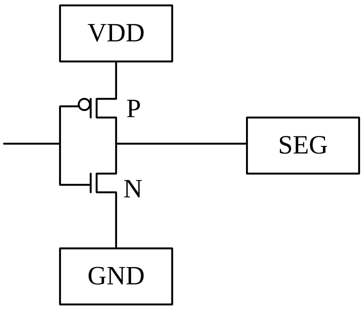


Figure 5-1 CMOS Output Type

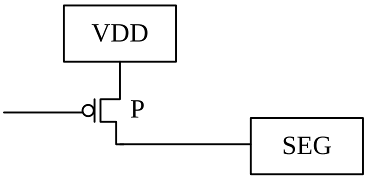


Figure 5-2 P Open-Drain Output Type

4.1.3 SEGMENT PLA CIRCUIT FOR LCD DISPLAY

4.1.3.1 PRINCIPLE OF OPERATION OF LCD DRIVER SECTION

Explained below is how the LCD driver section operates when the instructions are executed.

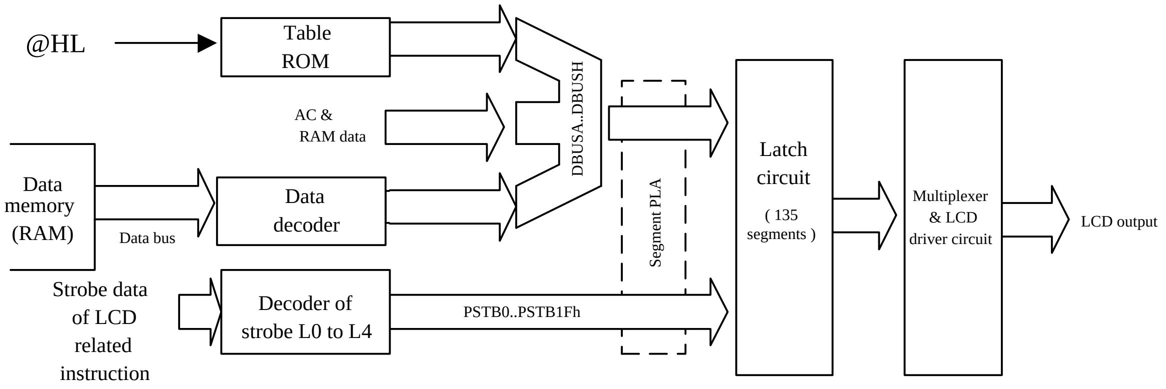


Figure 5-3 Principal Drawing of LCD Driver Section

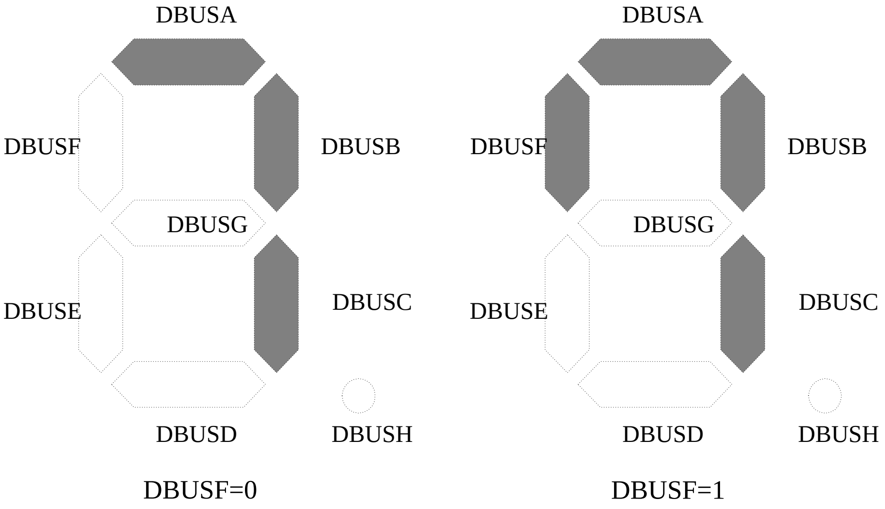
The LCD driver section consists of the following units:

- Data decoder to decode data supplied from RAM or table ROM
- Latch circuit to store LCD lighting information
- L0 to L4 decoder to decode the Lz-specified data in the LCD-related instructions which specifies the strobe of the latch circuit
- Multiplexer to select 1/2duty, 1/3duty, 1/4duty, 1/5duty
- LCD driver circuitry
- Segment PLA circuit connected between data decoder, L0 to L4 decoder and latch circuit.

The data decoder is used for decoding the content of the working register specified in LCD-related instructions as 7-segment pattern on LCD panel. The decoding table is shown below:

Content of data memory	Output of data decoder							
	DBUSA	DBUSB	DBUSC	DBUSD	DBUSE	DBUSF	DBUSG	DBUSH
0	1	1	1	1	1	1	0	1
1	0	1	1	0	0	0	0	1
2	1	1	0	1	1	0	1	1
3	1	1	1	1	0	0	1	1
4	0	1	1	0	0	1	1	1
5	1	0	1	1	0	1	1	1
6	1	0	1	1	1	1	1	1
7	1	1	1	0	0	*note	0	1
8	1	1	1	1	1	1	1	1
9	1	1	1	1	0	1	1	1
A-F	0	0	0	0	0	0	0	0

*** Note:** The DBUSF of decoded output can be selected as 0 or 1 by mask option. The LCD pattern of this option is shown below:



The following table shows the option table for displaying digit “7” pattern:

MASK OPTION table:

Mask Option name	Selected item
F SEGMENT FOR DISPLAY “ 7 “	(1) ON
F SEGMENT FOR DISPLAY “ 7 “	(2) OFF

Both LCT and LCB instructions use the data-decoder table to decode the content of data memory that specified. When the content of data memory that specified by LCB instruction is “0”, the decoded output of DBUSA ~ DBUSH are all “0”. (this is used for blanking the leading digit ”0” on LCD panel).

The LCP instruction transferred the data of the RAM (Rx) and accumulator (AC) directly from ” DBUSA” to ” DBUSH” without passing through the data decoder.

The LCD instruction transfers the table ROM data (T@HL) directly from ”DBUSA” to ”DBUSH” without passing through the data decoder.

Table 2- 2 The mapping table of LCP and LCD instructions

	DBUSA	DBUSB	DBUSC	DBUSD	DBUSE	DBUSF	DBUSG	DBUSH
LCP	Rx0	Rx1	Rx2	Rx3	AC0	AC1	AC2	AC3
LCD	T@HL0	T@HL1	T@HL2	T@HL3	T@HL4	T@HL5	T@HL6	T@HL7

There are 8 data decoder outputs of ”DBUSA” to ”DBUSH” and 32 L0 to L4 decoder outputs of PSTB 0h to PSTB 1Fh. The input data and clock signal of the latch circuit are ”DBUSA” to ”DBUSH” and PSTB 0h to PSTB 1Fh, respectively. Each segment pin has 5 latches corresponding to COM1-5.

The segment PLA performs the function of combining ”DBUSA” to ”DBUSH” inputs to each latch and strobe PSTB 0h to PSTB 1Fh is selected freely by mask option.

Of 256 signals obtainable by combining ”DBUSA” to ”DBUSH” and PSTB 0h to PSTB 1Fh, any 175 (corresponding to the number of latch circuits incorporated in the hardware) signals can be selected by programming and the above-mentioned segment PLA. Table 2-7 shows the PSTB 0h to PSTB 1Fh signals concretely.

Table 2- 3 Strobe Signal for LCD Latch in Segment PLA and Strobe in LCT Instruction

strobe signal for LCD latch	Strobe in LCT, LCB, LCP, LCD instructions The values of Lz in"LCT Lz, Q": *
PSTB 0	0H
PSTB 1	1H
PSTB 2	2H
PSTB 3	3H
PSTB 4	4H
PSTB 5	5H
…………	…………….
PSTB 1Ah	1AH
PSTB 1Bh	1BH
PSTB 1Ch	1CH
PSTB 1Dh	1DH
PSTB 1Eh	1EH
PSTB 1Fh	1FH

Note: The values of Q are the addresses of the working register in the data memory (RAM). In the LCD instruction, Q is the index address in the table ROM.

The LCD outputs could be turned off without changing the segment data. Executed SF2 4h instruction could turn off the display simultaneously and executed RF2 4h could turn on the display with the patterns before turned off. These two instructions will not affect the content stored in the latch circuitry. When the LCD is turned off by executing RF2 4h instruction, the program could still execute LCT, LCB, LCP and LCD instructions to update the content in the latch circuitry and the new content will be outputted to the LCD while the display is turned on again.

In stop state, all COM and SEG outputs of LCD driver will automatically switch to the GND state to avoid the DC voltage bias on the LCD panel.

4.1.3.2 Instructions

1. LCT Lz, Ry

Decodes the content specified in Ry with the data decoder and transfers the DBUSA ~ H to LCD latch specified by Lz.

2. LCB Lz, Ry

Decodes the content specified in Ry with the data decoder and transfers the DBUSA ~ H to LCD latch specified by Lz. The "DBUSA" to "DBUSH" are all 0 when the input data of the data decoder is 0.

3. LCD Lz, @HL

Transfers the table ROM data specified by @HL directly to "DBUSA" to "DBUSH" without passing through the data decoder. The mapping table is shown in table 2-32.

4. LCP Lz, Ry

The data of the RAM and accumulator (AC) are transferred directly to ”DBUSA” to ”DBUSH” without passing through the data decoder. The mapping table is shown below:

5. LCT Lz, @HL

Decodes the index RAM data specified in @HL with the data decoder and transfers the DBUSA ~ H to LCD latch specified by Lz.

6. LCB Lz, @HL

Decodes the index RAM data specified in @HL with the data decoder and transfers the DBUSA ~ H to LCD latch specified by Lz. The “DBUSA” to “DBUSH” are all 0 when the input data of the data decoder is 0.

7. LCP Lz, @HL

The data of the index RAM and accumulator (AC) are transferred directly to ”DBUSA” to ”DBUSH” without passing through the data decoder. The mapping table is shown below:

Table 2- 4 The mapping table of LCP and LCD instructions

	DBUSA	DBUSB	DBUSC	DBUSD	DBUSE	DBUSF	DBUSG	DBUSH
LCP	Rx0	Rx1	Rx2	Rx3	AC0	AC1	AC2	AC3
LCD	T@HL0	T@HL1	T@HL2	T@HL3	T@HL4	T@HL5	T@HL6	T@HL7

8. SF2 4h

Turns off the LCD display.

9. RF2 4h

Turns on the LCD display.

4.1.3.3 CONCRETE EXPLANATION

Each LCD driver output corresponds to the LCD 1/5 duty panel and has 5 latches (refer to Figure : Sample Organization of Segment PLA Option). Since the latch input and the signal to be applied to the clock (strobe) are selected with the segment PLA, the combination of the segments in the LCD driver outputs is flexible. In other words, one of the data decoder outputs “DBUSA” to “DBUSH” is applied to the latch input L, and one of the PSTB0 to PSTB1Fh outputs are applied to clock CLK.

Refer to Chapter 5 for detail description of these instructions.

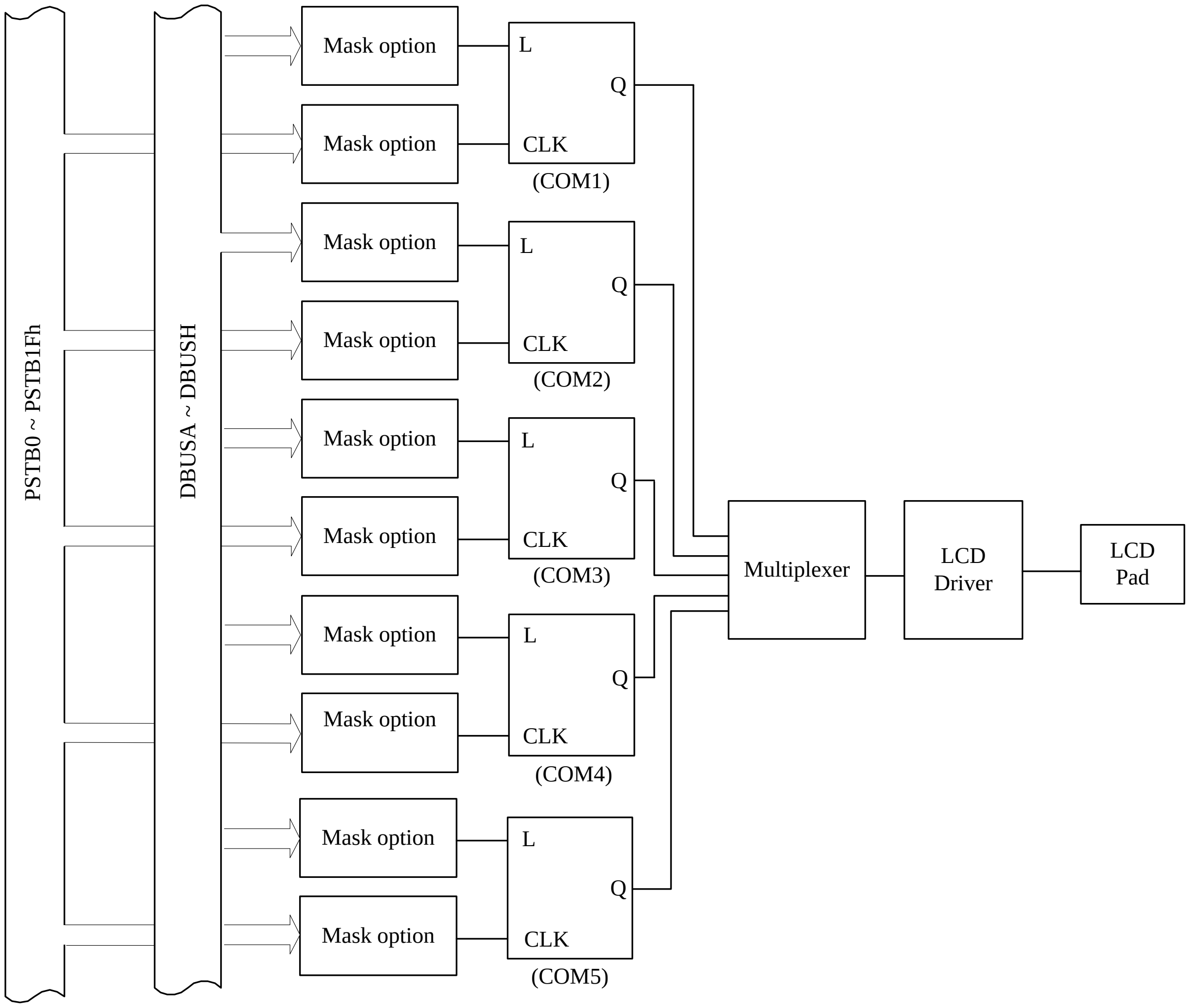


Figure : Sample Organization of Segment PLA Option

4.1.3.4 THE CONFIGURATION FILE FOR MASK OPTION

When configuring the mask option of LCD PLA, the *.cfg file provides the necessary format for editing the LCD/LED configuration.

The syntax in *.cfg file is as follows:

SEG COM PSTB DBUS

SEG: Specifies the segment pin

COM: Specifies the corresponding latch in each segment pin. Only 0, 1, 2, 3, 4, 5 and 9 can be entered into this field.”0”is for CMOS type DC output option and”9”is for P open-drain DC output option.

PSTB: Specifies the strobe data for the latch.

DBUS: Specifies the DBUS data for the latch.

4.2 LED DRIVER OUTPUT

When the LED mode is selected by the mask option, all of the segments and common pins will be used for LED display. The number of the LED driver outputs is 27 segment pins with 5 common pins (COM).

For LED driver outputs (COM), mask option can be used to select active low LED display or active high LED display, and there are static, 1/2 duty, 1/3 duty, 1/4duty or 1/5 duty lighting systems could be used. There is no bias issue in LED mode, so please select the 1/2 bias or No bias as the bias system.

In LED mode, all of the segment output pins (SEG) work only in low active mode.

MASK OPTION table:

When COM pins have been used to drive high active LED panel

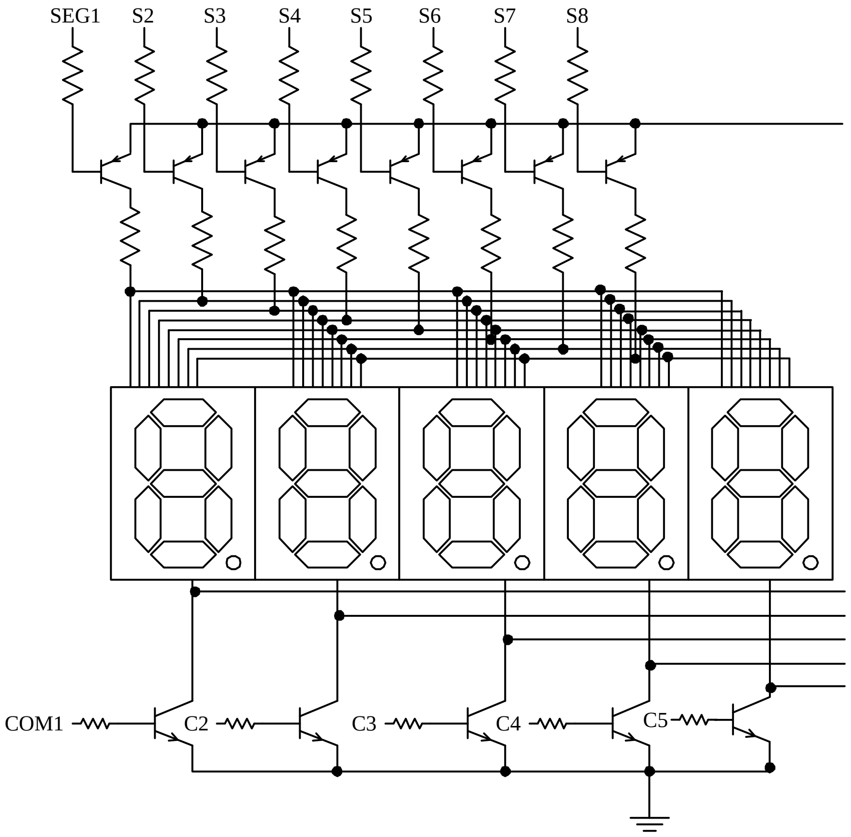
Mask Option name	Selected item
LCD/LED ACTIVE TYPE	(2) LED HIGH ACTIVE

When COM pins have been used to drive low active LED panel

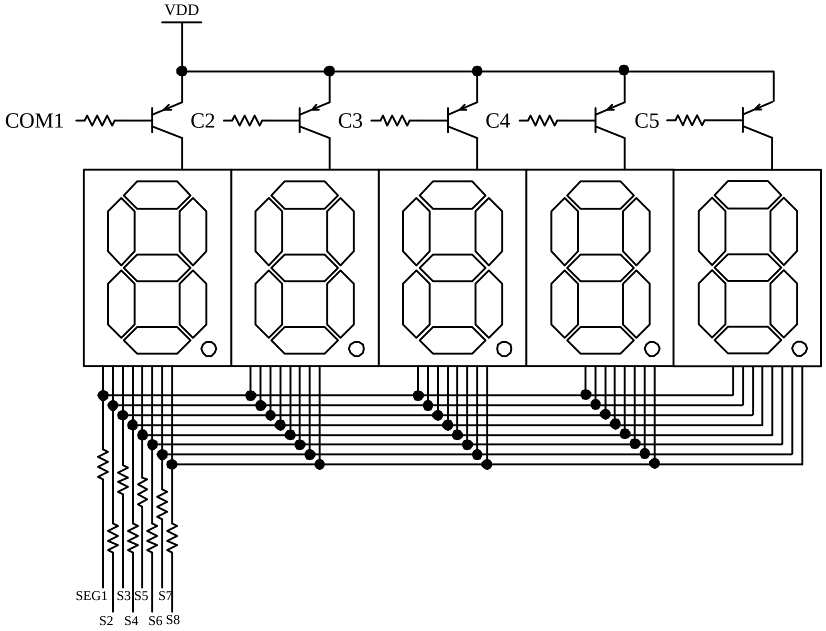
Mask Option name	Selected item
LCD/LED ACTIVE TYPE	(3) LED LOW ACTIVE

The following schema will illustrate the application difference between high active mode and low active mode:

(1) High Active Mode



(2) Low Active Mode



Note : Due to the sink current capability, if all segment more than “8” turn on at Low Active Mode in the same time, the IC may be burn down (if one segment has 5mA, then eight segments have 40mA).

Mask option can also be used to select the alternating frequency. (All of the LED alternating frequencies based on the clock source frequency of the pre-divider are 32768 Hz.)

The LED alternating frequency in 1/2 duty mode

LED duty cycle	1/2 duty		
Mask option	Slow	Typ.	Fast
LED alternating frequency	32 Hz	64 Hz	128 Hz

The LED alternating frequency in 1/3 duty mode

LED duty cycle	1/3 duty		
Mask option	Slow	Typ.	Fast
LED alternating frequency	43 Hz	84 Hz	120 Hz

The LED alternating frequency in 1/4 duty mode

LED duty cycle	1/4 duty		
Mask option	Slow	Typ.	Fast
LED alternating frequency	32 Hz	64 Hz	128 Hz

The LED alternating frequency in 1/5 mode

LED duty cycle	1/5 duty		
Mask option	Slow	Typ.	Fast
LED alternating frequency	51 Hz	102 Hz	205 Hz

Mask option permits LED driver output pins (SEG) to be used for CMOS type DC output or P open-drain DC output ports. In this case, it is possible to use some LED driver output pins for DC output ports and the remaining LED driver output pins for LED driver outputs.

In the LCD configuration file (*.cfg), with the”0”data listed in the ”COM” column, the segment pin will be defined as the CMOS type output port. When the “9” data is listed in the ”COM” column, the segment pin will be defined as the P open-drain type output port.

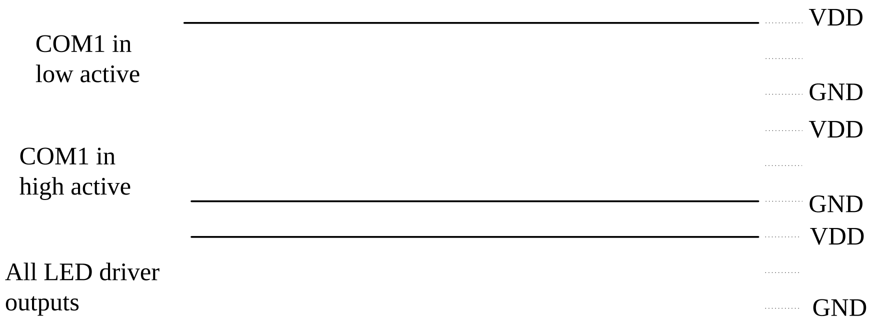
The SEG1 ~ SEG12,SEG21~SEG35 of LED driver outputs can be selected as CMOS type or P open-drain type output by mask option. When the LED driver output pin (SEG) has been defined as the DC output port, the output data will not be affected when the STOP mode or LED turn-off mode is active.

During reset mode all of LED's outputs will be unlighted by default setting as this setting may prevent large power consumption during the initial clear cycle. All of the LED output data will keep the initial setting until the LED related instructions are executed in the program.

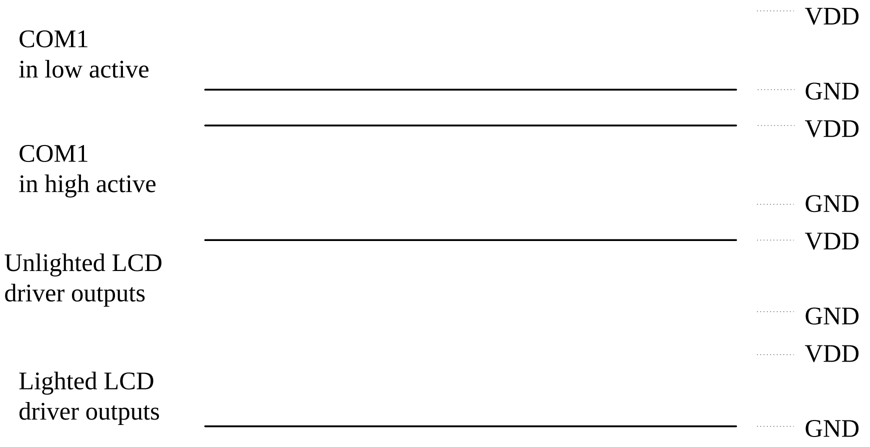
The output waveform of the common output and LED driver output for each LED lighting system are shown below.

4.2.1 STATIC LIGHTING SYSTEM FOR LED DRIVER

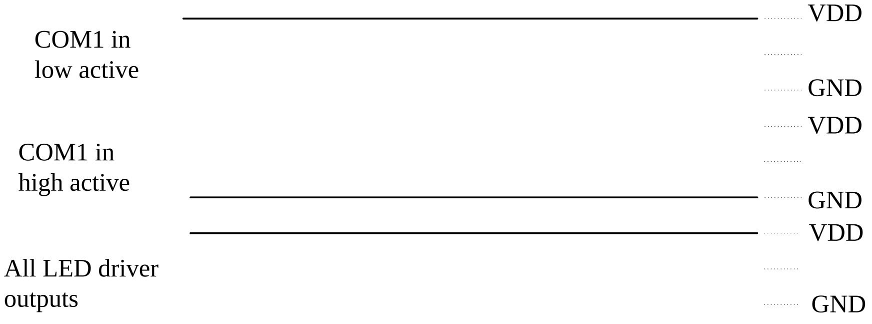
(i) Initial clear mode



(ii) Normal operation mode



(iii)Display Turn Off



(iv)STOP Mode

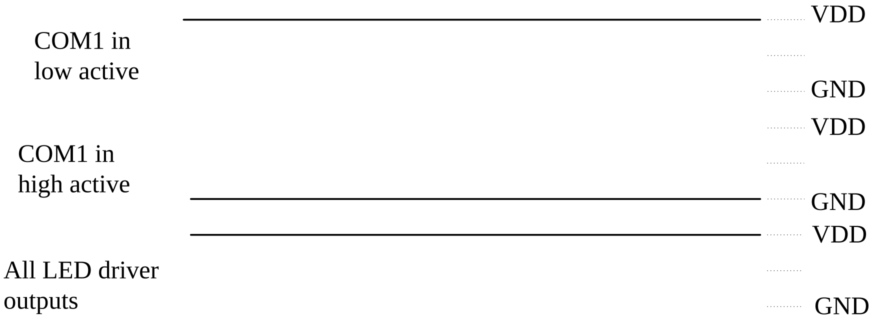
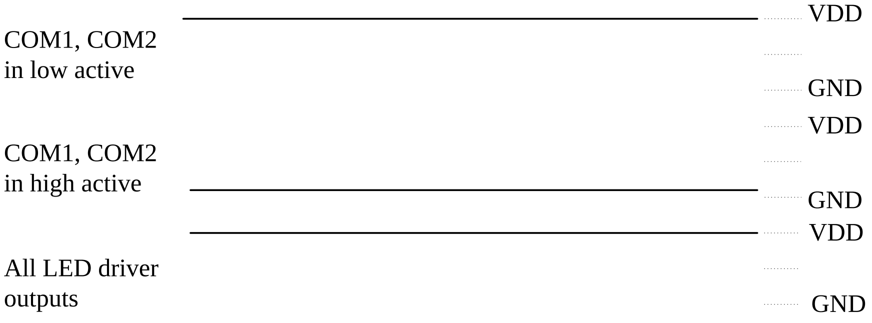


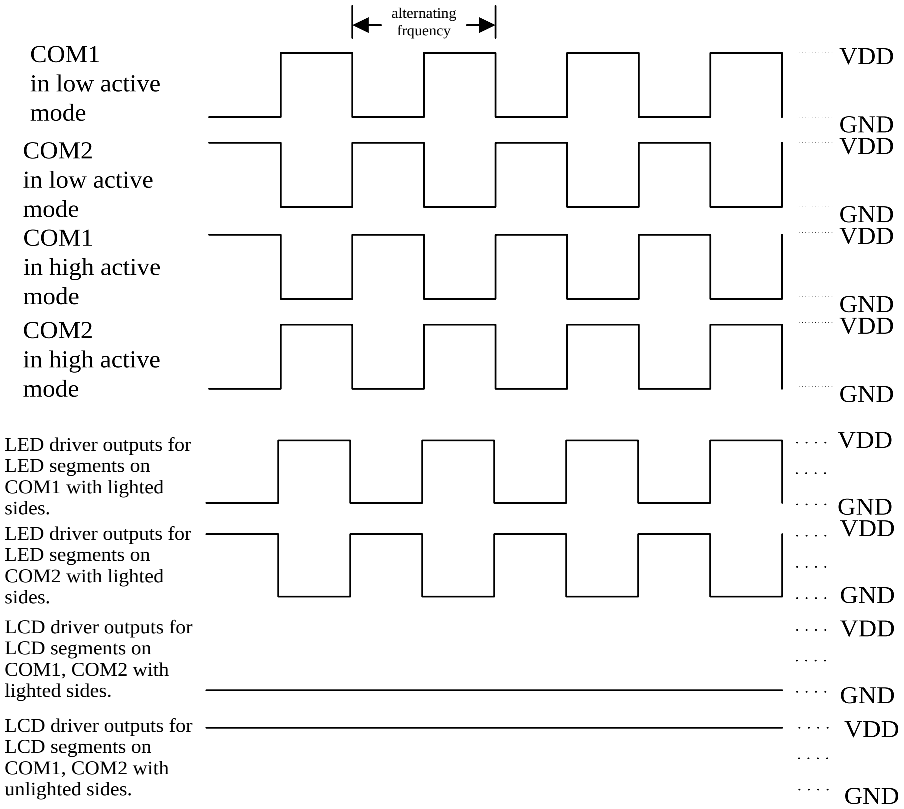
Figure 2- 39 Static LED Waveform

4.2.2 1/2 DUTY LIGHTING SYSTEM FOR LED DRIVER

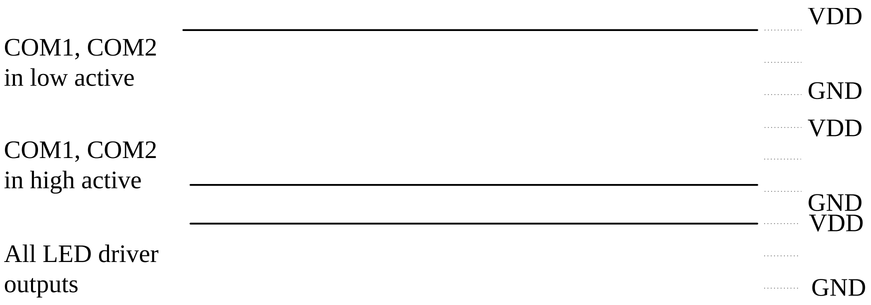
(i) Initial clear mode



(ii) Normal operation mode


alternating
frquency
COM1
in low active
mode
COM2
in low active
mode
COM1
in high active
mode
COM2
in high active
mode
LED driver outputs for LED segments on COM1 with lighted sides.
LED driver outputs for LED segments on COM2 with lighted sides.
LCD driver outputs for LCD segments on COM1, COM2 with lighted sides.
LCD driver outputs for LCD segments on COM1, COM2 with unlighted sides.
VDD
GND
VDD
GND
VDD
GND
VDD
GND
VDD
GND
VDD
GND
VDD
GND
VDD
GND

(iii)Display Turn Off



(iv)STOP Mode

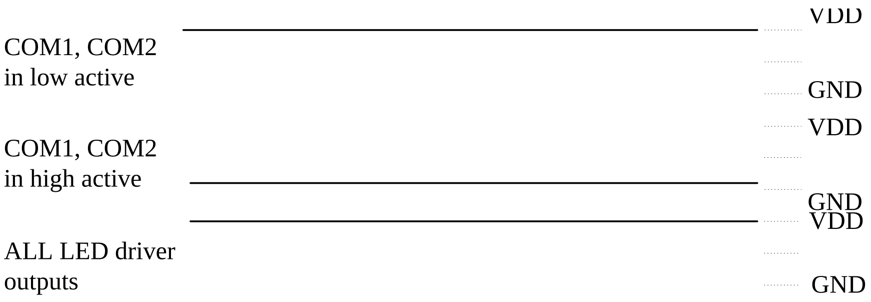
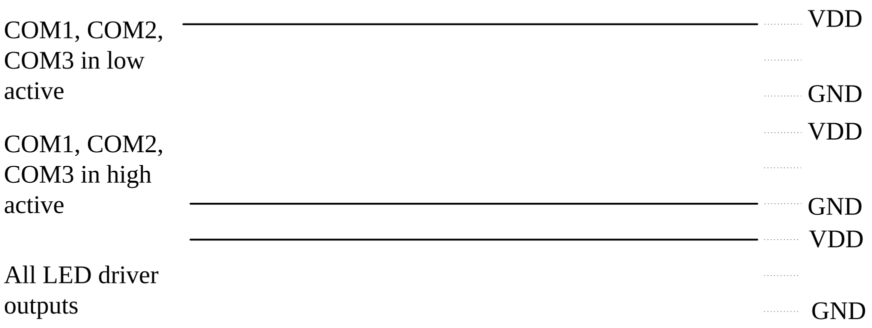


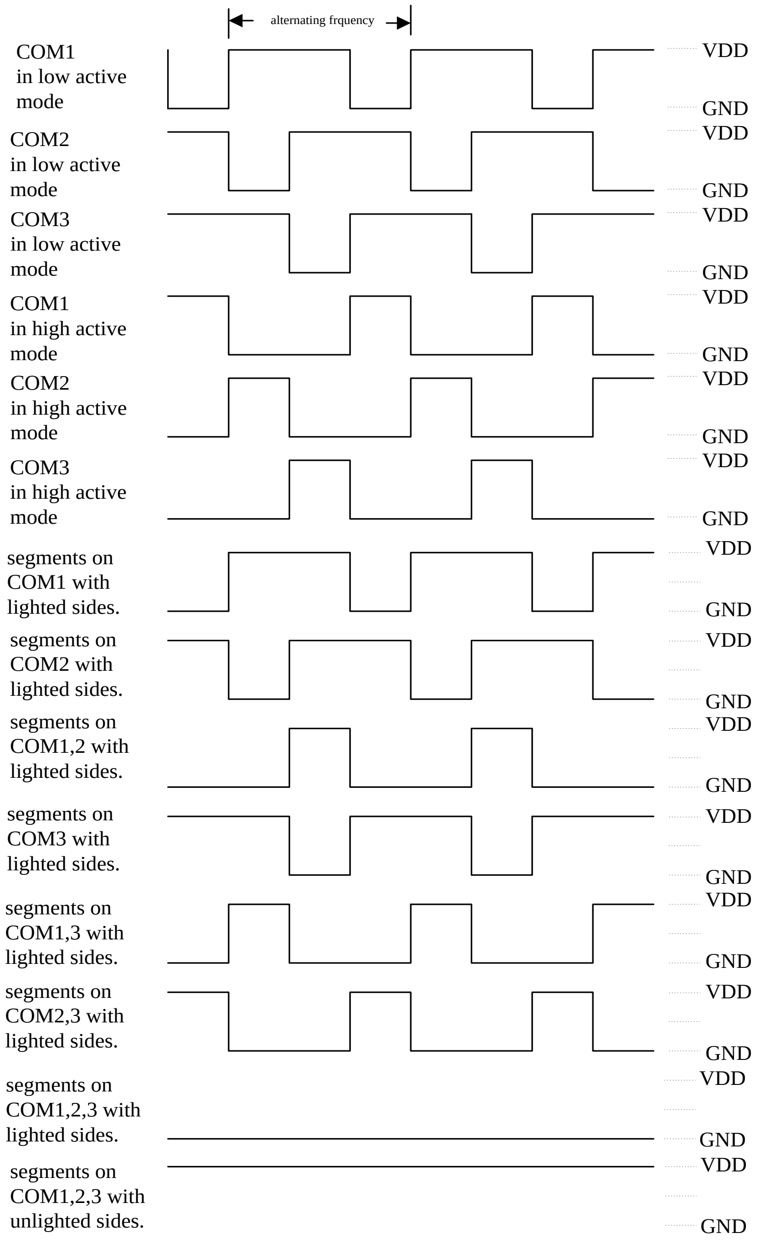
Figure 2- 40 Duplex (1/2 duty) LED Waveform

4.2.3 1/3 DUTY LIGHTING SYSTEM FOR LED DRIVER

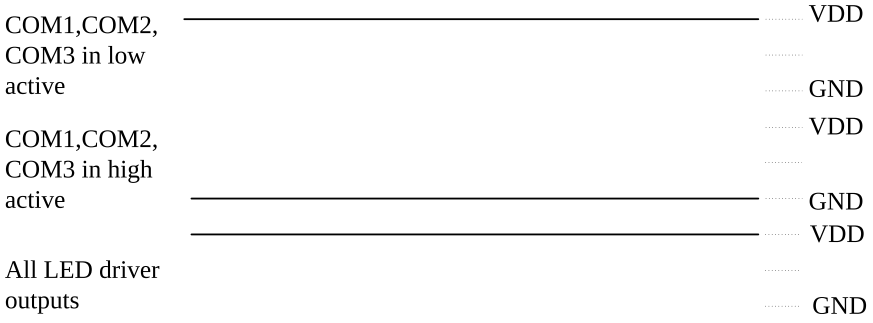
(i) Initial clear mode


COM1, COM2, COM3 in low active
VDD
GND
COM1, COM2, COM3 in high active
VDD
GND
All LED driver outputs
VDD
GND

(ii) Normal operation mode


alternating frquency
COM1 in low active mode
VDD
GND
COM2 in low active mode
VDD
GND
COM3 in low active mode
VDD
GND
COM1 in high active mode
VDD
GND
COM2 in high active mode
VDD
GND
COM3 in high active mode
VDD
GND
segments on COM1 with lighted sides.
VDD
GND
segments on COM2 with lighted sides.
VDD
GND
segments on COM1,2 with lighted sides.
VDD
GND
segments on COM3 with lighted sides.
VDD
GND
segments on COM1,3 with lighted sides.
VDD
GND
segments on COM2,3 with lighted sides.
VDD
GND
segments on COM1,2,3 with lighted sides.
VDD
GND
segments on COM1,2,3 with unlighted sides.
VDD
GND

(iii) Display Turn Off



(iv) STOP mode

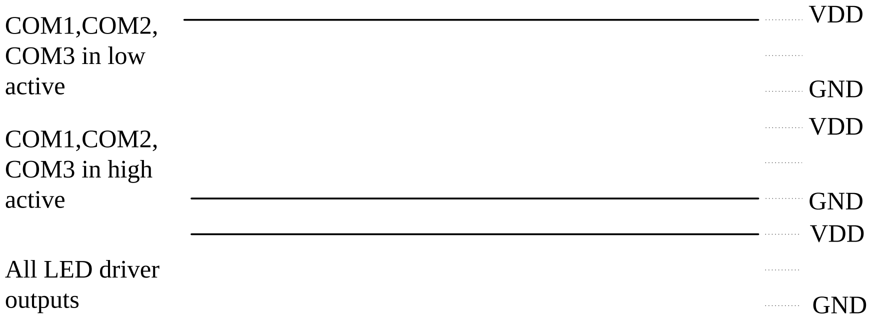
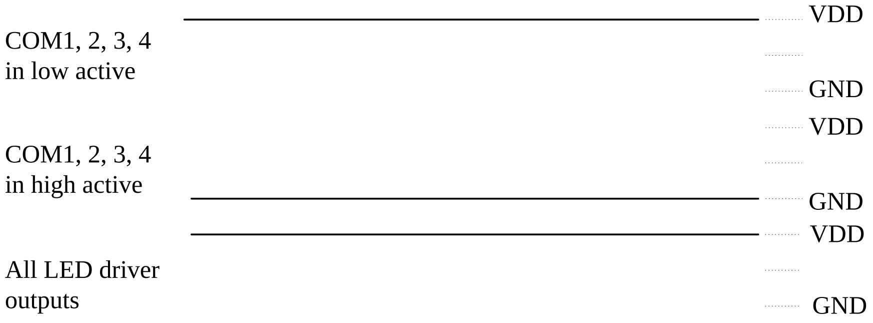


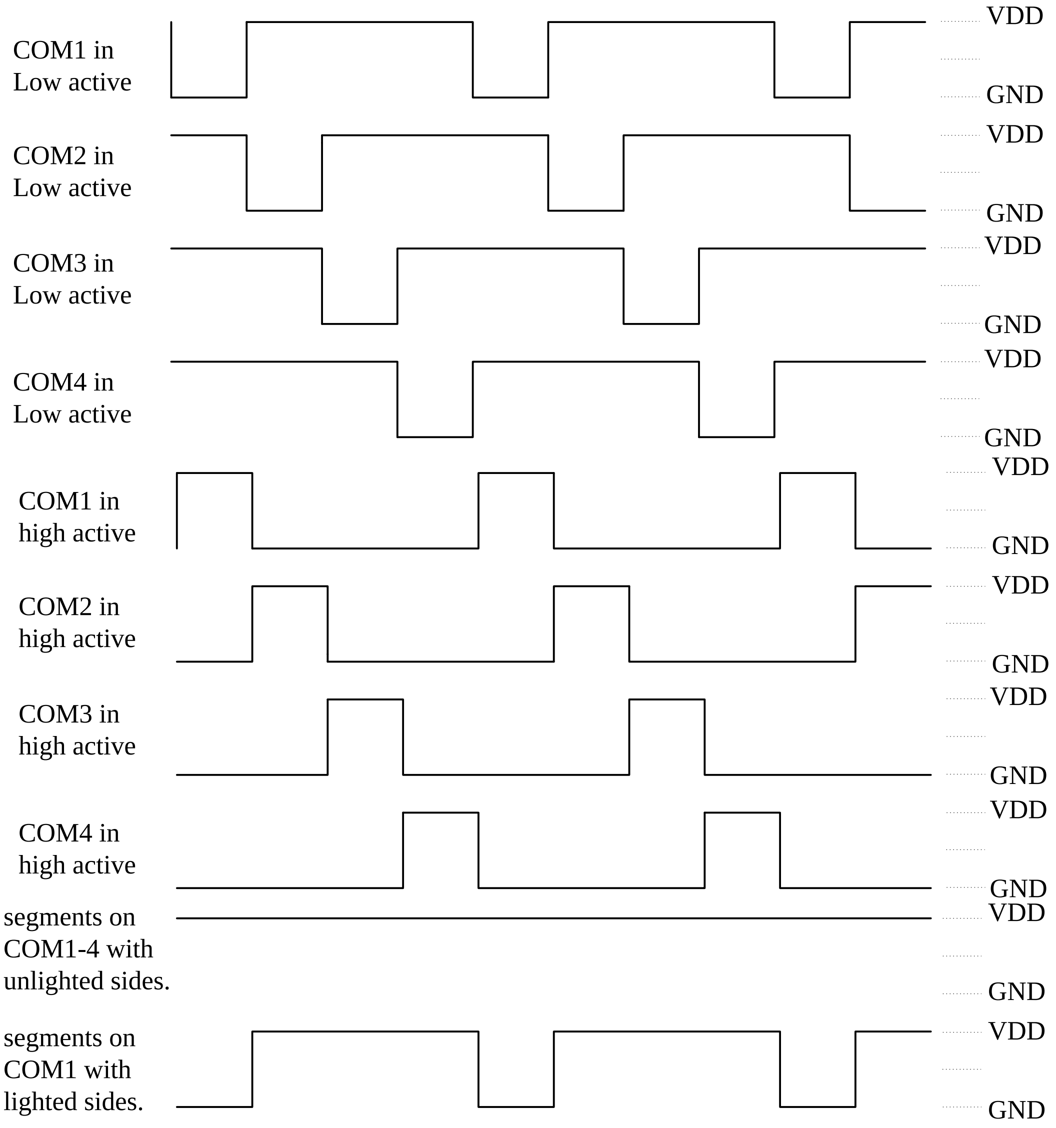
Figure 2- 41 1/3duty LED Waveform

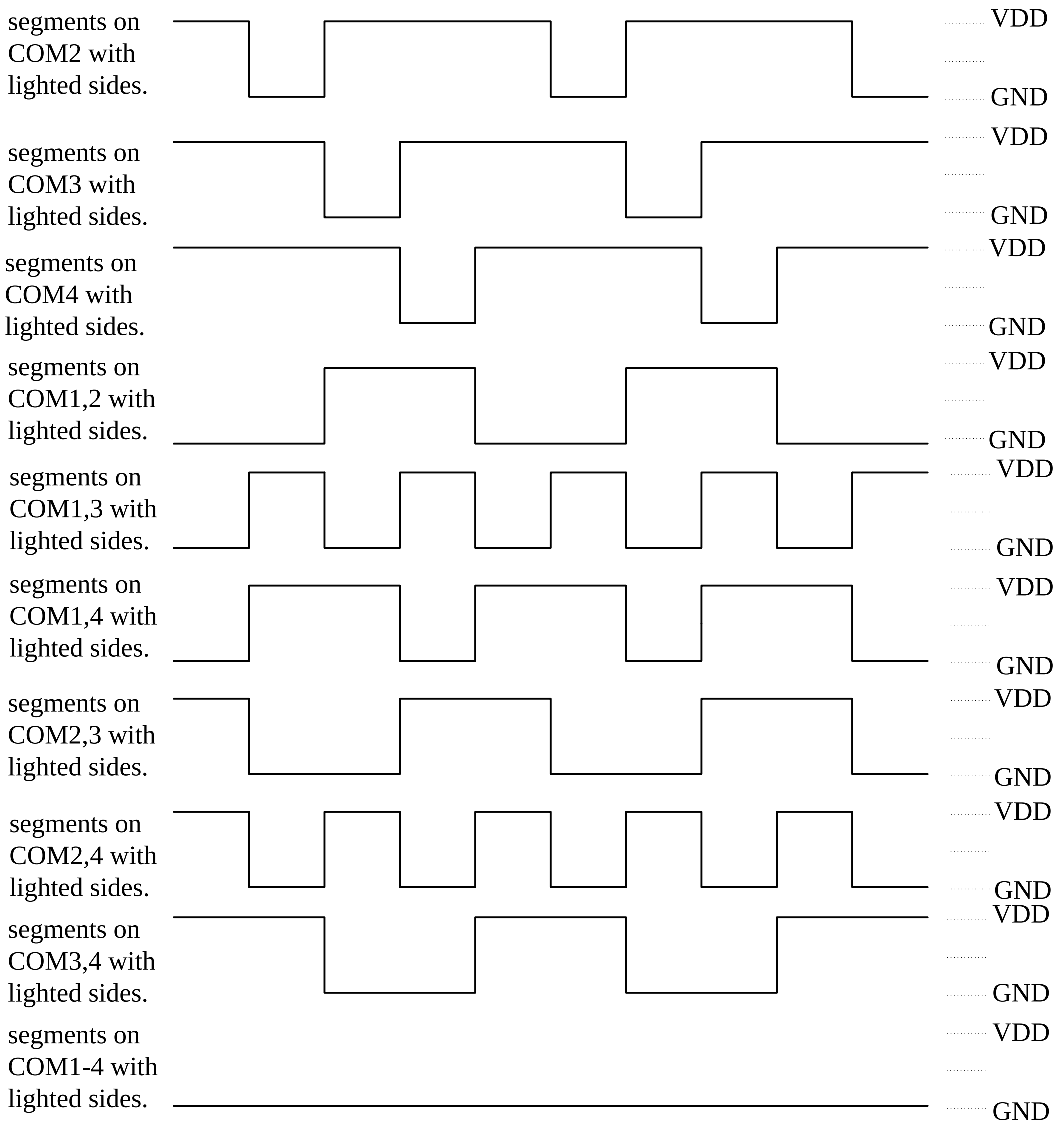
4.2.4 1/4 DUTY LIGHTING SYSTEM FOR LED DRIVER

(i) Initial clear mode (lighting)

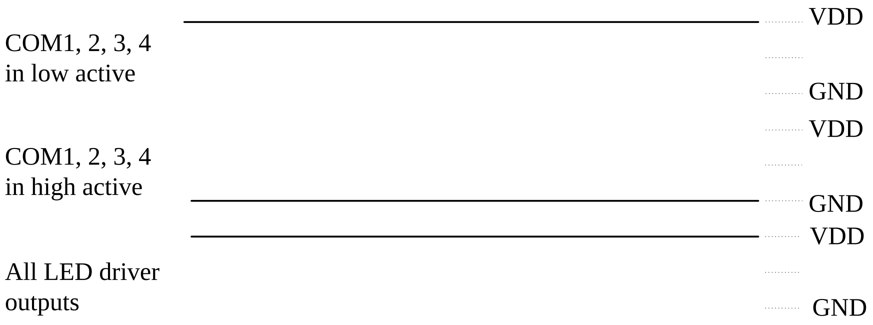


(ii) Normal operation mode




segments on COM2 with lighted sides.
VDD
GND
segments on COM3 with lighted sides.
VDD
GND
segments on COM4 with lighted sides.
VDD
GND
segments on COM1,2 with lighted sides.
VDD
GND
segments on COM1,3 with lighted sides.
VDD
GND
segments on COM1,4 with lighted sides.
VDD
GND
segments on COM2,3 with lighted sides.
VDD
GND
segments on COM2,4 with lighted sides.
VDD
GND
segments on COM3,4 with lighted sides.
VDD
GND
segments on COM1-4 with lighted sides.
VDD
GND

(iii) Display Turn Off



(iv) Stop mode

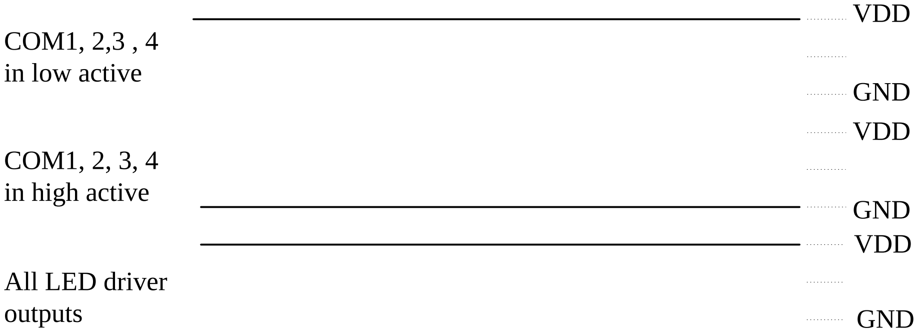
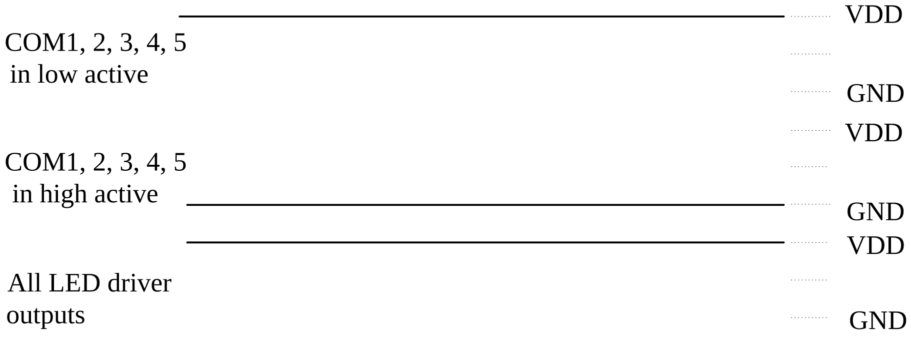


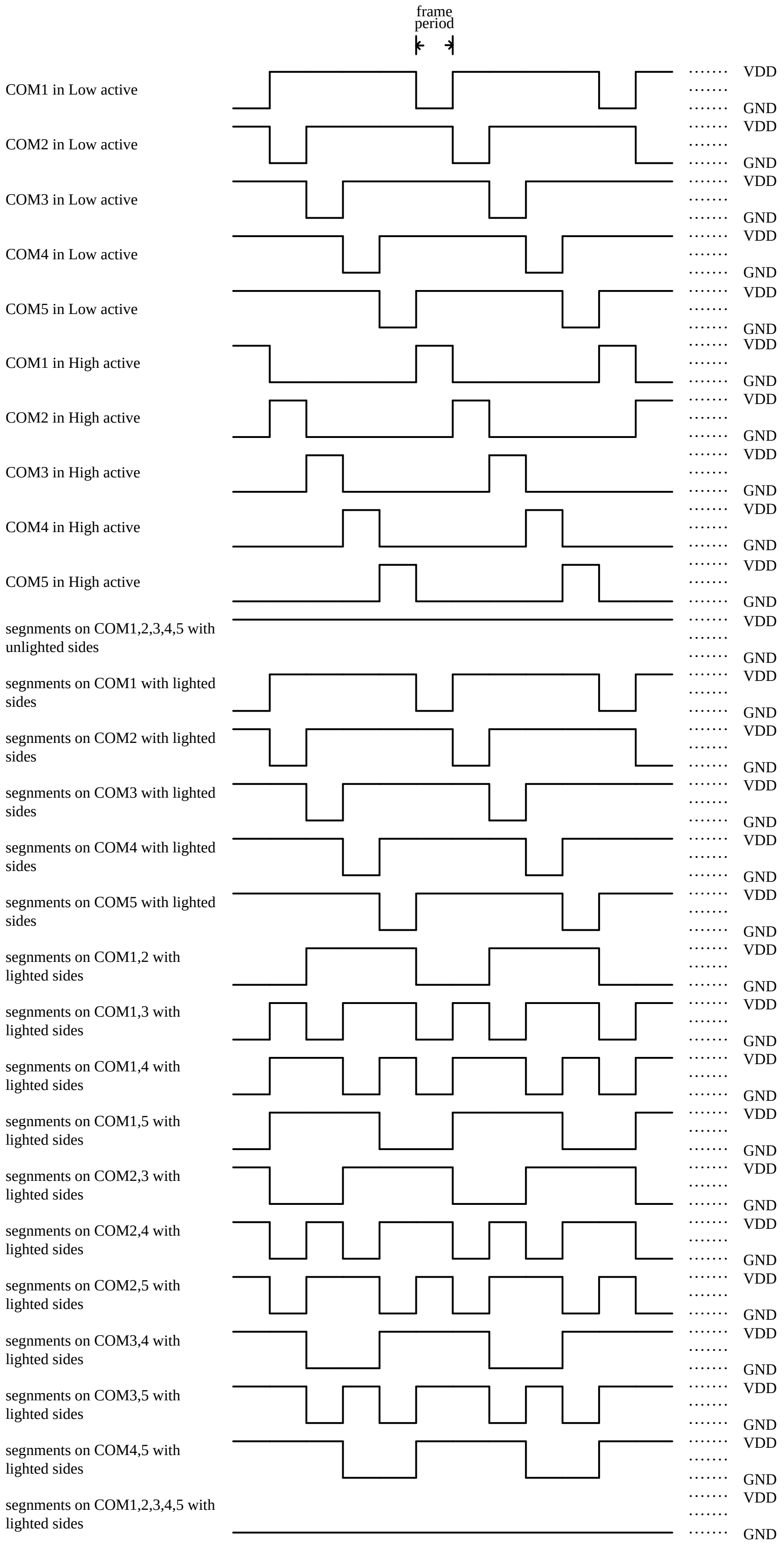
Figure 2- 42 1/4 duty LED Waveform

4.2.5 1/5 DUTY LIGHTING SYSTEM FOR LED DRIVER

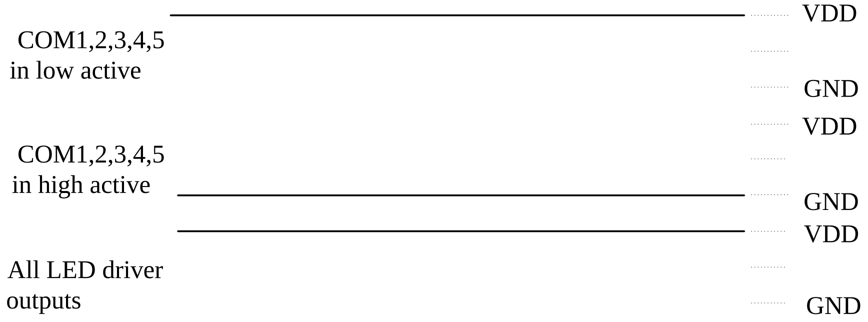
(i) Initial clear mode (lighting)



(ii) Normal operation mode


frame period
COM1 in Low active
COM2 in Low active
COM3 in Low active
COM4 in Low active
COM5 in Low active
COM1 in High active
COM2 in High active
COM3 in High active
COM4 in High active
COM5 in High active
segnments on COM1,2,3,4,5 with unlighted sides
segnments on COM1 with lighted sides
segnments on COM2 with lighted sides
segnments on COM3 with lighted sides
segnments on COM4 with lighted sides
segnments on COM5 with lighted sides
segnments on COM1,2 with lighted sides
segnments on COM1,3 with lighted sides
segnments on COM1,4 with lighted sides
segnments on COM1,5 with lighted sides
segnments on COM2,3 with lighted sides
segnments on COM2,4 with lighted sides
segnments on COM2,5 with lighted sides
segnments on COM3,4 with lighted sides
segnments on COM3,5 with lighted sides
segnments on COM4,5 with lighted sides
segnments on COM1,2,3,4,5 with lighted sides
VDD
GND

(iii) Display Turn Off



(iv) Stop mode

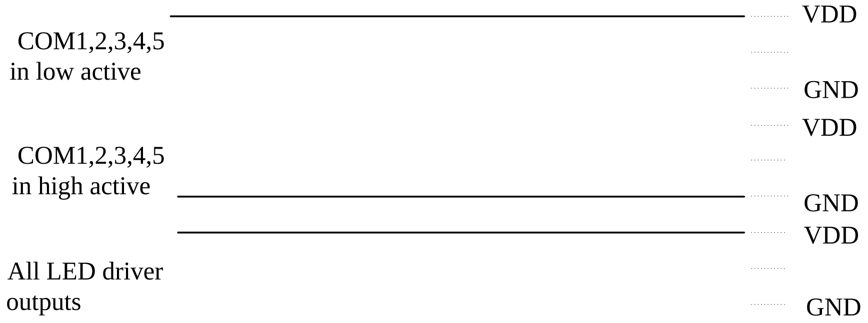


Figure 2- 43 1/5 duty LED Waveform

5. Detail Explanation of TM8763 Instructions

- Before using the data memory, it is necessary to initiate the content of data memory because the initial value is unknown.

- The working registers are part of the data memory (RAM), and the relationship between them can be shown as follows:

[The absolute address of working register Rx=Ry+70H]*

Note: Ry: Address of working register, the range of addresses specified by Rx is from 00H to 7FH.
Rx: Address of data memory, the range of addresses specified by Ry is from 0H to FH.

Address of working registers specified by Ry	Absolute address of data memory (Rx)
0H	70H
1H	71H
2H	72H
. . . .	. . . .
DH	7DH
EH	7EH
FH	7FH

- @HL is an 8-bit index address buffer. This buffer may address all data memory and table ROM. The contents of the index address buffer may be specified by two instructions: MVH and MVL. MVH transfers the contents of data memory Rx to the higher nibble (4-bits) and MVL to the lower nibble (4-bits).

  The organization of the index address buffer (@HL) is shown below.

Index Address Buffer	
Higher nibble @H	Lower nibble @L
H7~0	L3~L0
Transferred by MVH	Transferred by MVL

- Lz represents the address of the latch of LCD PLA; the address range specified by Lz is from 00H to 1FH.

5.1 INPUT / OUTPUT INSTRUCTIONS

LCT Lz, Ry

Function: LCD latch Lz ← data decoder ← (Ry)

Description: The working register contents specified by Ry are loaded to the LCD latch specified by Lz through the data decoder.

LCB Lz, Ry

Function: LCD latch Lz ← data decoder ← (Ry)

Description: The working register contents specified by Ry are loaded to the LCD latch specified by Lz through the data decoder.
If the content of Ry is "0", the outputs of the data decoder are all "0".

LCP Lz, Ry

Function: LCD latch Lz ← (Ry), (AC)

Description: The working register contents specified by Ry and the contents of AC are loaded to the LCD latch specified by Lz.

LCD Lz, @HL

Function: LCD latch Lz ← (T@HL)

Description: @HL indicates an index address of table ROM.
The contents of table ROM specified by @HL is loaded to the LCD latch specified by Lz directly.

LCT Lz, @HL

Function: LCD latch Lz ← data decoder ← (R@HL)

Description: The contents of index RAM specified by @HL is loaded to the LCD latch specified by Lz through the data decoder.

LCB Lz, @HL

Function: LCD latch Lz ← data decoder ← (R@HL)

Description: The contents of index RAM specified by @HL is loaded to the LCD latch specified by Lz through the data decoder.
If the content of @HL is "0", the outputs of the data decoder are all "0".

LCP Lz, @HL

Function: LCD latch Lz ← (R@HL), (AC)

Description: The contents of index RAM specified by @HL and the contents of AC are loaded to the LCD latch specified by Lz.

SPA X

Function: Defines the input/output mode of each pin for IOA port and enables / disables the pull-low device.

Description: Sets the I/O mode and turns on/off the pull-low device. The input pull-low device will be enabled when the I/O pin was set as input mode. The meaning of each bit of X (X3 X2 X1 X0) is shown below:

Bit pattern	Setting	Bit pattern	Setting
X4=1	Enable IOA pull low R	X4=0	Disable IOA pull low R
X3=1	IOA4 as output mode	X3=0	IOA4 as input mode
X2=1	IOA3 as output mode	X2=0	IOA3 as input mode
X1=1	IOA2 as output mode	X1=0	IOA2 as input mode
X0=1	IOA1 as output mode	X0=0	IOA1 as input mode

OPA Rx

Function: I/OA ← (Rx)

Description: The content of Rx is outputted to I/OA port.

OPAS Rx, D

Function: IOA1,2 ← (Rx), IOA3 ← D, IOA4 ← pulse

Description: Content of Rx is outputted to IOA port. D is outputted to IOA3, pulse is outputted to IOA4.
D=0 or 1

IPA Rx

Function: Rx, AC ← (IOA)

Description: The data of I/OA port is loaded to AC and data memory Rx.

SPB X

Function: Defines the input/output mode of each pin for IOB port and enables/disables the pull-low device.

Description: Sets the I/O mode and turns on/off the pull-low device. The input pull-low device will be enabled when the I/O pin was set as input mode. The meaning of each bit of X (X3 X2 X1 X0) is shown below:

Bit pattern	Setting	Bit pattern	Setting
X4=1	Enable IOB pull low R	X4=0	Disable IOB pull low R
X3=1	IOB4 as output mode	X3=0	IOB4 as input mode
X2=1	IOB3 as output mode	X2=0	IOB3 as input mode
X1=1	IOB2 as output mode	X1=0	IOB2 as input mode
X0=1	IOB1 as output mode	X0=0	IOB1 as input mode

OPB Rx

Function: I/OB ← (Rx)

Description: The contents of Rx are outputted to I/OB port.

IPB Rx

Function: Rx, AC ← (IOB)

Description: The data of I/OB port is loaded to AC and data memory Rx.

SPC X

Function: Defines the input/output mode of each pin for IOC port and enables/disables the pull-low device or low-level-hold device.

Description: Sets the I/O mode and turns on/off the pull-low device. The input pull-low device will be enabled when the I/O pin was set as input mode.

The meaning of each bit of X (X4 X3 X2 X1 X0) is shown below:

Bit pattern	Setting	Bit pattern	Setting
X4=1	Enables all of the pull-low and disables the low-level hold devices	X4=0	Disables all of the pull-low and enables the low-level hold devices
X3=1	IOC4 as output mode	X3=0	IOC4 as input mode
X2=1	IOC3 as output mode	X2=0	IOC3 as input mode
X1=1	IOC2 as output mode	X1=0	IOC2 as input mode
X0=1	IOC1 as output mode	X0=0	IOC1 as input mode

OPC Rx

Function: I/OC ← (Rx)

Description: The content of Rx is outputted to I/OC port.

IPC Rx

Function: Rx, AC ← (IOC)

Description: The data of I/OC port is loaded to AC and data memory Rx.

SPK X

Function: Sets Key Matrix scanning output state.

Description: When any of SEG1~12 is used for LCD/LED by mask option, the output state in scanning interval is set by X(X7~0)

Bit Patten	Setting : Halt Release by
X6=0	Normal Key Scanning
X6=1	Scanning Cycle

The bit pattern of X (for Key Matrix scanning output to SEG1~12)

X7	X5	X4	X3	X2	X1	X0	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	K11	K12
0	0	0	0	0	0	0	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
0	0	0	0	0	0	1	Hi-z	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
0	0	0	0	0	1	0	Hi-z	Hi-z	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
0	0	0	0	0	1	1	Hi-z	Hi-z	Hi-z	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
0	0	0	0	1	0	0	Hi-z	Hi-z	Hi-z	Hi-z	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
0	0	0	0	1	0	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
0	0	0	0	1	1	0	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
0	0	0	0	1	1	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	Hi-z	Hi-z	Hi-z	Hi-z
0	0	0	1	0	0	0	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	Hi-z	Hi-z	Hi-z
0	0	0	1	0	0	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	Hi-z	Hi-z
0	0	0	1	0	1	0	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	Hi-z
0	0	0	1	0	1	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1
0	0	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	1	0	0	0	0	0	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
1	0	X	0	X	X	X	1	1	1	1	1	1	1	1	Hi-z	Hi-z	Hi-z	Hi-z
1	0	X	1	X	X	X	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	1	1	1
1	1	0	0	0	X	X	1	1	1	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
1	1	0	0	1	X	X	Hi-z	Hi-z	Hi-z	Hi-z	1	1	1	1	Hi-z	Hi-z	Hi-z	Hi-z
1	1	0	1	0	X	X	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	1	1	1
1	1	0	1	1	X	X	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
1	1	1	0	0	0	X	1	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
1	1	1	0	0	1	X	Hi-z	Hi-z	1	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
1	1	1	0	1	0	X	Hi-z	Hi-z	Hi-z	Hi-z	1	1	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z
1	1	1	0	1	1	X	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	1	Hi-z	Hi-z	Hi-z	Hi-z
1	1	1	1	0	0	X	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	1	Hi-z	Hi-z
1	1	1	1	0	1	X	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	1	1
1	1	1	1	1	X	X	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z	Hi-z

Notes: 1. 1=H/L (LED/LCD)

2. K1~12 = SEG1~12 output in scanning interval

ALM X

Function: Sets buzzer output frequency.

Description: The waveform specified by X (X8 ~ X0) is delivered to the BZ and BZB pins.
The output frequency could be any combination in the following table.
The bit pattern of X (for higher frequency clock source):

X8	X7	X6	clock source (higher frequency)
1	1	1	FREQ*
1	0	0	DC1
0	1	1	ϕ3 (4 KHz)
0	1	0	ϕ4 (2 KHz)
0	0	1	ϕ5 (1 KHz)
0	0	0	DC0

The bit pattern of X (for lower frequency clock source)*:

Bit	clock source(lower frequency)
X5	ϕ15 (1 Hz)
X4	ϕ14 (2 Hz)
X3	ϕ13 (4 Hz)
X2	ϕ12 (8 Hz)
X1	ϕ11 (16 Hz)
X0	ϕ10 (32 Hz)

Notes: 1. FREQ is the output of frequency generator.
2. When the buzzer output does not need the envelope waveform, X5 ~ X0 should be set to 0.
3.The frequency inside the () bases on the ϕ0 is 32768Hz.

ELC X

Function: The bit control of EL-light driver.

Description: The meaning of each bit specified by X (X8~X0) is shown below:
For ELP pin setting:

(X8, X7, X6)	Pumping clock frequency	(X5, X4)	Duty cycle
000	ϕ0	00	3/4 duty
100	BCLK	01	2/3 duty
101	BCLK/2	10	1/2 duty
110	BCLK/4	11	1/1 duty (original)
111	BCLK/8		

For ELC pin setting:

(X3, X2)	Discharge pulse frequency	(X1, X0)	Duty cycle
00	ϕ8	00	1/4 duty
01	ϕ7	01	1/3 duty
10	ϕ6	10	1/2 duty
11	ϕ5	11	1/1 duty(original)

5.2 ACCUMULATOR MANIPULATION INSTRUCTIONS AND MEMORY MANIPULATION INSTRUCTIONS

MRW Ry, Rx

Function: AC, Rx ← (Rx)

Description: The content of Rx is loaded to AC and the working register specified by Ry.

MRW @HL, Rx

Function: AC, R@HL ← (Rx)

Description: The content of data memory specified by Rx is loaded to AC and data memory specified by @HL.

MWR Rx, Ry

Function: AC, Rx ← (Ry)

Description: The content of working register specified by Ry is loaded to AC and data memory specified by Rx.

MWR Rx, @HL

Function: AC, Rx ← (R@HL)

Description: The content of data memory specified by @HL is loaded to AC and data memory specified by Rx.

SR0 Rx

Function: Rxn, ACn ← Rx (n+1), AC (n+1)
Rx3, AC3 ← 0

Description: The Rx content is shifted right and 0 is loaded to the MSB.
The result is loaded to the AC.
0 → Rx3 → Rx2 → Rx1 → Rx0 →

SR1 Rx

Function: Rxn, ACn ← Rx (n+1), AC (n+1)
Rx3, AC3 ← 1

Description: The Rx content is shifted right and 1 is loaded to the MSB. The result is loaded to the AC.
1 → Rx3 → Rx2 → Rx1 → Rx0 →

SL0 Rx

Function: Rxn, ACn ← Rx (n-1), AC (n-1)
Rx0, AC0 ← 0

Description: The Rx content is shifted left and 0 is loaded to the LSB. The results are loaded to the AC.
← Rx3 ← Rx2 ← Rx1 ← Rx0 ← 0

SL1 Rx

Function: Rxn, ACn ← Rx (n-1), AC (n-1)
Rx0, AC0 ← 1

Description: The Rx content is shifted left and 1 is loaded to the LSB. The results are loaded to the AC.
← Rx3 ← Rx2 ← Rx1 ← Rx0 ← 1

MRA Rx

Function: CF ← (Rx) 3

Description: Bit3 of the content of Rx is loaded to carry flag (CF).

MAF Rx

Function: AC, Rx ← CF

Description: The content of CF is loaded to AC and Rx. The content of AC and meaning of bit after execution of this instruction are as follows:

Bit 3 .... CF

Bit 2 .... (AC) =0, zero flag

Bit 1 .... (No Use)

Bit 0 .... (No Use)

5.3 OPERATION INSTRUCTIONS

INC* Rx

Function: Rx, AC ← (Rx) +1

Description: Add 1 to the content of Rx; the result is loaded to data memory Rx and AC.

* Carry flag (CF) will be affected.

INC* @HL

Function: R@HL, AC ← (R@HL) +1

Description: Add 1 to the content of data memory specified by @HL; the result is loaded to data memory specified by @HL and AC.

* Carry flag (CF) will be affected.

DEC* Rx

Function: Rx, AC ← (Rx) -1

Description: Substrate 1 from the content of Rx; the result is loaded to data memory Rx and AC.

• Carry flag (CF) will be affected.

DEC* @HL

Function: R@HL, AC ← (R@HL) -1

Description: Substrate 1 from the content of data memory specified by @HL; the result is loaded to data memory specified by @HL and AC.

* Carry flag (CF) will be affected.

ADC Rx

Function: AC ← (Rx)+(AC) +CF

Description: The contents of Rx, AC and CF are binary-added; the result is loaded to AC.

* Carry flag (CF) will be affected.

ADC @HL

Function: AC ← (R@HL) + (AC) +CF

Description: The contents of data memory specified by @HL, AC and CF are binary-added; the result is loaded to AC.

* Carry flag (CF) will be affected.

ADC* Rx
Function: AC, Rx ← (Rx) + (AC) +CF
Description: The contents of Rx, AC and CF are binary-added; the result is loaded to AC and data memory Rx.
* Carry flag (CF) will be affected.

ADC* @HL
Function: AC, R@HL ← (R@HL) + (AC) +CF
Description: MThe contents of data memory specified by @HL, AC and CF are binary-added; the result is loaded to AC and data memory specified by @HL.
* Carry flag (CF) will be affected.

SBC Rx
Function: AC ← (Rx) + (AC) B+CF
Description: The contents of AC and CF are binary-subtracted from content of Rx; the result is loaded to AC.
. Carry flag (CF) will be affected.

SBC @HL
Function: AC ← (R@HL) + (AC) B+CF
Description: The contents of AC and CF are binary-subtracted from content of data memory specified by @HL; the result is loaded to AC.
* Carry flag (CF) will be affected.

SBC* Rx
Function: AC, Rx ← (Rx) + (AC) B+CF
Description: The contents of AC and CF are binary-subtracted from content of Rx; the result is loaded to AC and data memory Rx.
. Carry flag (CF) will be affected.

SBC* @HL
Function: AC, R@HL ← (R@HL) + (AC) B+CF
Description: The contents of AC and CF are binary-subtracted from content of data memory specified by @HL; the result is loaded to AC and data memory specified by @HL.
* Carry flag (CF) will be affected.

ADD Rx
Function: AC ← (Rx)+(AC)
Description: The contents of Rx and AC are binary-added; the result is loaded to AC.
. Carry flag (CF) will be affected.

ADD @HL
Function: AC ← (R@HL) + (AC)
Description: The contents of data memory specified by @HL and AC are binary-added; the result is loaded to AC.
* Carry flag (CF) will be affected.

ADD* Rx
Function: AC, Rx ← (Rx) + (AC)
Description: The contents of Rx and AC are binary-added; the result is loaded to AC and data memory Rx.
. Carry flag (CF) will be affected.

ADD* @HL

Function: AC, R@HL ← (R@HL) + (AC)

Description: The contents of data memory specified by @HL and AC are binary-added; the result is loaded to AC and data memory specified by @HL.
* Carry flag (CF) will be affected.

SUB Rx

Function: AC ← (Rx) + (AC) B+1

Description: The content of AC is binary-subtracted from content of Rx; the result is loaded to AC.
. Carry flag (CF) will be affected.

SUB @HL

Function: AC ← (R@HL) + (AC) B+1

Description: The content of AC is binary-subtracted from content of data memory specified by @HL; the result is loaded to AC.
* Carry flag (CF) will be affected.

SUB* Rx

Function: AC,Rx ← (Rx) + (AC) B+1

Description: The content of AC is binary-subtracted from content of Rx; the result is loaded to AC and Rx.
* Carry flag (CF) will be affected.

SUB* @HL

Function: AC, R@HL ← (R@HL) + (AC) B+1

Description: The content of AC is binary-subtracted from content of data memory specified by @HL; the result is loaded to AC and data memory specified by @HL.
* Carry flag (CF) will be affected.

ADN Rx

Function: AC ← (Rx) + (AC)

Description: The contents of Rx and AC are binary-added; the result is loaded to AC.
* The result will not affect the carry flag (CF).

ADN @HL

Function: AC ← (R@HL) + (AC)

Description: The contents of data memory specified by @HL and AC are binary-added; the result is loaded to AC.
* The result will not affect the carry flag (CF).

ADN* Rx

Function: AC, Rx ← (Rx) + (AC)

Description: The contents of Rx and AC are binary-added; the result is loaded to AC and data memory Rx.
* The result will not affect the carry flag (CF).

ADN* @HL

Function: AC, R@HL ← (R@HL) + (AC)

Description: The contents of data memory specified by @HL and AC are binary-added; the result is loaded to AC and data memory specified by @HL.
* The result will not affect the carry flag (CF).

AND Rx

Function: AC ← (Rx) & (AC)

Description: The contents of Rx and AC are binary-ANDed; the result is loaded to AC.

1

AND @HL

Function: AC ← (R@HL) & (AC)

Description: The contents of data memory specified by @HL and AC are binary-ANDed; the result is loaded to AC.

AND* Rx

Function: AC, Rx ← (Rx) & (AC)

Description: The contents of Rx and AC are binary-ANDed; the result is loaded to AC and data memory Rx.

AND* @HL

Function: AC, R@HL ← (R@HL) & (AC)

Description: The contents of data memory specified by @HL and AC are binary-ANDed; the result is loaded to AC and data memory specified by @HL.

EOR Rx

Function: AC ← (Rx) ⊕ (AC)

Description: The contents of Rx and AC are exclusive-Ored; the result is loaded to AC.

EOR @HL

Function: AC ← (R@HL) ⊕ (AC)

Description: The contents of data memory specified by @HL and AC are exclusive-Ored; the result is loaded to AC.

EOR* Rx

Function: AC, Rx ← (Rx) ⊕ (AC)

Description: The contents of Rx and AC are exclusive-Ored; the result is loaded to AC and data memory Rx.

EOR* @HL

Function: AC, R@HL ← (R@HL) ⊕ (AC)

Description: The contents of data memory specified by @HL and AC are exclusive-Ored; the result is loaded to AC and data memory data memory specified by @HL.

OR Rx

Function: AC ← (Rx) | (AC)

Description: The contents of Rx and AC are binary-Ored; the result is loaded to AC.

OR @HL

Function: AC ← (R@HL) | (AC)

Description: The contents of @HL and AC are binary-Ored; the result is loaded to AC.
. @HL indicates an index address of data memory.

OR* Rx

Function: AC, Rx ← (Rx) | (AC)

Description: The contents of Rx and AC are binary-Ored; the result is loaded to AC data memory Rx.

OR*   @HL

Function: AC, R@HL ← (R@HL) | (AC)

Description: The contents of data memory specified by @HL and AC are binary-Ored; the result is loaded to AC and data memory specified by @HL.

ADCI   Ry, D

Function: AC ← (Ry)+D+CF

Description: D represents an immediate data.
The contents of Ry, D and CF are binary-ADDed; the result is loaded to AC.
*The carry flag (CF) will be affected.

D=0H ~ FH

ADCI* Ry, D

Function: AC, Rx ← (Ry) +D+CF

Description: D represents an immediate data.
The contents of Ry, D and CF are binary-ADDed; the result is loaded to AC and working register Ry.
* The carry flag (CF) will be affected.

D=0H ~ FH

SBCI   Ry, D

Function: AC ← (Rx) + (D) B+CF

Description: D represents an immediate data.
The CF and immediate data D are binary-subtracted from working register Ry; the result is loaded to AC.
* The carry flag (CF) will be affected.

D=0H ~ FH

SBCI* Ry, D

Function: AC, Rx ← (Ry) + (D) B+CF

Description: D represents an immediate data.
The CF and immediate data D are binary-subtracted from working register Ry; the result is loaded to AC and working register Ry.
* The carry flag (CF) will be affected.

D=0H ~ FH

ADDI   Ry, D

Function: AC ← (Ry) +D

Description: D represents an immediate data.
The contents of Ry and D are binary-ADDed; the result is loaded to AC.
* The carry flag (CF) will be affected.

D=0H ~ FH

ADDI* Ry, D

Function: AC, Rx ← (Ry) +D

Description: D represents an immediate data.
The contents of Ry and D are binary-ADDed; the result is loaded to AC and working register Ry.
* The carry flag (CF) will be affected.

D=0H ~ FH

SUBI Ry, D

Function: AC ← (Ry) + (D)B+1

Description: D represents an immediate data.
The immediate data D is binary-subtracted from working register Ry; the result is loaded to AC.
* The carry flag (CF) will be affected.

D=0H ~ FH

SUBI* Ry, D

Function: AC, Rx ← (Ry) + (D) B+1

Description: D represents an immediate data.
The immediate data D is binary-subtracted from working register Ry; the result is loaded to AC and working register Ry.
* The carry flag (CF) will be affected.

D=0H ~ FH

ADNI Ry, D

Function: AC ← (Ry) +D

Description: D represents an immediate data.
The contents of Ry and D are binary-ADDed; the result is loaded to AC.
* The result will not affect the carry flag (CF).

D=0H ~ FH

ADNI* Ry, D

Function: AC, Rx ← (Ry) +D

Description: D represents an immediate data.
The contents of Ry and D are binary-ADDed; the result is loaded to AC and working register Ry.
* The result will not affect the carry flag (CF).

D=0H ~ FH

ANDI Ry, D

Function: AC ← (Ry) & D

Description: D represents an immediate data.
The contents of Ry and D are binary-ANDed; the result is loaded to AC.

D=0H ~ FH

ANDI* Ry, D

Function: AC, Rx ← (Ry) & D

Description: D represents an immediate data.
The contents of Ry and D are binary-ANDed; the result is loaded to AC and working register Ry.

D=0H ~ FH

EORI Ry, D

Function: AC ← (Ry) ⊕ D

Description: D represents an immediate data.
The contents of Ry and D are exclusive-OREd; the result is loaded to AC.

D=0H ~ FH

EORI* Ry, D

Function: AC, Rx ← (Ry) ⊕ D

Description: D represents an immediate data.
The contents of Ry and D are exclusive-OREd; the result is loaded to AC and working register Ry.

D = 0H ~ FH

ORI Ry, D

Function: AC ← (Ry) | D

Description: D represents an immediate data.
The contents of Ry and D are binary-OREd; the result is loaded to AC.

D=0H ~ FH

ORI* Ry, D

Function: AC, Rx ← (Ry) | D

Description: D represents an immediate data.
The contents of Ry and D are binary-OREd; the result is loaded to AC and working register Ry.

D=0H ~ FH

5.4 LOAD/STORE INSTRUCTIONS

STA Rx

Function: Rx ← (AC)

Description: The content of AC is loaded to data memory specified by Rx.

STA @HL

Function: R@HL ← (AC)

Description: The content of AC is loaded to data memory specified by @HL.

LDS Rx, D

Function: AC, Rx ← D

Description: Immediate data D is loaded to the AC and data memory specified by Rx.
D = 0H ~ FH

LDA Rx

Function: AC ← (Rx)

Description: The content of Rx is loaded to AC.

LDA @HL

Function: AC ← (R@HL)

Description: The content of data memory specified by @HL is loaded to AC.

LDH Rx, @HL

Function: Rx, AC ← H (T@HL)

Description: The higher nibble data of Table ROM specified by @HL is loaded to data memory specified by Rx.

LDH* Rx, @HL

Function: Rx , AC ← H (T@HL), @HL ← (@HL) +1

Description: The higher nibble data of Table ROM specified by @HL is loaded to data memory specified by Rx and then is increased in @HL.

LDL Rx, @HL

Function: Rx , AC ← L (T@HL)

Description: The lower nibble data of Table ROM specified by @HL is loaded to the data memory specified by Rx.

LDL* Rx, @HL

Function: Rx, AC ← L (T@HL), @HL ← (@HL) +1

Description: The lower nibble data of Table ROM specified by @HL is loaded to the data memory specified by Rx and then incremented the content of @HL.

5.5 CPU CONTROL INSTRUCTIONS

NOP

Function: no operation

Description: no operation

HALT

Function: Enters halt mode

Description: The following 3 conditions cause the halt mode to be released.
1) An interrupt is accepted.
2) The signal change specified by the SCA instruction is applied to IOC.
3) The halt release condition specified by SHE instruction is met.
When an interrupt is accepted to release the halt mode, the halt mode returns by executing the RTS instruction after completion of interrupt service.

STOP

Function: Enters stop mode and stops all oscillators

Description: Before executing this instruction, all signals on IOC port must be set to low.
The following 3 conditions cause the stop mode to be released.
1) One of the signal on KI1~4 is “H”/”L”(LED/LCD) in scanning interval.
2) A signal change in the INT pin.
3) One of the signals on IOC port is "H".

SCA X

Function: The data specified by X causes the halt mode to be released.

Description: The signal change at ports IOA, IOC is specified. The bit meaning of X (X4) is shown below:

Bit pattern	Description
X4=1	Halt mode is released when signal applied to IOC

X7~5,X3~0 is reserved

SIE* X

Function: Set/Reset interrupt enable flag

Description:

X0=1	The IEF0 is set so that interrupt 0(Signal change at port IOC specified by SCA) is accepted.
X1=1	The IEF1 is set so that interrupt 1 (underflow from timer 1) is accepted.
X2=1	The IEF2 is set so that interrupt 2 (the signal change at the INT pin) is accepted.
X3=1	The IEF3 is set so that interrupt 3 (overflow from the predivider) is accepted.
X5=1	The IEF5 is set so that interrupt 5 (key scanning) is accepted.

X7, 6, 4 is reserved

SHE X

Function: Set/Reset halt release enables flag

Description:

X1=1	The HEF1 is set so that the halt mode is released by TMR1 underflow.
X2=1	The HEF2 is set so that the halt mode is released by signal changed on INT pin.
X3=1	The HEF3 is set so that the halt mode is released by predivider overflow.
X5=1	The HEF5 is set so that the halt mode is released by the signal is “H”/”L” (LED/LCD) on KI1~4 in scanning interval.

X7, 6,4 is reserved

SRE X

Function: Set/Reset stop release enable flag

Description:

X4=1	The SRF4 is set so that the stop mode is released by the signal changed on IOC port.
X5=1	The SRF5 is set so that the stop mode is released by the signal changed on INT pin.
X7=1	The SRF6 is set so that the stop mode is released by the signal is “H”/”L” (LED/LCD) on KI1~4 in scanning interval.

X6, X3~0 is reserved

FAST

Function: Switches the system clock to CFOSC clock.

Description: Starts up the CFOSC (high-speed osc.) and then switches the system clock to high-speed clock.

SLOW

Function: Switches the system clock to XTOSC clock (low speed osc).

Description: Switches the system clock to low speed clock, and then stops the CFOSC.

MSB Rx

Function: AC, Rx ← SCF1, BCF2, BCF

Description: The SCF1, SCF2 and BCF flag contents are loaded to AC and the data memory specified by Rx.

The content of AC and meaning of bit after execution of this instruction are as follows:

Bit 3	Bit 2	Bit 1	Bit 0
NA	Start condition flag 2 (SCF2)	Start condition flag 1 (SCF1)	Backup flag (BCF)
	Halt release caused by SCF4,5,7,8	Halt release caused by the IOC port	The backup mode status in TM8763

MSC Rx
Function: AC, Rx ← SCF4..7
Description: The SCF4 to SCF7 contents are loaded to AC and the data memory specified by Rx.
The content of AC and meaning of bit after execution of this instruction are as follows:

Bit 3	Bit 2	Bit 1	Bit 0
Start condition flag 7 (SCF7)	The content of 15th stage of the predivider	Start condition flag 5 (SCF5)	Start condition flag 4 (SCF4)
Halt release caused by predivider overflow		Halt release caused by TM1 underflow	Halt release caused by INT pin

MCX Rx
Function: AC, Rx ← SCF8
Description: The SCF8 content is loaded to AC and the data memory specified by Rx.

The content of AC and meaning of bit after execution of this instruction are as follows:

Bit 3	Bit 2	Bit 1	Bit 0
NA	NA	NA	Start condition flag 8 (SCF8)
NA	NA	NA	Halt release caused by the signal change to “H”/”L”(LED/LCD) on KI1~4 in scanning interval

MSD Rx
Function: Rx, AC ← WDF, CSF, RFOVF
Description: The watchdog flag, system clock status and overflow flag of RFC counter are loaded to data memory specified by Rx and AC.
The content of AC and meaning of bit after execution of this instruction are as follows:

Bit 3	Bit 2	Bit 1	Bit 0
Reserved	Reserved	Watchdog timer enable flag (WDF)	System clock selection flag (CSF)

5.6 INDEX ADDRESS INSTRUCTIONS

MVH Rx
Function: (@H) ← (Rx),(AC)
Description: Loads content of Rx to higher nibble of index address buffer @H.
@H7=AC3, @H6=AC2, @H5=AC1, @H4=AC0,
@H3=Rx3, @H2=Rx2, @H1=Rx1, @H0=Rx0,

MVL Rx
Function: (@L) ← (Rx)
Description: Loads content of Rx to lower nibble of index address buffer @L.
@L3=Rx3, @L2=Rx2, @L1=Rx1, @L0=Rx0

5.7 DECIMAL ARITHMETIC INSTRUCTIONS

DAA

Function: AC ← BCD (AC)

Description: Converts the content of AC to binary format, and then restores to AC.
When this instruction is executed, the AC must be the result of any added instruction.
* The carry flag (CF) will be affected.

DAA* Rx

Function: AC, Rx ← BCD (AC)

Description: Converts the content of AC to binary format, and then restores to AC and data memory specified by Rx.
When this instruction is executed, the AC must be the result of any added instruction.
* The carry flag (CF) will be affected.

DAA* @HL

Function: AC, R@HL ← BCD (AC)

Description: Converts the content of AC to decimal format, and then restores to AC and data memory specified by @HL.
When this instruction is executed, the AC must be the result of any added instruction.
* The carry flag (CF) will be affected.

AC data before DAA execution	CF data before DAA execution	AC data after DAA execution	CF data after DAA execution
0 ≤ AC ≤9	CF=0	no change	no change
A ≤ AC ≤ F	CF=0	AC=AC+6	CF=1
0 ≤ AC ≤ 3	CF=1	AC=AC+6	no change

DAS

Function: AC ← BCD (AC)

Description: Converts the content of AC to decimal format, and then restores to AC.
When this instruction is executed, the AC must be the result of any subtracted instruction.
* The carry flag (CF) will be affected.

DAS* Rx

Function: AC, Rx ← BCD (AC)

Description: Converts the content of AC to decimal format, and then restores to AC and data memory specified by Rx.
When this instruction is executed, the AC must be the result of any subtracted instruction.
* The carry flag (CF) will be affected.

DAS* @HL

Function: AC, @HL ← BCD (AC)

Description: Converts the content of AC to decimal format, and then restores to AC and data memory @HL.
When this instruction is executed, the AC must be the result of any subtracted instruction.
* The carry flag (CF) will be affected.

AC data before DAS execution	CF data before DAS execution	AC data after DAS execution	CF data after DAS execution
0 ≤ AC ≤ 9	CF=1	No change	no change
6 ≤ AC ≤ F	CF=0	AC=AC+A	no change

5.8 JUMP INSTRUCTIONS

JB0 X

Function: Program counter jumps to X if AC0=1.

Description: If bit0 of AC is 1, jump occurs.
If 0, the PC increases by 1.
The range of X is from 000H to 5FFH.

JB1 X

Function: Program counter jumps to X if AC1=1.

Description: If bit1 of AC is 1, jump occurs.
If 0, the PC increases by 1.
The range of X is from 000H to 5FFH.

JB2 X

Function: Program counter jumps to X if AC2=1.

Description: If bit2 of AC is 1, jump occurs.
If 0, the PC increases by 1.
The range of X is from 000H to 5FFH.

JB3 X

Function: Program counter jumps to X if AC3=1.

Description: If bit3 of AC is 1, jump occurs.
If 0, the PC increases by 1.
The range of X is from 000H to 5FFH.

JNZ X

Function: Program counter jumps to X if (AC) ! =0.

Description: If the content of AC is not 0, jump occurs.
If 0, the PC increases by 1.
The range of X is from 000H to 5FFH.

JNC X

Function: Program counter jumps to X if CF=0.

Description: If the content of CF is 0, jump occurs.
If 1, the PC increases by 1.
The range of X is from 000H to 5FFH.

JZ X

Function: Program counter jumps to X if (AC) =0.

Description: If the content of AC is 0, jump occurs.
If 1, the PC increases by 1.
The range of X is from 000H to 5FFH.

JC X

Function: Program counter jumps to X if CF=1.

Description: If the content of CF is 1, jump occurs.
If 0, the PC increases by 1.
The range of X is from 000H to 5FFH.

JMP X

Function: Program counter jumps to X.

Description: Unconditional jump.
The range of X is from 000H to 5FFH.

CALL X

Function: STACK ← (PC) +1
Program counter jumps to X.

Description: A subroutine is called.
The range of X is from 000H to 5FFH.

RTS

Function: PC ← (STACK)

Description: A return from a subroutine occurs.

5.9 MISCELLANEOUS INSTRUCTIONS

SCC X

Function: Setting the clock source for IOA, IOC chattering prevention, PWM output and frequency generator.

Description: The following table shows the meaning of each bit for this instruction:

Bit pattern	Clock source setting	Bit pattern	Clock source setting
X6=1	The clock source comes from the system clock (BCLK).	X6=0	The clock source comes from the ϕ0. Refer to section 3-3-4 for ϕ0.
(X2, X1, X0) =001	Chattering prevention clock=ϕ10	(X2, X1, X0) =010	Chattering prevention clock=ϕ8
(X2, X1, X0) =100	Chattering clock = ϕ6		

X7, 5,4,3 is reserved

FRQ D, Rx

Function: Frequency generator ← D, (Rx), (AC)

Description: Loads the content of AC and data memory specified by Rx and D to frequency generator to set the duty cycle and initial value. The following table shows the preset data and the duty cycle setting:

	The bit pattern of preset letter N							
Programming divider	Bit7	Bit6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FRQ D, Rx	AC3	AC2	AC1	AC0	Rx3	Rx2	Rx1	Rx0

Preset Letter D		Duty Cycle
D1	D0	
0	0	1/4 duty
0	1	1/3 duty
1	0	1/2 duty
1	1	1/1 duty

FRQ D, @HL

Function: Frequency generator ← D, (T@HL)

Description: Loads the content of Table ROM specified by @HL and D to frequency generator to set the duty cycle and initial value. The following table shows the preset data and the duty cycle setting:

	The bit pattern of preset letter N							
Programming divider	Bit7	Bit6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FRQ D,@HL	T7	T6	T5	T4	T3	T2	T1	T0

Note: T0 ~ T7 represents the data of table ROM.

Preset Letter D		Duty Cycle
D1	D0	
0	0	1/4 duty
0	1	1/3 duty
1	0	1/2 duty
1	1	1/1 duty

FRQX D, X

Function: Frequency generator ← D, X

Description: Loads the data X (X7 ~ X0) and D to frequency generator to set the duty cycle and initial value. The following table shows the preset data and the duty cycle setting:

	The bit pattern of preset letter N							
Programming divider	Bit7	Bit6	Bit 5	Bit 4	Bit 3	Bit 2	bit 1	bit 0
FRQX D,X	X7	X6	X5	X4	X3	X2	X1	X0

Note: X0 ~ X7 represents the data specified in operand X.

Preset Letter D		Duty Cycle
D1	D0	
0	0	1/4 duty
0	1	1/3 duty
1	0	1/2 duty
1	1	1/1 duty

1. FRQ D, Rx
   The content of Rx and AC as preset data N.
2. FRQ D, @HL
   The content of tables TOM specified by index address buffer as preset data N.
3. FRQX D, X
   The data of operand in the instruction assigned as preset data N.

TMS Rx

Function: Select timer 1 clock source and preset timer 1.

Description: The content of data memory specified by Rx and AC are loaded to timer 1 to start the timer.
The following table shows the bit pattern for this instruction:

	Select clock		Setting value					
TMS Rx	AC3	AC2	AC1	AC0	Rx3	Rx2	Rx1	Rx0

The clock source option for timer 1

AC3	AC2	Clock source
0	0	ϕ9
0	1	ϕ3
1	0	ϕ15
1	1	FREQ

TMS @HL

Function: Select timer 1 clock source and preset timer 1.

Description: The content of table ROM specified by @HL is loaded to timer 1 to start the timer.
The following table shows the bit pattern for this instruction:

	Select clock		Setting value					
TMS @HL	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0

The clock source option for timer 1

Bit7	Bit6	Clock source
0	0	ϕ9
0	1	ϕ3
1	0	ϕ15
1	1	FREQ

TMSX X

Function: Selects timer 1 clock source and preset timer 1.

Description: The data specified by X (X7 ~ X0) is loaded to timer 1 to start the timer.
The following table shows the bit pattern for this instruction:

	Select clock		Setting value					
TMSX X	X7	X6	X5	X4	X3	X2	X1	X0

The clock source option for timer 1

X7	X6	Clock source
0	0	ϕ9
0	1	ϕ3
1	0	ϕ15
1	1	FREQ

SF X

Function: Sets flag

Description: Description of each flag

X0: "1" The CF is set to 1.

X1: "1" The chip enters backup mode and BCF is set to 1.

X2: "1" The EL-light driver output pin is active.

X3: "1" For X2=1 , when the SF instruction is executed at X3=1 , the EL-light driver is active and the halt request signal is outputted, then the chip enters halt mode.

X4: "1" The watchdog timer is initiated and active.

X7: "1" Enables the re-load function of timer 1.

X6, 5 is reserved

RF X

Function: Resets flag

Description: Description of each flag

X0: "1" The CF is reset to 0.
X1: "1" The chip is out of backup mode and BCF is reset to 0.
X2: "1" The EL-light driver is inactive.
X4: "1" The watchdog timer is inactive.
X7: "1" Disables the re-load function of timer 1.
X6, 5,3 is reserved

SF2 X

Function: Sets flag

Description: Description of each flag

X3: “1” Enable INT powerful pull-low
X2: "1" Disables the LCD/LED segment output.
X7~4,1,0 is reserved

RF2 X

Function: Resets flag

Description: Description of each flag

X3: “1” Disable INT powerful pull-low
X2: "1" Enables the LCD/LED segment output.
X7~4, 1, 0 is reserved

PLC

Function: Pulse control

Description: The pulse corresponding to the data specified by X is generated.

X0: "1" Halt release request flag HRF0 caused by the signal at I/O port C is reset.
X1: "1" Halt release request flag HRF1 caused by underflow from the timer 1 is reset and stops the operating of timer 1(TM1).
X2: "1" Halt or stop release request flag HRF2 caused by the signal change at the INT pin is reset.
X3: "1" Halt release request flag HRF3 caused by overflow from the predivider be reset.
X5: "1" Halt release request flag HRF5 caused by the signal change to “H”/”L” (LED/LCD) on KI1~4 in scanning interval is reset.
X8: "1" The last 5 bits of the predivider (15 bits) are reset. When executing this instruction, X3 must be set to "1".
X4, X8 are reserved.

ORDERING INFORMATION

The ordering information:

Ordering number	Package
TM8763-COD	Wafer/Dice with code

Appendix A TM8763 Instruction Table

Instruction		Machine Code	Function		Flag/Remark
NOP		0000 0000 0000 0000	No Operation		
LCT	Lz, Ry	0000 001Z ZZZZ YYYY	(Lz)	←7SEG ← (Ry)	
LCB	Lz, Ry	0000 010Z ZZZZ YYYY	(Lz)	←7SEG ←(Ry)	Blank Zero
LCP	Lz, Ry	0000 011Z ZZZZ YYYY	(Lz)	← (Ry) , (AC)	
LCD	Lz, @HL	0000 100Z ZZZZ 0000	(Lz)	← (R@HL)	
LCT	Lz, @HL	0000 100Z ZZZZ 0001	(Lz)	←7SEG ←(R@HL)	
LCB	Lz, @HL	0000 100Z ZZZZ 0010	(Lz)	←7SEG ←(R@HL)	Blank Zero
LCP	Lz, @HL	0000 100Z ZZZZ 0011	(Lz)	←(R@HL) , (AC)	
OPA	Rx	0000 1010 0XXX XXXX	(IOA)	← (Rx)	
OPAS	Rx, D	0000 1011 DXXX XXXX	IOA1, 2, 3, 4	← (Rx)0, (Rx)1, D, Pulse	
OPB	Rx	0000 1100 0XXX XXXX	(IOB)	←(Rx)	
OPC	Rx	0000 1101 0XXX XXXX	(IOC)	←(Rx)	
FRQ	D, Rx	0001 00DD 0XXX XXXX	FREQ D=00 D=01 D=10 D=11	← (Rx) , (AC) : 1/4 Duty : 1/3 Duty : 1/2 Duty : 1/1 Duty	
FRQ	D, @HL	0001 01DD 0000 0000	FREQ	←(T@HL)	
FRQX	D, X	0001 10DD XXXX XXXX	FREQ	← X	
MVL	Rx	0001 1100 0XXX XXXX	(@L)	← (Rx)	
MVH	Rx	0001 1101 0XXX XXXX	(@H)	← (Rx) , (AC)	
ADC	Rx	0010 0000 0XXX XXXX	(AC)	← (Rx) + (AC) + CF	CF
ADC	@HL	0010 0000 1000 0000	(AC)	← (R@HL) + (AC) + CF	CF
ADC*	Rx	0010 0001 0XXX XXXX	(AC), (Rx)	← (Rx) + (AC) + CF	CF
ADC*	@HL	0010 0001 1000 0000	(AC), (R@HL)	← (R@HL) + (AC) + CF	CF
SBC	Rx	0010 0010 0XXX XXXX	(AC)	← (Rx) + (AC)B + CF	CF
SBC	@HL	0010 0010 1000 0000	(AC)	← (R@HL) + (AC)B + CF	CF
SBC*	Rx	0010 0011 0XXX XXXX	(AC), (Rx)	← (Rx) + (AC)B + CF	CF
SBC*	@HL	0010 0011 1000 0000	(AC), (R@HL)	← (R@HL) + (AC)B + CF	CF
ADD	Rx	0010 0100 0XXX XXXX	(AC)	← (Rx) + (AC)	CF
ADD	@HL	0010 0100 1000 0000	(AC)	← (R@HL) + (AC)	CF
ADD*	Rx	0010 0101 0XXX XXXX	(AC), (Rx)	← (Rx) + (AC)	CF
ADD*	@HL	0010 0101 1000 0000	(AC), (R@HL)	← (R@HL) + (AC)	CF
SUB	Rx	0010 0110 0XXX XXXX	(AC)	← (Rx) + (AC)B + 1	CF
SUB	@HL	0010 0110 1000 0000	(AC)	← (R@HL) + (AC)B + 1	CF
SUB*	Rx	0010 0111 0XXX XXXX	(AC), (Rx)	← (Rx) + (AC)B + 1	CF
SUB*	@HL	0010 0111 1000 0000	(AC), (R@HL)	← (R@HL) + (AC)B + 1	CF
ADN	Rx	0010 1000 0XXX XXXX	(AC)	← (Rx) + (AC)	
ADN	@HL	0010 1000 1000 0000	(AC)	← (R@HL) + (AC)	
ADN*	Rx	0010 1001 0XXX XXXX	(AC), (Rx)	← (Rx) + (AC)	
ADN*	@HL	0010 1001 1000 0000	(AC), (R@HL)	← (R@HL) + (AC)	
AND	Rx	0010 1010 0XXX XXXX	(AC)	← (Rx) AND (AC)	
AND	@HL	0010 1010 1000 0000	(AC)	← (R@HL) AND (AC)	
AND*	Rx	0010 1011 0XXX XXXX	(AC), (Rx)	← (Rx) AND (AC)	
AND*	@HL	0010 1011 1000 0000	(AC), (R@HL)	← (R@HL) AND (AC)	
EOR	Rx	0010 1100 0XXX XXXX	(AC)	← (Rx) EOR (AC)	

Instruction		Machine Code	Function		Flag/Remark
EOR	@HL	0010 1100 1000 0000	(AC)	← (R@HL) EOR (AC)	
EOR*	Rx	0010 1101 0XXX XXXX	(AC),(Rx)	← (Rx) EOR (AC)	
EOR*	@HL	0010 1101 1000 0000	(AC),(R@HL)	← (R@HL) EOR (AC)	
OR	Rx	0010 1110 0XXX XXXX	(AC)	← (Rx) OR (AC)	
OR	@HL	0010 1110 1000 0000	(AC)	← (R@HL) OR (AC)	
OR*	Rx	0010 1111 0XXX XXXX	(AC),(Rx)	← (Rx) OR (AC)	
OR*	@HL	0010 1111 1000 0000	(AC),(R@HL)	← (R@HL) OR (AC)	
ADCI	Ry, D	0011 0000 DDDD YYYY	(AC)	← (Ry) + D + CF	CF
ADCI*	Ry, D	0011 0001 DDDD YYYY	(AC),(Ry)	← (Ry) + D + CF	CF
SBCI	Ry, D	0011 0010 DDDD YYYY	(AC)	← (Ry) + DB + CF	CF
SBCI*	Ry, D	0011 0011 DDDD YYYY	(AC),(Ry)	← (Ry) + DB + CF	CF
ADDI	Ry, D	0011 0100 DDDD YYYY	(AC)	← (Ry) + D	CF
ADDI*	Ry, D	0011 0101 DDDD YYYY	(AC),(Ry)	← (Ry) + D	CF
SUBI	Ry, D	0011 0110 DDDD YYYY	(AC)	← (Ry) + DB + 1	CF
SUBI*	Ry, D	0011 0111 DDDD YYYY	(AC),(Ry)	← (Ry) + DB + 1	CF
ADNI	Ry, D	0011 1000 DDDD YYYY	(AC)	← (Ry) + D	
ADNI*	Ry, D	0011 1001 DDDD YYYY	(AC),(Ry)	← (Ry) + D	
ANDI	Ry, D	0011 1010 DDDD YYYY	(AC)	← (Ry) AND D	
ANDI*	Ry, D	0011 1011 DDDD YYYY	(AC),(Ry)	← (Ry) AND D	
EORI	Ry, D	0011 1100 DDDD YYYY	(AC)	← (Ry) EOR D	
EORI*	Ry, D	0011 1101 DDDD YYYY	(AC),(Ry)	← (Ry) EOR D	
ORI	Ry, D	0011 1110 DDDD YYYY	(AC)	← (Ry) OR D	
ORI*	Ry, D	0011 1111 DDDD YYYY	(AC),(Ry)	← (Ry) OR D	
INC*	Rx	0100 0000 0XXX XXXX	(AC),(Rx)	← (Rx) + 1	CF
INC*	@HL	0100 0000 1000 0000	(AC),(R@HL)	← (R@HL) + 1	CF
DEC*	Rx	0100 0001 0XXX XXXX	(AC),(Rx)	← (Rx) - 1	CF
DEC*	@HL	0100 0001 1000 0000	(AC),(R@HL)	← (R@HL) - 1	CF
IPA	Rx	0100 0010 0XXX XXXX	(AC),(Rx)	← (IOA)	
IPB	Rx	0100 0100 0XXX XXXX	(AC),(Rx)	← (IOB)	
IPC	Rx	0100 0111 0XXX XXXX	(AC),(Rx)	← (IOC)	
MAF	Rx	0100 1010 0XXX XXXX	(AC),(Rx)	← STS1	B3: CF B2: ZERO B1: (No use) B0: (No use)
MSB	Rx	0100 1011 0XXX XXXX	(AC),(Rx)	← STS2	B3: (No use) B2: SCF2 (HRx) B1: SCF1 (CPT) B0: BCF
MSC	Rx	0100 1100 0XXX XXXX	(AC),(Rx)	← STS3	B3: SCF7 (PDV) B2: PH15 B1: SCF5 (TM1) B0: SCF4 (INT)
MCX	Rx	0100 1101 0XXX XXXX	(AC),(Rx)	← STS3X	B3: (No use) B2: (No use) B1: (No use) B0: SCF8(SKI)
MSD	Rx	0100 1110 0XXX XXXX	(AC),(Rx)	← STS4	B3: (No use) B2: (No use) B1: WDF B0: CSF
SR0	Rx	0101 0000 0XXX XXXX	(AC) n, (Rx) n (AC) 3, (Rx)3	← (Rx) (n+1) ← 0	

Instruction		Machine Code	Function		Flag/Remark
SR1	Rx	0101 0001 0XXX XXXX	(AC)n, (Rx)n (AC)3, (Rx)3	← (Rx) (n+1) ← 1	
SL0	Rx	0101 0010 0XXX XXXX	(AC)n, (Rx)n (AC)0, (Rx)0	← (Rx) (n-1) ← 0	
SL1	Rx	0101 0011 0XXX XXXX	(AC)n, (Rx)n (AC)0, (Rx)0	← (Rx) (n-1) ← 1	
DAA		0101 0100 0000 0000	(AC)	← BCD (AC)	CF
DAA*	Rx	0101 0101 0XXX XXXX	(AC),(Rx)	← BCD (AC)	CF
DAA*	@HL	0101 0101 1000 0000	(AC),(R@HL)	← BCD (AC)	CF
DAS		0101 0110 0000 0000	(AC)	← BCD (AC)	CF
DAS*	Rx	0101 0111 0XXX XXXX	(AC),(Rx)	← BCD (AC)	CF
DAS*	@HL	0101 0111 1000 0000	(AC),(R@HL)	← BCD (AC)	CF
LDS	Rx, D	0101 1DDD DXXX XXXX	(AC),(Rx)	← D	
LDH	Rx, @HL	0110 0000 0XXX XXXX	(AC),(Rx)	← H (T@HL)	
LDH*	Rx, @HL	0110 0001 0XXX XXXX	(AC),(Rx) (@HL)	← H (T@HL) ← (@HL) + 1	
LDL	Rx, @HL	0110 0010 0XXX XXXX	(AC),(Rx)	← L (T@HL)	
LDL*	Rx, @HL	0110 0011 0XXX XXXX	(AC),(Rx) (@HL)	← L (T@HL) ← (@HL) + 1	
STA	Rx	0110 1000 0XXX XXXX	(Rx)	← (AC)	
STA	@HL	0110 1000 1000 0000	(R@HL)	← (AC)	
LDA	Rx	0110 1100 0XXX XXXX	(AC)	← (Rx)	
LDA	@HL	0110 1100 1000 0000	(AC)	← (R@HL)	
MRA	Rx	0110 1101 0XXX XXXX	CF	← (Rx) 3	
MRW	@HL, Rx	0110 1110 0XXX XXXX	(AC),(R@HL)	← (Rx)	
MWR	Rx, @HL	0110 1111 0XXX XXXX	(AC),(Rx)	← (R@HL)	
MRW	Ry, Rx	0111 0YYY YXXX XXXX	(AC),(Ry)	← (Rx)	
MWR	Rx, Ry	0111 1YYY YXXX XXXX	(AC),(Rx)	← (Ry)	
JB0	X	1000 0XXX XXXX XXXX	PC	← X	if (AC) 0=1
JB1	X	1000 1XXX XXXX XXXX	PC	← X	if (AC) 1=1
JB2	X	1001 0XXX XXXX XXXX	PC	← X	if (AC) 2=1
JB3	X	1001 1XXX XXXX XXXX	PC	← X	if (AC) 3=1
JNZ	X	1010 0XXX XXXX XXXX	PC	← X	if (AC) ≠0
JNC	X	1010 1XXX XXXX XXXX	PC	← X	if CF=0
JZ	X	1011 0XXX XXXX XXXX	PC	← X	if (AC) =0
JC	X	1011 1XXX XXXX XXXX	PC	← X	if CF=1
CALL	X	1100 0XXX XXXX XXXX	STACK (PC)	← (PC) + 1 ← X	
JMP	X	1101 0XXX XXXX XXXX	(PC)	← X	
RTS		1101 1000 0000 0000	(PC)	← STACK	CALL Return
SCC	X	1101 1001 0X00 0XXX	X6=1 X6=0 X2, 1, 0=001 X2, 1, 0=010 X2, 1, 0=100	: Cfq=BCLK : Cfq=PH0 : Cch=PH10 : Cch=PH8 : Cch=PH6	
SCA	X	1101 1010 000X 0000	X4	: Enable SEF4	C1-4
SPA	X	1101 1100 000X XXXX	X4 X3~0	: Set IOA4-1 Pull-Low : Set IOA4-1 I/O	
SPB	X	1101 1101 000X XXXX	X4 X3~0	: Set IOB4-1 Pull-Low : Set IOB4-1 I/O	
SPC	X	1101 1110 000X XXXX	X4 X3-0	: Set IOC4-1 Pull-Low /Low-Level-Hold : Set IOC4-1 I/O	

Instruction		Machine Code	Function		Flag/Remark
TMS	Rx	1110 0000 0XXX XXXX	Timer1	← (Rx) & (AC)	
TMS	@HL	1110 0001 0000 0000	Timer1	← (T@HL)	
TMSX	X	1110 0010 XXXX XXXX	X7, 6=11 X7, 6=10 X7, 6=01 X7, 6=00 X5~0	: Ctm=FREQ : Ctm=PH15 : Ctm=PH3 : Ctm=PH9 : Set Timer1 Value	
SPK	X	1110 0011 XXXX XXXX	X6=1	: KEY_S release by scanning cycle	IOC=normal IOC=KEY SCAN IOC=KEY SCAN
			X6=0	: KEY_S release by normal key scanning	
			X7, 5, 4=000	: Set one of KO1~12=1 by X3~0	
			X7, 5, 4=001	: Set all=1	
			X7, 5, 4=010	: Set all Hi-z	
			X7, 5, 4=10X	: Set eight of KO1~12 =1 by X3 X3=0=>KO1~8 X3=1=>KO9~12	
			X7, 5, 4=110	: Set four of KO1~12 =1 by X3, 2 X3, 2=00= KO1~4 X3, 2=01=>KO5~8 X3, 2=10=>KO9~12	
			X7, 5, 4=111	: Set two of KO1~12 =1 by X3, 2, 1 X3~1=000=>KO1, 2 X3~1=001=>KO3, 4 X3~1=010=>KO5, 6 X3~1=011=>KO7, 8 X3~1=100=>KO9, 10 X3~1=101=>KO11, 12	
SHE	X	1110 1000 00X0 XXX0	X5 X3 X2 X1	: Enable HEF5 : Enable HEF3 : Enable HEF2 : Enable HEF1	KEY_S PDV INT TMR1
SIE*	X	1110 1001 00X0 XXXX	X5 X3 X2 X1 X0	: Enable IEF5 : Enable IEF3 : Enable IEF2 : Enable IEF1 : Enable IEF0	KEY_S PDV INT TMR1 CPT
PLC	X	1110 101X 00X0 XXXX	X8 X5, 3-0	: Reset PH15~11 : Reset HRF5, 3-0	
SRE	X	1110 1101 X0XX 0000	X7 X5 X4	: Enable SRF7 : Enable SRF5 : Enable SRF4	SRF7 (KEY_S) SRF5 (INT) SRF4 (C Port)
FAST		1110 1110 0000 0000	SCLK	: High Speed Clock	
SLOW		1110 1111 0000 0000	SCLK	: Low Speed Clock	
SF	X	1111 0000 X00X XXXX	X7 X4 X3 X2 X1 X0	: Reload 1 Set : WDT Enable : HALT after EL : EL LIGHT On : BCF Set : CF Set	RL1 WDF BCF CF
RF	X	1111 0100 X00X 0XXX	X7 X4 X2 X1 X0	: Reload 1 Reset : WDT Reset : EL LIGHT Off : BCF Reset : CF Reset	RL1 WDF BCF CF
SF2	X	1111 1000 0000 XX00	X3 X2	: Enable INT powerful Pull-low : Close all Segments	INTPL RSOFF
RF2	X	1111 1001 0000 XX00	X3 X2	: Disable INT powerful Pull-low : Release Segments	INTPL RSOFF
ALM	X	1111 101X XXXX XXXX	X8, 7, 6=111 X8, 7, 6=100	: FREQ : DC1	

Instruction		Machine Code	Function		Flag/Remark
			X8, 7, 6=011	: PH3	
			X8, 7, 6=010	: PH4	
			X8, 7, 6=001	: PH5	
			X8, 7, 6=000	: DC0	
			X5~0	← PH15~10	
ELC	X	1111 110X XXXX XXXX	X8=1	BCLKX	ELP - CLK
			X8=0	PH0	
			X7, 6=11	BCLK/8	BCLKX
			X7, 6=10	BCLK/4	
			X7, 6=01	BCLK/2	
			X7, 6=00	BCLK	
			X5, 4=11	1/1	ELP - DUTY
			X5, 4=10	1/2	
			X5, 4=01	2/3	
			X5, 4=00	3/4	
			X3, 2=11	PH5	ELC - CLK
			X3, 2=10	PH6	
			X3, 2=01	PH7	
			X3, 2=00	PH8	
			X1, 0=11	1/1	ELC - DUTY
			X1, 0=10	1/2	
			X1, 0=01	1/3	
			X1, 0=00	1/4	
HALT		1111 1110 0000 0000	Halt Operation		
STOP		1111 1111 0000 0000	Stop Operation		