HT95CXXX
CID Type Phone 8-Bit MCU

Features

- Provide MASK type and OTP type version
- Operating voltage range:
  - FSK: 3.0V~5.5V
  - Others: 2.4V~5.5V
- Program ROM
  - HT95C400/40P: 16K×16 bits
  - HT95C300/30P: 8K×16 bits
  - HT95C200/20P: 8K×16 bits
- Data RAM
  - HT95C400/40P: 2880×8 bits
  - HT95C300/30P: 2112×8 bits
  - HT95C200/20P: 1152×8 bits
- Bidirectional I/O lines
  - HT95C400/40P: 40~28 I/O lines
  - HT95C300/30P: 28~16 I/O lines
  - HT95C200/20P: 28~20 I/O lines
- 16-bit table read instructions
- Subroutine nesting
  - HT95C400/40P: 12 levels
  - HT95C300/30P: 8 levels
  - HT95C200/20P: 8 levels
- Timer
  - Two 16-bit programmable Timer/Event Counter
  - Real time clock (RTC)
  - Watchdog Timer (WDT)
- Programmable frequency divider (PFD)
- Dual system clock: 32768Hz, 3.58MHz
- Four operating modes: Idle mode, Sleep mode, Green mode and Normal mode
- Up to 1.117μs instruction cycle with 3.58MHz system clock
- All instructions in one or two machine cycles
- Built-in 3.58MHz DTMF Generator
- Built-in FSK decoder:
  - Supports Bell 202 and V.23
  - Supports ring and line reversal detection
- Built-in dialer I/O
- Built-in low battery detector
- LCD driver
  - LCD contrast can be adjusted by software or external resistor
  - Support two LCD frame frequency 64Hz, 128Hz
  - Support 16 or 8 common driver pins
  - Some segments or commons can option to bidirectional I/O lines
  - HT95C400/40P: 48 seg.×16 com.
  - HT95C300/30P: 48 seg.×16 com.
  - HT95C200/20P: 24 seg.×16 com.
- 128-pin QFP package

Applications

- Deluxe Feature Phone
- Caller ID Phone
- Cordless Phone
- Fax and answering machines
- Other communication system

General Description

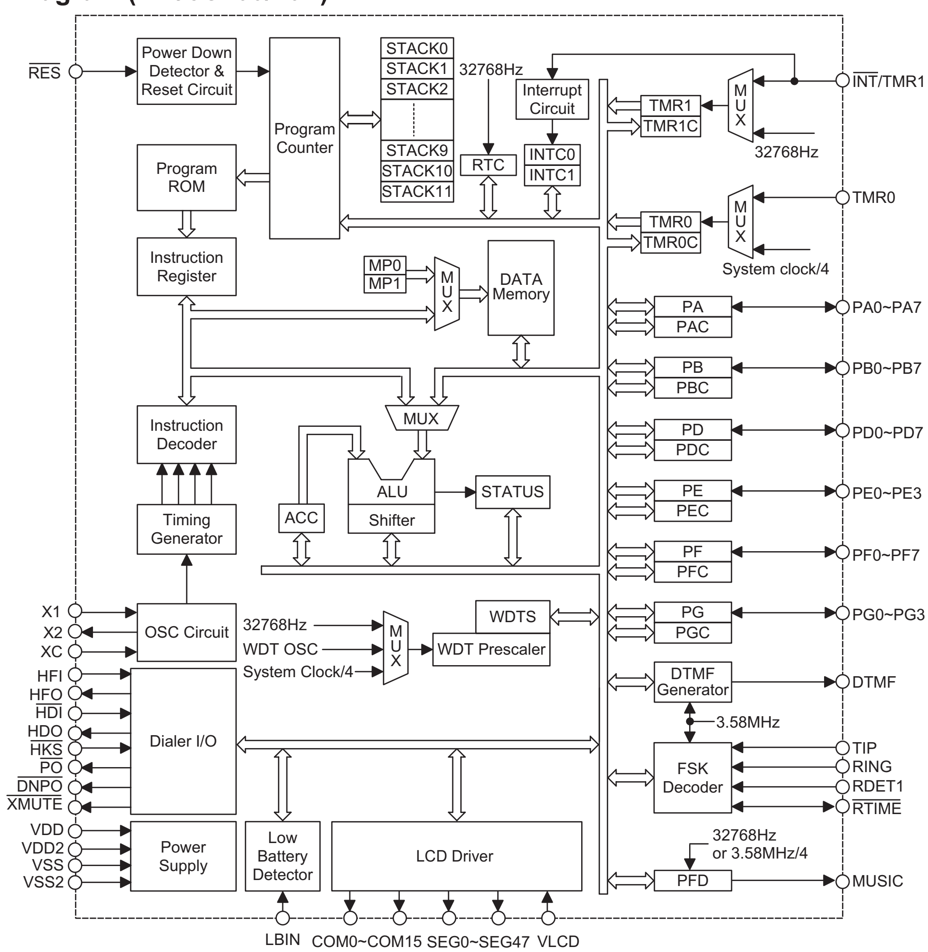
The HT95CXXX family MCU are 8-bit high performance RISC-like microcontrollers with built-in DTMF generator, FSK decoder and dialer I/O which provide MCU dialer implementation or system control features for telecom product applications. The phone controller has a built-in program ROM, data RAM, LCD driver and I/O lines for high end products design. In addition, for power management purpose, it has a built-in frequency up conversion circuit (32768Hz to 3.58MHz) which provides dual system clock and four types of operation modes. For example, it can operate with low speed system clock rate of 32768Hz in green mode with little power consumption. It can also operate with high speed system clock rate of 3.58MHz in normal mode for high performance operation. To ensure smooth dialer function and to avoid MCU shut-down in extreme low voltage situation, the dialer I/O circuit is built-in to generate hardware dialer signals such as on-hook, hold-line and hand-free. Built-in real time clock and programmable frequency divider are provided for additional fancy features in product developments. The device is best suited for feature phone products that comply with versatile dialer specification requirements of different areas or countries.

Selection Table

Part No.	Operating Voltage	Program Memory	Data Memory	Normal I/O	Dialer I/O	LCD	Timer	Stack	External Interrupt	DTMF Generator	FSK Receiver	Package
HT95A100 HT95A10P	2.4V~5.5V	4K×16	384×8	20	6	—	16-bit×2	4	3	√	—	28SOP
HT95A200 HT95A20P	2.4V~5.5V	4K×16	1152×8	28	8	—	16-bit×2	8	4	√	—	48SSOP
HT95A300 HT95A30P	2.4V~5.5V	8K×16	2112×8	28	8	—	16-bit×2	8	4	√	—	48SSOP
HT95A400 HT95A40P	2.4V~5.5V	16K×16	2880×8	44	8	—	16-bit×2	12	4	√	—	64QFP
HT95L000 HT95L00P	2.4V~5.5V	4K×16	384×8	14~18	6	12×8~16×8	16-bit×2	4	3	√	—	56SSOP
HT95L100 HT95L10P	2.4V~5.5V	4K×16	1152×8	16~20	8	16×8~20×8	16-bit×2	8	4	√	—	64QFP
HT95L200 HT95L20P	2.4V~5.5V	8K×16	1152×8	20~28	8	24×8~24×16	16-bit×2	8	4	√	—	100QFP
HT95L300 HT95L30P	2.4V~5.5V	8K×16	2112×8	16~28	8	36×16~48×16	16-bit×2	8	4	√	—	100QFP
HT95L400 HT95L40P	2.4V~5.5V	16K×16	2880×8	28~40	8	36×16~48×16	16-bit×2	12	4	√	—	128QFP
HT95C200 HT95C20P	2.4V~5.5V	8K×16	1152×8	20~28	8	24×8~24×16	16-bit×2	8	4	√	√	128QFP
HT95C300 HT95C30P	2.4V~5.5V	8K×16	2112×8	16~28	8	36×16~48×16	16-bit×2	8	4	√	√	128QFP
HT95C400 HT95C40P	2.4V~5.5V	16K×16	2880×8	28~40	8	36×16~48×16	16-bit×2	12	4	√	√	128QFP

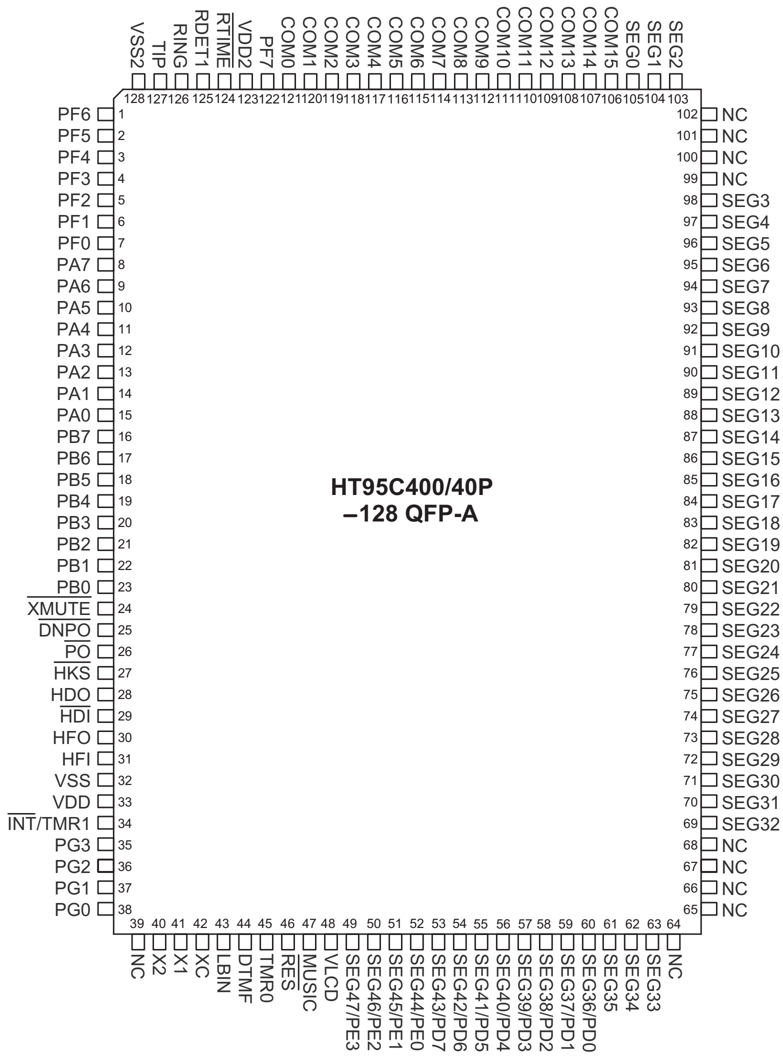
Note: Part numbers suffixed with "P" are OTP devices, all others are mask version devices.

Block Diagram (HT95C400/40P)

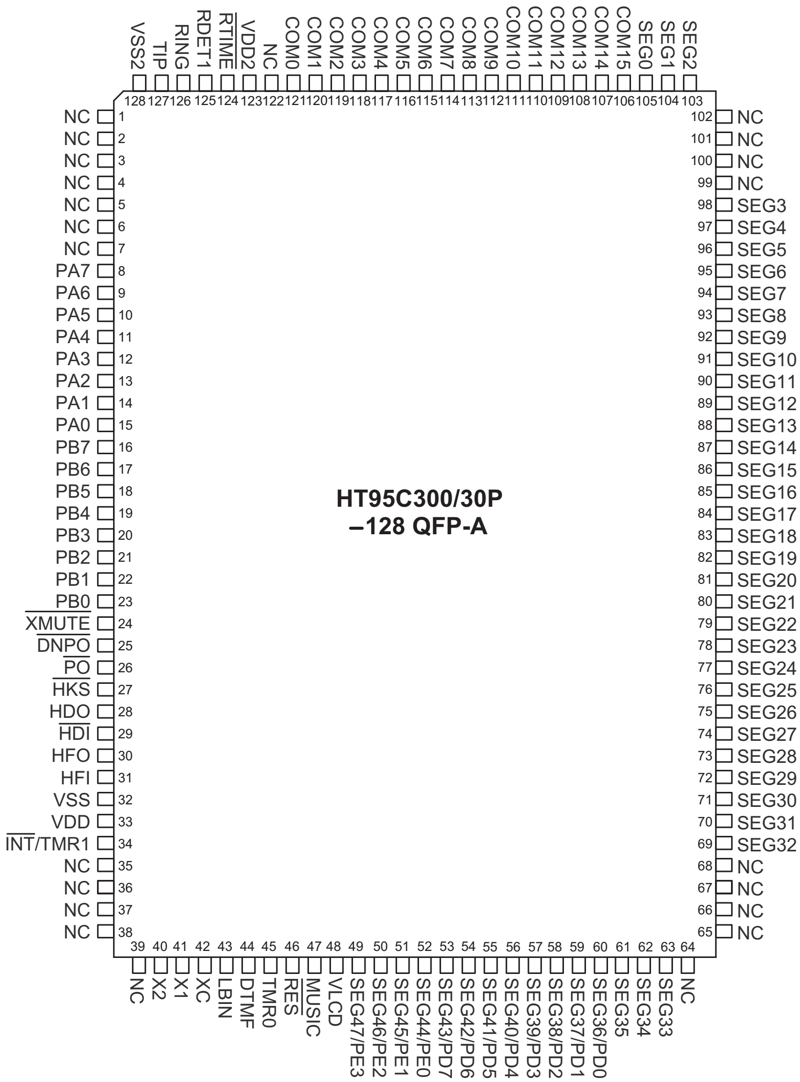


Pin Assignment

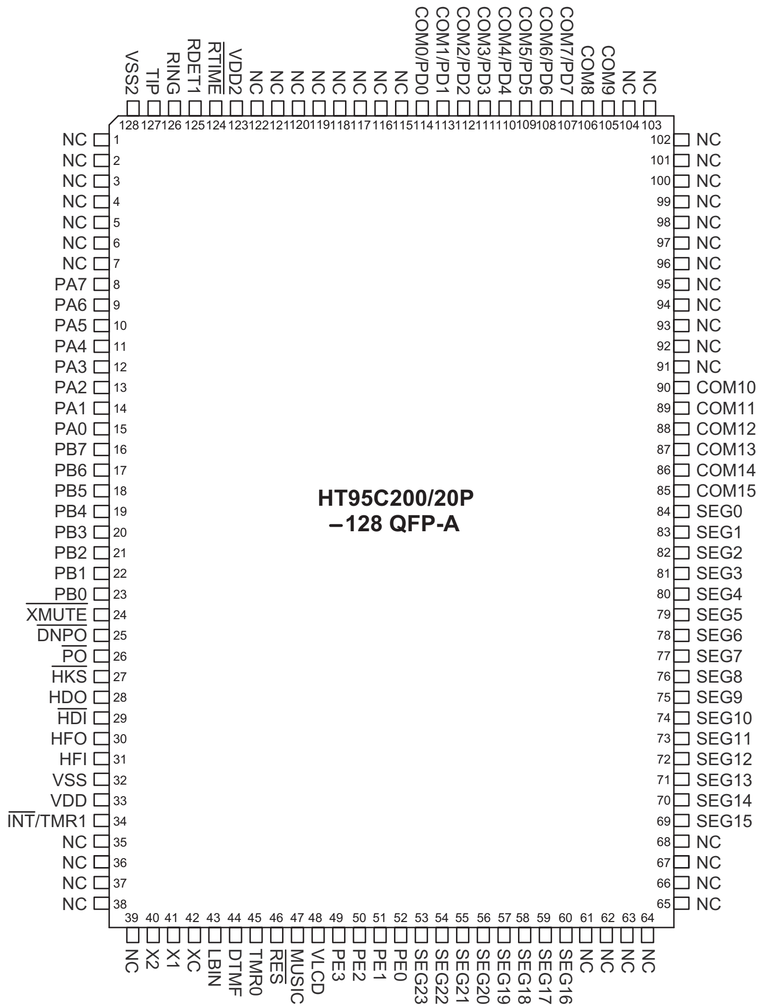
HT95C400/40P



HT95C300/30P



HT95C200/20P



Pin Description

Pin Name	I/O	Description
CPU		
VDD	—	Positive power supply
VDD2		Positive power supply for FSK decoder
VSS	—	Negative power supply, ground
VSS2		Negative power supply for FSK decoder, ground
X1	I	A 32768Hz crystal (or resonator) should be connected to this pin and X2.
X2	O	A 32768Hz crystal (or resonator) should be connected to this pin and X1.
XC	I	External low pass filter used for frequency up conversion circuit.
RES	I	Schmitt trigger reset input, active low.
INT/TMR1	I	Schmitt trigger input for external interrupt or Timer/Event Counter 1. No internal pull-high resistor. For INT: Edge trigger activated on a falling edge. For TMR1: Activated on falling or rising transition edge, selected by software.
TMR0	I	Schmitt trigger input for Timer/Event Counter 0. No internal pull-high resistor. Activated on falling or rising transition edge, selected by software.

Pin Name	I/O	Description
LCD Driver		
SEG47~SEG0	O or I/O	LCD panel segment outputs. Some segment outputs can be optioned to Bidirectional input/output ports by software. (See the ″LCD Driver″ function)
COM15~COM0	O or I/O	LCD panel common outputs. Some common outputs can be optioned to Bidirectional input/output ports by software. (See the ″LCD Driver″ function)
VLCD	I	LCD driver power source.
Normal I/O		
PA7~PA0	I/O	Bidirectional input/output ports. Schmitt trigger input and CMOS output. See mask option table for pull-high and wake-up function
PB7~PB0	I/O	Bidirectional input/output ports. Schmitt trigger input and CMOS output. See mask option table for pull-high function
PD7~PD0	I/O	Bidirectional input/output ports. Schmitt trigger input and CMOS output. See mask option table for pull-high function Port D could be optioned to LCD signal output, see the ″Input/Output Ports″ function
PE3~PE0	I/O	Bidirectional input/output ports. Schmitt trigger input and CMOS output. See mask option table for pull-high function Port E could be optioned to LCD signal output, see the ″Input/Output Ports″ function
PF7~PF0	I/O	Bidirectional input/output ports. Schmitt trigger input and CMOS output. See mask option table for pull-high function
PG3~PG0	I/O	Bidirectional input/output ports. Schmitt trigger input and CMOS output. See mask option table for pull-high function
Dialer I/O (See the ″Dialer I/O function″)		
HFI	I	Schmitt trigger input structure. An external RC network is recommended for input debouncing. This pin is pulled low with internal resistance of 200kΩ typ.
HFO	O	CMOS output structure.
$\overline{\text{HDI}}$	I	Schmitt trigger input structure. An external RC network is recommended for input debouncing. This pin is pulled high with internal resistance of 200kΩ typ.
HDO	O	CMOS output structure.
$\overline{\text{HKS}}$	I	This pin detects the status of the hook-switch and its combination with HFI/$\overline{\text{HDI}}$ can control the $\overline{\text{PO}}$ pin output to make or break the line.
$\overline{\text{PO}}$	O	CMOS output structure controlled by $\overline{\text{HKS}}$ and HFI/$\overline{\text{HDI}}$ pins and which determines whether the dialer connects or disconnects the telephone line.
$\overline{\text{DNPO}}$	O	NMOS output structure.
$\overline{\text{XMUTE}}$	O	NMOS output structure. Usually, $\overline{\text{XMUTE}}$ is used to mute the speech circuit when transmitting the dialer signal.

Pin Name	I/O	Description
Peripherals		
DTMF	O	This pin outputs dual tone signals to dial out the phone number. The load resistor should not be less than 5kΩ.
MUSIC	O	This pin outputs the single tone that is generated by the PFD generator.
TIP	I	Input pin connected to the tip side of the twisted pair wires. It is internally biased to 1/2 VDD when the device is in power-up mode. This pin must be DC isolated from the line.
RING	I	Input pin connected to the ring side of the twisted pair wires. It is internally biased to 1/2 VDD when the device is in power-up mode. This pin must be DC isolated from the line.
RDET1	I	This pin detects ring energy on the line through an attenuating network.
$\overline{\text{RTIME}}$	I/O	Schmitt trigger input and NMOS output pin which functions with RDET1 pin to make an RC network that performs ring detection function.
LBIN	I	This pin detects battery low through external R1/R2 to determine threshold voltage.

Absolute Maximum Ratings

Supply Voltage .... V_{SS}–0.3V to V_{SS}+5.5V

Input Voltage .... V_{SS}–0.3 to V_{DD}+0.3V

Storage Temperature ....–50°C to 125°C

Operating Temperature ....–20°C to 70°C

Note: These are stress ratings only. Stresses exceeding the range specified under "Absolute Maximum Ratings" may cause substantial damage to the device. Functional operation of this device at other conditions beyond those listed in the specification is not implied and prolonged exposure to extreme conditions may affect device reliability.

Electrical Characteristics

Ta=25°C

Symbol	Parameter	Test Conditions		Min.	Typ.	Max.	Unit
		V_{DD}	Conditions				
CPU							
I_{IDL}	Idle Mode Current	5V	32768Hz off, 3.58MHz off, CPU off, LCD off, WDT off, no load	—	—	2	μA
I_{SLP}	Sleep Mode Current	5V	32768Hz on, 3.58MHz off, CPU off, LCD off, WDT off, no load	—	17	30	μA
I_{GRN}	Green Mode Current	5V	32768Hz on, 3.58MHz off, CPU on, LCD off, WDT off, no load	—	28	50	μA
I_{NOR}	Normal Mode Current	5V	32768Hz on, 3.58MHz on, CPU on, LCD on, WDT on, DTMF generator off, FSK decoder off, no load	—	1.8	3	mA
V_{IL}	I/O Port Input Low Voltage	5V	—	0	—	1	V
V_{IH}	I/O Port Input High Voltage	5V	—	4	—	5	V
I_{OL}	I/O Port Sink Current	5V	—	4	6	—	mA
I_{OH}	I/O Port Source Current	5V	—	–2	–3	—	mA
R_{PH}	Pull-high Resistor	5V	—	10	30	—	kΩ
V_{LBIN}	Low Battery Detection Reference Voltage	5V	—	1.05	1.15	1.25	V

Symbol	Parameter	Test Conditions		Min.	Typ.	Max.	Unit
		V_{DD}	Conditions				
LCD Driver							
V_{LCD}	LCD Panel Power Supply	—	—	—	3	5	V
I_{LCD}	LCD Operation Current	—	V_{LCD}=5V, 32768Hz, no load	—	—	100	μA
Dialer I/O							
I_{XMO}	$\overline{XMUTE}$ Leakage Current	2.5V	$\overline{XMUTE}$ pin=2.5V	—	—	1	μA
I_{OLXM}	$\overline{XMUTE}$ Sink Current	2.5V	$\overline{XMUTE}$ pin=0.5V	1	—	—	mA
$I_{\overline{HKS}}$	$\overline{HKS}$ Input Current	2.5V	$\overline{HKS}$ pin=2.5V	—	—	0.1	μA
R_{HFI}	HFI Pull-low Resistance	2.5V	V_{HFI}=2.5V	—	200	—	kΩ
$R_{\overline{HDI}}$	$\overline{HDI}$ Pull-high Resistance	2.5V	$V_{\overline{HDI}}$=0V	—	200	—	kΩ
I_{OH2}	HFO Source Current	2.5V	V_{OH}=2V	−1	—	—	mA
I_{OL2}	HFO Sink Current	2.5V	V_{OL}=0.5V	1	—	—	mA
I_{OH3}	HDO Source Current	2.5V	V_{OH}=2V	−1	—	—	mA
I_{OL3}	HDO Sink Current	2.5V	V_{OL}=0.5V	1	—	—	mA
I_{OH4}	$\overline{PO}$ Source Current	2.5V	V_{OH}=2V	−1	—	—	mA
I_{OL4}	$\overline{PO}$ Sink Current	2.5V	V_{OL}=0.5V	1	—	—	mA
I_{OL5}	$\overline{DNPO}$ Sink Current	2.5V	V_{OL}=0.5V	1	—	—	mA
DTMF Generator							
V_{TDC}	DTMF Output DC Level	—	—	$0.45V_{DD}$	—	$0.7V_{DD}$	V
V_{TOL}	DTMF Sink Current	—	V_{DTMF}=0.5V	0.1	—	—	mA
V_{TAC}	DTMF Output AC Level	—	Row group, R_L=5kΩ	120	155	180	mVrms
R_L	DTMF Output Load	—	THD≤−23dB	5	—	—	kΩ
A_{CR}	Column Pre-emphasis	—	Row group=0dB	1	2	3	dB
THD	Tone Signal Distortion	—	R_L=5kΩ	—	−30	−23	dB
FSK Decoder							
	Input Sensitivity: TIP, RING	—	—	−40	−45	—	dBm
	Transmission Rate	5V	—	1188	1200	1212	baud
S/N	Signal to Noise Ratio	—	—	—	20	—	dB
	Band-pass Filter Frequency Response Relative to 1700Hz at 0dBm ≤60Hz 550Hz 2700Hz ≥3300Hz	—	—	— — — —	−64 −4 −3 −34	— — — —	dB
	Carrier Detect Sensitivity	—	—	—	−48	—	dBm
t_{SUPD}	Power Up to FSK Signal Set Up Time	—	—	15	—	—	ms

Functional Description

Execution Flow

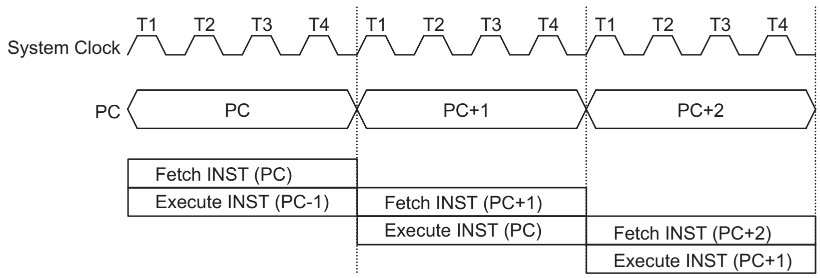
The system clock for the telephone controller is derived from a 32768Hz crystal oscillator. A built-in frequency up conversion circuit provides dual system clock, namely; 32768Hz and 3.58MHz. The system clock is internally divided into four non-overlapping clocks. One instruction cycle consists of four system clock cycles. Instruction fetching and execution are pipelined in such a way that a fetch takes an instruction cycle while decoding and execution takes the next instruction cycle. The pipelining scheme causes each instruction to be effectively executed in a instruction cycle. If an instruction changes the program counter, two instruction cycles are required to complete the instruction.

Program Counter – PC

The program counter (PC) controls the sequence in which the instructions stored in the program ROM are executed and its contents specify a full range of program memory. After accessing a program memory word to fetch an instruction code, the contents of the program counter are incremented by 1. The program counter then points to the memory word containing the next instruction code.

When executing a jump instruction, conditional skip execution, loading PCL register, subroutine call, initial reset, internal interrupt, external interrupt or return from subroutine, the program counter manipulates the program transfer by loading the address corresponding to each instruction. The conditional skip is activated by instructions. Once the condition is met, the next instruction, fetched during the current instruction execution, is discarded and a dummy cycle replaces it to get the proper instruction. Otherwise proceed to the next instruction.

The program counter lower order byte register (PCL:06H) is a readable and write-able register. Moving data into the PCL performs a short jump. The destination will be within 256 locations. When a control transfer takes place, an additional dummy cycle is required.



Execution Flow

Mode	Program Counter													
	*13	*12	*11	*10	*9	*8	*7	*6	*5	*4	*3	*2	*1	*0
Initial reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0
External interrupt	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Timer/Event Counter 0 overflow	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Timer/Event Counter 1 overflow	0	0	0	0	0	0	0	0	0	0	1	1	0	0
Peripheral interrupt	0	0	0	0	0	0	0	0	0	1	0	0	0	0
RTC interrupt	0	0	0	0	0	0	0	0	0	1	0	1	0	0
Dialer I/O interrupt	0	0	0	0	0	0	0	0	0	1	1	0	0	0
Skip	Program Counter+2 (within current bank)													
Loading PCL	*13	*12	*11	*10	*9	*8	@7	@6	@5	@4	@3	@2	@1	@0
Jump, call branch	BP.5	#12	#11	#10	#9	#8	#7	#6	#5	#4	#3	#2	#1	#0
Return from subroutine	S13	S12	S11	S10	S9	S8	S7	S6	S5	S4	S3	S2	S1	S0

Program ROM Address

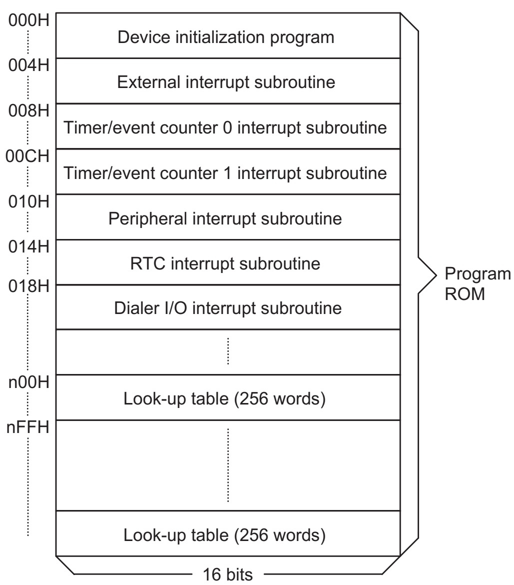
Note: *13~*0: Program counter bits
S13~S0: Stack register bits
#12~#0: Instruction code bits
@7~@0: PCL bits

Available bits of program counter for HT95C400/40P: Bit 13~Bit 0
Available bits of program counter for HT95C300/30P: Bit 12~Bit 0
Available bits of program counter for HT95C200/20P: Bit 12~Bit 0

Program Memory – ROM

The program memory is used to store the program instructions which are to be executed. It also contains data, table, and interrupt entries, and is organized into 8K×16 bits×2 banks (HT95C400/40P) or 8K×16 bits (HT95C300/30P, HT95C200/20P) addressed by the program counter and table pointer.

For the HT95C400/40P, the program memory is divided into 2 banks, each bank having a ROM Size 8K×16 bits. To move from the present ROM bank to a different ROM bank, the higher 1 bits of the ROM address are set by the BP (Bank Pointer), while the remaining 13 bits of the PC are set in the usual way by executing the appropriate jump or call instruction. As the 14 address bits are latched during the execution of a call or jump instruction, the correct value of the BP must first be setup before a jump or call is executed. When either a software or hardware interrupt is received, note that no matter which ROM bank the program is in, the program will always jump to the appropriate interrupt service address in Bank 0. The original 14 bits address will be stored on the stack and restored when the relevant RET/RETI instruction is executed, automatically returning the program to the original ROM bank. This eliminates the need for programmers to manage the BP when interrupts occur. Certain locations in the program memory are reserved for special usage:



Note: The Last page for HT95C400/40P is 3F00H~3FFFH
The Last page for HT95C300/30P is 1F00H~1FFFH
The Last page for HT95C200/20P is 1F00H~1FFFH

Program Memory

- Location 0000H (Bank0)

  This area is reserved for the initialization program. After chip power-on reset or external reset or WDT time-out reset, the program always begins execution at location 0000H.

- Location 0004H (Bank0)

  This area is reserved for the external interrupt service program. If the $\overline{\text{INT}}$/TMR1 input pin is activated, the external interrupt is enabled and the stack is not full, the program begins execution at location 0004H.

- Location 0008H (Bank0)

  This area is reserved for the Timer/Event Counter 0 interrupt service program. If a timer interrupt results from a Timer/Event Counter 0 overflow, the Timer/Event Counter 0 interrupt is enabled and the stack is not full, the program begins execution at location 0008H.

- Location 000CH (Bank0)

  This location is reserved for the Timer/Event Counter 1 interrupt service program. If a timer interrupt results from a Timer/Event Counter 1 overflow, the Timer/Event Counter 1 interrupt is enabled and the stack is not full, the program begins execution at location 000CH.

- Location 0010H (Bank0)

  This location is reserved for the peripherals interrupt service program. When the FSK decoder detects a ringer or line reversal or FSK carrier signal or FSK packet data, the FSK interrupt is generated. If these interrupts occurred, the peripheral interrupt is enabled and the stack is not full, the program begins execution at location 0010H. The programmer could distinguish from these interrupts from the FSKS register.

- Location 0014H (Bank0)

  This location is reserved for real time clock (RTC) interrupt service program. When RTC generator is enabled and time-out occurs, the RTC interrupt is enabled and the stack is not full, the program begins execution at location 0014H.

- Location 0018H (Bank0)

  This location is reserved for the $\overline{\text{HKS}}$ pin edge transition or $\overline{\text{HDI}}$ pin falling edge transition or HFI pin rising edge transition. If this condition occurs, the dialer I/O interrupt is enabled and the stack is not full, the program begins execution at location 18H.

Table Location

Any location in the ROM space can be used as look-up tables. The instructions ″TABRDC [m]″ (the current page, one page=256 words) and ″TABRDL [m]″ (the last page) transfer the contents of the lower-order byte to the specified data memory, and the higher-order byte to TBLH (08H). For the HT95C400/40P, the instruction ″TABRDC [m]″ is used for any page of any bank. Only the destination of the lower-order byte in the table is well-defined, and the higher-order byte of the table word is transferred to TBLH. The table pointer (TBLP) or (TBHP, TBLP for the HT95C400/40P) is a read/write register (07H) or (1FH, 07H for the HT95C400/40P), which indicates the table location. Before accessing the table, the location must be placed in the (TBLP) or (TBHP, TBLP for the HT95C400/40P). The TBLH is read only and cannot be restored. If the main routine and the ISR (Interrupt Service Routine) both employ the table read instruction, the contents of the TBLH in the main routine are likely to be changed by the table read instruction used in the ISR. Errors will then occur. Hence, simultaneously using the table read instruction in the main routine and the ISR should be avoided. However, if the table read instruction has to be applied in both the main routine and the ISR, the interrupt should be disabled prior to the table read instruction. It will not be enabled until the TBLH has been backed-up. All table related instructions require two cycles to complete the operation. These areas may function as normal program memory depending on the requirements.

Stack Register

This is a special part of the memory which is used to save the contents of the program counter only. The stack is organized into 12 levels (HT95C400/40P) or 8 levels (HT95C300/30P, HT95C200/20P) and is neither part of the data nor part of the program space, and is neither readable nor writable. The activated level is indexed by the stack pointer (SP) and is neither readable nor writable. At a subroutine call or interrupt acknowledge signal, the contents of the program counter are pushed onto the stack. At the end of a subroutine or an interrupt routine, signaled by a return instruction (RET or RETI), the program counter is restored to its previous value from the stack. After a chip reset, the SP will point to the top of the stack. If the stack is full and an interrupt takes place, the interrupt request flag will be recorded but the acknowledge signal will be inhibited even if this interrupt is enabled. When the stack pointer is decremented (by RET or RETI), the interrupt will be serviced. This feature prevents stack overflow allowing the programmer to use the structure more easily. If the stack is full and a ″CALL″ is subsequently executed, stack overflow occurs and the first entry will be lost (only the most recent 12 or 8, depending on various MCU type, returned addresses are stored).

Data Memory

The data memory is divided into four functional groups: special function registers, embedded control register, LCD display memory and general purpose memory. Most are read/write, but some are read only.

The special function registers are located from 00H to 1FH. The embedded control registers are located in the memory areas from 20H to 3FH. The remaining spaces which are not specified in the following table before the 40H are reserved for future expanded usage and reading these locations will get ″00H″. The general purpose data memory is divided into 15 banks (HT95C400/40P), 11 banks (HT95C300/30P) or 6 banks (HT95C200/20P). The banks in the RAM are all addressed from 40H to 0FFH and they are selected by setting the value of the bank pointer (BP).

HT95C400/40P

Instruction(s)	Table Location													
	*13	*12	*11	*10	*9	*8	*7	*6	*5	*4	*3	*2	*1	*0
TABRDC [m]	#5	#4	#3	#2	#1	#0	@7	@6	@5	@4	@3	@2	@1	@0
TABRDL [m]	1	1	1	1	1	1	@7	@6	@5	@4	@3	@2	@1	@0

HT95C300/30P, HT95C200/20P

Instruction(s)	Table Location												
	*12	*11	*10	*9	*8	*7	*6	*5	*4	*3	*2	*1	*0
TABRDC [m]	P12	P11	P10	P9	P8	@7	@6	@5	@4	@3	@2	@1	@0
TABRDL [m]	1	1	1	1	1	@7	@6	@5	@4	@3	@2	@1	@0

Note: *13~*0: Table location bits
@7~@0: TBLP register bit7~bit0
#7~#0: TBHP register bit7~bit0
P12~P8: Current program counter bits

All of the data memory areas can handle arithmetic, logic, increment, decrement and rotate operations directly. Except for some dedicated bits, each bit in the data memory can be set and reset by ″SET [m].i″ and ″CLR [m].i″. They are also indirectly accessible through memory pointer registers (MP0 or MP1). The bank1~bank14 and bank27 are only indirectly accessible through memory pointer 1 register (MP1).

The LCD display memory is located at bank 1BH. They can be read and written to by the indirect addressing mode using memory pointer 1 (MP1). To turn the display On or Off, a ″1″ or ″0″ is written to the corresponding bit of the memory area.

Special Register, Embedded Control Register, LCD Display Memory and General Purpose RAM

BP (RAM Bank)	Address	Function	Description	Supported for HT95CXXX		
				400/P	300/P	200/P
Special Function Register						
00H	00H	IAR0	Indirect addressing register 0	√	√	√
00H	01H	MP0	Memory pointer register 0	√	√	√
00H	02H	IAR1	Indirect addressing register 1	√	√	√
00H	03H	MP1	Memory pointer register 1	√	√	√
00H	04H	BP	Bank Pointer register	√	√	√
00H	05H	ACC	Accumulator	√	√	√
00H	06H	PCL	Program counter lower-order byte register	√	√	√
00H	07H	TBLP	Table pointer	√	√	√
00H	08H	TBLH	Table higher-order byte register	√	√	√
00H	09H	WDTS	Watchdog Timer option setting register	√	√	√
00H	0AH	STATUS	Status register	√	√	√
00H	0BH	INTC0	Interrupt control register 0	√	√	√
00H	0CH	TMR0H	Timer/Event Counter 0 high-order byte register	√	√	√
00H	0DH	TMR0L	Timer/Event Counter 0 low-order byte register	√	√	√
00H	0EH	TMR0C	Timer/Event Counter 0 control register	√	√	√
00H	0FH	TMR1H	Timer/Event Counter 1 high-order byte register	√	√	√
00H	10H	TMR1L	Timer/Event Counter 1 low-order byte register	√	√	√
00H	11H	TMR1C	Timer/Event Counter 1 control register	√	√	√
00H	12H	PA	Port A data register	√	√	√
00H	13H	PAC	Port A control register	√	√	√
00H	14H	PB	Port B data register	√	√	√
00H	15H	PBC	Port B control register	√	√	√
00H	16H	DIALERIO	Dialer I/O register	√	√	√
00H	18H	PD	Port D data register	√	√	√
00H	19H	PDC	Port D control register	√	√	√
00H	1AH	PE	Port E data register	√	√	√
00H	1BH	PEC	Port E control register	√	√	√
00H	1EH	INTC1	Interrupt control register 1	√	√	√
00H	1FH	TBHP	Table high-order byte pointer	√	—	—

BP (RAM Bank)	Address	Function	Description	Supported for HT95CXXX		
				400/P	300/P	200/P
Embedded Control Register						
00H	20H	DTMFC	DTMF generator control register	√	√	√
00H	21H	DTMFD	DTMF generator data register	√	√	√
00H	22H	LINE	Line control register	√	√	√
00H	24H	RTCC	Real time clock control register	√	√	√
00H	26H	MODE	Operation mode control register	√	√	√
00H	28H	LCDIO	LCD segment and I/O option register	√	√	—
00H	29H	FSKC	FSK decoder control register	√	√	√
00H	2AH	FSKS	FSK decoder status register	√	√	√
00H	2BH	FSKD	FSK packet data register	√	√	√
00H	2DH	LCDC	LCD driver control register	√	√	√
00H	2EH	PFDC	PFD control register	√	√	√
00H	2FH	PFDD	PFD data register	√	√	√
00H	34H	PF	Port F data register	√	—	—
00H	35H	PFC	Port F control register	√	—	—
00H	36H	PG	Port G data register	√	—	—
00H	37H	PGC	Port G control register	√	—	—
General Purpose RAM						
00H	40H~FFH	BANK0 RAM	General purpose RAM space	√	√	√
01H	40H~FFH	BANK1 RAM	General purpose RAM space	√	√	√
02H	40H~FFH	BANK2 RAM	General purpose RAM space	√	√	√
03H	40H~FFH	BANK3 RAM	General purpose RAM space	√	√	√
04H	40H~FFH	BANK4 RAM	General purpose RAM space	√	√	√
05H	40H~FFH	BANK5 RAM	General purpose RAM space	√	√	√
06H	40H~FFH	BANK6 RAM	General purpose RAM space	√	√	—
07H	40H~FFH	BANK7 RAM	General purpose RAM space	√	√	—
08H	40H~FFH	BANK8 RAM	General purpose RAM space	√	√	—
09H	40H~FFH	BANK9 RAM	General purpose RAM space	√	√	—
0AH	40H~FFH	BANK10 RAM	General purpose RAM space	√	√	—
0BH	40H~FFH	BANK11 RAM	General purpose RAM space	√	—	—
0CH	40H~FFH	BANK12 RAM	General purpose RAM space	√	—	—
0DH	40H~FFH	BANK13 RAM	General purpose RAM space	√	—	—
0EH	40H~FFH	BANK14 RAM	General purpose RAM space	√	—	—
LCD RAM Display Memory						
1BH	40H~9FH	LCD RAM	LCD RAM mapping space for COM0~COM15 (see "LCD Driver" function)			

Indirect Addressing Register

Location 00H and 02H are indirect addressing registers that are not physically implemented. Any read/write operation of [00H] and [02H] will access the memory pointed to by MP0 and MP1, respectively. Reading location [00H] or [02H] indirectly returns the result 00H, while writing it leads to no operation. MP0 is indirectly addressable in bank0, but MP1 is available for all banks by switch BP [04H]. If BP is unequal to 00H, the indirect addressing mode to read/write operation from 00H~3FH will return the result as same as the value of bank0.

The memory pointer registers MP0 and MP1 are 8-bits registers, and the bank pointer register BP is 6-bits register for the HT95C400/40P or 5-bits for the other devices in the series.

Accumulator

The accumulator is closely related to ALU operations. It is also mapped to location 05H of the data memory and can operate with immediate data. All data movement between two data memory locations must pass through the accumulator.

Arithmetic and Logic Unit – ALU

This circuit performs 8-bit arithmetic and logic operations and provides the following functions:

- Arithmetic operations (ADD, ADC, SUB, SBC, DAA)
- Logic operations (AND, OR, XOR, CPL)
- Rotation (RL, RR, RLC, RRC)
- Increment and Decrement (INC, DEC)
- Branch decision (SZ, SNZ, SIZ, SDZ, etc.)

The ALU not only saves the results of a data operation but also changes the status register.

Status Register – STATUS

This status register contains the carry flag (C), auxiliary carry flag (AC), zero flag (Z), overflow flag (OV), power down flag (PDF), and watchdog time-out flag (TO). It also records the status information and controls the operation sequence.

Except for the TO and PDF flags, bits in the status register can be altered by instructions, similar to the other registers. Data written into the status register will not change the TO or PDF flag. Operations related to the status register may yield different results from those intended. The TO flag can be affected only by system power-up, a WDT time-out or executing the ″CLR WDT″ or ″HALT″ instruction. The PDF flag can be affected only by executing the ″HALT″ or ″CLR WDT″ instruction or during a system power-up.

The Z, OV, AC and C flags generally reflect the status of the latest operations.

On entering the interrupt sequence or executing the subroutine call, the status register will not be automatically pushed onto the stack.

If the contents of the status are important and if the subroutine can corrupt the status register, precautions must be taken to save it.

Interrupt

The telephone controller provides an external interrupt, internal timer/event counter interrupt, a peripheral interrupt, an internal real time clock interrupt and internal dialer I/O interrupt. The Interrupt Control Registers 0 and Interrupt Control Register 1 both contains the interrupt control bits that set the enable/disable and the interrupt request flags.

Once an interrupt subroutine is serviced, all the other interrupts will be blocked (by hardware clearing the EMI bit). This scheme may prevent any further interrupt nesting. Other interrupt requests may occur during this interval but only the interrupt request flag is recorded. If a certain interrupt requires servicing within the service routine, the EMI bit and the corresponding bit of the INTC0 (INTC1) may be set to allow interrupt nesting.

Bit No.	Label	Function
0	C	C is set if the operation results in a carry during an addition operation or if a borrow does not take place during a subtraction operation; otherwise C is cleared. Also it is affected by a rotate through carry instruction.
1	AC	AC is set if the operation results in a carry out of the low nibbles in addition or no borrow from the high nibble into the low nibble in subtraction; otherwise AC is cleared.
2	Z	Z is set if the result of an arithmetic or logic operation is 0; otherwise Z is cleared.
3	OV	OV is set if the operation results in a carry into the highest-order bit but not a carry out of the highest-order bit, or vice versa; otherwise OV is cleared.
4	PDF	PDF is cleared when either a system power-up or executing the ″CLR WDT″ instruction. PDF is set by executing the ″HALT″ instruction.
5	TO	TO is cleared by a system power-up or executing the ″CLR WDT″ or ″HALT″ instruction. TO is set by a WDT time-out.
6, 7	—	Unused bit, read as ″0″

STATUS (0AH) Register

If the stack is full, any other interrupt request will not be acknowledged, even if the related interrupt is enabled, until the stack pointer is decremented. If immediate service is desired, the stack must be prevented from becoming full.

All these kinds of interrupts have a wake-up capability. As an interrupt is serviced, a control transfer occurs by pushing the program counter onto the stack, followed by a branch to a subroutine at specified location in the program memory. Only the program counter is pushed onto the stack. If the contents of the register or status register (STATUS) are altered by the interrupt service program which corrupts the desired control sequence, the contents should be saved in advance.

External interrupt is triggered by a high to low transition of the $\overline{\text{INT}}$/TMR1 pin and the interrupt request flag EIF will be set. When the external interrupt is enabled, the stack is not full and the external interrupt is active, a subroutine call to location 04H will occur. The interrupt request flag EIF and EMI bits will be cleared to disable other interrupts.

The Timer/Event Counter 0 interrupt is generated by a timeout overflow and the interrupt request flag T0F will be set. When the Timer/Event Counter 0 interrupt is enabled, the stack is not full and the T0F bit is set, a subroutine call to location 08H will occur. The interrupt request flag T0F and EMI bits will be cleared to disable further interrupts.

The Timer/Event Counter 1 interrupt is generated by a timeout overflow and the interrupt request flag T1F will be set. When the Timer/Event Counter 1 interrupt is enabled, the stack is not full and the T1F bit is set, a subroutine call to location 0CH will occur. The interrupt request flag T1F and EMI bits will be cleared to disable further interrupts.

The peripheral interrupt is activated when the FSK decoder detect the ring signal or line reversal or FSK carrier signal or FSK packet data. When these interrupts occurred, the interrupt request flag PERF will be set. When the peripheral interrupt is enabled, the stack is not full and the PERF is set, a subroutine call to location 10H will occur. The interrupt request flag PERF and EMI bits will be cleared to disable other interrupts.

The real time clock interrupt is generated by a 1Hz RTC generator. When the RTC time-out occurs, the interrupt request flag RTCF will be set. When the RTC interrupt is enabled, the stack is not full and the RTCF is set, a subroutine call to location 14H will occur. The interrupt request flag RTCF and EMI bits will be cleared to disable other interrupts.

The dialer I/O interrupt is triggered by any edge transition onto $\overline{\text{HKS}}$ pin or a falling edge transition onto $\overline{\text{HDI}}$ pin or a rising edge transition onto HFI pin, the interrupt request flag DRF will be set. When the dialer I/O interrupt is enabled, the stack is not full and the DRF is set, a subroutine call to location 18H will occur. The interrupt request flag DRF and EMI bits will be cleared to disable other interrupts.

Bit No.	Label	R/W	Function
0	EMI	RW	Controls the master (global) interrupt (1=enabled; 0=disabled)
1	EEI	RW	Controls the external interrupt (1=enabled; 0=disabled)
2	ET0I	RW	Controls the Timer/Event Counter 0 interrupt (1=enabled; 0=disabled)
3	ET1I	RW	Controls the Timer/Event Counter1 interrupt (1=enabled; 0=disabled)
4	EIF	RW	External interrupt request flag (1=active; 0=inactive)
5	T0F	RW	Timer/Event Counter 0 request flag (1=active; 0=inactive)
6	T1F	RW	Timer/Event Counter1 request flag (1=active; 0=inactive)
7	—	RO	Unused bit, read as ″0″

INTC0 (0BH) Register

Bit No.	Label	R/W	Function
0	EPERI	RW	Control the peripheral interrupt (1=enable; 0=disable)
1	ERTCI	RW	Control the real time clock interrupt (1=enable; 0=disable)
2	EDRI	RW	Control the dialer I/O interrupt (1=enable; 0=disable)
3	—	RO	Unused bit, read as ″0″
4	PERF	RW	Peripheral interrupt request flag (1=active; 0=inactive)
5	RTCF	RW	Internal real time clock interrupt request flag (1=active; 0=inactive)
6	DRF	RW	Internal dialer I/O interrupt request flag (1=active: 0=inactive)
7	—	RO	Unused bit, read as ″0″

INTC1 (1EH) Register

Note: 1. If the dialer status is on-hook and hold-line, the falling edge transition onto $\overline{HDI}$ pin will not generate the dialer I/O interrupt.

2. The dialer I/O interrupt will be disabled when the operation mode is in Idle mode.

During the execution of an interrupt subroutine, other interrupt acknowledge signals are held until the RETI instruction is executed or the EMI bit and the related interrupt control bit are set to 1 (if the stack is not full). To return from the interrupt subroutine, ″RET″ or ″RETI″ may be invoked. RETI will set the EMI bit to enable an interrupt service, but RET will not.

Interrupts, occurring in the interval between the rising edges of two consecutive T2 pulses, will be serviced on the latter of the two T2 pulses, if the corresponding interrupts are enabled. In the case of simultaneous requests the following table shows the priority that is applied. These can be masked by resetting the EMI bit.

Interrupt Source	Priority	Vector
External interrupt	1	04H
Timer/Event Counter 0 interrupt	2	08H
Timer/Event Counter 1 interrupt	3	0CH
Peripheral interrupt	4	10H
Real time clock interrupt	5	14H
Dialer I/O interrupt	6	18H

Priority of the Interrupt

EMI, EEI, ET0I, ET1I, EPERI, ERTCI and EDRI are used to control the enabling/disabling of interrupts. These bits prevent the requested interrupt from being serviced. Once the interrupt request flags (EIF, T0F, T1F, PERF, RTCF, DRF) are set by hardware or software, they will remain in the INTC0 or INTC1 registers until the interrupts are serviced or cleared by a software instruction.

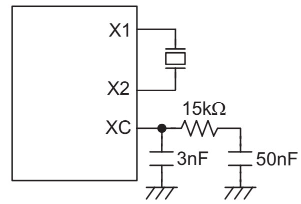
It is recommended that a program should not use the ″CALL subroutine″ within the interrupt subroutine. Interrupts often occur in an unpredictable manner or need to be serviced immediately in some applications. If only one stack is left and enabling the interrupt is not well controlled, the original control sequence will be damaged once the ″CALL″ operates in the interrupt subroutine.

Oscillator Configuration

There are two oscillator circuits in the controller, the external 32768Hz crystal oscillator and internal WDT OSC.

The 32768Hz crystal oscillator and frequency-up conversion circuit (32768Hz to 3.58MHz) are designed for dual system clock source. It is necessary for frequency conversion circuit to add external RC components to make up the low pass filter that stabilize the output frequency 3.58MHz (see the oscillator circuit).

The WDT OSC is a free running on-chip RC oscillator, and no external components are required. Even if the system enters the Idle mode (the system clock is stopped), the WDT OSC still works within a period of 78μs normally. When the WDT is disabled or the WDT source is not this RC oscillator, the WDT OSC will be disabled.



System Oscillator Circuit

Watchdog Timer – WDT

The WDT clock source is implemented by a WDT OSC or external 32768Hz or an instruction clock (system clock divided by 4), determined by the mask option. This timer is designed to prevent a software malfunction or sequence from jumping to an unknown location with unpredictable results. The Watchdog Timer can be disabled by mask option. If the Watchdog Timer is disabled, all the executions related to the WDT result in no operation.

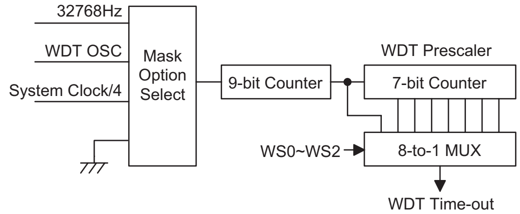
If the device operates in a noisy environment, using the on-chip WDT OSC or 32768Hz crystal oscillator is strongly recommended.

When the WDT clock source is selected, it will be first divided by 512 (9-stage) to get the nominal time-out period. By invoking the WDT prescaler, longer time-out periods can be realized. Writing data to WS2, WS1, WS0 can give different time-out periods.

The WDT OSC period is 78μs. This time-out period may vary with temperature, VDD and process variations. The WDT OSC always works for any operation mode.

If the instruction clock is selected as the WDT clock source, the WDT operates in the same manner except in the Sleep mode or Idle mode. In these two modes, the WDT stops counting and lose its protecting purpose. In this situation the logic can only be re-started by external logic.

If the WDT clock source is the 32768Hz, the WDT also operates in the same manner except in the Idle mode.



Watchdog Timer

Bit No.	Label	R/W	Function
0 1 2	WS0 WS1 WS2	RW	Watchdog Timer division ratio selection bits Bit 2, 1, 0=000, Division ratio=1:1 Bit 2, 1, 0=001, Division ratio=1:2 Bit 2, 1, 0=010, Division ratio=1:4 Bit 2, 1, 0=011, Division ratio=1:8 Bit 2, 1, 0=100, Division ratio=1:16 Bit 2, 1, 0=101, Division ratio=1:32 Bit 2, 1, 0=110, Division ratio=1:64 Bit 2, 1, 0=111, Division ratio=1:128
7~3	—	RW	Unused bit. These bits are read/write-able.

WDTS (09H) Register

When in the Idle mode, the 32768Hz stops, the WDT stops counting and lose its protecting purpose. In this situation the logic can only be re-started by external logic.

The high nibble and bit3 of the WDTS are reserved for user defined flags, which can be used to indicate some specified status.

The WDT time-out under Normal mode or Green mode will initialize ″chip reset″ and set the status bit ″TO″. But in the Sleep mode or Idle mode, the time-out will initialize a ″warm reset″ and only the program counter and stack pointer are reset to 0. To clear the WDT contents (including the WDT prescaler), three methods are adopted; external reset (a low level to $\overline{RES}$ pin), software instruction and a ″HALT″ instruction.

The software instruction include ″CLR WDT″ and the other set ″CLR WDT1″ and ″CLR WDT2″. Of these two types of instruction, only one can be active depending on the mask option ″WDT instr″. If the ″CLR WDT″ is selected (i.e. One clear instruction), any execution of the CLR WDT instruction will clear the WDT. In the case that ″CLR WDT1″ and ″CLR WDT2″ are chosen (i.e. Two clear instructions), these two instructions must be executed to clear the WDT; otherwise, the WDT may reset the chip as a result of time-out.

Controller Operation Mode

Holtek′s telephone controllers support two system clock and four operation modes. The system clock could be 32768Hz or 3.58MHz and operation mode could be Normal, Green, Sleep or Idle mode. These are all selected by the software.

The following conditions will force the operation mode to change to Green mode:

- Any reset condition from any operation mode
- Any interrupt from Sleep mode or Idle mode
- Port A wake-up from Sleep mode or Idle mode

How to change the Operation Mode

- Normal mode to Green mode:
  Clear MODE1 to 0, then operation mode is changed to Green mode but the UPEN status is not changed. However, UPEN can be cleared by software.

Bit No.	Label	R/W	Function
4~0	—	RO	Unused bit, read as ″0″
5	UPEN	RW	1: Enable frequency up conversion function to generate 3.58MHz 0: Disable frequency up conversion function to generate 3.58MHz
6	MODE0	RW	1: Disable 32768Hz oscillator while the HALT instruction is executed (Idle mode) 0: Enable 32768Hz oscillator while the HALT instruction is executed (Sleep mode)
7	MODE1	RW	1: Select 3.58MHz as CPU system clock 0: Select 32768Hz as CPU system clock

MODE (26H) Register

Operation Mode Description

HALT Instruction	MODE1	MODE0	UPEN	Operation Mode	32768Hz	3.58MHz	System Clock
Not execute	1	X	1	Normal	ON	ON	3.58MHz
Not execute	0	X	0	Green	ON	OFF	32768Hz
Be executed	0	0	0	Sleep	ON	OFF	HALT
Be executed	0	1	0	Idle	OFF	OFF	HALT

Note: ″X″ means don′t care

- Normal mode or Green mode to Sleep mode:
  Step 1: Clear MODE0 to 0
  Step 2: Clear MODE1 to 0
  Step 3: Clear UPEN to 0
  Step 4: Execute HALT instruction
  After Step 4, operation mode is changed to Sleep mode.
- Normal mode or Green mode to Idle mode:
  Step 1: Set MODE0 to 1
  Step 2: Clear MODE1 to 0
  Step 3: Clear UPEN to 0
  Step 4: Execute HALT instruction
  After Step 4, operation mode is changed to Idle mode.
- Green mode to Normal mode:
  Step 1: Set UPEN to 1
  Step 2: Software delay 20ms
  Step 3: Set MODE1 to 1
  After Step 3, operation mode is changed to Normal mode.
- Sleep mode or Idle mode to Green mode:
  Method 1: Any reset condition occurred
  Method 2: Any interrupt is active
  Method 3: Port A wake-up

Note The Timer0, Timer1, RTC and dialer I/O interrupt function will not work at the Idle mode because the 32768Hz crystal is stopped.

The reset conditions include power on reset, external reset, WDT time-out reset. By examining the processor status flag, PDF and TO, the program can distinguish between different ″reset conditions″. Refer to the Reset function for detailed description.

The port A wake-up and interrupt can be considered as a continuation of normal execution. Each bit in port A can be independently selected to wake-up the device by mask option. Awakening from Port A stimulus, the program will resume execution of the next instruction.

Any valid interrupts from Sleep mode or Idle mode may cause two sequences. One is if the related interrupt is disabled or the interrupt is enabled but the stack is full, the program will resume execution at the next instruction. The other is if the interrupt is enabled and the stack is not full, the regular interrupt response takes place. It is necessary to mention that if an interrupt request flag is set to ″1″ before entering the Sleep mode or Idle mode, the wake-up function of the related interrupt will be disabled.

Once a Sleep mode or Idle mode wake-up event occurs, it will take SST delay time (1024 system clock period) to resume to Green mode. In other words, a dummy period is inserted after a wake-up. If the wake-up results from an interrupt acknowledge signal, the actual interrupt subroutine execution will be delayed by one or more cycles. If the wake-up results in the next instruction execution, this will be executed immediately after the dummy period is finished.

To minimize power consumption, all the I/O pins should be carefully managed before entering the Sleep mode or Idle mode.

The Sleep mode or Idle mode is initialized by the HALT instruction and results in the following.

- The system clock will be turned off.
- The WDT function will be disabled if the WDT clock source is the instruction clock.
- The WDT function will be disabled if the WDT clock source is the 32768Hz in Idle mode.
- The WDT will still function if the WDT clock source is the WDT OSC.
- If the WDT function is still enabled, the WDT counter and WDT prescaler will be cleared and recounted again.
- The contents of the on chip RAM and registers remain unchanged.
- All the I/O ports maintain their original status.
- The flag PDF is set and the flag TO is cleared by hardware.

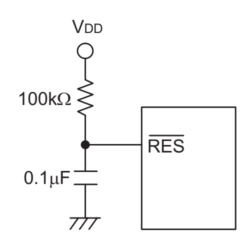
Reset

There are three ways in which a reset can occur.

- Power on reset.
- A low pulse onto $\overline{RES}$ pin.
- WDT time-out.

After these reset conditions, the Program Counter and Stack Pointer will be cleared to 0.

To guarantee that the system oscillator is started and stabilized, the SST (System Start-up Timer) provides an extra-delay of 1024 system clock pulses when the system is reset or awakes from the Sleep or Idle operation mode.

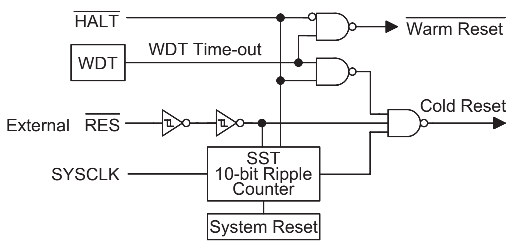


Reset Circuit

By examining the processor status flags PDF and TO, the software program can distinguish between the different "chip resets".

TO	PDF	Reset Condition
0	0	Power on reset
u	u	External reset during Normal mode or Green mode
0	1	External reset during Sleep mode or Idle mode
1	u	WDT time-out during Normal mode or Green mode
1	1	WDT time-out during Sleep mode or Idle mode

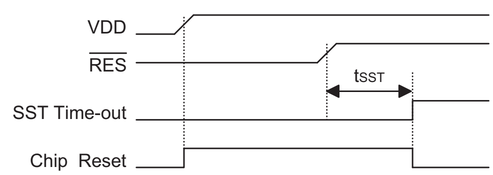
Note: "u" means "unchanged"



Reset Configuration

The functional units chip reset status are shown below:

Program Counter	000H
Interrupt	Disabled
Prescaler	Cleared
WDT	Cleared After a master reset, WDT begins counting. (If WDT function is enabled by mask option)
Timer/Event Counter 0/1	Off
Input/output Port	Input mode
Stack Pointer	Points to the top of the stack



Reset Timing Chart

When the reset conditions occurred, some registers may be changed or unchanged. (HT95C400/40P)

Register	Addr.	Reset Conditions				
		Power On	RES Pin	RES Pin (Sleep/Idle)	WDT	WDT (Sleep/Idle)
IAR0	00H	xxxx xxxx	uuuu uuuu	uuuu uuuu	uuuu uuuu	uuuu uuuu
MP0	01H	xxxx xxxx	uuuu uuuu	uuuu uuuu	uuuu uuuu	uuuu uuuu
IAR1	02H	xxxx xxxx	uuuu uuuu	uuuu uuuu	uuuu uuuu	uuuu uuuu
MP1	03H	xxxx xxxx	uuuu uuuu	uuuu uuuu	uuuu uuuu	uuuu uuuu
BP	04H	---0 0000	---0 0000	---0 0000	---0 0000	---u uuuu
ACC	05H	xxxx xxxx	uuuu uuuu	uuuu uuuu	uuuu uuuu	uuuu uuuu
PCL	06H	0000H	0000H	0000H	0000H	0000H
TBLP	07H	xxxx xxxx	uuuu uuuu	uuuu uuuu	uuuu uuuu	uuuu uuuu
TBLH	08H	xxxx xxxx	uuuu uuuu	uuuu uuuu	uuuu uuuu	uuuu uuuu
WDTS	09H	0000 0111	0000 0111	0000 0111	0000 0111	uuuu uuuu
STATUS	0AH	--00 xxxx	--uu uuuu	--01 uuuu	--1u uuuu	--11 uuuu
INTC0	0BH	-000 0000	-000 0000	-000 0000	-000 0000	uuuu uuuu
TMR0H	0CH	xxxx xxxx	xxxx xxxx	xxxx xxxx	xxxx xxxx	uuuu uuuu
TMR0L	0DH	xxxx xxxx	xxxx xxxx	xxxx xxxx	xxxx xxxx	uuuu uuuu
TMR0C	0EH	00-0 1---	00-0 1---	00-0 1---	00-0 1---	uu-u u---
TMR1H	0FH	xxxx xxxx	xxxx xxxx	xxxx xxxx	xxxx xxxx	uuuu uuuu
TMR1L	10H	xxxx xxxx	xxxx xxxx	xxxx xxxx	xxxx xxxx	uuuu uuuu
TMR1C	11H	00-0 1---	00-0 1---	00-0 1---	00-0 1---	uu-u u---

Register	Addr.	Reset Conditions				
		Power On	$\overline{RES}$ Pin	$\overline{RES}$ Pin (Sleep/Idle)	WDT	WDT (Sleep/Idle)
PA	12H	1111 1111	1111 1111	1111 1111	1111 1111	uuuu uuuu
PAC	13H	1111 1111	1111 1111	1111 1111	1111 1111	uuuu uuuu
PB	14H	1111 1111	1111 1111	1111 1111	1111 1111	uuuu uuuu
PBC	15H	1111 1111	1111 1111	1111 1111	1111 1111	uuuu uuuu
DialerIO	16H	111x xxxx	111x xxxx	111x xxxx	111x xxxx	uuuu uuuu
PD	18H	1111 1111	1111 1111	1111 1111	1111 1111	uuuu uuuu
PDC	19H	1111 1111	1111 1111	1111 1111	1111 1111	uuuu uuuu
PE	1AH	---- 1111	---- 1111	---- 1111	---- 1111	---- uuuu
PEC	1BH	---- 1111	---- 1111	---- 1111	---- 1111	---- uuuu
INTC1	1EH	-000 -000	-000 -000	-000 -000	-000 -000	-uuu -uuu
TBHP	1FH	--xx xxx	--uu uuuu	--uu uuuu	--uu uuuu	--uu uuuu
DTMFC	20H	---- -0-1	---- -0-1	---- -0-1	---- -0-1	---- -u-u
DTMFD	21H	0000 0000	0000 0000	0000 0000	0000 0000	uuuu uuuu
LINE	22H	0--- ----	u--- ----	u--- ----	u--- ----	u--- ----
RTCC	24H	0-0- ----	u-u- ----	u-u- ----	u-u- ----	u-u- ----
MODE	26H	000- ----	00u- ----	000- ----	00u- ----	000- ----
LCDIO	28H	000- ----	uuu- ----	uuu- ----	uuu- ----	uuu- ----
FSKC	29H	--11 11-1	--11 11-1	--11 11-1	--11 11-1	--uu uu-u
FSKS	2AH	-x0- 1100	-x0- 1100	-x0- 1100	-x0- 1100	-xu- uuuu
FSKD	2BH	0000 0000	0000 0000	0000 0000	0000 0000	uuuu uuuu
LCDC	2DH	0000 -000	uuuu -uuu	uuuu -uuu	uuuu -uuu	uuuu -uuu
PFDC	2EH	0000 ----	0000 ----	0000 ----	0000 ----	uuuu ----
PFDD	2FH	0000 0000	0000 0000	0000 0000	0000 0000	uuuu uuuu
PF	34H	1111 1111	1111 1111	1111 1111	1111 1111	uuuu uuuu
PFC	35H	1111 1111	1111 1111	1111 1111	1111 1111	uuuu uuuu
PG	36H	---- 1111	---- 1111	---- 1111	---- 1111	---- uuuu
PGC	37H	---- 1111	---- 1111	---- 1111	---- 1111	---- uuuu
RAM (Data & LCD)		x	u	u	u	u

Note: "u" means "unchanged"
"x" means "unknown"
"-" means "unused"

Timer/Event Counter

Two timer/event counters (TMR0, TMR1) are implemented in the telephone controller series. The Timer/Event Counter 0 and Timer/Event Counter 1 contain 16-bits programmable count-up counter and the clock may come from an external or internal source. For TMR0, the internal source is the instruction clock (system clock/4). For TMR1, the internal source is 32768Hz.

Using the 32768Hz clock or instruction clock, there is only one reference time-base. The external clock input allows the user to count external events, measure time intervals or pulse width, or generate an accurate time base.

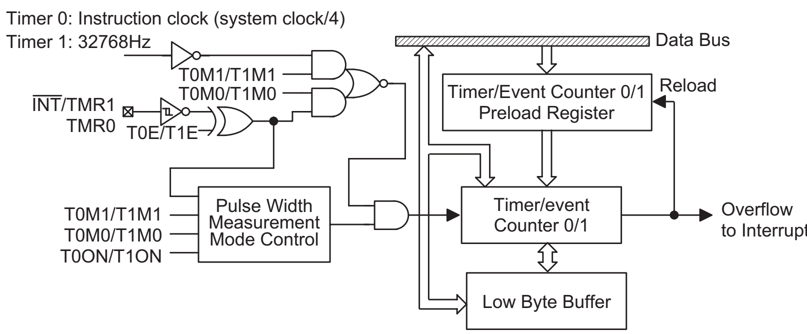
There are 3 registers related to the Timer/Event Counter 0; TMR0H, TMR0L and TMR0C. Writing TMR0L only writes the data into a low byte buffer, but writing TMR0H simultaneously writes the data along with the contents of the low byte buffer into the Timer/Event Counter 0 preload register (16-bit). The Timer/Event Counter 0 preload register is changed by writing TMR0H operations. Writing TMR0L will keep the Timer/Event Counter 0 preload register unchanged.

Reading TMR0H latches the TMR0L into the low byte buffer to avoid a false timing problem. Reading TMR0L returns the contents of the low byte buffer. In other words, the low byte of the Timer/Event Counter 0 can not be read directly. It must read the TMR0H first to make the low byte contents of Timer/Event Counter 0 be latched into the buffer.

There are 3 registers related to the Timer/Event Counter 1; TMR1H, TMR1L and TMR1C. The Timer/Event Counter 1 operates in the same manner as the Timer/Event Counter 0.

The TMR0C is the Timer/Event Counter 0 control register, which defines the Timer/Event Counter 0 options. The Timer/Event Counter 1 has the same options as the Timer/Event Counter 0 and is defined by TMR1C. The timer/event counter control registers define the operating mode, counting enable or disable and active edge.

The T0M0/T1M0, T0M1/T1M1 bits define the operating mode. The event count mode is used to count external events, which means the clock source comes from an external (TMR0 or $\overline{INT}$/TMR1) pin. The timer mode functions as a normal timer with the clock source coming from instruction clock (TMR0) or 32768Hz (TMR1). The pulse width measurement mode can be used to count the high or low level duration of the external signal (TMR0 or $\overline{INT}$/TMR1). The counting is based on the 32768Hz clock for TMR1 or instruction clock for TMR0.



Timer/Event Counter 0/1

Bit No.	Label	R/W	Function
0~2	—	RO	Unused bit, read as ″0″
3	T0E/T1E	RW	To define the TMR0/TMR1 active edge of timer For event count or Timer mode (0=active on low to high; 1=active on high to low) For pulse width measurement mode (0=measures low pulse width; 1=measures high pulse width)
4	T0ON/T1ON	RW	To enable/disable timer counting (0=disabled; 1=enabled)
5	—	RO	Unused bit, read as ″0″
6 7	T0M0/T1M0 T0M1/T1M1	RW	To define the operating mode Bit 7, 6=01, Event count mode (external clock) Bit 7, 6=10, Timer mode Bit 7, 6=11, Pulse width measurement mode Bit 7, 6=00, Unused

TMR0C (0EH)/TMR1C (11H) Register

Register	Bit No.	R/W	Function
TMR0H (0CH)	0~7	RW	Timer/Event Counter 0 higher-order byte register
TMR0L (0DH)	0~7	RW	Timer/Event Counter 0 lower-order byte register
TMR1H (0FH)	0~7	RW	Timer/Event Counter 1 higher-order byte register
TMR1L (10H)	0~7	RW	Timer/Event Counter 1 lower-order byte register

In the event count or timer mode, once the timer/event counter starts counting, it will count from the current contents in the timer/event counter to FFFFH. If an overflow occurs, the counter is reloaded from the timer/event counter preload register and generates the corresponding interrupt request flag (T0F/T1F) at the same time.

In pulse width measurement mode with the T0ON/T1ON and T0E/T1E bits equal to 1, once the TMR0/TMR1 pin has received a transient from low to high (or high to low; if the T0E/T1E bit is 0) it will start counting until the TMR0/TMR1 pin returns to the original level and resets the T0ON/T1ON. The measured result will remain in the timer/event counter even if the activated transient occurs again. In other words, only 1 cycle measurement can be done. Until setting the T0ON/T1ON, the cycle measurement will function again as long as it receives further transient pulse. Note that, in this operating mode, the timer/event counter starts counting not according to the logic level but according to the transient edges. In the case of counter overflows, the counter is reloaded from the timer/event counter preload register and continue to measure the width and issues the interrupt request just like the other two modes.

To enable the counting operation, the timer on bit (T0ON/T1ON) should be set to 1. In the pulse width measurement mode, the T0ON/T1ON will be cleared automatically after the measurement cycle is completed. But in the other two modes the T0ON/T1ON can only be reset by instruction. The overflow of the timer/event counter is one of the wake-up sources. No matter what the operation mode is, writing a 0 to ET0I/ET1I can disable the corresponding interrupt service.

In the case of timer/event counter off condition, writing data to the timer/event counter preload register also reloads that data to the timer/event counter. But if the timer/event counter is turned on, data written to the timer/event counter is reserved only in the timer/event counter preload register. The timer/event counter will go on operating until an overflow occurs.

Input/Output Ports

There is a maximum of 40 bidirectional input/output lines in the HT95CXXX family MCU, labeled as PA, PB, PD, PE, PF and PG. All of these I/O ports can be used for input and output operations. For input operation, these ports are non-latching, that is, the inputs must be ready at the T2 rising edge of instruction "MOV A,[m]" (m=12H, 14H, 18H, 1AH, 34H or 36H). For output operation, all the data is latched and remains unchanged until the output latch is rewritten.

Each I/O line has its own control register (PAC, PBC, PDC, PEC, PFC, PGC) to control the input/output configuration. With this control register, CMOS output or Schmitt trigger input can be reconfigured dynamically under software control. To make one I/O line to function as an input line, the corresponding latch of the control register must be written with a ″1″. The pull-high resistance shows itself automatically if the pull-high option is selected. The input source also depends on the control register. If the control register bit is ″1″, the input will read the pad state. If the control register bit is ″0″, the contents of the latches will move to the internal bus. The latter is possible in the ″read-modify-write″ instruction. For output function, CMOS is the only configuration. Each bit of these input/output latches can be set or cleared by ″SET [m].i″ and ″CLR [m].i″ (m=12H, 14H, 18H, 1AH, 34H or 36H) instructions.

Some instructions first input data and then follow the output operations. For example, ″SET [m].i″, ″CLR [m].i″, ″CPL [m]″, ″CPLA [m]″ read the entire port states into the CPU, execute the defined operations (bit-operation), and then write the results back to the latches or the accumulator.

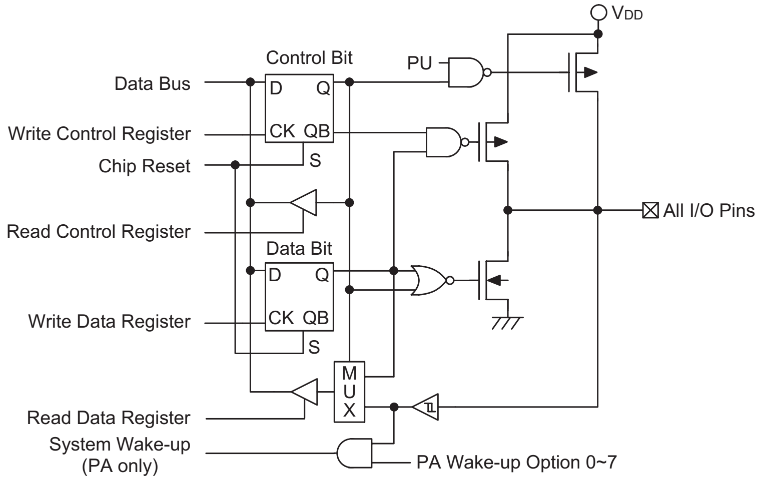
Each line of port A has the capability of waking-up the device. They are selected by mask option per bit.

There is a pull-high option available for all I/O lines. Once the pull-high option of an I/O line is selected, the I/O lines have pull-high resistor. Otherwise, the pull-high resistor is absent. It should be noted that a non-pull-high I/O line operating in input mode may cause a floating state.

I/O port pull-high, wake-up function are selected by mask option

I/O Port	Output	Input		Supported for HT95CXXX		
		Pull-high Resistor	Wake-up Function	400/40P	300/30P	200/20P
PA7~PA0	CMOS	Selected per bit	Selected per bit	√	√	√
PB7~PB0	CMOS	Selected per bit	—	√	√	√
PD7~PD0	CMOS	Selected per nibble	—	√	√	√
PE3~PE0	CMOS	Selected per nibble	—	√	√	√
PF7~PF0	CMOS	Selected per nibble	—	√	—	—
PG3~PG0	CMOS	Selected per nibble	—	√	—	—

Note: ″—″ means unavailable



Input/Output Ports

Some input/output pins can be optioned to LCD outputs by software.

Bit No.	Label	R/W	Value	400/40P	300/30P	200/20P
5	SPE0	RW	0	SEG47~SEG44		—
			1	PE3~PE0		—
7	SPD1	RW	0	SEG43~SEG40		—
			1	PD7~PD4		—
6	SPD0	RW	0	SEG39~SEG36		—
			1	PD3~PD0		—
1	VBIAS	RW	0	COM7~COM0		COM7~COM0
			1	COM7~COM0 are unavailable		PD7~PD0

LCDIO (28H) Register

Bit No.	Label	R/W	Value	400/40P	300/30P	200/20P
1	VBIAS	RW	0	COM7~COM0		COM7~COM0
			1	COM7~COM0 are unavailable		PD7~PD0

LCDC (2DH) Register

When the PD0~PD7 or the PE0~PE3 are not selected, the I/O port control register PDC(19H), PEC(1BH) could be readable/writable and be used as a general user RAM, but this function is not available for register PD (18H) and PE (1AH).

FSK Decoder

The FSK decoder supports three interrupt sources to the peripheral interrupt vector. There are ring detect or line reversal detect, FSK carrier detect and FSK packet data. Write 0 to the control flag, RMSK, CMSK and FMSK will enable these interrupt. When any of these interrupt occurs, its interrupt flag (RDETF, CDETF, FSKF) will be set to 1 by hardware even if the interrupt is disabled. These interrupts will cause a peripheral interrupt if the peripheral interrupt is enabled.When the peripheral interrupt occurs, the interrupt request flag PERF will be set and a subroutine call to location 10H will occur. Returning from the interrupt subroutine, the interrupt flag RDETF, CDETF or FSKF will not be cleared by hardware, the user should clear it by software. If interrupt flag RDETF is not cleared, next ring detect interrupt will be inhibited, other interrupt flags CDETF, FSKF have the same behavior. The power down mode (F_PWDN=1) will terminate all the FSK decoder function, however, the registers FSKC, FSKS and FSKD are accessible at this power down mode.

Bit No.	Label	R/W	Function
0	F_PWDN	RW	FSK decoder power down 1: FSK decoder is at power down mode 0: FSK decoder is at operation mode
1	—	RO	Unused bit, read as ″0″
2	FMSK	RW	FSK packet data interrupt mask 1: Disable FSK packet data interrupt 0: Enable FSK packet data interrupt
3	RMSK	RW	Ring or line reversal detect interrupt mask 1: Disable ring or line reversal detect interrupt 0: Enable ring or line reversal detect interrupt
4	CMSK	RW	Carrier detect interrupt mask 1: Disable carrier detect interrupt 0: Enable carrier detect interrupt
5	FSKSEL	RW	Select FSK packet data source 1: FSK packet data source is DOUTC 0: FSK packet data source is DOUT
6, 7	—	RO	Unused bit, read as ″0″

FSKC (29H) Register

Bit No.	Label	R/W	Function
0	RDETF	RW	Ring or line reversal detect interrupt flag 1: Ring or line reversal detected 0: No ring or line reversal detected This flag is set by hardware and cleared by software.
1	CDETF	RW	FSK carrier detect interrupt flag 1: An FSK carrier signal is detected 0: No valid FSK carrier signal is detected This flag is set by hardware and cleared by software.
2	DOUT	RO	This flag presents the FSK decoder output when the decoder is at operation mode. This data stream includes the alternate 1 and 0 pattern, the marking and the data.
3	DOUTC	RO	This flag present the FSK decoder output like as the DOUT flag but does not include the alternate 1 and 0 pattern.
4	—	RO	Unused bit, read as ″0″
5	FSKF	RW	FSK packet data interrupt flag 1: FSK packet data is ready 0: FSK packet data is not ready This flag is set by hardware and cleared by software.
6	RINGF	RO	This flag presents the ring coming signal. Refer to the following figure.
7	—	RO	Unused bit, read as ″0″

FSKS (2AH) Register

Bit No.	Label	R/W	Function
7~0	—	RO	FSK packet data register

FSKD (2BH) Register

Ring or Line Reversal Detect

When no signal is present on the telephone line, RDET1 will be at GND and $\overline{\text{RTIME}}$ is pulled to VDD by R1. If a line reversal occurs, the RDET1 pin will become high. This causes $\overline{\text{RTIME}}$ and internal signal R_DET to be pulled low. The C1 and R1 ensure that the R_DET signal is low during such a time, so that processor can detect it.

When a ring occurs on the line, internal signal R_DET is permanently low, indicating the envelope of the ring. If the frequency of the ring must be measured, C1 may be removed, $\overline{\text{RTIME}}$ and R_DET inverter follow RDET1.

The flag RDETF will go high when the R_DET signal falling edge is detected. This may cause a peripheral interrupt if RMSK is 0 and the peripheral interrupt is enabled (EPERI=1).

FSK Data Output

The FSK decoder will decode the FSK signal on the TIP and RING line and produce two kinds of data formats, the serial data and the 8-bit packet data. It also provides the FSK carrier detection signal.

To enable the FSK decoder, the F_PWDN should be written as 0. Once the FSK carrier signal is detected, the flag CDETF will be set to 1. This may cause a peripheral interrupt if CMSK is 0 and the peripheral interrupt is enabled.

The serial FSK data is present in two formats: RAW data and COOK data, and could be monitored by the flag DOUT, DOUTC, respectively.

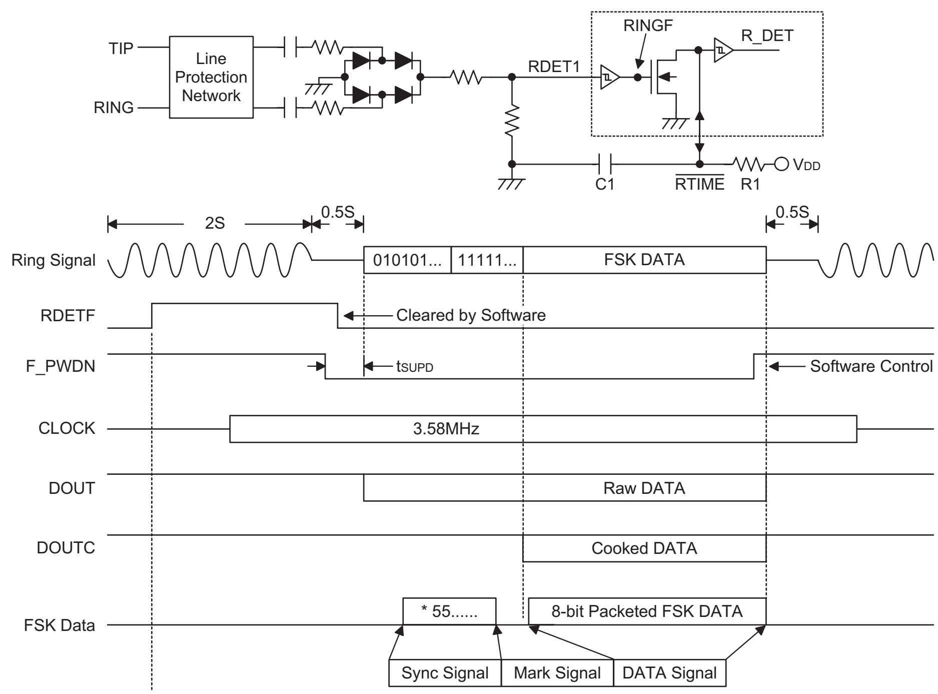
The flag DOUT presents the output of the decoder when the decoder is at operation mode. This data stream includes the alternate 1 and 0 pattern, the marking and the data.

The flag DOUTC presents the output of the decoder when the decoder is at operation mode. This data stream is like the DOUT flag but does not include the alternate 1 and 0 pattern.

If the FSK data is not detected, the DOUT and DOUTC are held high.

Beside the serial data, the decoder also provides FSK packet data. When decoder receives an FSK signal, it will packet 10 bits data to 8 bits data, the first and 10th bits will be discarded. When the 8-bit packet data is valid, it will be stored in the FSK data register FSKD, the FSK packet data interrupt flag FSKF will be set to 1. This may cause a peripheral interrupt if FMSK is 0 and the peripheral interrupt is enabled. The FSK packet source could be DOUT or DOUTC, selected by FSKSEL. Note that the start bit of the 10 packet bit should be 0, so the MARK signal (one of the FSK data signals) will not be packeted.

To detect the carrier signal or decode the serial data or packet 10-bit data to 8-bit data, the operation mode of the controller must be selected in Normal mode (processor running with 3.58MHz). When the operation mode is Green or Sleep, FSK decoder will decode the wrong signal. However, when the operation mode is Green or Sleep mode and the FSK decoder is at power down mode (F_PWDN=1), the ring and line reversal detect is still functional.



Note: "*" If the flag FSKSEL=1, the sync signal data will not be packeted.

DTMF Generator

The DTMF (Dual Tone Multiple-Frequency) signal generator is implemented in the telephone controller. It can generate 16 dual tones and 8 single tones from the DTMF pin. This generator also supports power down, tone on/off function. The DTMF generator clock source is 3.58MHz, before using this function, the system operation mode must be at Normal mode.

The power down mode (D_PWDN=1) will terminate all the DTMF generator function, however, the registers DTMFC and DTMFD are accessible at this power down mode. The duration of DTMF output should be handled by the software. DTMFD register value could be changed as desired, the DTMF pin will output the new dual-tone simultaneously.

Bit No.	Label	R/W	Function
0	D_PWDN		DTMF generator power down 1: DTMF generator is at power down mode. 0: DTMF generator is at operation mode.
1	—	RO	Unused bit, read as "0"
2	TONE	RW	Tone output enable 1: DTMF signal output is enabled. 0: DTMF signal output is disabled.
3	—	RW	Reserved, inhibit using.
4	—	RW	Reserved, inhibit using.
5	—	RO	Unused bit, read as "0"
6	—	RW	Reserved, inhibit using.
7	—	RO	Unused bit, read as "0"

DTMFC (20H) Register

Note: Bit3, 4, 6 of DTMFC are reserved, always keep the initial value.

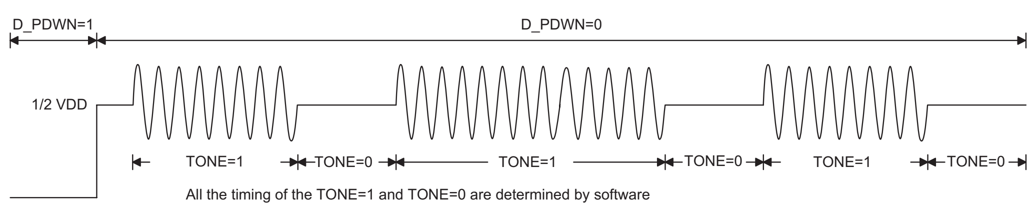
Bit No.	Label	R/W	Function
3~0	TC4~TC1	RW	To set high group frequency
7~4	TR4~TR1	RW	To set low group frequency

DTMFD (21H) Register

Note: Bit3, 4, 6 of DTMFC are reserved, always keep the initial value.

The DTMF pin output is controlled by the combination of the D_PWDN, TONE, TR~TC value.

Control Register Bits			DTMF Pin Output Status
D_PWDN	TONE	TR4~TR1/TC4~TC1	
1	x	x	0
0	0	x	1/2 VDD
0	1	0	1/2 VDD
0	1	Any valid value	16 dual tones or 8 signal tones, bias with 1/2 VDD



DTMF Output

Tone frequency

Output Frequency (Hz)		% Error
Specified	Actual	
697	699	+0.29%
770	766	–0.52%
852	847	–0.59%
941	948	+0.74%
1209	1215	+0.50%
1336	1332	–0.30%
1477	1472	–0.34%

% Error does not contain the crystal frequency shift

DTMF frequency selection table: register DTMFD[21H]

Low Group				High Group				DTMF Output		DTMF Code
TR4	TR3	TR2	TR1	TC4	TC3	TC2	TC1	Low	High	
0	0	0	1	0	0	0	1	697	1209	1
0	0	0	1	0	0	1	0	697	1336	2
0	0	0	1	0	1	0	0	697	1477	3
0	0	0	1	1	0	0	0	697	1633	A
0	0	1	0	0	0	0	1	770	1209	4
0	0	1	0	0	0	1	0	770	1336	5
0	0	1	0	0	1	0	0	770	1477	6
0	0	1	0	1	0	0	0	770	1633	B
0	1	0	0	0	0	0	1	852	1209	7
0	1	0	0	0	0	1	0	852	1336	8
0	1	0	0	0	1	0	0	852	1477	9
0	1	0	0	1	0	0	0	852	1633	C
1	0	0	0	0	0	0	1	941	1209	*
1	0	0	0	0	0	1	0	941	1336	0
1	0	0	0	0	1	0	0	941	1477	#
1	0	0	0	1	0	0	0	941	1633	D
Single tone for testing only										
0	0	0	1	0	0	0	0	697		
0	0	1	0	0	0	0	0	770		
0	1	0	0	0	0	0	0	852		
1	0	0	0	0	0	0	0	941		
0	0	0	0	0	0	0	1		1209	
0	0	0	0	0	0	1	0		1336	
0	0	0	0	0	1	0	0		1477	
0	0	0	0	1	0	0	0		1633	

Writing other values to TR4~TR1, TC4~TC1 may generate an unpredictable tone.

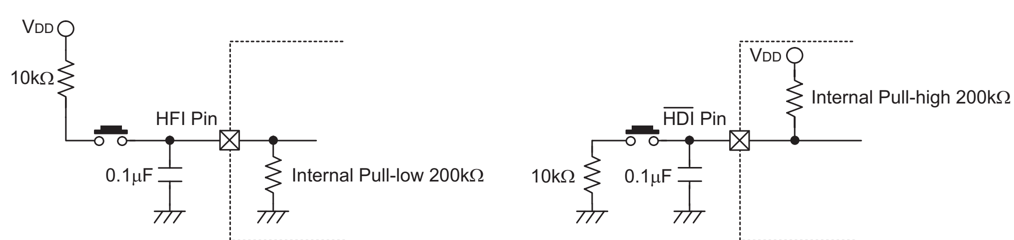
Dialer I/O Function

A special dialer I/O circuit is built into the telephone controller for dialing application. These specially designed I/O cells allows the controller to work under a low voltage condition that usually happens when the subscriber's loop is long.

Dialer I/O pin function:

Name	I/O	Description
$\overline{XMUTE}$	NMOS Output	$\overline{XMUTE}$ pin output is controlled by software. This is an NMOS open drain structure pulled to VSS during dialing signal transmission. Otherwise, it is an open circuit. $\overline{XMUTE}$ is used to mute the speech circuit when transmitting the dialer signal.
$\overline{DNPO}$	NMOS Output	$\overline{DNPO}$ pin is an NMOS output, usually by means of software to make/break the line. This pin is only controlled by software.
$\overline{PO}$	CMOS Output	This pin is controlled by the $\overline{HKS}$, HFI and $\overline{HDI}$ pins. When $\overline{PO}$ pin is high, the telephone line is make. When $\overline{PO}$ pin is low, the telephone line is break.
$\overline{HKS}$	Schmitt Trigger Input	This pin controls the $\overline{PO}$ pin directly. This pin is used to monitor the status of the hook-switch and its combination with HFI/$\overline{HDI}$ can control the $\overline{PO}$ pin output to make or break the line. A rising edge to $\overline{HKS}$ pin will cause the dialer I/O to be on-hook status and generate an interrupt, its vector is 18H. A falling edge to $\overline{HKS}$ pin will cause the dialer I/O to be off-hook status and clear HFO and HDO flags to 0. This falling edge will also generate an interrupt, its vector is 18H.
HDO	CMOS Output	This pin is controlled directly by $\overline{HDI}$, $\overline{HKS}$ and HFI pin. When HDO pin is high, the hold-line function is enabled and $\overline{PO}$ outputs a high signal to make the line.
$\overline{HDI}$	Schmitt Trigger Input	A low pulse to $\overline{HDI}$ pin (hold-line function request) will clear HFO to 0 and toggle HDO and generates an interrupt, its vector is 18H. This pin controls the HFO and HDO pins directly. This pin is functional only when the line is made, that is, off-hook or hand-free ($\overline{PO}$ output high signal).
HFO	CMOS Output	This pin is controlled directly by HFI, $\overline{HDI}$ and $\overline{HKS}$ pins. When HFO pin is high, the hand-free function is enabled and $\overline{PO}$ outputs a high signal to make the line.
HFI	Schmitt Trigger Input	A high pulse to HFI pin (hand-free function request) will clear HDO to 0 and toggle HFO and generates an interrupt, its vector is 18H. This pin controls the $\overline{PO}$, HFO and HDO pins directly.

The following are the recommended circuit for HFI and $\overline{HDI}$ pins.



Phone controller also supports the dialer I/O flag to monitor the dialer status.

Bit No.	Label	R/W	Function
0	HFI	RO	1: The HFI pin level is 1. 0: The HFI pin level is 0.
1	HFO	RO	1: The HFO pin level is 1. 0: The HFO pin level is 0.
2	$\overline{HDI}$	RO	1: The $\overline{HDI}$ pin level is 1. 0: The $\overline{HDI}$ pin level is 0.
3	HDO	RO	1: The HDO pin level is 1. 0: The HDO pin level is 0.
4	$\overline{HKS}$	RO	1: The $\overline{HKS}$ pin level is 1. 0: The $\overline{HKS}$ pin level is 0.
5	SPO	RW	1: The $\overline{PO}$ pin is controlled by the combination of the $\overline{HKS}$, HFI and $\overline{HDI}$ pin. 0: The $\overline{PO}$ pin level is set to 0 by software.
6	SDNPO	RW	1: The $\overline{DNPO}$ pin level is set to floating by software. 0: The $\overline{DNPO}$ pin level is set to 0 by software.
7	$\overline{XMUTE}$	RW	1: The $\overline{XMUTE}$ pin is set to floating by software. 0: The $\overline{XMUTE}$ pin is set to 0 by software.

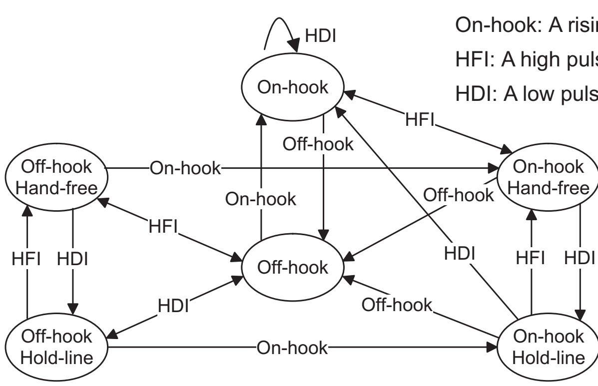
DIALERIO (16H) Register

The SPO flag is special designed to control the $\overline{PO}$. When the flag SPO is set to 1, the $\overline{PO}$ pin is controlled by the combination of the $\overline{HKS}$ pin, HFI pin and $\overline{HDI}$ pin. The $\overline{PO}$ pin will always be 0 if the flag SPO=0.

The relation between the Dialer I/O function (SPO=1)

Dialer Function	Dialer I/O Pin (Flag) Status			Result		
	$\overline{HKS}$	HFO	HDO	$\overline{PO}$	$\overline{DNPO}$	Telephone Line
On-hook	1	0	0	0	floating	break
On-hook & Hand-free	1	1	0	1	floating	make
On-hook & Hold-line	1	0	1	1	floating	make
Off-hook	0	0	0	1	floating	make
Off-hook & Hand-free	0	1	0	1	floating	make
Off-hook & Hold-line	0	0	1	1	floating	make

The following describes the dialer I/O function status machine figure (Available on Normal mode, Green mode or Sleep mode):



Off-hook: A falling edge to $\overline{HKS}$ pin

On-hook: A rising edge to $\overline{HKS}$ pin

HFI: A high pulse to HFI pin (Hand-free request is generated.)

HDI: A low pulse to $\overline{HDI}$ pin (Hold-line request is generated.)

Note: 1. If the dialer status is on-hook and hold-line, the falling edge transition onto $\overline{HDI}$ pin will not generate the dialer I/O interrupt.

2. Dialer I/O function is not available in Idle mode

Line Control Function

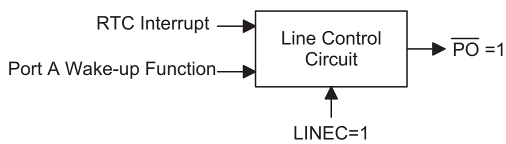
Bit No.	Label	R/W	Function
6~0	—	RO	Unused bit, read as "0"
7	LINEC	RW	1: Enable the line control function 0: Disable the line control function

LINE (22H) Register

The line control function is enabled by the flag LINEC

Conditions		Source to Enable Line Control Function
LINEC	Operation Mode	
1	Normal or Green mode	RTC time out interrupt
1	Sleep mode	Port A wake-up RTC time out interrupt
1	Idle mode	Port A wake-up

When the line control source is activated, the $\overline{PO}$ pin will be set to high signal. Clearing LINEC to 0 will terminate the line control function and drive $\overline{PO}$ pin outputs low signal.



RTC Function

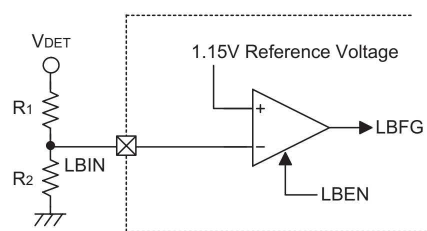
Bit No.	Label	R/W	Function
6, 4~0	—	RO	Unused bit, read as "0"
5	RTCEN	RW	1: Enable RTC function 0: Disable RTC function
7	RTCTO	RW	1: RTC time-out occurs 0: RTC time-out not occurs

RTCC (24H) Register

The real time clock (RTC) is used to supply a regular internal interrupt. Its time-out period is 1000ms. If the RTC time-out occurs, the interrupt request flag RTCF and the RTCTO flag will be set to 1. The interrupt vector for the RTC is 14H. When the interrupt subroutine is serviced, the interrupt request flag (RTCF) will be cleared to 0, but the flag RTCTO remain in its original value. If the RTCTO flag is not cleared, next RTC time-out interrupt will occur.

Low Battery Detection

The phone controller provides a circuit that detects the LBIN pin voltage level. To enable this detection function, the LBEN should be written as 1. Once this function is enabled, the detection circuit needs 50μs to be stable. After that, the user could read the result from LBFG. The low battery detect function will consume power. For power saving, write 0 to LBEN if the low battery detection function is unnecessary.



The battery low threshold is determined by external R1 and R2 resistors.

$$1.15=\frac{V_{DET}xR2}{R1+R2} \rightarrow V_{DET}=\frac{1.15x(R1+R2)}{R2}$$

If we want to detect V_{DET}=2.4V

$$\text{then } 2.4V=\frac{1.15x(R1+R2)}{R2} \rightarrow R1=1.087R2$$

LCD Driver

The LCD driver can directly drive an LCD panel with 1/8 duty and 1/4 bias or with 1/16 duty and 1/5 bias, this function is selected by the flag VBIAS. The frame of this LCD driver may select a 64Hz or 128Hz by flag FRAME.

LCD driver uses the voltage of the VLCD pin as the power source. To adjust the view angle, the programmer can select the real LCD power by the flags VCON0 and VCON1. The flag LCDON is used to turn On/Off the LCD display. Note that the VLCD voltage must equal or be less than VDD.

Segment/Common to I/O Selection

For the flexible purpose, some of the LCD COMMON and SEGMENT pins are shared with the input/output port.

Both of the HT95C400/40P and HT95C300/30P provide 12 pins to be selected to SEGMENT output pins or I/O pins. HT95C200/20P provides 8 pins to be selected for COMMON output pins or I/O pins.

All of the HT95C400/40P, HT95C300/30P and HT95C200/20P provide the LCD COMMON output pins for 8 COMMON or 16 COMMON. The description of the relation between segment pins, common pins and I/O pins are shown on the below.

Bit No.	Label	R/W	Function
0	FRAME	RW	LCD frame selection 0: LCD frame is 64Hz 1: LCD frame is 128Hz
1	VBIAS	RW	LCD BIAS selection 0: select 1/16 duty and 1/5 bias, COM15~COM0 are available 1: select 1/8 duty and 1/4 bias, only COM15~COM8 are available When the 8 COM is selected HT95C400/40P: COM7~COM0 will be optioned to unused pins HT95C300/30P: COM7~COM0 will be optioned to unused pins HT95C200/20P: COM7~COM0 are disabled, PD7~PD0 are available
2	LBEN	RW	Low battery detection switch 0: disable the low battery detection 1: enable the low battery detection
3	—	RO	Unused bit, read as "0"
4	LBFG	RO	Low battery detection flag 1: LBIN pin voltage is less than 1.15V 0: LBIN pin voltage is not less than 1.15V
5 6	VCON0 VCON1	RW	LCD contrast adjusting Bit6,5=00: LCD voltage supply is 0.66×VLCD Bit6,5=10: LCD voltage supply is 0.82×VLCD Bit6,5=01: LCD voltage supply is 0.93×VLCD Bit6,5=11: LCD voltage supply is 1.00×VLCD
7	LCDON	RW	1: Turn on the LCD display 0: Turn off the LCD display

LCDC (2DH) Register

Bit No.	Label	R/W	Function
0~4	—	RO	Unused bit, read as "0"
5	SPE0	RW	Supported for HT95C400/40P, HT95C300/30P Bit value is 0: HT95C400/40P: SEG47~SEG44 output are available HT95C300/30P: SEG47~SEG44 output are available Bit value is 1: HT95C400/40P: PE3~PE0 output are available HT95C300/30P: PE3~PE0 output are available
6	SPD0	RW	Supported for HT95C400/40P, HT95C300/30P Bit value is 0: SEG39~SEG36 output are available Bit value is 1: PD3~PD0 output are available
7	SPD1	RW	Supported for HT95C400/40P, HT95C300/30P Bit value is 0: SEG43~SEG40 output are available Bit value is 1: PD7~PD4 output are available

LCDIO (28H) Register

LCD Display Memory

The phone controller provides an area on embedded data memory for LCD display. The LCD display memory are located at bank 1BH and can be read and written to, only by indirect addressing mode using MP1. When data is written into the display data area it is automatically read by the LCD driver which then generates the corresponding LCD driving signals, to turn the display On or Off, a ″1″ or ″0″ is written to the corresponding bit of the display memory, respectively. All of the LCD display memories are with random values after the power on reset and unchanged after other reset conditions.

COM7 to COM0 for HT95C400/40P, HT95C300/30P									
Address	**Register Name**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
40H	SEG0	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
41H	SEG1	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
—	—	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
6EH	SEG46	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
6FH	SEG47	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
COM15 to COM8 for HT95C400/40P, HT95C300/30P									
Address	**Register Name**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
70H	SEG0	COM15	COM14	COM13	COM12	COM11	COM10	COM9	COM8
71H	SEG1	COM15	COM14	COM13	COM12	COM11	COM10	COM9	COM8
—	—	COM15	COM14	COM13	COM12	COM11	COM10	COM9	COM8
9EH	SEG46	COM15	COM14	COM13	COM12	COM11	COM10	COM9	COM8
9FH	SEG47	COM15	COM14	COM13	COM12	COM11	COM10	COM9	COM8

Note: When VBIAS bit set to 1 for 8 COM operation (48×8), the LCD RAM only map to (70H~9FH).

COM7 to COM0 for HT95C200/20P									
Address	**Register Name**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
40H	SEG0	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
41H	SEG1	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
—	—	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
56H	SEG22	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
57H	SEG23	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
COM15 to COM8 for HT95C200/20P									
Address	**Register Name**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
70H	SEG0	COM15	COM14	COM13	COM12	COM11	COM10	COM9	COM8
71H	SEG1	COM15	COM14	COM13	COM12	COM11	COM10	COM9	COM8
—	—	COM15	COM14	COM13	COM12	COM11	COM10	COM9	COM8
86H	SEG22	COM15	COM14	COM13	COM12	COM11	COM10	COM9	COM8
87H	SEG23	COM15	COM14	COM13	COM12	COM11	COM10	COM9	COM8

Note: When VBIAS bit is set to 1 for 8 COM operation (24×8), the LCD RAM only map to (70H~87H).

PFD Generator

Bit No.	Label	R/W	Function
3~0	—	RO	Unused bit, read as "0"
4	PFDEN	RW	1: Enable PFD output 0: Disable PFD output, the MUSIC pin output low level.
5 6	PRES0 PRES1	RW	Bit6, 5=00: Prescaler output= PFD frequency source/1 Bit6, 5=01: Prescaler output= PFD frequency source/2 Bit6, 5=10: Prescaler output= PFD frequency source/4 Bit6, 5=11: Prescaler output= PFD frequency source/8
7	FPFD	RW	1: The PFD frequency source is 3.58MHz/4 0: The PFD frequency source is 32768Hz

PFDC (2EH) Register

Bit No.	Label	R/W	Function
7~0	—	RW	PFD data register

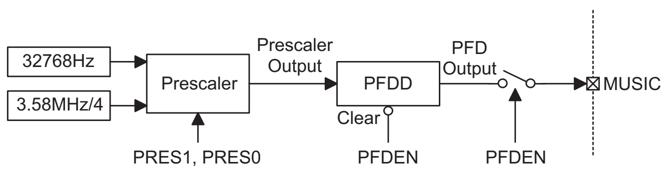
PFDD (2FH) Register

The PFD (programmable frequency divider) is implemented in the phone controller. It is composed of two portions: a prescaler and a general counter.

The prescaler is controlled by the register bits, PRES0 and PRES1. The general counter is programmed by an 8-bit register PFDD.

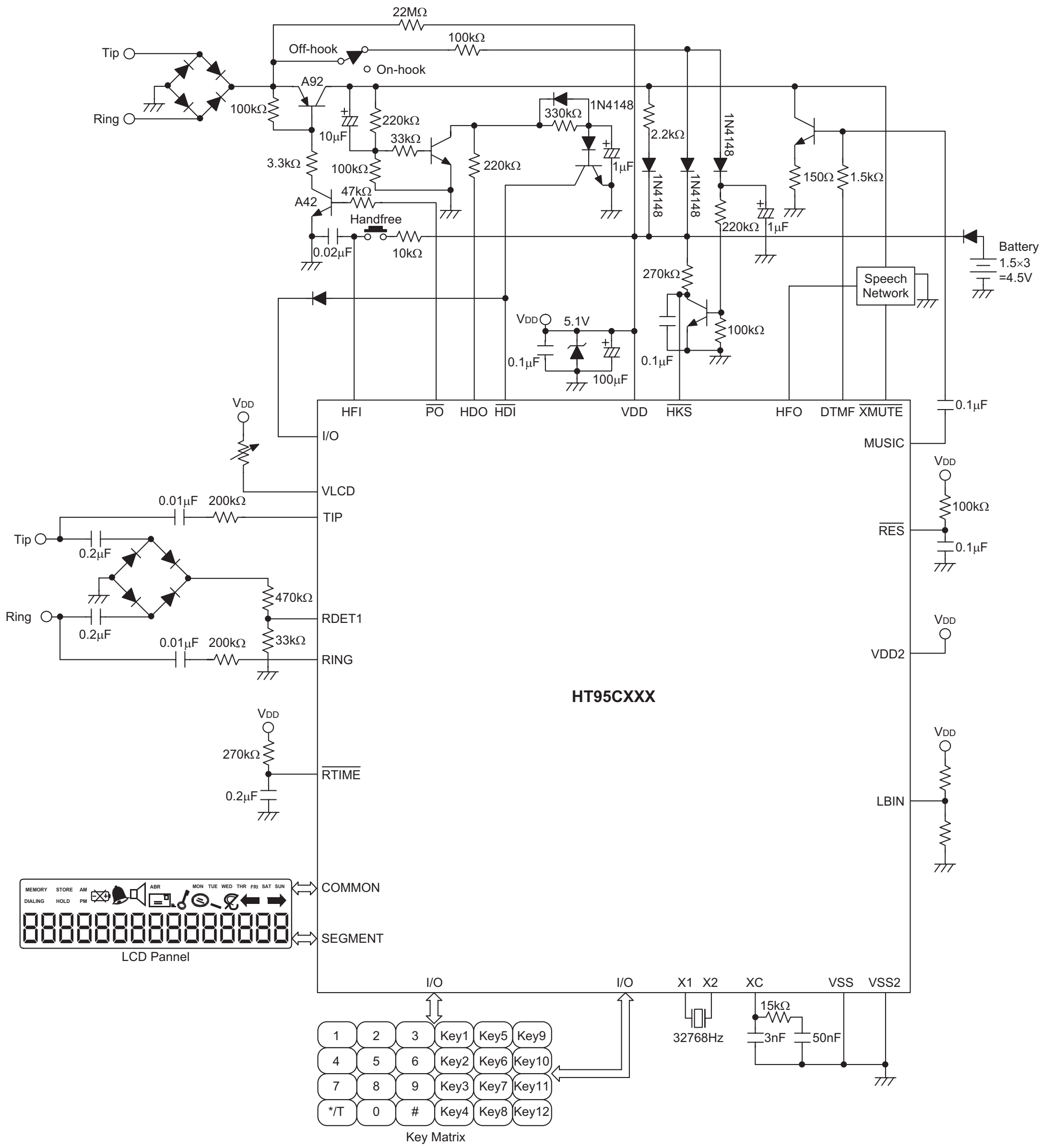
The source for this generator can be selected from 3.58MHz/4 or 32768Hz. To enable the PFD output, write 1 to the PFDEN bit.

The PFDD is inhibited to write while the PFD is disabled. To modify the PFDD contents, the PFD must be enabled. When the generator is disabled, the PFDD is cleared by hardware.



PFD output frequency= $\frac{\text{Prescaler output}}{2x(N+1)}$, where N=the value of the PFDD

Application Circuits



Note: Some floating input pins ($\overline{\text{INT}}$/TMR1, TMR0, etc.) are not shown in this circuit.

Instruction Set Summary

Mnemonic	Description	Instruction Cycle	Flag Affected
Arithmetic			
ADD A,[m]	Add data memory to ACC	1	Z,C,AC,OV
ADDM A,[m]	Add ACC to data memory	1(1)	Z,C,AC,OV
ADD A,x	Add immediate data to ACC	1	Z,C,AC,OV
ADC A,[m]	Add data memory to ACC with carry	1	Z,C,AC,OV
ADCM A,[m]	Add ACC to data memory with carry	1(1)	Z,C,AC,OV
SUB A,x	Subtract immediate data from ACC	1	Z,C,AC,OV
SUB A,[m]	Subtract data memory from ACC	1	Z,C,AC,OV
SUBM A,[m]	Subtract data memory from ACC with result in data memory	1(1)	Z,C,AC,OV
SBC A,[m]	Subtract data memory from ACC with carry	1	Z,C,AC,OV
SBCM A,[m]	Subtract data memory from ACC with carry and result in data memory	1(1)	Z,C,AC,OV
DAA [m]	Decimal adjust ACC for addition with result in data memory	1(1)	C
Logic Operation			
AND A,[m]	AND data memory to ACC	1	Z
OR A,[m]	OR data memory to ACC	1	Z
XOR A,[m]	Exclusive-OR data memory to ACC	1	Z
ANDM A,[m]	AND ACC to data memory	1(1)	Z
ORM A,[m]	OR ACC to data memory	1(1)	Z
XORM A,[m]	Exclusive-OR ACC to data memory	1(1)	Z
AND A,x	AND immediate data to ACC	1	Z
OR A,x	OR immediate data to ACC	1	Z
XOR A,x	Exclusive-OR immediate data to ACC	1	Z
CPL [m]	Complement data memory	1(1)	Z
CPLA [m]	Complement data memory with result in ACC	1	Z
Increment & Decrement			
INCA [m]	Increment data memory with result in ACC	1	Z
INC [m]	Increment data memory	1(1)	Z
DECA [m]	Decrement data memory with result in ACC	1	Z
DEC [m]	Decrement data memory	1(1)	Z
Rotate			
RRA [m]	Rotate data memory right with result in ACC	1	None
RR [m]	Rotate data memory right	1(1)	None
RRCA [m]	Rotate data memory right through carry with result in ACC	1	C
RRC [m]	Rotate data memory right through carry	1(1)	C
RLA [m]	Rotate data memory left with result in ACC	1	None
RL [m]	Rotate data memory left	1(1)	None
RLCA [m]	Rotate data memory left through carry with result in ACC	1	C
RLC [m]	Rotate data memory left through carry	1(1)	C
Data Move			
MOV A,[m]	Move data memory to ACC	1	None
MOV [m],A	Move ACC to data memory	1(1)	None
MOV A,x	Move immediate data to ACC	1	None
Bit Operation			
CLR [m].i	Clear bit of data memory	1(1)	None
SET [m].i	Set bit of data memory	1(1)	None

Mnemonic	Description	Instruction Cycle	Flag Affected
Branch			
JMP addr	Jump unconditionally	2	None
SZ [m]	Skip if data memory is zero	1(2)	None
SZA [m]	Skip if data memory is zero with data movement to ACC	1(2)	None
SZ [m].i	Skip if bit i of data memory is zero	1(2)	None
SNZ [m].i	Skip if bit i of data memory is not zero	1(2)	None
SIZ [m]	Skip if increment data memory is zero	1(3)	None
SDZ [m]	Skip if decrement data memory is zero	1(3)	None
SIZA [m]	Skip if increment data memory is zero with result in ACC	1(2)	None
SDZA [m]	Skip if decrement data memory is zero with result in ACC	1(2)	None
CALL addr	Subroutine call	2	None
RET	Return from subroutine	2	None
RET A,x	Return from subroutine and load immediate data to ACC	2	None
RETI	Return from interrupt	2	None
Table Read			
TABRDC [m]	Read ROM code (current page) to data memory and TBLH	2(1)	None
TABRDL [m]	Read ROM code (last page) to data memory and TBLH	2(1)	None
Miscellaneous			
NOP	No operation	1	None
CLR [m]	Clear data memory	1(1)	None
SET [m]	Set data memory	1(1)	None
CLR WDT	Clear Watchdog Timer	1	TO,PDF
CLR WDT1	Pre-clear Watchdog Timer	1	TO(4),PDF(4)
CLR WDT2	Pre-clear Watchdog Timer	1	TO(4),PDF(4)
SWAP [m]	Swap nibbles of data memory	1(1)	None
SWAPA [m]	Swap nibbles of data memory with result in ACC	1	None
HALT	Enter power down mode	1	TO,PDF

Note: x: Immediate data

m: Data memory address

A: Accumulator

i: 0~7 number of bits

addr: Program memory address

√: Flag is affected

–: Flag is not affected

(1): If a loading to the PCL register occurs, the execution cycle of instructions will be delayed for one more cycle (four system clocks).

(2): If a skipping to the next instruction occurs, the execution cycle of instructions will be delayed for one more cycle (four system clocks). Otherwise the original instruction cycle is unchanged.

(3): (1) and (2)

(4): The flags may be affected by the execution status. If the Watchdog Timer is cleared by executing the ″CLR WDT1″ or ″CLR WDT2″ instruction, the TO and PDF are cleared.
Otherwise the TO and PDF flags remain unchanged.

Instruction Definition

ADC A,[m] Add data memory and carry to the accumulator

Description: The contents of the specified data memory, accumulator and the carry flag are added simultaneously, leaving the result in the accumulator.

Operation: ACC ← ACC+[m]+C

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	√	√	√	√

ADCM A,[m] Add the accumulator and carry to data memory

Description: The contents of the specified data memory, accumulator and the carry flag are added simultaneously, leaving the result in the specified data memory.

Operation: [m] ← ACC+[m]+C

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	√	√	√	√

ADD A,[m] Add data memory to the accumulator

Description: The contents of the specified data memory and the accumulator are added. The result is stored in the accumulator.

Operation: ACC ← ACC+[m]

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	√	√	√	√

ADD A,x Add immediate data to the accumulator

Description: The contents of the accumulator and the specified data are added, leaving the result in the accumulator.

Operation: ACC ← ACC+x

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	√	√	√	√

ADDM A,[m] Add the accumulator to the data memory

Description: The contents of the specified data memory and the accumulator are added. The result is stored in the data memory.

Operation: [m] ← ACC+[m]

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	√	√	√	√

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

Description: Data in the accumulator and the specified data memory perform a bitwise logical_AND operation. The result is stored in the accumulator.

Operation: ACC ← ACC ″AND″ [m]

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	√	—	—

AND A,x Logical AND immediate data to the accumulator

Description: Data in the accumulator and the specified data perform a bitwise logical_AND operation. The result is stored in the accumulator.

Operation: ACC ← ACC ″AND″ x

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	√	—	—

ANDM A,[m] Logical AND data memory with the accumulator

Description: Data in the specified data memory and the accumulator perform a bitwise logical_AND operation. The result is stored in the data memory.

Operation: [m] ← ACC ″AND″ [m]

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	√	—	—

CALL addr Subroutine call

Description: The instruction unconditionally calls a subroutine located at the indicated address. The program counter increments once to obtain the address of the next instruction, and pushes this onto the stack. The indicated address is then loaded. Program execution continues with the instruction at this address.

Operation: Stack ← PC+1
PC ← addr

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

CLR [m] Clear data memory

Description: The contents of the specified data memory are cleared to 0.

Operation: [m] ← 00H

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

CLR [m].i Clear bit of data memory

Description: The bit i of the specified data memory is cleared to 0.

Operation: [m].i ← 0

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

CLR WDT Clear Watchdog Timer

Description: The WDT is cleared (clears the WDT). The power down bit (PDF) and time-out bit (TO) are cleared.

Operation: WDT ← 00H
PDF and TO ← 0

Affected flag(s)

TO	PDF	OV	Z	AC	C
0	0	—	—	—	—

CLR WDT1 Preclear Watchdog Timer

Description: Together with CLR WDT2, clears the WDT. PDF and TO are also cleared. Only execution of this instruction without the other preclear instruction just sets the indicated flag which implies this instruction has been executed and the TO and PDF flags remain unchanged.

Operation: WDT ← 00H*
PDF and TO ← 0*

Affected flag(s)

TO	PDF	OV	Z	AC	C
0*	0*	—	—	—	—

CLR WDT2 Preclear Watchdog Timer

Description: Together with CLR WDT1, clears the WDT. PDF and TO are also cleared. Only execution of this instruction without the other preclear instruction, sets the indicated flag which implies this instruction has been executed and the TO and PDF flags remain unchanged.

Operation: WDT ← 00H*
PDF and TO ← 0*

Affected flag(s)

TO	PDF	OV	Z	AC	C
0*	0*	—	—	—	—

CPL [m] Complement data memory

Description: Each bit of the specified data memory is logically complemented (1′s complement). Bits which previously contained a 1 are changed to 0 and vice-versa.

Operation: $[m] \leftarrow [\overline{m}]$

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	√	—	—

CPLA [m] Complement data memory and place result in the accumulator

Description: Each bit of the specified data memory is logically complemented (1′s complement). Bits which previously contained a 1 are changed to 0 and vice-versa. The complemented result is stored in the accumulator and the contents of the data memory remain unchanged.

Operation: ACC ← $[\overline{m}]$

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	√	—	—

DAA [m] Decimal-Adjust accumulator for addition

Description: The accumulator value is adjusted to the BCD (Binary Coded Decimal) code. The accumulator is divided into two nibbles. Each nibble is adjusted to the BCD code and an internal carry (AC1) will be done if the low nibble of the accumulator is greater than 9. The BCD adjustment is done by adding 6 to the original value if the original value is greater than 9 or a carry (AC or C) is set; otherwise the original value remains unchanged. The result is stored in the data memory and only the carry flag (C) may be affected.

Operation:
If ACC.3~ACC.0 >9 or AC=1
then [m].3~[m].0 ← (ACC.3~ACC.0)+6, AC1=$\overline{AC}$
else [m].3~[m].0 ← (ACC.3~ACC.0), AC1=0
and
If ACC.7~ACC.4+AC1 >9 or C=1
then [m].7~[m].4 ← ACC.7~ACC.4+6+AC1,C=1
else [m].7~[m].4 ← ACC.7~ACC.4+AC1,C=C

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	√

DEC [m] Decrement data memory

Description: Data in the specified data memory is decremented by 1.

Operation: [m] ← [m]−1

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	√	—	—

DECA [m] Decrement data memory and place result in the accumulator

Description: Data in the specified data memory is decremented by 1, leaving the result in the accumulator. The contents of the data memory remain unchanged.

Operation: ACC ← [m]−1

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	√	—	—

HALT Enter power down mode

Description: This instruction stops program execution and turns off the system clock. The contents of the RAM and registers are retained. The WDT and prescaler are cleared. The power down bit (PDF) is set and the WDT time-out bit (TO) is cleared.

Operation: PC ← PC+1
PDF ← 1
TO ← 0

Affected flag(s)

TO	PDF	OV	Z	AC	C
0	1	—	—	—	—

INC [m] Increment data memory

Description: Data in the specified data memory is incremented by 1

Operation: [m] ← [m]+1

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	√	—	—

INCA [m] Increment data memory and place result in the accumulator

Description: Data in the specified data memory is incremented by 1, leaving the result in the accumulator. The contents of the data memory remain unchanged.

Operation: ACC ← [m]+1

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	√	—	—

JMP addr Directly jump

Description: The program counter are replaced with the directly-specified address unconditionally, and control is passed to this destination.

Operation: PC ←addr

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

MOV A,[m] Move data memory to the accumulator

Description: The contents of the specified data memory are copied to the accumulator.

Operation: ACC ← [m]

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

MOV A,x Move immediate data to the accumulator

Description The 8-bit data specified by the code is loaded into the accumulator.

Operation ACC ← x

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

MOV [m],A Move the accumulator to data memory

Description The contents of the accumulator are copied to the specified data memory (one of the data memories).

Operation [m] ←ACC

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

NOP No operation

Description No operation is performed. Execution continues with the next instruction.

Operation PC ← PC+1

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

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

Description Data in the accumulator and the specified data memory (one of the data memories) perform a bitwise logical_OR operation. The result is stored in the accumulator.

Operation ACC ← ACC ″OR″ [m]

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	√	—	—

OR A,x Logical OR immediate data to the accumulator

Description Data in the accumulator and the specified data perform a bitwise logical_OR operation. The result is stored in the accumulator.

Operation ACC ← ACC ″OR″ x

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	√	—	—

ORM A,[m] Logical OR data memory with the accumulator

Description Data in the data memory (one of the data memories) and the accumulator perform a bitwise logical_OR operation. The result is stored in the data memory.

Operation [m] ←ACC ″OR″ [m]

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	√	—	—

RET Return from subroutine

Description: The program counter is restored from the stack. This is a 2-cycle instruction.

Operation: PC ← Stack

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

RET A,x Return and place immediate data in the accumulator

Description: The program counter is restored from the stack and the accumulator loaded with the specified 8-bit immediate data.

Operation: PC ← Stack
ACC ← x

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

RETI Return from interrupt

Description: The program counter is restored from the stack, and interrupts are enabled by setting the EMI bit. EMI is the enable master (global) interrupt bit.

Operation: PC ← Stack
EMI ← 1

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

RL [m] Rotate data memory left

Description: The contents of the specified data memory are rotated 1 bit left with bit 7 rotated into bit 0.

Operation: [m].(i+1) ← [m].i; [m].i:bit i of the data memory (i=0~6)
[m].0 ← [m].7

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

RLA [m] Rotate data memory left and place result in the accumulator

Description: Data in the specified data memory is rotated 1 bit left with bit 7 rotated into bit 0, leaving the rotated result in the accumulator. The contents of the data memory remain unchanged.

Operation: ACC.(i+1) ← [m].i; [m].i:bit i of the data memory (i=0~6)
ACC.0 ← [m].7

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

RLC [m] Rotate data memory left through carry

Description The contents of the specified data memory and the carry flag are rotated 1 bit left. Bit 7 replaces the carry bit; the original carry flag is rotated into the bit 0 position.

Operation [m].(i+1) ← [m].i; [m].i:bit i of the data memory (i=0~6)
[m].0 ← C
C ← [m].7

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	√

RLCA [m] Rotate left through carry and place result in the accumulator

Description Data in the specified data memory and the carry flag are rotated 1 bit left. Bit 7 replaces the carry bit and the original carry flag is rotated into bit 0 position. The rotated result is stored in the accumulator but the contents of the data memory remain unchanged.

Operation ACC.(i+1) ← [m].i; [m].i:bit i of the data memory (i=0~6)
ACC.0 ← C
C ← [m].7

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	√

RR [m] Rotate data memory right

Description The contents of the specified data memory are rotated 1 bit right with bit 0 rotated to bit 7.

Operation [m].i ← [m].(i+1); [m].i:bit i of the data memory (i=0~6)
[m].7 ← [m].0

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

RRA [m] Rotate right and place result in the accumulator

Description Data in the specified data memory is rotated 1 bit right with bit 0 rotated into bit 7, leaving the rotated result in the accumulator. The contents of the data memory remain unchanged.

Operation ACC.(i) ← [m].(i+1); [m].i:bit i of the data memory (i=0~6)
ACC.7 ← [m].0

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

RRC [m] Rotate data memory right through carry

Description The contents of the specified data memory and the carry flag are together rotated 1 bit right. Bit 0 replaces the carry bit; the original carry flag is rotated into the bit 7 position.

Operation [m].i ← [m].(i+1); [m].i:bit i of the data memory (i=0~6)
[m].7 ← C
C ← [m].0

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	√

RRCA [m] Rotate right through carry and place result in the accumulator

Description: Data of the specified data memory and the carry flag are rotated 1 bit right. Bit 0 replaces the carry bit and the original carry flag is rotated into the bit 7 position. The rotated result is stored in the accumulator. The contents of the data memory remain unchanged.

Operation: ACC.i ← [m].(i+1); [m].i:bit i of the data memory (i=0~6)
ACC.7 ← C
C ← [m].0

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	√

SBC A,[m] Subtract data memory and carry from the accumulator

Description: The contents of the specified data memory and the complement of the carry flag are subtracted from the accumulator, leaving the result in the accumulator.

Operation: ACC ← ACC+$[\overline{m}]$+C

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	√	√	√	√

SBCM A,[m] Subtract data memory and carry from the accumulator

Description: The contents of the specified data memory and the complement of the carry flag are subtracted from the accumulator, leaving the result in the data memory.

Operation: [m] ← ACC+$[\overline{m}]$+C

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	√	√	√	√

SDZ [m] Skip if decrement data memory is 0

Description: The contents of the specified data memory are decremented by 1. If the result is 0, the next instruction is skipped. If the result is 0, the following instruction, fetched during the current instruction execution, is discarded and a dummy cycle is replaced to get the proper instruction (2 cycles). Otherwise proceed with the next instruction (1 cycle).

Operation: Skip if ([m]−1)=0, [m] ← ([m]−1)

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

SDZA [m] Decrement data memory and place result in ACC, skip if 0

Description: The contents of the specified data memory are decremented by 1. If the result is 0, the next instruction is skipped. The result is stored in the accumulator but the data memory remains unchanged. If the result is 0, the following instruction, fetched during the current instruction execution, is discarded and a dummy cycle is replaced to get the proper instruction (2 cycles). Otherwise proceed with the next instruction (1 cycle).

Operation: Skip if ([m]−1)=0, ACC ← ([m]−1)

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

SET [m] Set data memory

Description Each bit of the specified data memory is set to 1.

Operation [m] ← FFH

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

SET [m]. i Set bit of data memory

Description Bit i of the specified data memory is set to 1.

Operation [m].i ← 1

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

SIZ [m] Skip if increment data memory is 0

Description The contents of the specified data memory are incremented by 1. If the result is 0, the following instruction, fetched during the current instruction execution, is discarded and a dummy cycle is replaced to get the proper instruction (2 cycles). Otherwise proceed with the next instruction (1 cycle).

Operation Skip if ([m]+1)=0, [m] ← ([m]+1)

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

SIZA [m] Increment data memory and place result in ACC, skip if 0

Description The contents of the specified data memory are incremented by 1. If the result is 0, the next instruction is skipped and the result is stored in the accumulator. The data memory remains unchanged. If the result is 0, the following instruction, fetched during the current instruction execution, is discarded and a dummy cycle is replaced to get the proper instruction (2 cycles). Otherwise proceed with the next instruction (1 cycle).

Operation Skip if ([m]+1)=0, ACC ← ([m]+1)

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

SNZ [m].i Skip if bit i of the data memory is not 0

Description If bit i of the specified data memory is not 0, the next instruction is skipped. If bit i of the data memory is not 0, the following instruction, fetched during the current instruction execution, is discarded and a dummy cycle is replaced to get the proper instruction (2 cycles). Otherwise proceed with the next instruction (1 cycle).

Operation Skip if [m].i≠0

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

SUB A,[m] Subtract data memory from the accumulator

Description: The specified data memory is subtracted from the contents of the accumulator, leaving the result in the accumulator.

Operation: ACC ← ACC+$\overline{[m]}$+1

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	√	√	√	√

SUBM A,[m] Subtract data memory from the accumulator

Description: The specified data memory is subtracted from the contents of the accumulator, leaving the result in the data memory.

Operation: [m] ← ACC+$\overline{[m]}$+1

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	√	√	√	√

SUB A,x Subtract immediate data from the accumulator

Description: The immediate data specified by the code is subtracted from the contents of the accumulator, leaving the result in the accumulator.

Operation: ACC ← ACC+$\overline{x}$+1

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	√	√	√	√

SWAP [m] Swap nibbles within the data memory

Description: The low-order and high-order nibbles of the specified data memory (1 of the data memories) are interchanged.

Operation: [m].3~[m].0 ↔ [m].7~[m].4

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

SWAPA [m] Swap data memory and place result in the accumulator

Description: The low-order and high-order nibbles of the specified data memory are interchanged, writing the result to the accumulator. The contents of the data memory remain unchanged.

Operation: ACC.3~ACC.0 ← [m].7~[m].4
ACC.7~ACC.4 ← [m].3~[m].0

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

SZ [m] Skip if data memory is 0

Description If the contents of the specified data memory are 0, the following instruction, fetched during the current instruction execution, is discarded and a dummy cycle is replaced to get the proper instruction (2 cycles). Otherwise proceed with the next instruction (1 cycle).

Operation Skip if [m]=0

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

SZA [m] Move data memory to ACC, skip if 0

Description The contents of the specified data memory are copied to the accumulator. If the contents is 0, the following instruction, fetched during the current instruction execution, is discarded and a dummy cycle is replaced to get the proper instruction (2 cycles). Otherwise proceed with the next instruction (1 cycle).

Operation Skip if [m]=0

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

SZ [m].i Skip if bit i of the data memory is 0

Description If bit i of the specified data memory is 0, the following instruction, fetched during the current instruction execution, is discarded and a dummy cycle is replaced to get the proper instruction (2 cycles). Otherwise proceed with the next instruction (1 cycle).

Operation Skip if [m].i=0

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

TABRDC [m] Move the ROM code (current page) to TBLH and data memory

Description The low byte of ROM code (current page) addressed by the table pointer (TBLP) is moved to the specified data memory and the high byte transferred to TBLH directly.

Operation [m] ← ROM code (low byte)
TBLH ← ROM code (high byte)

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

TABRDL [m] Move the ROM code (last page) to TBLH and data memory

Description The low byte of ROM code (last page) addressed by the table pointer (TBLP) is moved to the data memory and the high byte transferred to TBLH directly.

Operation [m] ← ROM code (low byte)
TBLH ← ROM code (high byte)

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	—	—	—

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

Description
Data in the accumulator and the indicated data memory perform a bitwise logical Exclusive_OR operation and the result is stored in the accumulator.

Operation
ACC ← ACC ″XOR″ [m]

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	√	—	—

XORM A,[m] Logical XOR data memory with the accumulator

Description
Data in the indicated data memory and the accumulator perform a bitwise logical Exclusive_OR operation. The result is stored in the data memory. The 0 flag is affected.

Operation
[m] ← ACC ″XOR″ [m]

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	√	—	—

XOR A,x Logical XOR immediate data to the accumulator

Description
Data in the accumulator and the specified data perform a bitwise logical Exclusive_OR operation. The result is stored in the accumulator. The 0 flag is affected.

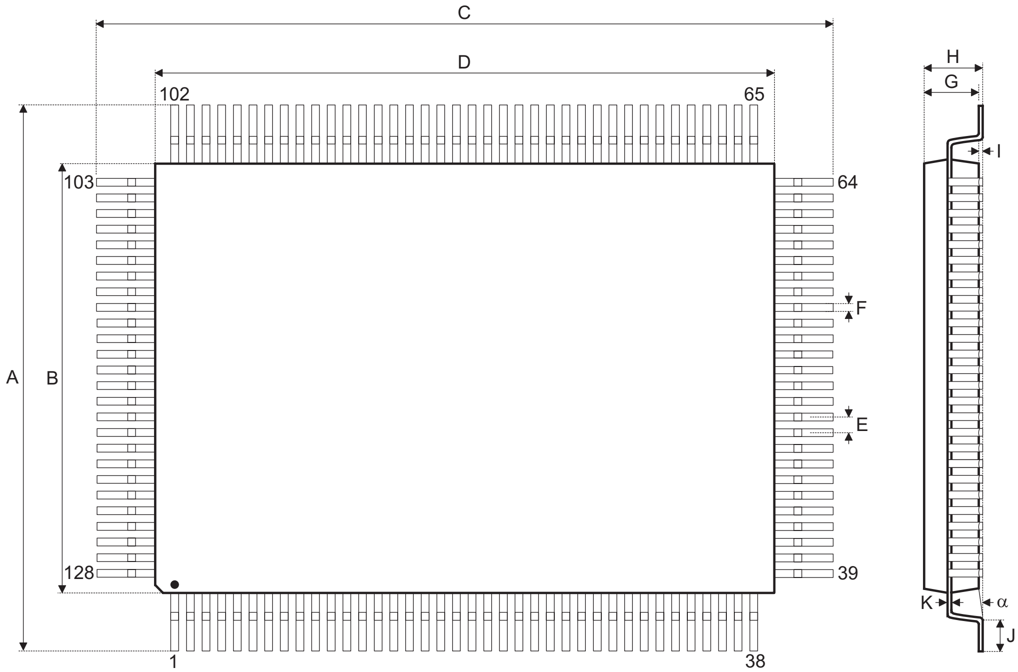
Operation
ACC ← ACC ″XOR″ x

Affected flag(s)

TO	PDF	OV	Z	AC	C
—	—	—	√	—	—

Package Information

128-pin QFP (14×20) Outline Dimensions



Symbol	Dimensions in mm		
	Min.	Nom.	Max.
A	17.00	—	17.50
B	13.90	—	14.10
C	23.00	—	23.50
D	19.90	—	20.10
E	—	0.50	—
F	—	0.20	—
G	2.50	—	3.10
H	—	—	3.40
I	—	0.10	—
J	0.65	—	0.95
K	0.10	—	0.20
α	0°	—	7°

Holtek Semiconductor Inc. (Headquarters)
No.3, Creation Rd. II, Science Park, Hsinchu, Taiwan
Tel: 886-3-563-1999
Fax: 886-3-563-1189
http://www.holtek.com.tw

Holtek Semiconductor Inc. (Taipei Sales Office)
4F-2, No. 3-2, YuanQu St., Nankang Software Park, Taipei 115, Taiwan
Tel: 886-2-2655-7070
Fax: 886-2-2655-7373
Fax: 886-2-2655-7383 (International sales hotline)

Holtek Semiconductor Inc. (Shanghai Sales Office)
7th Floor, Building 2, No.889, Yi Shan Rd., Shanghai, China 200233
Tel: 021-6485-5560
Fax: 021-6485-0313
http://www.holtek.com.cn

Holtek Semiconductor Inc. (Shenzhen Sales Office)
5/F, Unit A, Productivity Building, Cross of Science M 3rd Road and Gaoxin M 2nd Road, Science Park, Nanshan District, Shenzhen, China 518057
Tel: 0755-8616-9908, 8616-9308
Fax: 0755-8616-9533

Holtek Semiconductor Inc. (Beijing Sales Office)
Suite 1721, Jinyu Tower, A129 West Xuan Wu Men Street, Xicheng District, Beijing, China 100031
Tel: 010-6641-0030, 6641-7751, 6641-7752
Fax: 010-6641-0125

Holtek Semiconductor Inc. (Chengdu Sales Office)
709, Building 3, Champagne Plaza, No.97 Dongda Street, Chengdu, Sichuan, China 610016
Tel: 028-6653-6590
Fax: 028-6653-6591

Holmate Semiconductor, Inc. (North America Sales Office)
46729 Fremont Blvd., Fremont, CA 94538
Tel: 510-252-9880
Fax: 510-252-9885
http://www.holmate.com