

KXD8P53

OTP-Based 8-Bit Microcontroller Series

High-Performance RISC CPU

- 8-bit wide data path
- Only 42 single-word instructions to learn
- All single-cycle instructions except for program branches, which will take two cycles.
- 13-bit wide instructions
- 5-level deep hardware stack
- GOTO instruction can reach all OTP area
- CALL instruction can reach all OTP area subroutine
- Direct, indirect addressing modes for data and instructions.
- Operating Speed: DC-20 MHz clock input
DC-8 MHz internal clock

Device	Program Memory (Word)	Data Memory (RAM)	Pins #	I/O	Timers (8 bits)
KXD8P53	1K	49	14	12	1

Special Microcontroller Features

- Internal Power-on Reset (POR)
- Power-on Reset Timer (PORT).
- Built-in Low Voltage Detector (LVD) for Brown-out Reset (BOR)
- Watchdog Timer (WDT) with dedicated on-chip oscillator and software Watchdog enable/disable control
- One Internal Interrupt Source: Timer0 Overflow; Two External Interrupt Sources: INT Pin, IOB input change
- Power-saving SLEEP mode
- Built-in 8MHz, 4MHz, 1MHz, and 500KHz Internal RC oscillator
- Programmable code protection
- Selectable oscillator options:
 - ERIC: External RC Oscillator
 - HF: High Frequency Crystal/Resonator Oscillator
 - LF: Low Frequency Crystal Oscillator
 - IRC: Internal RC Oscillator
- Operating Voltage range: 2.2V to 5.5V

Peripheral Features

- Two I/O Ports IOA and IOB with individual direction control
- Software I/O Pull-High/Pull-Low or open-drain control
- Wake-up from SLEEP on INT pin or IOB input change
- 8-bit Real-Time Clock/Counter (Timer0) with 3-bit Programmable Pre-scaler

GENERAL DESCRIPTION

The KXD8P53 device is a high performance, low-cost, high noise immunity, OTP-based 8-bit CMOS technology microcontroller. Its operation kernel is implemented with RISC architecture with only 42 instructions. All instructions are single cycle except for program branches, which will take two cycles. The easy-to-use instruction set makes development easy.

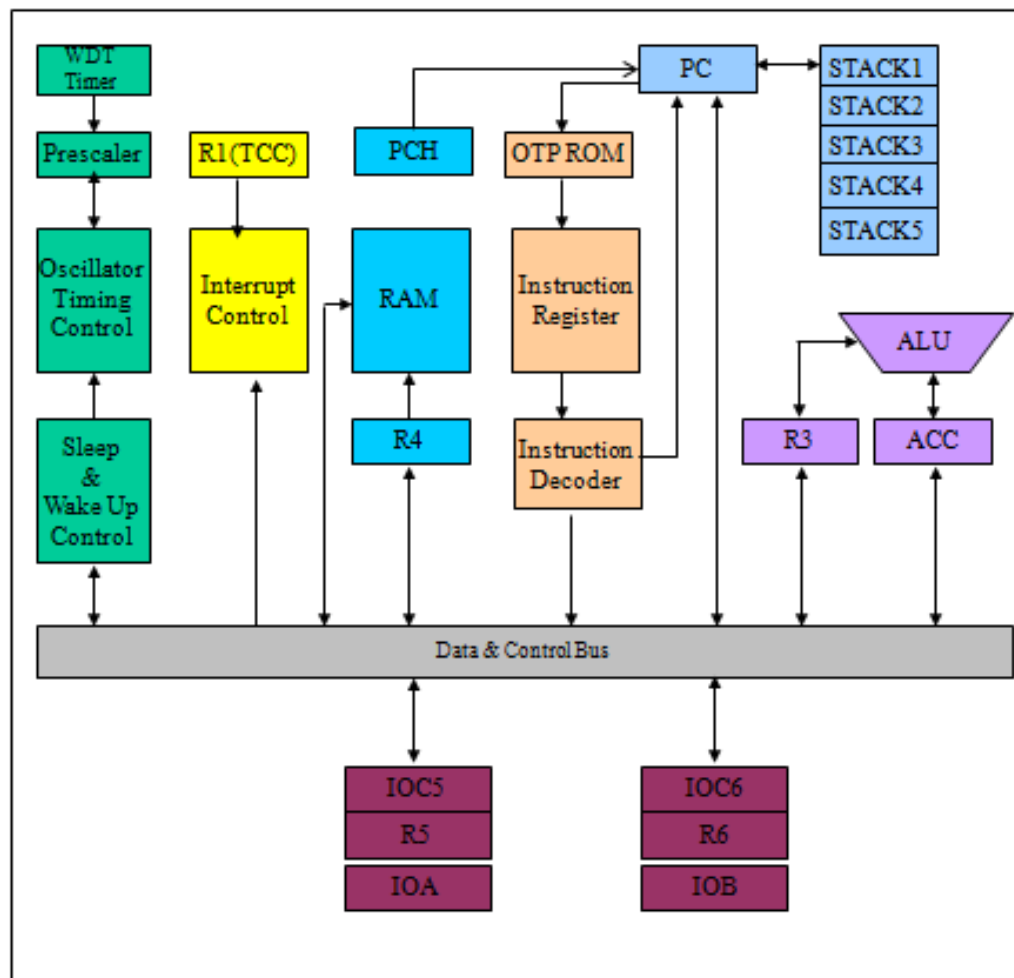
The KXD8P53 product is equipped with Power-on Reset (POR), Power-on Reset Timer (PORT), Brown-out Reset (BOR), Watchdog Timer, OTP, SRAM, tri-state I/O port, I/O pull-high/ pull-low/ open-drain control, Power saving SLEEP mode, real time programmable clock/counter, Interrupt, Wake-up from SLEEP mode, and Code Protection for OTP products. With the 1K×13 OTP-ROM feature the KXD8P53 provides a convenient way of developing and verifying

KXD8P53

user programs.

The KXD8P53 can directly or indirectly address its register files and data memory. All special function registers including the program counter are mapped in the data memory.

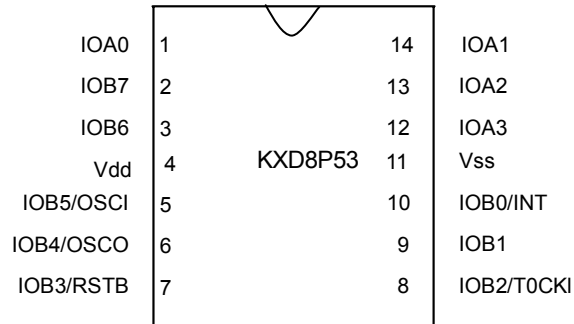
BLOCK DIAGRAM



KXD8P53

PIN CONNECTION

PDIP



PIN DESCRIPTIONS

Name	I/O	Description
IOA0 ~ IOA3	I/O	IOA0 ~ IOA3 as bi-direction I/O pin (software-controlled pull-down)
IOB0/INT	I/O	Bi-direction I/O pin with system wake-up function Software-controlled pull-high/pull-down/open-drain/ external interrupt input
IOB1	I/O	Bi-direction I/O pin with system wake-up function Software-controlled pull-high/open-drain/pull-down
IOB2/T0CKI	I/O	Bi-direction I/O pin with system wake-up function Software-controlled pull-high/open-drain/pull-down / External clock input to Timer0
IOB3/RSTB	I	IOB3 is input pin only with system wake-up function / System reset input. Active low RESET to the device. If it is configured as RSTB, the weak pull-high will apply to this pin.
IOB4/OSCO	I/O	Bi-direction I/O pin with system wake-up function (RCOUT optional in IRC/ERIC, ERC mode) Software-controlled pull-high/open-drain / Oscillator crystal output (XT, LP mode) Outputs with the instruction cycle rate (RCOUT optional in IRC/ERIC, ERC mode)
IOB5/OSCI	I/O	Bi-direction I/O pin with system wake-up function (IRC mode) Software-controlled pull-high/open-drain / Oscillator crystal input (XT, LP mode) External clock source input (ERIC, ERC mode)
IOB6 ~ IOB7	I/O	Bi-direction I/O pin with system wake-up function Software-controlled pull-high/open-drain
Vdd	-	Positive supply
Vss	-	Ground

Legend: I=input, O=output, I/O=input/output

1.0 MEMORY ORGANIZATION

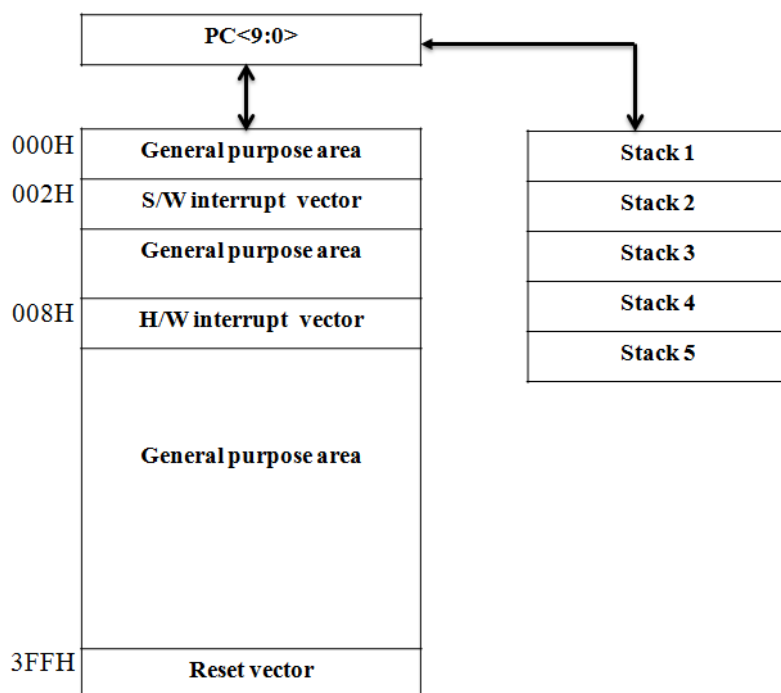
KXD8P53 memory is organized into program memory and data memory.

1.1 Program Memory Organization

The KXD8P53 ROM includes Reset vector, S/W interrupt vector, H/W interrupt vector and General purpose area. The RESET vector, 3FFh, is the program beginning address.

The H/W interrupt vector is at 008h. And the S/W interrupt vector is at 002h. KXD8P53 supports all OTP area CALL/GOTO instructions without page.

FIGURE 1.1: 1K words ROM and STACK



1.2 Data Memory Organization

Data memory is composed of registers or bytes of RAM. The register file is divided into two functional groups: Special Function Registers (SFR) and General Purpose Registers (GPR).

The General Purpose Registers are accessed either directly or indirectly through the FSR register.

The Special Function Registers are registers used by the CPU and peripheral functions to control the operation of the device.

TABLE 1.1: Registers Map for KXD8P53 Series

Address	R Page Registers	IOC Page Registers
00	R0: INDF	
01	R1: TMR0	
02	R2: PCL	
03	R3: STATUS	
04	R4: FSR	
05	R5: IOA	IOC5: IOSTA
06	R6: IOB	IOC6: IOSTB
07	R7: GPR	
08	R8: PCON	
09	R9: WUCON	
0A	RA: PCHBUF	
0B	RB: PDCON	
0C	RC: ODCON	
0D	RD: PHCON	
0E	RE: INTEN	
0F	RF: INTFLAG	
10	General Purpose Registers	
3F		
OPTION		

TABLE 1.2: The Registers Controlled by OPTION or IOST Instructions

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
N/A (w)	OPTION	*	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
05h (w)	IOC5	IOA I/O Control Register							
06h (w)	IOC6	IOB I/O Control Register							

TABLE 1.3: Operational Registers Map

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
00h (r/w)	INDF	Uses contents of FSR to address data memory (not a physical register)							
01h (r/w)	TMR0	8-bit real-time clock/counter							
02h (r/w)	PCL	Low order 8 bits of PC							
03h (r/w)	STATUS	RST	GP1	GP0	TO	PD	Z	DC	C
04h (r/w)	FSR	*	*	Indirect data memory address pointer					
05h (r/w)	PORTA					IOA3	IOA2	IOA1	IOA0
06h (r/w)	PORTB	IOB7	IOB6	IOB5	IOB4	IOB3	IOB2	IOB1	IOB0
07h (r/w)	SRAM	General Purpose Register							
08h (r/w)	PCON	WDTE	EIS	LVDTE	*	*	*	*	*
09h (r/w)	WUCON	WUB7	WUB6	WUB5	WUB4	WUB3	WUB2	WUB1	WUB0
0Ah (r/w)	PCHBUF	-	-	-	-	-	-	2 MSBs Buffer of PC	
0Bh (r/w)	PDCON		/PDB2	/PDB1	/PDB0	/PDA3	/PDA2	/PDA1	/PDA0
0Ch (r/w)	ODCON	ODB7	ODB6	ODB5	ODB4		ODB2	ODB1	ODB0
0Dh (r/w)	PHCON	/PHB7	/PHB6	/PHB5	/PHB4		/PHB2	/PHB1	/PHB0
0Eh (r/w)	INTEN	GIE	*	*	*	*	INTIE	PBIE	TOIE
0Fh (r/w)	INTFLAG	-	-	-	-	-	INTIF	PBIF	TOIF

Legend: - = unimplemented, read as '0', * = unimplemented, read as '1',

2.0 FUNCTIONAL DESCRIPTIONS

2.1 Operational Registers

2.1.1 INDF (Indirect Addressing Register)

00h(r/w)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INDF	Uses contents of FSR to address data memory (not a physical register)							

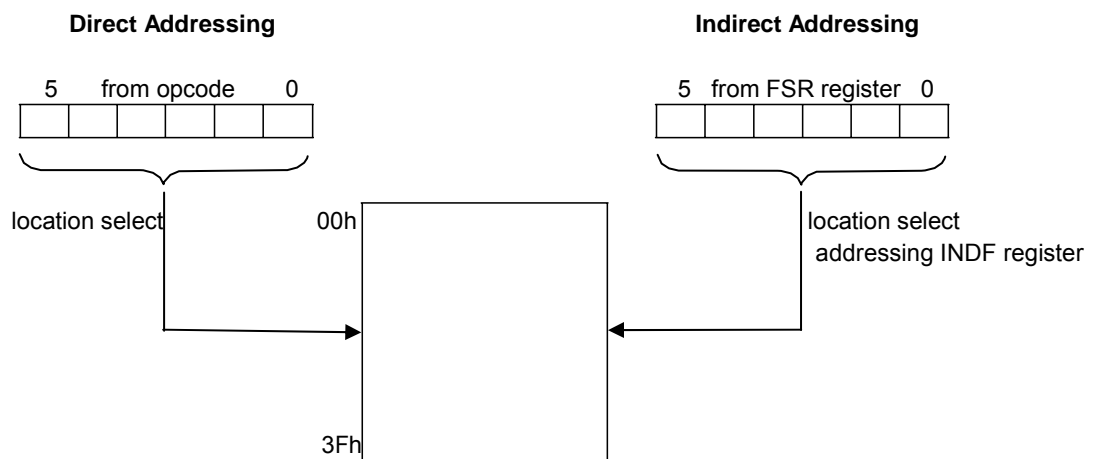
The INDF Register is not a physical register. Addressing INDF actually addresses the register pointed by FSR Register. Reading the INDF register itself indirectly (FSR="0") will produce 00h.

Writing to the INDF register indirectly results in a no-operation (although status bits may be affected). The bits 5-0 of FSR register are used to select up to 64 registers (address: 00h ~ 3Fh).

EXAMPLE 2.1: INDIRECT ADDRESSING

- Register R22 contains the value 25h
- Register R23 contains the value 3Ah
- Load the value 22 into the FSR Register
- A read of the INDF Register will return the value of 25h
- Increment the value of the FSR Register by one (@FSR=23h)
- A read of the INDR register now will return the value of 3Ah.

FIGURE 2.1: Direct/Indirect Addressing



2.1.2 **TMR0 (Time Clock/Counter register)**

01h (r/w)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TMR0	8-bit real-time clock/counter							

The Timer0 is an 8-bit timer/counter. The TMR0 register is readable and writable. T0CS bit (OPTION<5>) chooses the clock source of Timer0 from the instruction cycle clock or by an external clock source (T0CKI pin). If T0CKI pin is selected, the Timer0 is increased by T0CKI signal rising/falling edge (selected by T0SE bit (OPTION<4>)).

The prescaler is assigned to Timer0 by clearing the PSA bit (OPTION<3>). In this case, the prescaler will be cleared when TMR0 register is written with a value.

2.1.3 **PCL (Low Bytes of Program Counter) & Stack**

02h (r/w)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCL	Low order 8 bits of PC							

KXD8P53 devices have a 10-bit wide Program Counter (PC) and a 5-deep 10-bit wide hardware push/pop stack. The low byte of PC is called the PCL register. This register is readable and writable. The high byte of PC is called the PCH register. This register contains the PC<9:8> bits and is not directly readable or writable. All updates to the PCH register go through the PCHBUF register. As a program instruction is executed, the Program Counter will contain the address of the next program instruction to be executed.

The PC value is increased by one, every instruction cycle, unless an instruction changes the PC. For a GOTO instruction, bits 9:0 of the PC are provided by the GOTO instruction word. The PCL register is mapped to PC <7:0>, and the PCHBUF register is not updated.

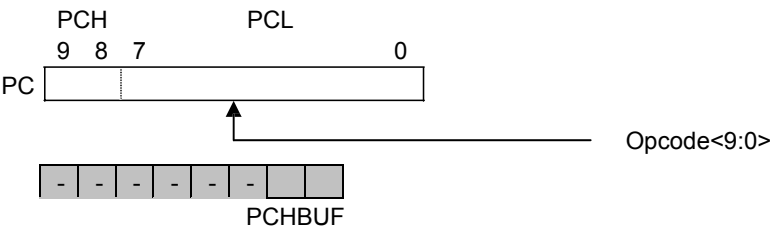
For a CALL instruction, bits 9:0 of the PC again are provided by the instruction word. The next PC will be loaded.
(PUSHed)onto the top of STACK. The PCL register is mapped to PC<7:0>, and the PCHBUF register is not updated.

“RETIA” (“RETFIE”, or “RETURN”) instruction loads the program counter with the contents of the top-level stack. The
PCL register is mapped to PC<7:0>, and the PCHBUF register is not updated.

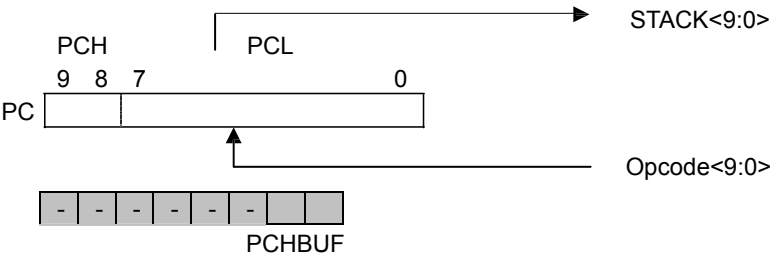
For any instruction where the PCL is the destination, the PC<7:0> is provided by the instruction word or ALU result. However, the PC<9:8> will come from the PCHBUF<1:0> bits (PCHBUF --> PCH).
PCHBUF register is never updated with the contents of PCH.

FIGURE 2.2: Loading of PC in Different Situations

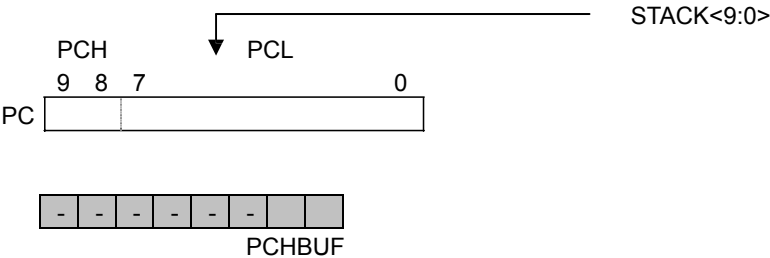
Situation 1: GOTO Instruction



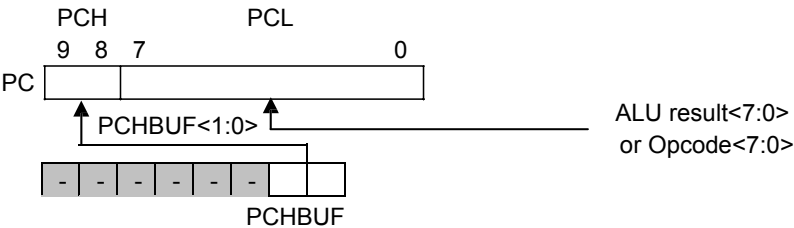
Situation 2: CALL Instruction



Situation 3: RETIA, RETFIE, or RETURN Instruction



Situation 4: Instruction with PCL as destination



Note: 1. PCHBUF is used only for instruction with PCL as destination for KXD8P53.

2.1.4 STATUS (Status Register)

03h (r/w)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STATUS	RST	GP1	GP0	TO	PD	Z	DC	C

This register contains the arithmetic status of the ALU, the RESET status. If the STATUS Register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the TO and PD bits are not writable. Therefore, the result of an instruction with the STATUS Register as destination may be different than intended. For example, CLRR STATUS will clear the upper three bits and set the Z bit. This leaves the STATUS Register as 000u u1uu (where u = unchanged).

RST : Bit for wake-up type.
= 1, Wake-up from SLEEP on Port B input change.
= 0, Wake-up from other reset types.

GP1:GP0 : General purpose read/write bits.

TO : Time overflow flag bit.
= 1, after power-up or by the CLRWDT or SLEEP instruction.
= 0, a watch-dog time overflow occurred.

PD : Power down flag bit.
= 1, after power-up or by the CLRWDT instruction.
= 0, by the SLEEP instruction.

Z : Zero bit.
= 1, the result of a logic operation is zero.
= 0, the result of a logic operation is not zero.

DC : Half carry/half borrow bit.
ADDAR, ADDIA
= 1, a carry from the 4th low order bit of the result occurred.
= 0, a carry from the 4th low order bit of the result did not occur.
SUBAR, SUBIA
= 1, a borrow from the 4th low order bit of the result did not occur.
= 0, a borrow from the 4th low order bit of the result occurred.

C : Carry/borrow bit.
ADDAR, ADDIA
= 1, a carry occurred.
= 0, a carry did not occur.
SUBAR, SUBIA
= 1, a borrow did not occur.
= 0, a borrow occurred.

Note : A subtraction is executed by adding the two's complement of the second operand. For rotate (RRR, RLR) instructions, this bit is loaded with either the high or low order bit of the source register.

2.1.5 FSR (Indirect Data Memory Address Pointer)

04h (r/w)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FSR	*	*	Indirect data memory address pointer					

Bit7:Bit6 : Not used. Read as “1”s.

Bit5:Bit0 : Select registers address in the indirect addressing mode. See 2.1.1 for detail description.

2.1.6 IOA, IOB (Port Data Registers)

05h (r/w)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IOA					IOA3	IOA2	IOA1	IOA0

06h (r/w)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IOB	IOB7	IOB6	IOB5	IOB4	IOB3	IOB2	IOB1	IOB0

Reading the port (IOA, IOB register) reads the status of the pins independent of the pin's input/output modes. Writing to these ports will write to the port data latch.

IOA is a 4-bit port data Register. Only the low order 4 bits are used (IOA<3:0>). Bits 7-4 are general purpose read/write bits. IOB is a 8-bit port data register. And IOB3 is input only.

2.1.7 PCON (Power Control Register)

08h (r/w)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCON	WDTE	EIS	LVDTE	*	*	*	*	*

WDTE : WDT (watch-dog timer) enable bit.

= 0, Disable WDT.

= 1, Enable WDT.

EIS : Define the function of IOB0/INT pin.

= 0, IOB0 (bi-directional I/O pin) is selected. The path of INT is masked.

= 1, INT (external interrupt pin) is selected. In this case, the I/O control bit of IOB0 must be set to “1”. The path of Port B input change of IOB0 pin is masked by hardware, the status of INT pin can also be read by way of reading PORTB.

LVDTE : LVDT (low voltage detector) enable bit.

= 0, Disable LVDT.

= 1, Enable LVDT.

Bit4:Bit0 : Not used. Read as “1”s.

2.1.8 WUCON (IOB Input Change Interrupt/ Wake-up Control Register)

09h (r/w)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WUCON	WUB7	WUB6	WUB5	WUB4	WUB3	WUB2	WUB1	WUB0

WUB7 : = 0, Disable Enable the input change interrupt/wake-up function of IOB7 pin.

= 1, Enable the input change interrupt/wake-up function of IOB7 pin.

WUB6 : = 0, Disable the input change interrupt/wake-up function of IOB6 pin.

= 1, Enable the input change interrupt/wake-up function of IOB6 pin.

WUB5 : = 0, Disable the input change interrupt/wake-up function of IOB5 pin.
= 1, Enable the input change interrupt/wake-up function of IOB5 pin.

WUB4 : = 0, Disable the input change interrupt/wake-up function of IOB4 pin.
= 1, Enable the input change interrupt/wake-up function of IOB4 pin.

WUB3 : = 0, Disable the input change interrupt/wake-up function of IOB3 pin.
= 1, Enable the input change interrupt/wake-up function of IOB3 pin.

WUB2 : = 0, Disable the input change interrupt/wake-up function of IOB2 pin.
= 1, Enable the input change interrupt/wake-up function of IOB2 pin.

WUB1 : = 0, Disable the input change interrupt/wake-up function of IOB1 pin.
= 1, Enable the input change interrupt/wake-up function of IOB1 pin.

WUB0 : = 0, Disable the input change interrupt/wake-up function of IOB0 pin.
= 1, Enable the input change interrupt/wake-up function of IOB0 pin.

2.1.9 PCHBUF (High Byte Buffer of Program Counter)

0Ah (r/w)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCHBUF	-	-	-	-	-	-	2 MSBs Buffer of PC	

Bit7:Bit2 : Not used. Read as "0"s.

Bit1:Bit0 : See 2.1.3 for detail description.

2.1.10 PDCON (Pull-down Control Register)

0Bh (r/w)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PDCON		/PDB2	/PDB1	/PDB0	/PDA3	/PDA2	/PDA1	/PDA0

Bit7 : General purpose read/write bit.

/PDB2 : = 0, Enable the internal pull-down of IOB2 pin.
= 1, Disable the internal pull-down of IOB2 pin.

/PDB1 : = 0, Enable the internal pull-down of IOB1 pin.
= 1, Disable the internal pull-down of IOB1 pin.

/PDB0 : = 0, Enable the internal pull-down of IOB0 pin.
= 1, Disable the internal pull-down of IOB0 pin.

/PDA3 : = 0, Enable the internal pull-down of IOA3 pin.
= 1, Disable the internal pull-down of IOA3 pin.

/PDA2 : = 0, Enable the internal pull-down of IOA2 pin.
= 1, Disable the internal pull-down of IOA2 pin.

/PDA1 : = 0, Enable the internal pull-down of IOA1 pin.
= 1, Disable the internal pull-down of IOA1 pin.

/PDA0 : = 0, Enable the internal pull-down of IOA0 pin.
= 1, Disable the internal pull-down of IOA0 pin.

2.1.11 ODCON (Open-drain Control Register)

0Ch (r/w)	B7	B6	B5	B4	B3	B2	B1	B0
ODCON	ODB7	ODB6	ODB5	ODB4		ODB2	ODB1	ODB0

ODB7 : = 0, Disable the internal open-drain of IOB7 pin.
= 1, Enable the internal open-drain of IOB7 pin.

ODB6 : = 0, Disable the internal open-drain of IOB6 pin.
= 1, Enable the internal open-drain of IOB6 pin.

ODB5 : = 0, Disable the internal open-drain of IOB5 pin.
= 1, Enable the internal open-drain of IOB5 pin.

ODB4 : = 0, Disable the internal open-drain of IOB4 pin.
= 1, Enable the internal open-drain of IOB4 pin.

Bit 3 : General purpose read/write bit.

ODB2 : = 0, Disable the internal open-drain of IOB2 pin.
= 1, Enable the internal open-drain of IOB2 pin.

ODB1 : = 0, Disable the internal open-drain of IOB1 pin.
= 1, Enable the internal open-drain of IOB1 pin.

ODB0 : = 0, Disable the internal open-drain of IOB0 pin.
= 1, Enable the internal open-drain of IOB0 pin.

2.1.12 PHCON (Pull-high Control Register)

0Dh (r/w)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PHCON	/PHB7	/PHB6	/PHB5	/PHB4		/PHB2	/PHB1	/PHB0

/PHB7 : = 0, Enable the internal pull-high of IOB7 pin.
= 1, Disable the internal pull-high of IOB7 pin.

/PHB6 : = 0, Enable the internal pull-high of IOB6 pin.
= 1, Disable the internal pull-high of IOB6 pin.

/PHB5 : = 0, Enable the internal pull-high of IOB5 pin.
= 1, Disable the internal pull-high of IOB5 pin.

/PHB4 : = 0, Enable the internal pull-high of IOB4 pin.
= 1, Disable the internal pull-high of IOB4 pin.

Bit3 : General purpose read/write bit.

/PHB2 : = 0, Enable the internal pull-high of IOB2 pin.
= 1, Disable the internal pull-high of IOB2 pin.

/PHB1 : = 0, Enable the internal pull-high of IOB1 pin.
= 1, Disable the internal pull-high of IOB1 pin.

/PHB0 : = 0, Enable the internal pull-high of IOB0 pin.
= 1, Disable the internal pull-high of IOB0 pin.

2.1.13 INTEN (Interrupt Mask Register)

0Eh (r/w)	B7	B6	B5	B4	B3	B2	B1	B0
INTEN	GIE	*	*	*	*	INTIE	PBIE	T0IE

GIE : Global interrupt enable bit.

= 0, Disable all interrupts. For wake-up from SLEEP mode through an interrupt event, the device will continue execution at the instruction after the SLEEP instruction.

= 1, Enable all un-masked interrupts. For wake-up from SLEEP mode through an interrupt event, the device will branch to the interrupt address (008h).

Note : When an interrupt event occur with the GIE bit and its corresponding interrupt enable bit are all set, the GIE bit will be cleared by hardware to disable any further interrupts. The RETFIE instruction will exit the interrupt routine and set the GIE bit to re-enable interrupt.

Bit6:BIT3 : Not used. Read as "1"s.

INTIE : External INT pin interrupt enable bit.

= 0, Disable the External INT pin interrupt.

= 1, Enable the External INT pin interrupt.

PBIE : Port B input change interrupt enable bit.

= 0, Disable the Port B input change interrupt.

= 1, Enable the Port B input change interrupt .

T0IE : Timer0 overflow interrupt enable bit.

= 0, Disable the Timer0 overflow interrupt.

= 1, Enable the Timer0 overflow interrupt.

2.1.14 INTFLAG (Interrupt Status Register)

0Fh (r/w)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTFLAG	-	-	-	-	-	INTIF	PBIF	T0IF

INTIF : External INT pin interrupt flag. Set by rising/falling (selected by INTEDG bit (OPTION<6>)) edge on INT pin, reset by software.

PBIF : Port B input change interrupt flag. Set when Port B input changes, reset by software.

T0IF : Timer0 overflow interrupt flag. Set when Timer0 overflows, reset by software.

Bit7:BIT3 : Not used. Read as "0"s.

2.1.15 ACC (Accumulator)

(r/w)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ACC	Accumulator							

Accumulator is an internal data transfer, or instruction operand holding. It can not be addressed.

2.1.16 OPTION Register

(w)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OPTION	*	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0

The OPTION register is not memory mapped and is therefore only addressable by OPTION instruction.

By executing the OPTION instruction, the contents of the ACC Register will be transferred to the OPTION Register.

The OPTION Register is a 7-bit wide, write-only register which contains various control bits to configure the Timer0/WDT prescaler, Timer0, and the external INT interrupt. The OPTION Register are “write-only” and are set all “1”s except INTEDG bit.

Bit 7 : Not used.

INTEDG : Interrupt edge select bit.

- = 1, interrupt on rising edge of INT pin.
- = 0, interrupt on falling edge of INT pin.

T0CS : TMR0 clock source select bit.

- = 1, External T0CKI pin. Pin IOB2/T0CKI is forced to be an input even if IOST IOB2 = “0”.
- = 0, internal instruction clock cycle.

T0SE : TMR0 source edge select bit.

- = 1, Falling edge on T0CKI pin.
- = 0, Rising edge on T0CKI pin.

PSA : Prescaler assign bit.

- = 1, WDT (watch-dog timer).
- = 0, TMR0 (Timer0).

PS2:PS0 : Prescaler rate select bits.

PS2:PS0	WDT Rate	Timer0 Rate
0 0 0	1:1	1:2
0 0 1	1:2	1:4
0 1 0	1:4	1:8
0 1 1	1:8	1:16
1 0 0	1:16	1:32
1 0 1	1:32	1:64
1 1 0	1:64	1:128
1 1 1	1:128	1:256

2.1.17 IOSTA & IOSTB (Port I/O Control Registers)

(w)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IOSTA	IOA I/O Control Register							
IOSTB	IOB I/O Control Register							

The Port I/O control registers are not memory mapped and are therefore only addressable by IOST instruction.

A “IOST R” (05h~06h) instruction will load the current ACC value into Port I/O Control Registers. A ‘1’ from a IOST Register bit puts the corresponding output driver in hi-impedance state (input mode). A ‘0’ enables the output buffer and puts the contents of the output data latch on the selected pins (output mode). The IOST Registers are “write-only” and are set (output drivers disabled) upon RESET.

2.2 I/O Ports

IOA and IOB are bi-directional tri-state I/O ports. IOA is a 4-pin I/O port. IOB is a 8-pin I/O port. Please note that IOB3 is an input only pin.

All I/O pins have data direction control registers (IOSTA, IOSTB) which can configure these pins as output or input.

The exceptions are IOB3 which is input only and IOB2 which may be controlled by the T0CS bit (OPTION<5>). IOB<7:4> and IOB<2:0> have its corresponding pull-high control bits (PHCON register) to enable the weak internal pull-high. The weak pull-high is automatically turned off when the pin is configured as an output pin.

IOA<3:0> and IOB<2:0> have its corresponding pull-down control bits (PDCON register) to enable the weak internal pull-down. The weak pull-down is automatically turned off when the pin is configured as an output pin. IOB<7:4> and IOB<2:0> have its corresponding open-drain control bits (ODCON register) to enable the open-drain output when these pins are configured to be an output pin.

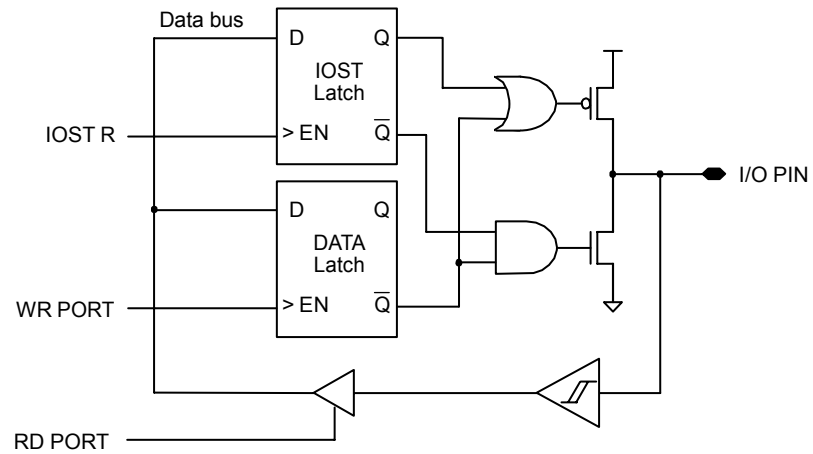
IOB<7:0> also provides the input change interrupt/wake-up function. Each pin has its corresponding input change interrupt/wake-up enable bits (WUCON) to select the input change interrupt/wake-up source.

The IOB0 is also an external interrupt input signal by setting the EIS bit (PCON<6>). In this case, IOB0 input change interrupt/wake-up function will be disabled by hardware even if it is enabled by software.

The CONFIGURATION words can set several I/Os to alternate functions. When acting as alternate functions the pins will read as "0" during port read.

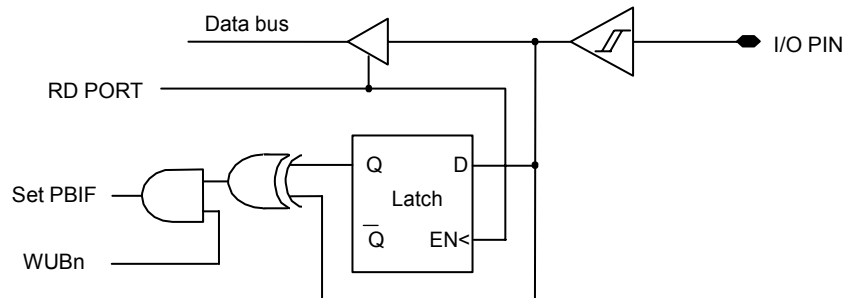
FIGURE 2.3: Block Diagram of I/O PINs

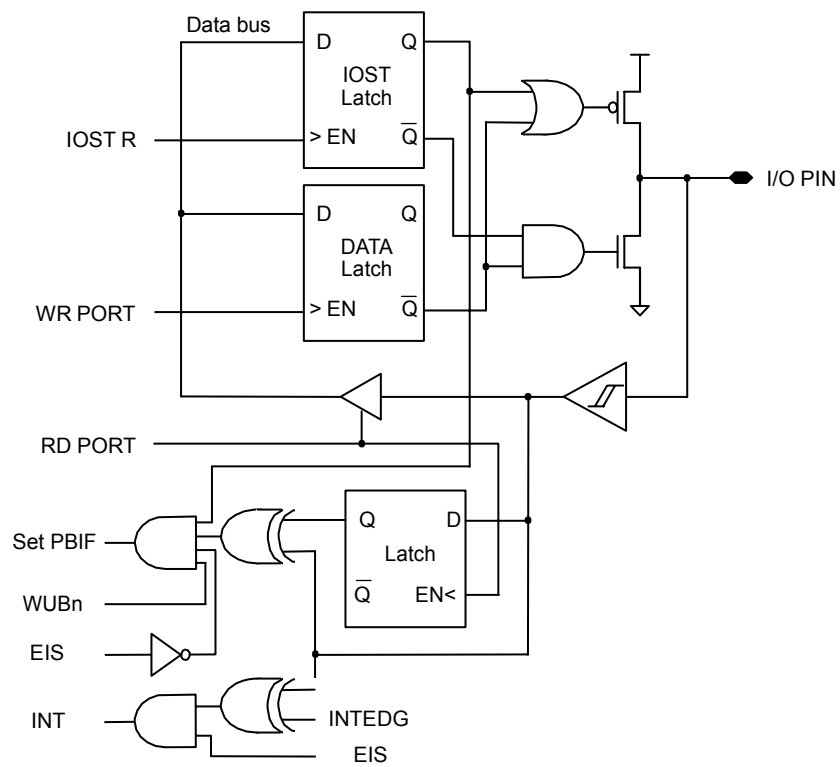
IOA3 ~ IOA0 :



Pull-down is not shown in the figure

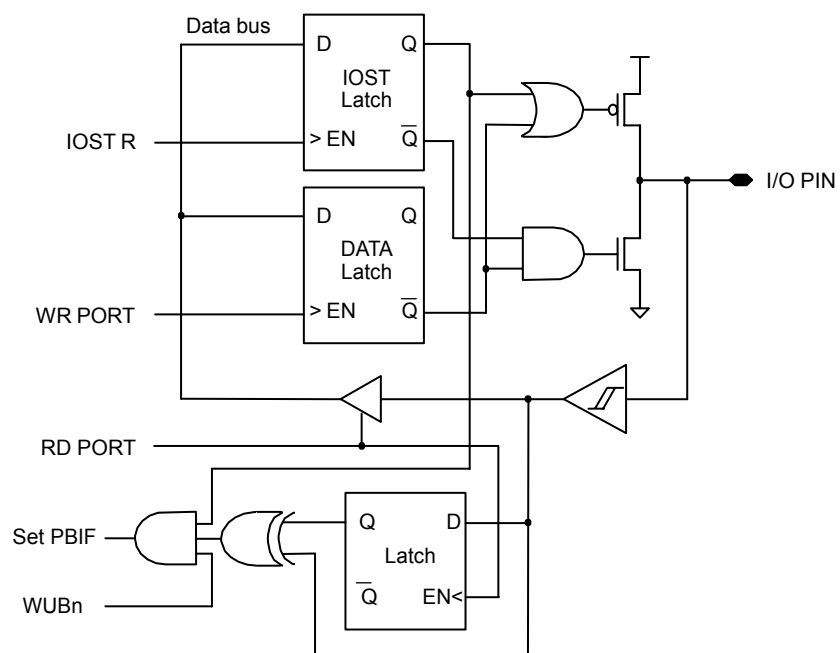
IOB3 :





Pull-high/pull-down and open-drain are not shown in the figure

IOB7 ~ IOB1 :



Pull-high/pull-down and open-drain are not shown in the figure

2.3 Timer0/WDT & Prescaler

2.3.1 Timer0

The Timer0 is a 8-bit timer/counter. The clock source of Timer0 can come from the internal clock or by an external clock source (T0CKI pin).

2.3.1.1 Using Timer0 with an Internal Clock : Timer mode

Timer mode is selected by clearing the T0CS bit (OPTION<5>). In timer mode, the timer0 register (TMR0) will increment every instruction cycle (without prescaler). If TMR0 register is written, the increment is inhibited for the following two cycles.

2.3.1.2 Using Timer0 with an External Clock : Counter mode

Counter mode is selected by setting the T0CS bit (OPTION<5>). In this mode, Timer0 will increment either on every rising or falling edge of pin T0CKI. The incrementing edge is determined by the source edge select bit T0SE (OPTION<4>).

The external clock requirement is due to internal phase clock (Tosc) synchronization. Also, there is a delay in the actual incrementing of Timer0 after synchronization.

When no prescaler is used, the external clock input is the same as the prescaler output. The synchronization of T0CKI with the internal phase clocks is accomplished by sampling the prescaler output on the T2 and T4 cycles of the internal phase clocks. Therefore, it is necessary for T0CKI to be high for at least 2 T_{osc} and low for at least 2 T_{osc}.

When a prescaler is used, the external clock input is divided by the asynchronous prescaler. For the external clock to meet the sampling requirement, the ripple counter must be taken into account. Therefore, it is necessary for T0CKI to have a period of at least 4 T_{osc} divided by the prescaler value.

2.3.2 Watchdog Timer (WDT)

The Watchdog Timer (WDT) is a free running on-chip RC oscillator which does not require any external components. So the WDT will still run even if the clock on the OSC1 and OSC0 pins is turned off, such as in SLEEP mode. During normal operation or in SLEEP mode, a WDT time-out will cause the device reset and the \overline{TO} bit (STATUS<4>) will be cleared.

The WDT can be disabled by clearing the control bit WDTE (PCON<7>) to "0". The WDT has a nominal time-out period of 18 ms, 4.5ms, 288ms or 72ms selected by SUT<1:0> bits of configuration word (without prescaler). If a longer time-out period is desired, a prescaler with a division ratio of up to 1:128 can be assigned to the WDT controlled by the OPTION register. Thus, the longest time-out period is approximately 36.8 seconds.

The CLRWDT instruction clears the WDT and the prescaler, if assigned to the WDT, and prevents it from timing out and generating a device reset.

The SLEEP instruction resets the WDT and the prescaler, if assigned to the WDT. This gives the maximum SLEEP time before a WDT Wake-up Reset.

2.3.3 Prescaler

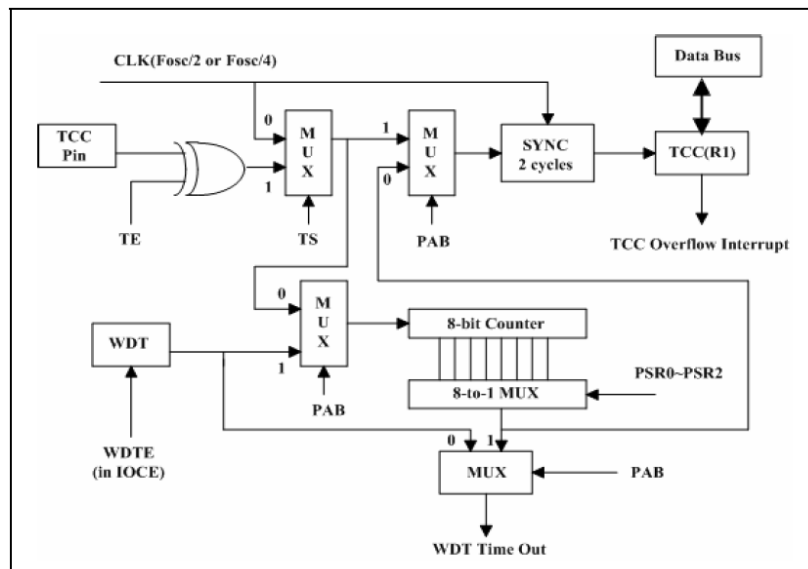
An 8-bit counter (down counter) is available as a prescaler for the Timer0, or as a postscaler for the Watchdog Timer (WDT).

Note that the prescaler may be used by either the Timer0 module or the WDT, but not both. Thus, a prescaler assignment for the Timer0 means that there is no prescaler for the WDT, and vice-versa. The PSA bit (OPTION<3>) determines prescaler assignment. The PS<2:0> bits (OPTION<2:0>) determine prescaler ratio.

When the prescaler is assigned to the Timer0 module, all instructions writing to the TMR0 register will clear the prescaler. When it is assigned to WDT, a CLRWDT instruction will clear the prescaler along with the WDT. The prescaler is neither readable nor writable. On a RESET, the prescaler contains all '1's.

To avoid an unintended device reset, CLRWDT or CLRR TMR0 instructions must be executed when changing the prescaler assignment from Timer0 to the WDT, and vice-versa.

FIGURE 2.4: Block Diagram of The Timer0/WDT Prescaler



2.4 Interrupts

The KXD8P53 has up to three sources of interrupt:

1. External interrupt INT pin.
2. TMR0 overflow interrupt.
3. IOB input change interrupt (pins IOB7:IOB0).

INTFLAG is the interrupt flag register that recodes the interrupt requests in the relative flags.

A global interrupt enable bit, GIE (INTEN<7>), enables (if set) all un-masked interrupts or disables (if cleared) all interrupts. Individual interrupts can be enabled/disabled through their corresponding enable bits in INTEN register regardless of the status of the GIE bit.

When an interrupt event occurs with the GIE bit and its corresponding interrupt enable bit are all set, the GIE bit will be cleared by hardware to disable any further interrupts, and the next instruction will be fetched from address 008h. The interrupt flag bits must be cleared by software before re-enabling GIE bit to avoid recursive interrupts. The RETFIE instruction exits the interrupt routine and sets the GIE bit to re-enable interrupt.

The flag bit (except PBIF bit) in INTFLAG register is set by interrupt event regardless of the status of its mask bit. Reading the INTFLAG register will be the logic AND of INTFLAG and INTEN. When an interrupt is generated by the INT instruction, the next instruction will be fetched from address 002h.

2.4.1 External INT Interrupt

External interrupt on INT pin is rising or falling edge triggered selected by INTEDG (OPTION<6>).

When a valid edge appears on the INT pin the flag bit INTIF (INTFLAG<2>) is set. This interrupt can be disabled by clearing INTIE bit (INTEN<2>).

The INT pin interrupt can wake-up the system from SLEEP condition, if bit INTIE was set before going to SLEEP. If GIE bit was set, the program will execute interrupt service routine after wake-up; or if GIE bit was cleared, the program will execute next PC after wake-up.

2.4.2 Timer0 Interrupt

An overflow (FFh --> 00h) in the TMR0 register will set the flag bit T0IF (INTFLAG<0>). This interrupt can be disabled by clearing T0IE bit (INTEN<0>).

2.4.3 IOB Input Change Interrupt

An input change on IOB<7:0> set flag bit PBIF (INTFLAG<1>). This interrupt can be disabled by clearing PBIE bit (INTEN<1>).

Before the IOB input change interrupt is enabled, reading IOB (any instruction accessed to IOB, including read/write instructions) is necessary.

Any pin which corresponding WUBn bit (WUCON<7:0>) is cleared to "0" or configured as output or IOB0 pin configured as INT pin will be excluded from this function.

The IOB input change interrupt also can wake-up the system from SLEEP condition, if bit PBIE was set before going to SLEEP. And GIE bit also decides whether or not the processor branches to the interrupt vector following wake-up.

If GIE bit was set, the program will execute interrupt service routine after wake-up; or if GIE bit was cleared, the program will execute next PC after wake-up.

2.5 Power-down Mode (SLEEP)

Power-down mode is entered by executing a SLEEP instruction.

When SLEEP instruction is executed, the \overline{PD} bit (STATUS<3>) is cleared, the \overline{TO} bit is set, the watchdog timer will be cleared and keeps running, and the oscillator driver is turned off. All I/O pins maintain the status they had before the SLEEP instruction was executed.

2.5.1 Wake-up from SLEEP Mode

The device can wake-up from SLEEP mode through one of the following events:

1. RSTB reset.
2. WDT time-out reset (if enabled).
3. Interrupt from RB0/INT pin, or PORTB change interrupt.

External RSTB reset and WDT time-out reset will cause a device reset. The \overline{PD} and \overline{TO} bits can be used to determine the cause of device reset. The \overline{PD} bit is set on power-up and is cleared when SLEEP instruction is executed. The \overline{TO} bit is cleared if a WDT time-out occurred.

For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be set. Wake-up is regardless of the GIE bit. If GIE bit is cleared, the device will continue execution at the instruction after the SLEEP instruction. If the GIE bit is set, the device will branch to the interrupt address (008h).

In HF or LF oscillation mode, the system wake-up delay time is 18/4.5/288/72ms (selected by SUT<1:0> bits of configuration word) plus 16 oscillator cycles time.

And in IRC/ERIC or ERC oscillation mode, the system wake-up delay time is 140us.

2.6 Reset

KXD8P53 device may be RESET in one of the following ways:

1. Power-on Reset (POR)
2. Brown-out Reset (BOR)
3. RSTB Pin Reset
4. WDT time-out Reset

Some registers are not affected in any RESET condition. Their status is unknown on Power-on Reset and unchanged in any other RESET. Most other registers are reset to a "reset state" on Power-on Reset, RSTB or WDT Reset.

A Power-on RESET pulse is generated on-chip when Vdd rise is detected. To use this feature, the user merely ties the RSTB pin to Vdd.

On-chip Low Voltage Detector (LVD) places the device into reset when Vdd is below a fixed voltage. This ensures that the device does not continue program execution outside the valid operation Vdd range. Brown-out RESET is typically used in AC line or heavy loads switched applications.

A RSTB or WDT Wake-up from SLEEP also results in a device RESET, and not a continuation of operation before SLEEP. The \overline{TO} and \overline{PD} bits (STATUS<4:3>) are set or cleared depending on the different reset conditions.

2.6.1 Power-up Reset Timer(PWRT)

The Power-up Reset Timer provides a nominal 18/4.5/288/72ms (selected by SUT<1:0> bits of configuration word) (or 140us, varies based on oscillator selection and reset condition) delay after Power-on Reset (POR), Brown-out Reset (BOR), RSTB Reset or WDT time-out Reset. The device is kept in reset state as long as the PWRT is active. The PWRT delay will vary from device to device due to Vdd, temperature, and process variation.

TABLE 2.1: PWRT Period

Oscillator Mode	Power-on Reset Brown-out Reset	RSTB Reset WDT time-out Reset
ERIC & IRC	18/4.5/288/72 ms	18 ms
HF & LF	18/4.5/288/72 ms	18/4.5/288/72ms

2.6.2 Oscillator Start-up Timer(OST)

The OST timer provides a 16 oscillator cycle delay (from OSCI input) after the PWRT delay (18/4.5/288/72ms) is over in HF or LF oscillation mode. This delay ensures that the X'tal oscillator or resonator has started and stabilized. The device is kept in reset state as long as the OST is active. This counter only starts incrementing after the amplitude of the OSCI signal reaches the oscillator input thresholds.

2.6.3 Reset Sequence

When Power-on Reset (POR), Brown-out Reset (BOR), RSTB Reset or WDT time-out Reset is detected, the reset sequence is as follows:

1. The reset latch is set and the PWRT & OST are cleared.
2. When the internal POR, BOR, RSTB Reset or WDT time-out Reset pulse is finished, then the PWRT begins counting.
3. After the PWRT time-out, the OST is activated.
4. And after the OST delay is over, the reset latch will be cleared and thus end the on-chip reset signal.

In HF or LF oscillation mode, the totally system reset delay time is 18/4.5/288/72ms plus 16 oscillator cycle time. And in IRC/ERIC or ERC oscillation mode, the totally system reset delay time is 18/4.5/288/72ms after Power-on Reset (POR), Brown-out Reset (BOR), or 140us after RSTB Reset or WDT time-out Reset.

FIGURE 2.5: Simplified Block Diagram of on-chip Reset Circuit

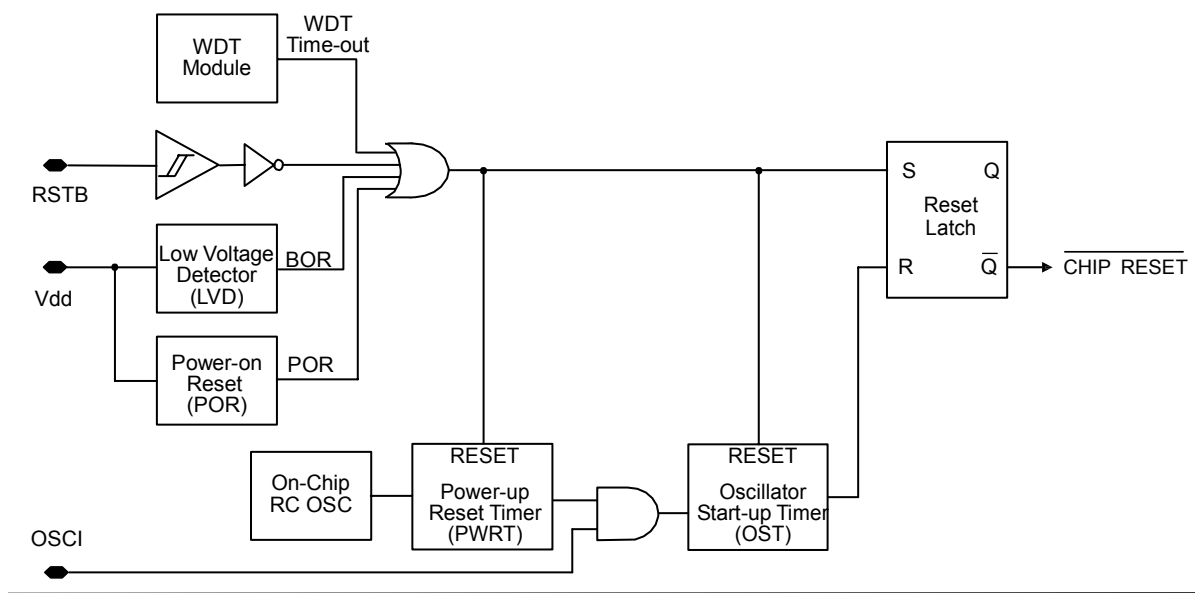


TABLE 2.2: Reset Conditions for All Registers

Register	Address	Power-on Reset Brown-out Reset	RSTB Reset WDT Reset
ACC	N/A	xxxx xxxx	uuuu uuuu
OPTION	N/A	-011 1111	-011 1111
IOSTA	N/A	---- 1111	---- 1111
IOSTB	N/A	1111 1111	1111 1111
INDF	00h	xxxx xxxx	uuuu uuuu
TMR0	01h	xxxx xxxx	uuuu uuuu
PCL	02h	1111 1111	1111 1111
STATUS	03h	0001 1xxx	000# #uuu
FSR	04h	11xx xxxx	11uu uuuu
PORTA	05h	xxxx xxxx	uuuu uuuu
PORTB	06h	xxxx xxxx	uuuu uuuu
General Purpose Register	07h	xxxx xxxx	uuuu uuuu
PCON	08h	101- ----	101- ----
WUCON	09h	0000 0000	0000 0000
PCHBUF	0Ah	---- --00	---- --00
PDCON	0Bh	1111 1111	1111 1111
ODCON	0Ch	0000 0000	0000 0000
PHCON	0Dh	1111 1111	1111 1111
INTEN	0Eh	0--- -000	0--- -000
INTFLAG	0Fh	---- -000	---- -000
General Purpose Registers	10 ~ 3Fh	xxxx xxxx	uuuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented,
= refer to the following table for possible values.

TABLE 2.3: RST/TO/PD Status after Reset or Wake-up

RST	TO	PD	RESET was caused by
0	1	1	Power-on Reset
0	1	1	Brown-out reset
0	u	u	RSTB Reset during normal operation
0	1	0	RSTB Reset during SLEEP
0	0	1	WDT Reset during normal operation
0	0	0	WDT Wake-up during SLEEP
1	1	0	Wake-up on pin change during SLEEP

Legend: u = unchanged

TABLE 2.4: Events Affecting TO/PD Status Bits

Event	TO	PD
Power-on	1	1
WDT Time-Out	0	u
SLEEP instruction	1	0
CLRWDT instruction	1	1

Legend: u = unchanged

2.7 Hexadecimal Convert to Decimal (HCD)

Decimal format is another number format for KXD8P53. When the content of the data memory has been assigned as decimal format, it is necessary to convert the results to decimal format after the execution of ALU instructions.

When the decimal converting operation is processing, all of the operand data (including the contents of the data memory (RAM), accumulator (ACC), immediate data, and look-up table) should be in the decimal format, or the results of conversion will be incorrect.

Instruction DAA can convert the ACC data from hexadecimal to decimal format after any addition operation and restored to ACC. The conversion operation is illustrated in example 2.2.

EXAMPLE 2.2: DAA CONVERSION

MOVIA	90h	;Set immediate data = decimal format number "90" (ACC Å 90h)
MOVAR	30h	;Load immediate data "90" to data memory address 30H
MOVIA	10h	;Set immediate data = decimal format number "10" (ACC Å 10h)
ADDAR	30h, 0	;Contents of the data memory address 30H and ACC are binary-added ;the result loads to the ACC (ACC Å A0h, C Å 0)
DAA		;Convert the content of ACC to decimal format, and restored to ACC ;The result in the ACC is "00" and the carry bit C is "1". This represents the ;decimal number "100"

Instruction DAS can convert the ACC data from hexadecimal to decimal format after any subtraction operation and restored to ACC. The conversion operation is illustrated in example 2.3.

EXAMPLE 2.3: DAS CONVERSION

MOVIA	10H	;Set immediate data = decimal format number "10" (ACC Å 10h)
MOVAR	30H	;Load immediate data "10" to data memory address 30H
MOVIA	20H	;Set immediate data = decimal format number "20" (ACC Å 20h)
SUBAR	30H,0	;Contents of the data memory address 30H and ACC are binary-subtracted ;the result loads to the ACC (ACC Å F0h, C Å 0)
DAS		;Convert the content of ACC to decimal format, and restored to ACC ;The result in the ACC is "90" and the carry bit C is "0". This represents the ;decimal number " -10"

2.8 Oscillator Configurations

H53 can be operated in six different oscillator modes. Users can program three configuration bits ($Fosc<2:0>$) to select the appropriate modes:

- ERIC: External RC Oscillator
- HF: High Frequency Crystal/Resonator Oscillator
- LF: Low Frequency Crystal Oscillator
- IRC: Internal RC Oscillator

In LF, XT, or HF modes, a crystal or ceramic resonator is connected to the OSCI and OSCO pins to establish oscillation. When in LF, or HF modes, the device can have an external clock source drive the OSCI pin. The ERIC device option offers additional cost savings for timing insensitive applications. The RC oscillator frequency is a function of the resistor (R_{ext}), the operating temperature, and the process parameter.

The IRC/ERIC device option offers largest cost savings for timing insensitive applications. These devices offer 4 different internal RC oscillator frequency, 8MHz, 4MHz, 1MHz, and 500KHz, which is selected by two configuration bits ($RCM<1:0>$). Or user can change the oscillator frequency with external resistor. The ERC oscillator frequency is a function of the resistor (R_{ext}), the operating temperature, and the process parameter.

FIGURE 2.6: HF, or LF Oscillator Modes (Crystal Operation or Ceramic Resonator)

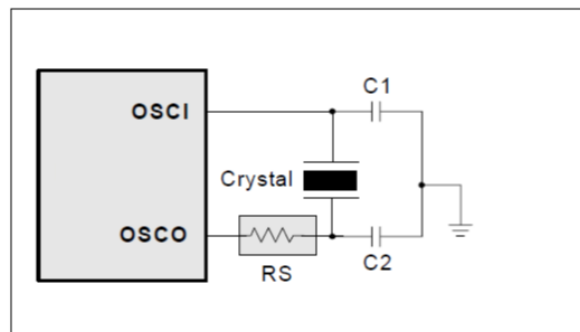


FIGURE 2.7: HF, or LF Oscillator Modes (External Clock Input Operation)

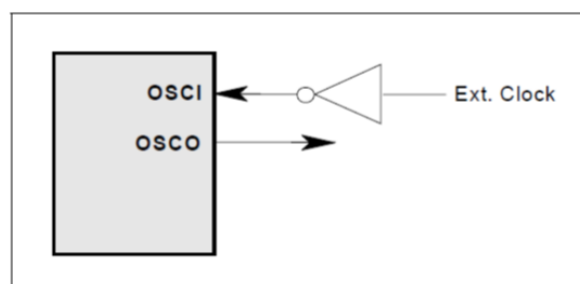
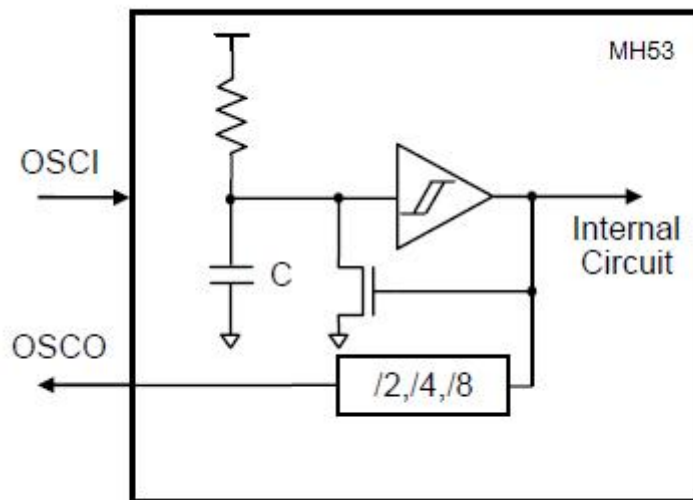
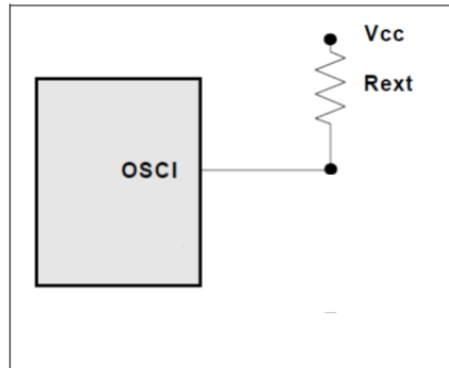


FIGURE 2.8: ERIC Oscillator Mode (External R, Internal C Oscillator)



3.0 INSTRUCTION SET

Mnemonic, Operands	Description	Operation	Cycles	Status Affected
BCR R, bit	Clear bit in R	$0 \rightarrow R$	1	-
BSR R, bit	Set bit in R	$1 \rightarrow R$	1	-
BTRSC R, bit	Test bit in R, Skip if Clear	Skip if $R = 0$	$1/2^{(1)}$	-
BTRSS R, bit	Test bit in R, Skip if Set	Skip if $R = 1$	$1/2^{(1)}$	-
NOP	No Operation	No operation	1	-
CLRWDT	Clear Watchdog Timer	$00h \rightarrow$ WDT, $00h \rightarrow$ WDT prescaler	1	TO, PD
OPTION	Load OPTION register	$ACC \rightarrow$ OPTION	1	-
SLEEP	Go into power-down mode	$00h \rightarrow$ WDT, $00h \rightarrow$ WDT prescaler	1	TO, PD
IOST R	Load IOST register	$ACC \rightarrow$ IOST register	1	-
DAA	Adjust ACC's data format from HEX to DEC after any addition operation	$ACC(hex) \rightarrow ACC(dec)$	1	C
DAS	Adjust ACC's data format from HEX to DEC after any subtraction operation	$ACC(hex) \rightarrow ACC(dec)$	1	-
INT	S/W interrupt	$PC + 1 \rightarrow$ Top of Stack, $002h \rightarrow PC$	2	-
RETURN	Return from subroutine	Top of Stack $\rightarrow PC$	2	-
RETFIE	Return from interrupt, set GIE bit	Top of Stack $\rightarrow PC$, $1 \rightarrow GIE$	2	-
CLRA	Clear ACC	$00h \rightarrow ACC$	1	Z
CLRR R	Clear R	$00h \rightarrow R$	1	Z
MOVAR R	Move ACC to R	$ACC \rightarrow R$	1	-
MOVR R, d	Move R	$R \rightarrow$ dest	1	Z
DECR R, d	Decrement R	$R - 1 \rightarrow$ dest	1	Z
DECRSZ R, d	Decrement R, Skip if 0	$R - 1 \rightarrow$ dest, Skip if result = 0	$1/2^{(1)}$	-
INCR R, d	Increment R	$R + 1 \rightarrow$ dest	1	Z
INCRSZ R, d	Increment R, Skip if 0	$R + 1 \rightarrow$ dest, Skip if result = 0	$1/2^{(1)}$	-
ADDAR R, d	Add ACC and R	$R + ACC \rightarrow$ dest	1	C, DC, Z
SUBAR R, d	Subtract ACC from R	$R - ACC \rightarrow$ dest	1	C, DC, Z
ADCAR R, d	Add ACC and R with Carry	$R + ACC + C \rightarrow$ dest	1	C, DC, Z
SBCAR R, d	Subtract ACC from R with Carry	$R + ACC + C \rightarrow$ dest	1	C, DC, Z
ANDAR R, d	AND ACC with R	ACC and $R \rightarrow$ dest	1	Z
IORAR R, d	Inclusive OR ACC with R	ACC or $R \rightarrow$ dest	1	Z
XORAR R, d	Exclusive OR ACC with R	R xor $ACC \rightarrow$ dest	1	Z
COMR R, d	Complement R	$R \rightarrow$ dest	1	Z
RLR R, d	Rotate left R through Carry	$R<7> \rightarrow C$, $R<6:0> \rightarrow$ dest<7:1>, $C \rightarrow$ dest<0>	1	C

Mnemonic, Operands	Description	Operation	Cycles	Status Affected
RRR R, d	Rotate right R through Carry	$C \rightarrow \text{dest}<7>$, $R<7:1> \rightarrow \text{dest}<6:0>$, $R<0> \rightarrow C$	1	C
SWAPR R, d	Swap R	$R<3:0> \rightarrow \text{dest}<7:4>$, $R<7:4> \rightarrow \text{dest}<3:0>$	1	-
MOVIA I	Move Immediate to ACC	$I \rightarrow \text{ACC}$	1	-
ADDIA I	Add ACC and Immediate	$I + \text{ACC} \rightarrow \text{ACC}$	1	C, DC, Z
SUBIA I	Subtract ACC from Immediate	$I - \text{ACC} \rightarrow \text{ACC}$	1	C, DC, Z
ANDIA I	AND Immediate with ACC	$\text{ACC and } I \rightarrow \text{ACC}$	1	Z
IORIA I	OR Immediate with ACC	$\text{ACC or } I \rightarrow \text{ACC}$	1	Z
XORIA I	Exclusive OR Immediate to ACC	$\text{ACC xor } I \rightarrow \text{ACC}$	1	Z
RETIA I	Return, place Immediate in ACC	$I \rightarrow \text{ACC}$, Top of Stack $\rightarrow \text{PC}$	2	-
CALL I	Call subroutine	$\text{PC} + 1 \rightarrow \text{Top of Stack}$, $I \rightarrow \text{PC}$	2	-
GOTO I	Unconditional branch	$I \rightarrow \text{PC}$	2	-

Note: 1. 2 cycles for skip, else 1 cycle

2. bit : Bit address within an 8-bit register R R :

Register address (00h to 3Fh)

I : Immediate data

ACC : Accumulator

d : Destination select;

=0 (store result in ACC)

=1 (store result in file register R)

dest : Destination

PC : Program Counter

PCHBUF : High Byte Buffer of Program Counter

WDT : Watchdog Timer Counter GIE : Global

interrupt enable bit

TO : Time-out bit

PD : Power-down bit

C : Carry bit

DC : Digital carry bit

Z : Zero bit

ADCAR	Add ACC and R with Carry
Syntax:	ADCAR R, d
Operands:	$0 \leq R \leq 63$ $d \in [0, 1]$
Operation:	$R + ACC + C \rightarrow \text{dest}$
Status Affected:	C, DC, Z
Description:	Add the contents of the ACC register and register 'R' with Carry. If 'd' is 0 the result is stored in the ACC register. If 'd' is '1' the result is stored back in register 'R'.
Cycles:	1
ADDAR	Add ACC and R
Syntax:	ADDAR R, d
Operands:	$0 \leq R \leq 63$ $d \in [0, 1]$
Operation:	$ACC + R \rightarrow \text{dest}$
Status Affected:	C, DC, Z
Description:	Add the contents of the ACC register and register 'R'. If 'd' is 0 the result is stored in the ACC register. If 'd' is '1' the result is stored back in register 'R'.
Cycles:	1
ADDIA	Add ACC and Immediate
Syntax:	ADDIA I
Operands:	$0 \leq I \leq 255$
Operation:	$ACC + I \rightarrow ACC$
Status Affected:	C, DC, Z
Description:	Add the contents of the ACC register with the 8-bit immediate 'I'. The result is placed in the ACC register.
Cycles:	1
ANDAR	AND ACC and R
Syntax:	ANDAR R, d
Operands:	$0 \leq R \leq 63$ $d \in [0, 1]$
Operation:	$ACC \text{ and } R \rightarrow \text{dest}$
Status Affected:	Z
Description:	The contents of the ACC register are AND'ed with register 'R'. If 'd' is 0 the result is stored in the ACC register. If 'd' is '1' the result is stored back in register 'R'.
Cycles:	1
ANDIA	AND Immediate with ACC
Syntax:	ANDIA I
Operands:	$0 \leq I \leq 255$
Operation:	$ACC \text{ AND } I \rightarrow ACC$
Status Affected:	Z
Description:	The contents of the ACC register are AND'ed with the 8-bit immediate 'I'. The result is placed in the ACC register.
Cycles:	1

BCR	Clear Bit in R
Syntax:	BCF R, b
Operands:	$0 \leq R \leq 63$ $0 \leq b \leq 7$
Operation:	$0 \rightarrow R[b] \text{ Status}$
Affected:	None
Description:	Clear bit 'b' in register 'R'.
Cycles:	1
BSR	Set Bit in R
Syntax:	BSR R, b
Operands:	$0 \leq R \leq 63$ $0 \leq b \leq 7$
Operation:	$1 \rightarrow R[b] \text{ Status}$
Affected:	None
Description:	Set bit 'b' in register 'R'.
Cycles:	1
BTRSC	Test Bit in R, Skip if Clear
Syntax:	BTRSC R, b
Operands:	$0 \leq R \leq 63$ $0 \leq b \leq 7$
Operation:	Skip if $R[b] = 0$
Status Affected:	None
Description:	If bit 'b' in register 'R' is 0 then the next instruction is skipped. If bit 'b' is 0 then next instruction fetched during the current instruction execution is discarded, and a NOP is executed instead making this a 2-cycle instruction.
Cycles:	1(2)
BTRSS	Test Bit in R, Skip if Set
Syntax:	BTRSS R, b
Operands:	$0 \leq R \leq 63$ $0 \leq b \leq 7$
Operation:	Skip if $R[b] = 1$
Status Affected:	None
Description:	If bit 'b' in register 'R' is '1' then the next instruction is skipped. If bit 'b' is '1', then the next instruction fetched during the current instruction execution, is discarded and a NOP is executed instead, making this a 2-cycle instruction.
Cycles:	1(2)
CALL	Subroutine Call
Syntax:	CALL I
Operands:	$0 \leq I \leq 1023$
Operation:	$PC + 1 \rightarrow \text{Top of Stack};$ $I \rightarrow PC$
Status Affected:	None
Description:	Subroutine call. First, return address (PC+1) is pushed onto the stack. The 10-bit immediate address is loaded into PC bits <9:0>. CALL is a two-cycle instruction.
Cycles:	2

CLRA	Clear ACC
Syntax:	CLRA Operands:
None Operation:	00h → ACC; 1 → Z
Status Affected:	Z
Description:	The ACC register is cleared. Zero bit (Z) is set.
Cycles:	1
CLRR	Clear R
Syntax:	CLRR R
Operands:	0 ≤ R ≤ 63
Operation:	00h → R; 1 → Z
Status Affected:	Z
Description:	The contents of register 'R' are cleared and the Z bit is set.
Cycles:	1
CLRWDT	Clear Watchdog Timer
Syntax:	CLRWDT
Operands:	None
Operation:	00h → WDT; 00h → WDT prescaler (if assigned); 1 → TO ; 1 → PD
Status Affected:	TO , PD
Description:	The CLRWDT instruction resets the WDT. It also resets the prescaler, if the prescaler is assigned to the WDT and not Timer0. Status bits TO and PD are set.
Cycles:	1
COMR	Complement R
Syntax:	COMR R, d
Operands:	0 ≤ R ≤ 63 d ∈ [0, 1]
Operation:	R → dest
Status Affected:	Z
Description:	The contents of register 'R' are complemented. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1
DAA	Adjust ACC's data format from HEX to DEC
Syntax:	DAA
Operands:	None
Operation:	ACC(hex) → ACC(dec)
Status Affected:	C
Description:	Convert the ACC data from hexadecimal to decimal format after any addition operation and restored to ACC.
Cycles:	1

DAS	Adjust ACC's data format from HEX to DEC
Syntax:	DAS
Operands:	None
Operation:	ACC(hex)→ACC(dec)
Status Affected:	None
	Description: Convert the ACC data from hexadecimal to decimal format after any subtraction operation and restored to ACC.
Cycles:	1
DECR	Decrement R
Syntax:	DECR R, d
Operands:	$0 \leq R \leq 63$ $d \in [0,1]$
Operation:	$R - 1 \rightarrow \text{dest}$
Status Affected:	Z
Description:	Decrement register 'R'. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1
DECRSZ	Decrement R, Skip if 0
Syntax:	DECRSZ R, d
Operands:	$0 \leq R \leq 63$ $d \in [0,1]$
Operation:	$R - 1 \rightarrow \text{dest}$; skip if result =0
Status Affected:	None
Description:	The contents of register 'R' are decremented. If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is placed back in register 'R'. If the result is 0, the next instruction, which is already fetched, is discarded and a NOP is executed instead making it a two-cycle instruction.
Cycles:	1(2)
GOTO	Unconditional Branch
Syntax:	GOTO I
Operands:	$0 \leq I \leq 1023$
Operation:	$I \rightarrow \text{PC}$
Status Affected:	None
Description:	GOTO is an unconditional branch. The 10-bit immediate value is loaded into PC bits <9:0>. GOTO is a two-cycle instruction.
Cycles:	2
INCR	Increment R
Syntax:	INCR R, d
Operands:	$0 \leq R \leq 63$ $d \in [0,1]$
Operation:	$R + 1 \rightarrow \text{dest}$
Status Affected:	Z
Description:	The contents of register 'R' are incremented. If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is placed back in register 'R'.
Cycles:	1

INCRSZ Increment R, Skip if 0

Syntax: INCRSZ R, d
Operands: $0 \leq R \leq 63$
 $d \in [0,1]$
Operation: $R + 1 \rightarrow \text{dest}$, skip if result = 0
Status Affected: None
Description: The contents of register 'R' are incremented. If 'd' is 0 the result is placed in the ACC register.
 If 'd' is the result is placed back in register 'R'.
 If the result is 0, then the next instruction, which is already fetched, is discarded and a NOP is
 executed instead making it a two-cycle instruction.
Cycles: 1(2)

INT S/W Interrupt

Syntax: INT
Operands: None
Operation: $PC + 1 \rightarrow \text{Top of Stack}$,
 $002h \rightarrow PC$
Status Affected: None
Description: Interrupt subroutine call. First, return address (PC+1) is pushed onto the stack. The address
 002h is loaded into PC bits <10:0>.
Cycles: 2

IORAR OR ACC with R

Syntax: IORAR R, d
Operands: $0 \leq R \leq 63$
 $d \in [0,1]$
Operation: $ACC \text{ or } R \rightarrow \text{dest}$
Status Affected: Z
Description: Inclusive OR the ACC register with register 'R'. If 'd' is 0 the result is placed in the ACC
 register. If 'd' is 1 the result is placed back in register 'R'.
Cycles: 1

IORIA OR Immediate with ACC

Syntax: IORIA I
Operands: $0 \leq I \leq 255$
Operation: $ACC \text{ or } I \rightarrow ACC$
Status Affected: Z
Description: The contents of the ACC register are OR'ed with the 8-bit immediate 'I'. The result is placed in
 the ACC register.
Cycles: 1

IOST Load IOST Register

Syntax: IOST R
Operands: R = 5 or 6
Operation: $ACC \rightarrow \text{IOST register R}$
Status Affected: None
Description: IOST register 'R' (R = 5 or 6) is loaded with the contents of the ACC register.
Cycles: 1

MOVAR	Move ACC to R
Syntax:	MOVAR R
Operands:	$0 \leq R \leq 63$
Operation:	ACC \rightarrow R
Status Affected:	None
Description:	Move data from the ACC register to register 'R'.
Cycles:	1
MOVIA	Move Immediate to ACC
Syntax:	MOVIA I
Operands:	$0 \leq I \leq 255$
Operation:	I \rightarrow ACC
Status Affected:	None
Description:	The 8-bit immediate 'I' is loaded into the ACC register. The don't cares will assemble as 0s.
Cycles:	1
MOVR	Move R
Syntax:	MOVR R, d
Operands:	$0 \leq R \leq 63$ $d \in [0,1]$
Operation:	R \rightarrow dest
Status Affected:	Z
Description:	The contents of register 'R' is moved to destination 'd'. If 'd' is 0, destination is the ACC register. If 'd' is 1, the destination is file register 'R'. 'd' is 1 is useful to test a file register since status flag Z is affected.
Cycles:	1
NOP	No Operation
Syntax:	NOP
Operands:	None
Operation:	No operation
Status Affected:	None
Description:	No operation.
Cycles:	1
OPTION	Load OPTION Register
Syntax:	OPTION
Operands:	None
Operation:	ACC \rightarrow OPTION
Status Affected:	None
Description:	The content of the ACC register is loaded into the OPTION register.
Cycles:	1
RETFIE	Return from Interrupt, Set 'GIE' Bit
Syntax:	RETFIE
Operands:	None
Operation:	Top of Stack \rightarrow PC
Status Affected:	None
Description:	The program counter is loaded from the top of the stack (the return address). The 'GIE' bit is

Cycles: set to 1. This is a two-cycle instruction.
2

RETIA Return with Immediate in ACC

Syntax: RETIA I
Operands: $0 \leq I \leq 255$
Operation: $I \rightarrow \text{ACC};$
Top of Stack $\rightarrow \text{PC}$
Status Affected: None
Description: The ACC register is loaded with the 8-bit immediate 'I'. The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.
Cycles: 2

RETURN Return from Subroutine

Syntax: RETURN
Operands: None
Operation: Top of Stack $\rightarrow \text{PC}$
Status Affected: None
Description: The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.
Cycles: 2

RLR Rotate Left R through Carry

Syntax: RLR R, d
Operands: $0 \leq R \leq 63$
 $d \in [0,1]$
Operation: $R<7> \rightarrow C;$
 $R<6:0> \rightarrow \text{dest}<7:1>;$
 $C \rightarrow \text{dest}<0>$
Status Affected: C
Description: The contents of register 'R' are rotated one bit to the left through the Carry Flag. If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles: 1

RRR Rotate Right R through Carry

Syntax: RRR R, d
Operands: $0 \leq R \leq 63$
 $d \in [0,1]$
Operation: $C \rightarrow \text{dest}<7>;$
 $R<7:1> \rightarrow \text{dest}<6:0>;$
 $R<0> \rightarrow C$
Status Affected: C
Description: The contents of register 'R' are rotated one bit to the right through the Carry Flag. If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is placed back in register 'R'.
Cycles: 1

SLEEP	Enter SLEEP Mode
Syntax:	SLEEP
Operands:	None
Operation:	00h-->WDT; 00h→WDT prescaler; 1→ TO ; 0→ PD
Status Affected:	TO , PD
Description:	Time-out status bit (TO) is set. The power-down status bit (PD) is cleared. The WDT and its prescaler are cleared. The processor is put into SLEEP mode.
Cycles:	1
SBCAR	Subtract ACC from R with Carry
Syntax:	SBCAR R, d
Operands:	$0 \leq R \leq 63$ $d \in [0,1]$
Operation:	$R + ACC + C \rightarrow \text{dest}$
Status Affected:	C, DC, Z
Description:	Add the 2's complement data of the ACC register from register 'R' with Carry. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1
SUBAR	Subtract ACC from R
Syntax:	SUBAR R, d
Operands:	$0 \leq R \leq 63$ $d \in [0,1]$
Operation:	$R - ACC \rightarrow \text{dest}$
Status Affected:	C, DC, Z
Description:	Subtract (2's complement method) the ACC register from register 'R'. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1
SUBIA	Subtract ACC from Immediate
Syntax:	SUBIA I
Operands:	$0 \leq I \leq 255$
Operation:	$I - ACC \rightarrow ACC$
Status Affected:	C, DC, Z
Description:	Subtract (2's complement method) the ACC register from the 8-bit immediate 'I'. The result is placed in the ACC register.
Cycles:	1
SWAPR	Swap nibbles in R
Syntax:	SWAPR R, d
Operands:	$0 \leq R \leq 63$ $d \in [0,1]$
Operation:	$R<3:0> \rightarrow \text{dest}<7:4>;$ $R<7:4> \rightarrow \text{dest}<3:0>$
Status Affected:	None
Description:	The upper and lower nibbles of register 'R' are exchanged. If 'd' is 0 the result is placed in

Cycles:	ACC register. If 'd' is 1 the result is placed in register 'R'. 1
---------	--

XORAR	Exclusive OR ACC with R
--------------	--------------------------------

Syntax:	XORAR R, d
Operands:	$0 \leq R \leq 63$ $d \in [0,1]$
Operation:	ACC xor R \rightarrow dest
Status Affected:	Z
Description:	Exclusive OR the contents of the ACC register with register 'R'. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

XORIA	Exclusive OR Immediate with ACC
--------------	--

Syntax:	XORIA I
Operands:	$0 \leq I \leq 255$
Operation:	ACC xor I \rightarrow ACC
Status Affected:	Z
Description:	The contents of the ACC register are XOR'ed with the 8-bit immediate 'I'. The result is placed in the ACC register.
Cycles:	1

4.0 ABSOLUTE MAXIMUM RATINGS

DC Supply Voltage (Vdd)	0V to +6.0V
Input Voltage with respect to Ground (Vss)	-0.3V to (Vdd + 0.3)V
Ambient Operating Temperature	0°C to +70°C
Store Temperature	-65°C to +150°C

5.0 OPERATING CONDITIONS

DC Supply Voltage	+2.2V to +5.5V
Operating Temperature	0°C to +70°C

6.0 ELECTRICAL CHARACTERISTICS

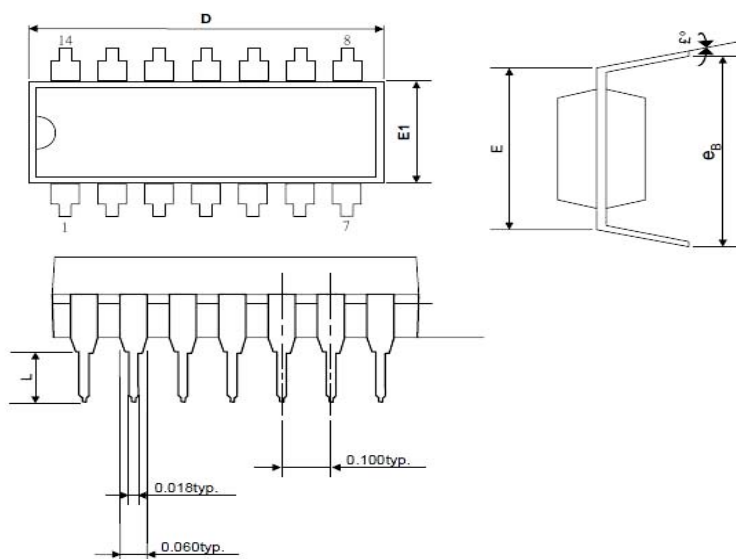
6.1 ELECTRICAL CHARACTERISTICS of KXD8P53

Under Operating Conditions, at four clock instruction cycles and WDT & LVDT are disabled

Symbol	Parameter	Condition	Min.	Typ.	Max.	Unit
HF	Crystal: Vdd to 2.2.V	4 clocks	DC	-	8.0	MHz
	Crystal: Vdd to 3.0 V	4 clocks	DC	-	20.0	MHz
	Crystal: Vdd to 5.0 V	4 clocks	DC	-	20.0	MHz
ERIC	ERIC: Vdd = 3.0 V	R: 390 K-ohm	-	1	-	MHz
	ERIC: Vdd = 3.0 V	R: 196 K-ohm	-	4	-	MHz
	ERIC: Vdd = 3.0 V	R: 45 K-ohm	-	8	-	MHz
IIL	Input leakage current for input pins	IOA, IOB	-	-	1	uA
VIHT1	Input High Voltage (Vdd = 5.0V)	IOA, IOB	2.0	-	-	V
VILT1	Input Low Voltage (Vdd = 5.0V)	IOA, IOB	-	-	0.8	V
IPH	Pull-high current	IOB pull-high active, Input pin at Vss	25	50	150	uA
IPD	Pull-down current	IOA, IOB pull-down active Input pin at Vdd	40	70	200	uA
VOH1	Output high voltage (Vdd=3.0V)	IOH = 7.5 mA	2.4			V
VOL1	Output low voltage (Vdd=3.0V)	IOL = 8.5 mA			0.4	V
VOH2	Output high voltage (Vdd=5.0V)	IOH = 25 mA	3.6			V
VOL2	Output low voltage (Vdd=5.0V)	IOL = 18 mA			0.6	V
Isb	Power down current	All input pin pull-down, All output pin floating, WDT disabled	-	-	1	uA

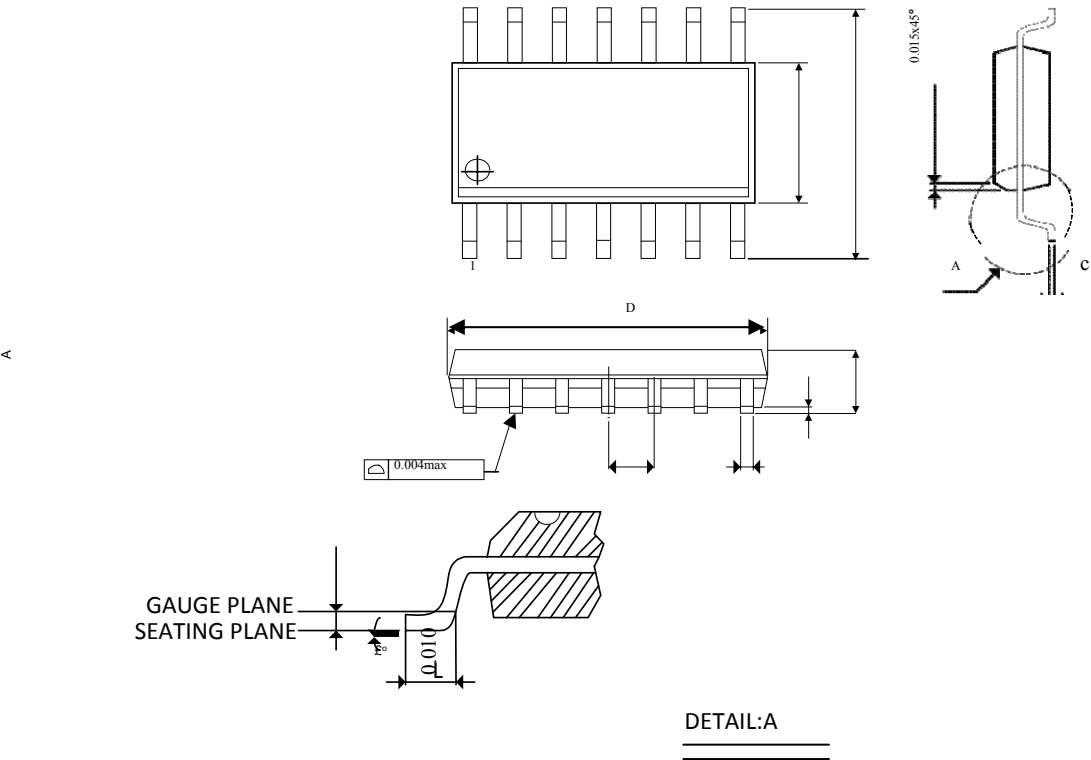
7.0 PACKAGE DIMENSION

7.1 14-PIN PDIP 300mil



Symbols	Dimension In Inches		
	Min	Nom	Max
A	-	-	0.210
A1	0.015	-	-
A2	0.125	0.130	0.135
D	0.735	0.750	0.775
E	0.300 BSC.		
E1	0.245	0.250	0.255
L	0.115	0.130	0.150
eB	0.335	0.355	0.375
θ°	0°	7°	15°

7.2 14-PIN SOP 150mil



Symbols	Dimension In Inches		
	Min	Nom	Max
A	0.058	0.064	0.068
A1	0.004	-	0.010
B	0.013	0.016	0.020
C	0.0075	0.008	0.0098
D	0.336	0.341	0.344
E	0.150	0.154	0.157
e	-	0.050	-
H	0.228	0.236	0.244
L	0.015	0.025	0.050
θ°	0°	-	8°