1. General Description

KL5C80A12CFP is a 8-bit high speed microcontroller developed with the state-of-the-art 0.8μm CMOS process. KL5C80A12CFP is compatible. In this data sheet. As CPU core it uses KC82, which is binary compatible with Zilog's Z80, and contains MMU to expand address space. The KC82 is a next generation 8-bit CPU core which executes instructions about four times faster than the Z80 (10MHz) and offers higher performance than typical 16-bit CPUs. In addition to the CPU core, KL5C80A12 contains a 512-byte high speed RAM, an interrupt controller, five16-bit high performance timer/counters, a synchronous/synchronous serial communication interface (USART)and 40 parallel ports, providing all requirements for high performance and compact system. It has also low power dissipation and is suitable for application to portable devices.

Features

1) Binary compatible with the Z80 CPU,4 times faster than Z80 at the same clock rate
2) Built-in MMU which expands address space to 512KB
3) High speed USART (Universal Synchronous Asynchronous Receiver Transmitter)
4) Two high performance 16-bit timer/counters
5) Three high performance 16-bit timer/counters with 8-bit prescaler
6) Eight internal interrupts and eight external interrupts (flexible priority), one non-maskable interrupt
7) 40 parallel ports
8) On-chip 512-byte high speed RAM
9) On-chip external memory chip select circuit
10) On-chip crystal oscillator buffer
11) Maximum operating frequency 10MHz
12) Low power dissipation

The KL5C8012's CPU core (KC82) uses synchronous system bus. Its internal I/Os and high speed RAMs include a bus interface suitable for the KC82's synchronous bus so that they are connected directly. When connecting I/O and memory to KL5C8012, it is required to convert KC82's synchronous bus signals to asynchronous bus signals which can be input to ordinary memories. For this purpose KL5C8012 contains a circuit (external bus interface unit), which outputs Read/Write signals (EMRD_, EMWR_,EIORD_, EIOWR_) appropriate for external asynchronous buses. Unlike Read/Write signals of internal bus cycle on the synchronous bus, these Read/Write signals of external bus cycle are strobe signals which can be directly connected to external memories. In this Manual the KC82's synchronous bus is referred to as internal bus cycle and the external asynchronous bus converted by the external bus interface unit as external bus cycle.

All bus cycles in the figures of the chapter "KC82 CPU" are shown in terms of internal bus cycle, that is, the operations of KC82 in that chapter are described as bus cycles inside KL5C8012 are described. Therefore, a wait state may be inserted by the external wait input or wait state controller when accessing external memories or I/Os in the same operations as shown in that chapter. Refer to the chapter "External Bus Interface Unit" for more information on external bus cycle.

The external bus interface unit contains a wait state controller to enable an efficient connection with external memories. When accessing external memory or I/O, this controller automatically inserts wait states according to two bits of SCR1 (System Control Register 1), and generates various external accesses such as external memory access (0 wait state), external memory access (1 wait state), external I/O access (1 wait state), and external I/O access (2 wait states). The controller is designed to connect two kinds of memories with different access time efficiently, such as high speed SRAM and EPROM, by controlling 1MB internal physical memory divided in two. Internal memory access is always executed in the internal memory cycle (0wait state) regardless of the SCR1 setting. It should be noted that a external wait signal (ERDY) is ignored in the external memory access (0 wait state).

In addition to SCR0 and SCR1, two external inputs (MODE[1:0]) are provided to set the mode. When MODE[1] = 0 and MODE[0] = 1 are selected and a Bug Finder adapter is connected to the external pin BFSIO, our simple debugging tool (Bug Finder) is started just after reset. If RAM is connected in place of external ROM, a program can be downloaded from a personal computer to the RAM and debugging operation can be performed on this RAM just like an ICE.

2. Block Diagram

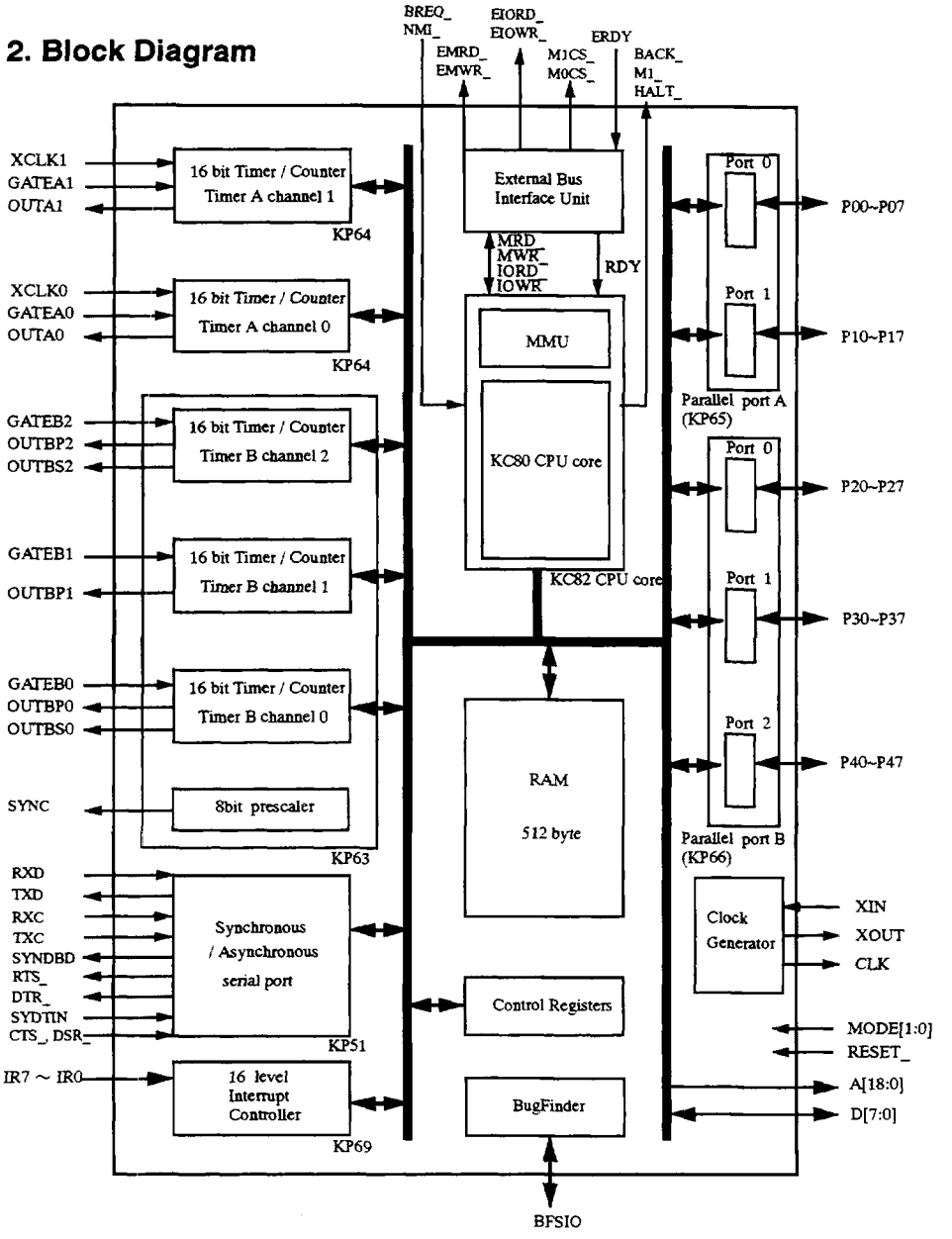


Figure 2-1. KL5C80A12 Block Diagram

3. Pin Description

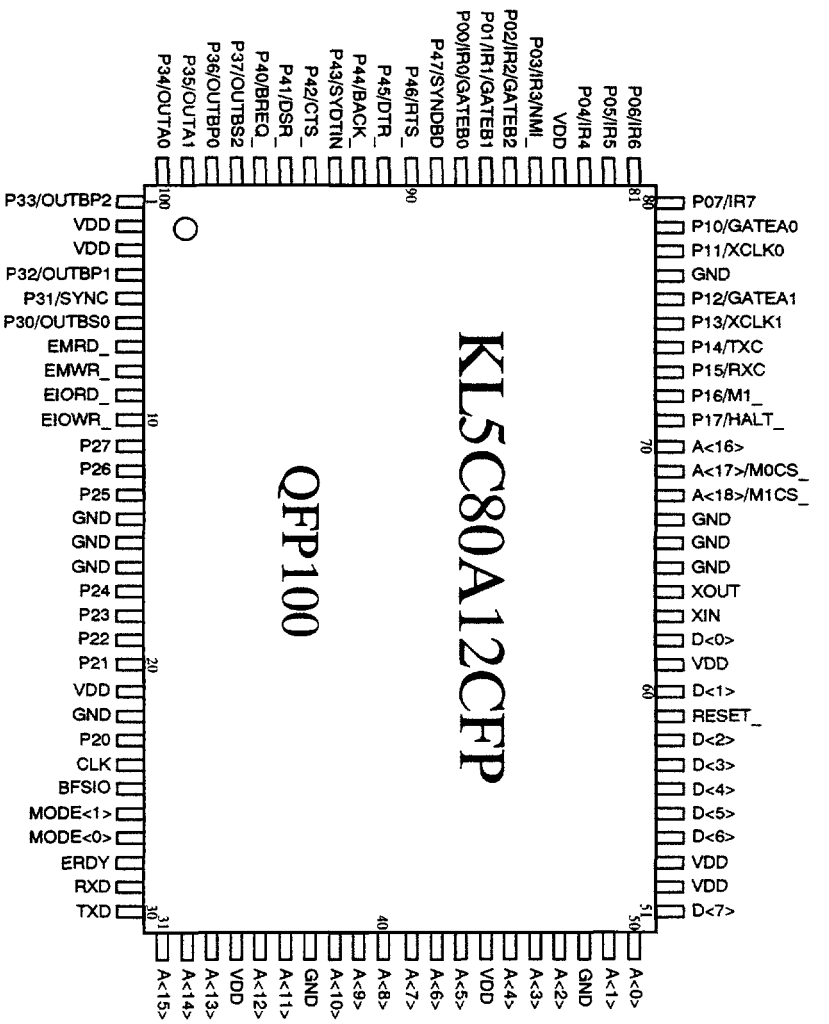


Figure 3-1. Pin Configuration (Top View)

The KL5C8012 pins are divided into two groups, dedicated pins and multiplexed pins. Please see the Table 3-1 for the dedicated pins. The pins in the Table 3-2 are multiplexed with parallel port pins in the Table 3-3. For more information on multiplexing refer to the Pin Configuration. All input/output pins and 3-state output pins (A[15:0], EMRD_, EMWR_, EIORD_, and EIOWR_) and some input pins are provided with internal pull-up resistors.

Table 3-1. Dedicated Pins

pin name	I/O	description
GND	power	supply Connect 0V.
VDD	power	supply Connect 5V.
EMRD_	O	External memory read signal. This signal is obtained by converting the KC80's internal bus signal MRD_ for external bus by the external bus interface unit.
EMWR_	O	External memory write signal. This signal is obtained by converting the KC80's internal bus signal MWR_ for external bus by the external bus interface unit.
EIORD_	O	External I/O device read signal. This signal is obtained by converting the KC80's internal bus signal IORD_ for external bus by the external bus interface unit.
EIOWR_	O	External I/O device write signal. This signal is obtained by converting the KC80's internal bus signal IOWR_ for external bus by the external bus interface unit.
CLK	O	Outputs the internal clock generated by crystal oscillator to external circuit.
BFSIO	I/O	Dedicated port for Bug Finder debugging tool. Leave open when the Bug Finder is not used.
MODE[1:0]	I	Input for setting the mode. Sets the KL5C8012's operation mode.
ERDY	I	External wait request input.
A[16:0]	O	Address output. CPU's A[16:0] are connected.
D[7:0]	I/O	External data bus.
RESET_	I	Reset input. The circuit is reset when this signal goes "L".
XIN	For crystal oscillator	Connects an external crystal oscillator to the built-in clock generator. Use a clock of twice the system clock frequency. Input a clock of twice the system clock frequency when a crystal oscillator is not used.
XOUT	For crystal oscillator	Connects an external crystal oscillator to the built-in clock generator. Use a clock of twice the system clock frequency.
A[17]/M0CS_	O	CPU address output A[17] or external ROM chip select output (address = 00000H ~ 1FFFFH).
A[18]/M1CS_	O	CPU address output A[18] or external RAM/ROM chip select output (address = E0000H ~ FFDFFH).
RXD	I	Receive data input of USART. Serial receive data is input. Receive data is sampled at the rising edge of RXC according to the set baud rate.

pin name	I/O	description
TXD	O	Transmit data output of USART. Transmit data is output serially.Transmit data is output in synchronization with the falling edge of TXC according to the set baud rate. If a send break command (sbrk= '1') is entered, the TXD output is forced to be masked in "L" state (break state). The break state continues until a break clear command (sbrk='1') is entered, and in the meantime the receive buffer, TXRDYPIN and TXEMPTY outputs function according to the initially set character length at the same timing as ordinary characters are sent. When CTS_= "L" and send is enabled (txen= '1'), transmit data can be output. If CTS_= "H" or send is disabled (txen= '0') during data transfer, it is placed in marking state when there is no data to be sent.

Table 3-2. Pins Multiplexed with Parallel Ports

pin name	I/O	description
IR7~0	I	External interrupt input levels 7 to 0. For more information see Chapter 6.
XCLK1	I	External counter clock input to the Counter A channel 1. For more information see Chapter 8.
XCLK0	I	External counter clock input to the Counter A channel 0. For more information see Chapter 8.
GATEA1	I	GATE input to the Counter A channel 1. For more information see Chapter 8.
GATEA0	I	GATE input to the Counter A channel 0. For more information see Chapter 8.
GATEB2	I	GATE input to the Counter B channel 2. For more information see Chapter 9.
GATEB1	I	GATE input to the Counter B channel 1. For more information see Chapter 9.
GATEB0	I	GATE input to the Counter B channel 0. For more information see Chapter 9.
OUTA1	O	OUT output of the Counter A channel 1. For more information see Chapter 8.
OUTA0	O	OUT output of the Counter A channel 0. For more information see Chapter 8.
OUTBS2	O	Strobe output of the Counter B channel 2. For more information see Chapter 9.
OUTBS0	O	Strobe output of the Counter B channel 0. For more information see Chapter 9.
OUTBP2	O	Pulse output of the Counter B channel 2. For more information see Chapter 9.
OUTBP1	O	Pulse output of the Counter B channel 1. For more information see Chapter 9.
OUTBP0	O	Pulse output of the Counter B channel 0. For more information see Chapter 9.
SYNC	O	Synchronous output of the Counter B's prescaler. For more information see Chapter 9.
HALT_	O	HALT signal output. Active Low. This signal indicates the CPU executes HALT instruction and is in HALT status.The CPU resumes from the HALT status by receiving active NMI_, INT_ or reset. During HALT status, the CPU contiguously executes NOP instructions.

pin name	I/O	description
M1_	O	M1 cycle output. Active Low. This signal indicates current machine cycle is opcode fetch cycle of current executing instruction. (During execution of two-byte instruction, "L" is output continuously.)
BACK_	O	Bus Acknowledge output. This signal indicates that the CUP released the bus for external bus master.
BREQ_	I	Bus request input. Active Low. When this signal goes active, the address bus and data bus go to high impedance state after completion of current instruction execution. This signal has higher priority than NMI_ and INT_.
NMI_	I	Non-Maskable interrupt input. Falling edge trigger. This input accepts the non-maskable interrupt. The priority of this input is higher than maskable interrupt but lower than BREQ_. As soon as current instructions execution is completed, it jumps to 0066H and executes the interrupt service routine regardless of the status of interrupt enable flag.
RXC	I	Receiver clock input of USART. This clock input controls the baud rate when data is read or received from the RXD input.
TXC	I	Transmitter clock input of USART. This clock input controls the baud rate when data is sent from the TXD output.
SYNDBD	O	Sync detect/break detect signal output of USART. This signal goes "L" after reset, but it has different functions depending on modes. If it is programmed to the internal sync detect of the sync mode, it goes "H" when a sync character is received and detected after issuing an Enter Hunt command. It is set to "H" at the center of final bit of the sync character (at the center of final bit of the second sync character in the bi-sync mode, or at the center of parity bit when parity is enabled). Then it is reset to "L" by status read. If it is programmed to the external sync detect of the sync mode, it goes "H" when SYDTIN= "H" is detected at RXC= "H". As with the internal sync detect, it is reset to "L" by status read. In the asynchronous mode, it is used as the output pin to indicate that a break signal is detected. When "L" of more than two character data blocks (start bit, parity bit and stop bit) is received from the RXD pin, it is regarded as a break signal and this signal goes "H". When "H" is received from the RXD pin, it goes "L".
RTS_	O	Request-to-send signal output of USART. It is usually used to output a request-to-send signal to a modem, but can be used as general-purpose output port.

pin name	I/O	description
DTR_	O	Data terminal ready signal output of USART. It is usually used to output a data terminal ready signal to a modem, but can be used as general-purpose output port.
SYDTIN	I	External sync detect signal input of USART. When it is programmed to the external sync detect of the sync mode and "H" is input to this pin during RXC= "H", the USART exits the Enter Hunt mode and starts sampling of receive data at the rising edge of the next RXC. If an Enter Hunt command was not executed, it does not receive data. "H" input must be longer than one cycle of RXC. This pin does not function in the internal sync detect mode.
CTS_	I	Clear-to-send signal input of USART. A transmission request signal to USART is input. transmission is being requested while it is "L". It is usually used as clear-to-send signal to the modem. If transmission is enabled (txen= '1'; the bit 0 of command register is set to '1') and there is data to be sent, data is sent from the TXD when this input is "L". When this input goes "H", all data in the buffer is sent and then the TXD goes into the marking state.
DSR_	I	Data set ready signal input. It is usually used as data set ready signal to check the status of modem ("L" = modem enabled to send, "H" = disabled to send), but can be used as general-purpose input port.

Table 3-3. Parallel Port Pins

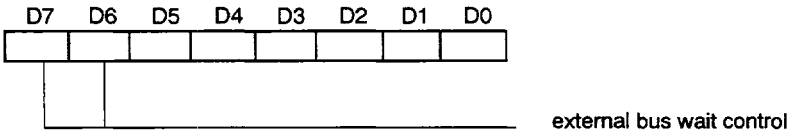
pin name	I/O	description
P00~P07	I/O	P0 port of the parallel port block A. For more information see Chapter 10.
P10~P17	I/O	P1 port of the parallel port block A. For more information see Chapter 10.
P20~P27	I/O	P0 port of the parallel port block B. For more information see Chapter 10.
P30~P37	I/O	P1 port of the parallel port block B. For more information see Chapter 10.
P40~P47	I/O	P2 port of the parallel port block B. For more information see Chapter 10.

4. External Bus Interface Unit

4. 1 General description

The80A12 external bus interface unit includes a wait state controller, and converts internal bus cycles to external bus cycles inserting specified number of wait states. The number of wait states is specified by bits 6 and 7 of SCR1 (System Control Register 1). This unit works only at the access of external I/O and external memory. The Read/Write signal of external bus stays "H" at the access of internal I/Os and the internal high speed RAM.

SCR1



	external memory (00000 - 7FFFF)	external memory (80000 -FFDFF)	external I/O
00	1 wait state	1 wait state	2 wait states
01	1 〃 *	1 〃 *	2 〃
10	1 〃	0 〃	1 〃
11	0 〃	0 〃	1 〃

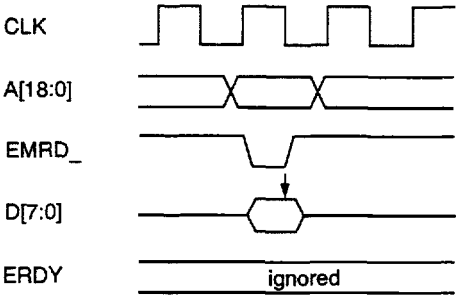
* wide write strobe option

The KL5C80A12's CPU core (KC82) uses synchronous system bus. Its internal I/Os and high speed RAMs include a bus interface suitable for the KC82's synchronous bus so that they are connected directly. When connecting I/O and memory to KL5C80A12, the external bus interface unit converts KC82's synchronous bus Read/Write signals (MRD_, MWR_, IORD_, IOWR_) to external asynchronous bus Read/Write signals (EMRD_, EMWR_, EIORD_,EIOWR_) described later. The external bus interface unit includes a wait state controller, and converts internal bus cycles to external bus cycles inserting specified number of wait states. The number of wait states is specified by bit 6 and 7 of SCR1. The external bus interface unit adds a wait request from the built-in wait state controller to an external wait request input from ERDY, and send them to the CPU.

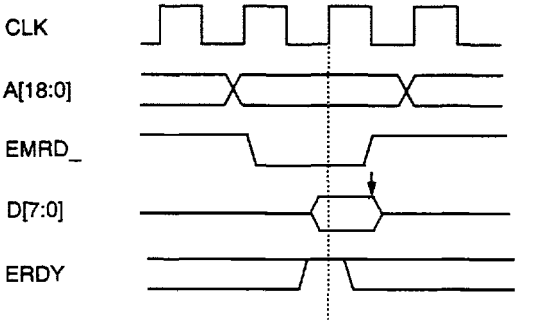
This controller is designed to connect two kinds of memories with different access time efficiently, such as high speed SRAM and EPROM, by controlling 1MB internal physical memory divided in two. Internal memory access is always executed in the internal memory cycle (0 wait status) regardless of the SCR1 setting. It should be noted that a external wait signal (ERDY) is ignored in the external memory access (0 wait status).

Timing Diagrams of External Bus Cycles

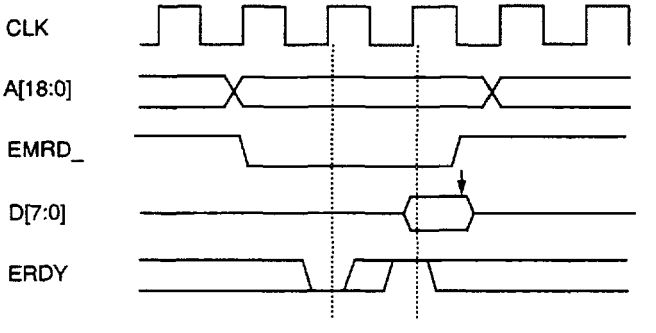
External memory read cycle (0 wait state from the wait state controller)



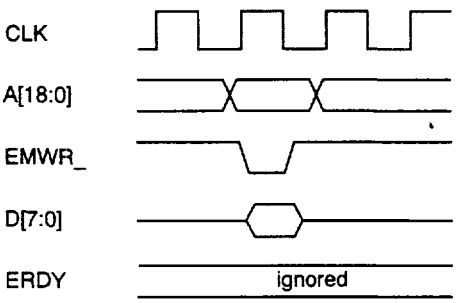
External memory read cycle (1 wait state from the wait state controller)



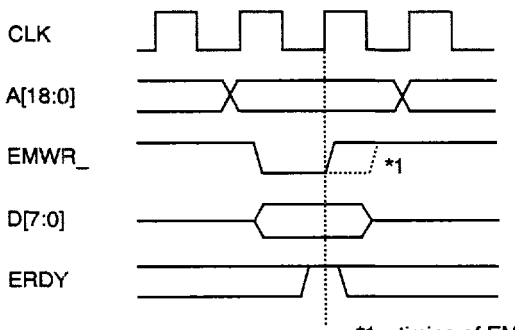
External memory read cycle (1 wait state from the wait state controller + 1 wait state from ERDY)



External memory write cycle (0 wait state from the wait state controller)

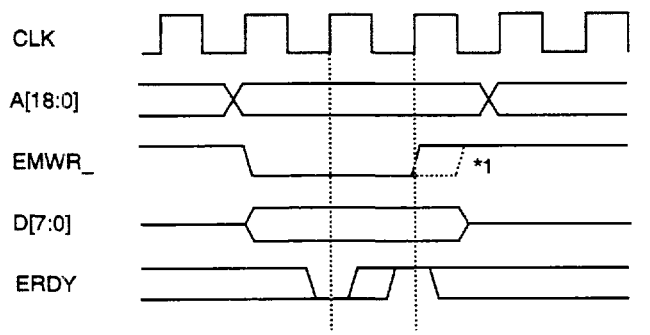


External memory write cycle (1 wait state from the wait state controller)

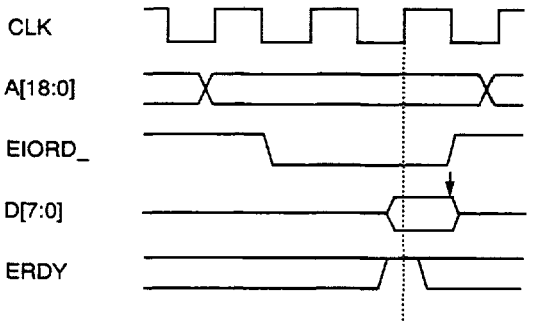


*1 timing of EMWR_ signal with the wide write strobe option selected

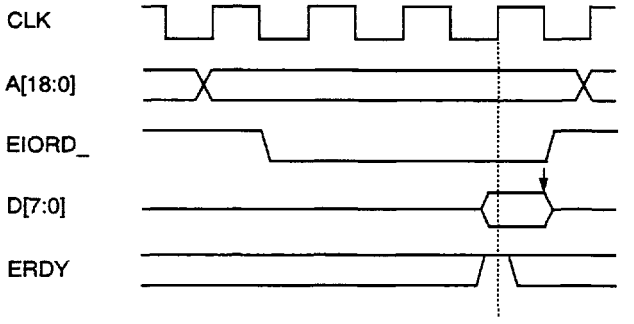
External memory write cycle (1 wait state from the wait state controller + 1 wait state from ERDY)



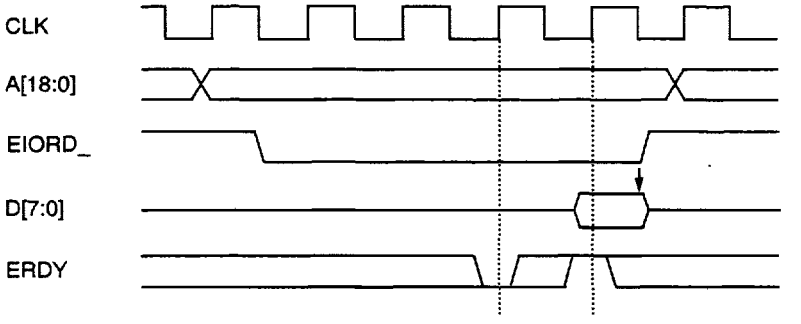
External I/O read cycle (1 wait state from the wait state controller)



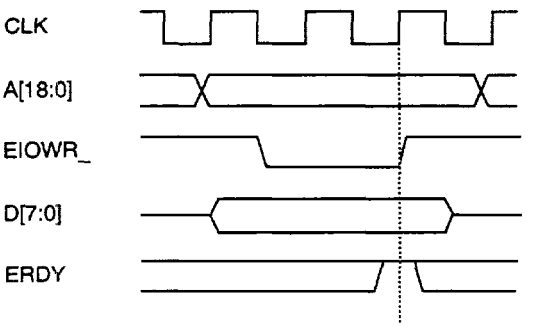
External I/O read cycle (2 wait states from the wait state controller)



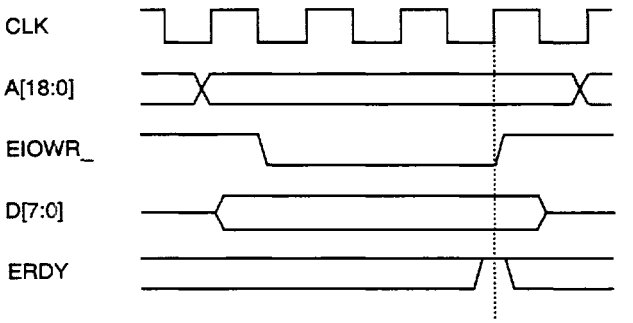
External I/O read cycle (2 wait states from the wait state controller + 1 wait state from ERDY)



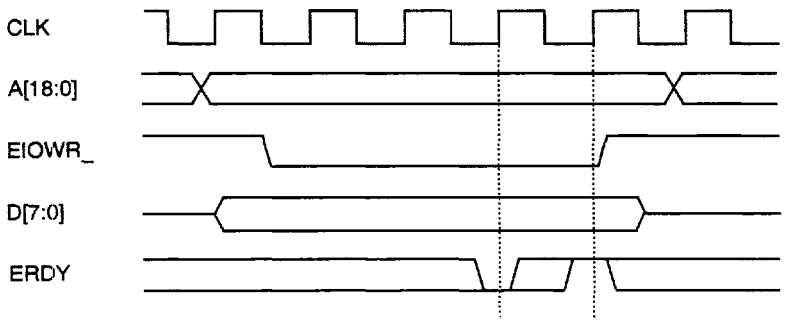
External I/O write cycle (1 wait state from the wait state controller)



External I/O write cycle (2 wait states from the wait state controller)



External I/O write cycle (2 wait states from the wait state controller + 1 wait state from ERDY)



5. KC82 CPU

5.1 General description

KC 82 is the CPU core which is object compatible with Zilog's Z80 CPU with MMU that extends the address space up to 1M byte. But its internal circuit is totally different. KC82 has RISC like architecture, 16 bit internal data path and synchronous bus, and offers higher performance than typical 16 bit CPU.

Feature

KC82 has the following features:

1)KC82 executes instructions 4 times faster than Zilog Z80 CPU at the same clock rate.

ex : instruction	Z80	KC82
LD r, r'	4 clocks	1 clock
ADD HL, ss	11 clocks	1 clock

2)With interface macro cell, it can be used with the Z80 peripheral and ordinary RAMs.

3)158 instructions and fully compatible with the Z80 CPU at object code level.

5.2 Block Diagram

The following figures 5-1 has the block diagram of the KC82.

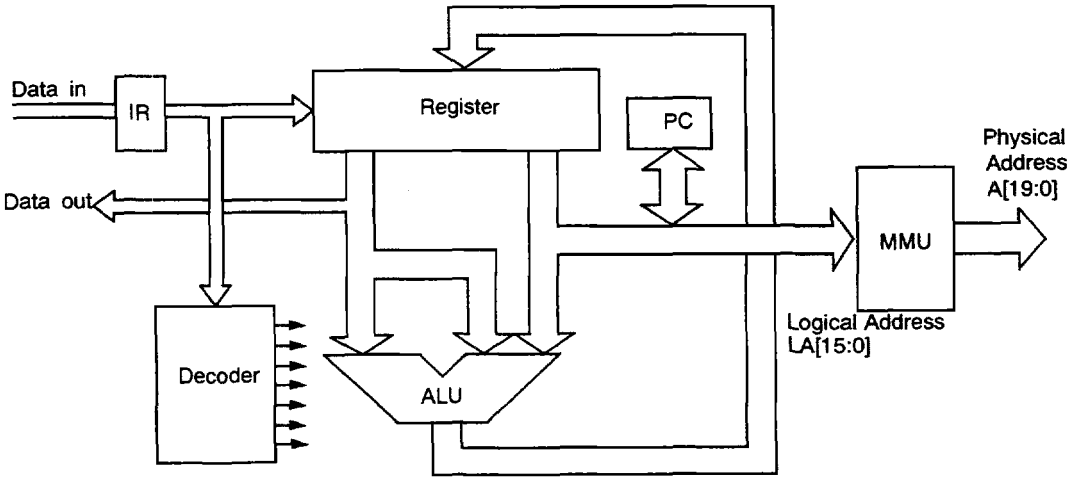


Figure 5-1. KC82 Block Diagram

5.3 CPU internal pins description

signal name	I/O	description
MRD_	O	Memory read. Active Low. This signal goes active while reading data from memory.
MWR_	O	Memory write. Active Low. This signal goes active while writing data to memory.
IORD_	O	I/O read. Active Low. This signal goes active for minimum of two clock cycles while reading data from I/O device.
IOWR_	O	I/O write. Active Low. This signal goes active for minimum of two clock cycles while writing data to I/O device.
WAIT_	I	Wait input. Active Low. This signal is to notify the CPU that the peripherals or memory is not ready for data transfer. While this signal stays active, the CPU is placed in the wait state. This input is connected to the external pin, ERDY through the wait state controller in the external bus interface unit.
INT_	I	Maskable interrupt input. Active Low. This input accepts the interrupt from peripherals. If the interrupt enable flag of the CPU is set, and BREQ_ is inactive, the CPU completes the execution of the current instruction, and starts interrupt service. This input is connected to the INT_ output of the interrupt controller.
IACK_	O	Interrupt Acknowledge output. Active Low. This signal indicates that the CPU acknowledges the interrupt, requesting interrupt vector or instruction from the I/O device. It stays active at minimum of two clock cycles. This output is connected to the interrupt controller. The CPU reads the interrupt vector synchronously with this signal.
EOI_	O	End of interrupt signal output. Active Low. This signal goes active at refetch of RETI instruction (code ED 4D). This output is connected to the interrupt controller.
LA [15:0]	O	Logical Address output. It outputs the physical address A[19:0] expanded by the MMU in KC82.

5.4 Register architecture

5.4.1 Special purpose registers

Program Counter (PC)

Program Counter holds the address of the next instruction. The next instruction is fetched from memory address Program Counter indicates.

Stack Pointer (SP)

Stack Pointer holds the current top address of stack area in RAM.

Index registers (IX, IY)

Index registers hold base address for indexed addressing. There are two index registers. The one is IX, and the other IY.

Interrupt page address register (I)

Interrupt page address register holds indirect upper 8 bit address for indirect jump in Mode 2.

Memory refresh register (R)

This register automatically increments by one on instruction fetch. Bit D7 of this register will not change by increments on D6-D0.

Accumulator (A, A'), and Flag register (F, F')

There are two 8 bit accumulators (A, A') and two flag registers. Accumulator holds the result of arithmetic and logical operation. Flag register holds status of 8 bit or 16 bit operation. The instruction, EX AF, AF' exchanges A, F and A', F'.

5.4.2 General purpose registers

There are two sets of general purpose registers. Each of them can be used as an 8 bit register independently (B, C, D, E, H, L or B', C', D', E', H', L'). Paired registers (BC, DE, HL, BC', DE' and HL') can be used as a 16 bit register. The instruction, EXX exchanges B,C, D, E, H, L and B',C', D', E', H', L'.

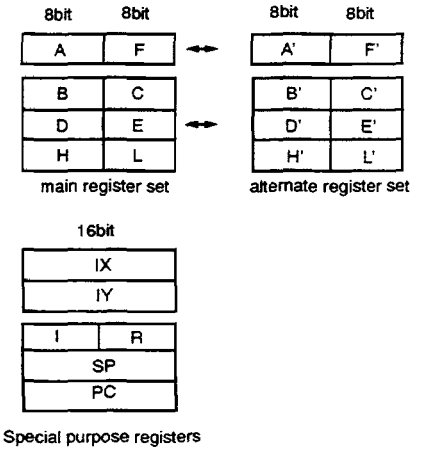


Figure 5-3. KC82 register architecture.

5.5 Flags

Bit assignment of the Flag register.

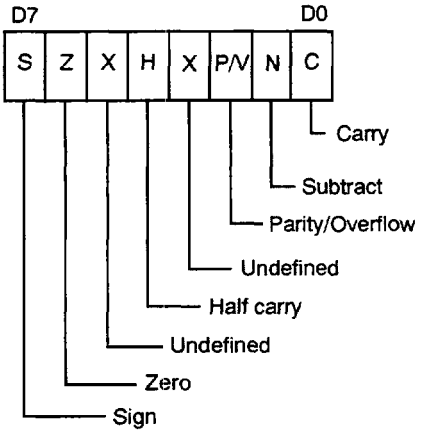


Figure 5-4. Flag register

The function of each bit in the Flag register is defined as below.

Carry flag (C)

Carry flag is set by carry produced by Accumulator from the MSB at execution of add instructions, subtract instructions, shift and rotate instructions, etc.

Zero flag (Z)

Zero flag is set if the operation result is zero, when add instructions, subtract instructions, logical instructions including INC, DEC, DAA instructions. This also occurs on block I/O instructions, rotate and shift instructions, string search instructions, bit test instructions and data detection of block search instructions.

Sign flag (S)

Sign flag is set when the result of sign number operations is negative.

Parity/Overflow flag (P/V)

Parity/Overflow flag has two functions. It indicates the parity on logical operations, and also the overflow on arithmetic operations. The overflow means that the content of Accumulator is less than -128 in 2's complement, or larger than 127. It is also set by block search instructions, block I/O instructions, LD A, R instruction and LD A, I instruction.

Half carry flag (H)

Half carry flag is set when the carry or borrow from lower 4 bits is produced.

Subtract flag (N)

Subtract flag is set on subtract instructions. KC82 checks if the previous instruction is addition or subtraction with this flag on DAA instruction.

5.6 Functional description and timing

In this section, functional description and timing are described. Note that the address bus in the timing diagrams is the logical address, LA[15:0], to describe easily. The logical address LA[15:0] is converted to the physical address by MMU.
All bus cycles are described using the internal bus. See the chapter 4 for more information of the external bus.

5.6.1 Basic operation (Instruction cycle)

Basic operation of the KC82 can be divided into the following 5 cycles.

1)Opcode fetch cycle - Fetches opcode of an instruction from memory.
2)Memory read cycle - reads data from memory
3)Memory write cycle - writes data to memory.
4)I/O read cycle - reads data from I/O device
5)I/O write cycle - writes data to I/O device

Follows are the description of the each machine cycles.

Opcode fetch cycle

As shown figure 5-5, KC82 fetches one byte of an instruction in a clock cycle unless there is a wait. During this cycle, MRD_ and M1_ go active. The difference between memory read cycle is whether if M1_ goes active or not. This cycle is minimum of one clock cycle and appears at the end of the instruction execution sequence.

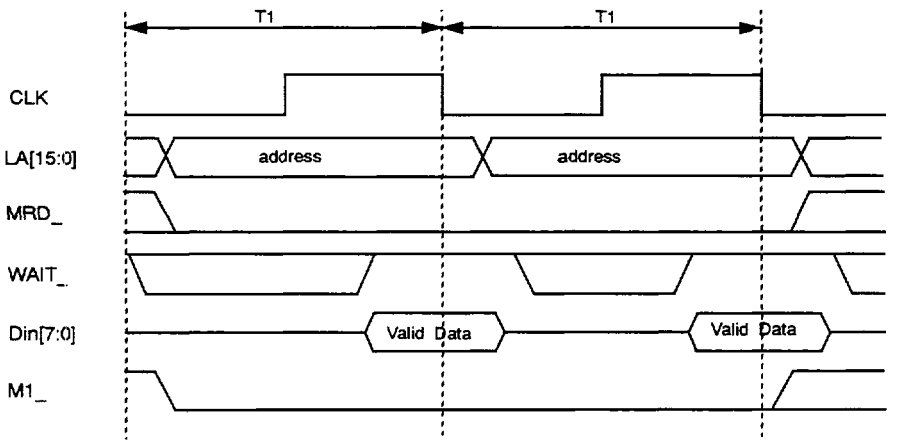


Figure 5-5. Opcode fetch cycle (Internal cycle)
(0 wait state, two contiguous reads.)

Internal bus memory read cycle

The difference between opcode fetch cycle is whether if M1_ goes active or not. This cycle is minimum of 1 clock cycle.

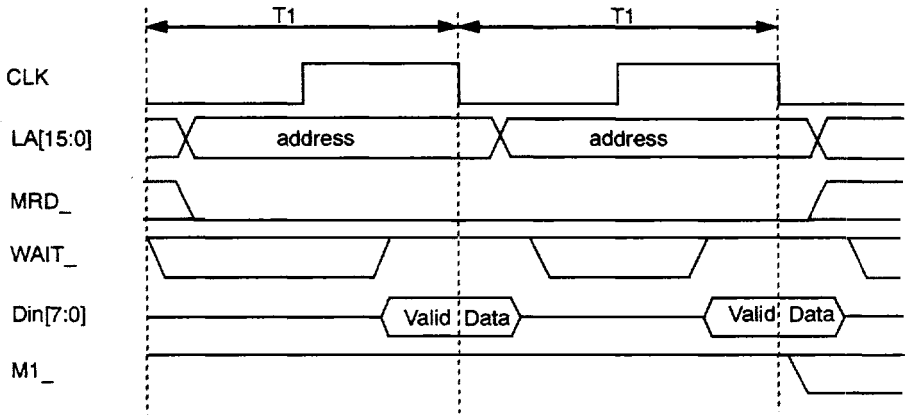


Figure 5-6. Memory read cycle (0 wait)
(0 wait state, two contiguous reads.)

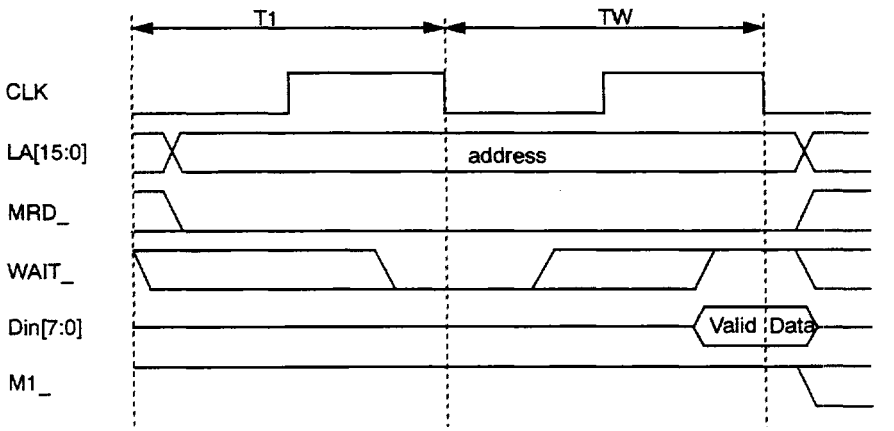


Figure 5-7. Memory read cycle (1 wait)
(One read with 1 wait state)

Internal bus memory write cycle

This cycle is minimum of 1 clock cycle.

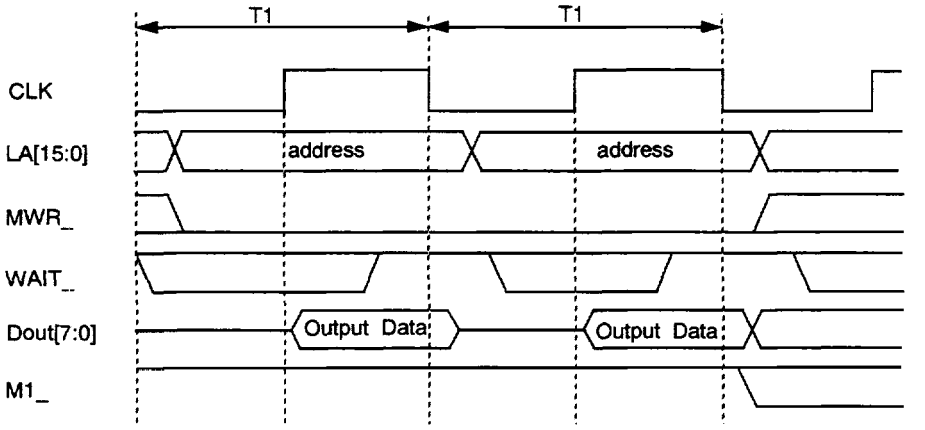


Figure 5-8. Memory write cycle (0 wait)

(0 wait state, two contiguous writes.)

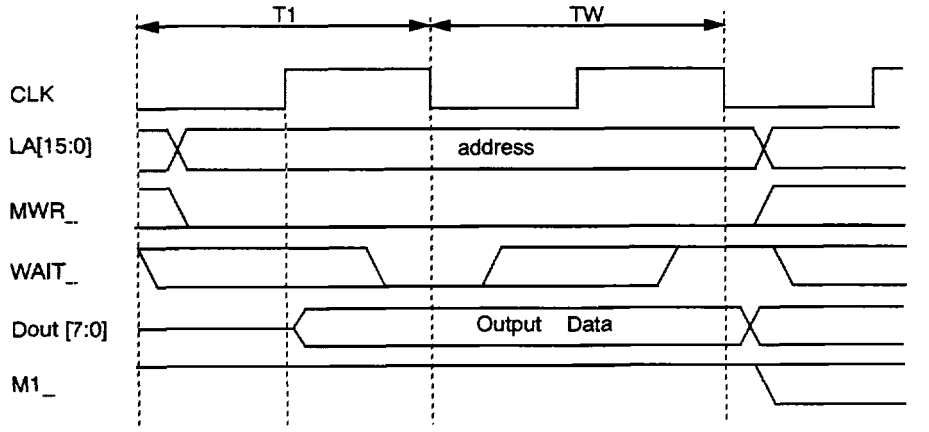


Figure 5-9. Memory write cycle (1 wait)

(One write with 1 wait state)

Internal bus I/O read cycle

This cycle is minimum of 2 clock cycles.

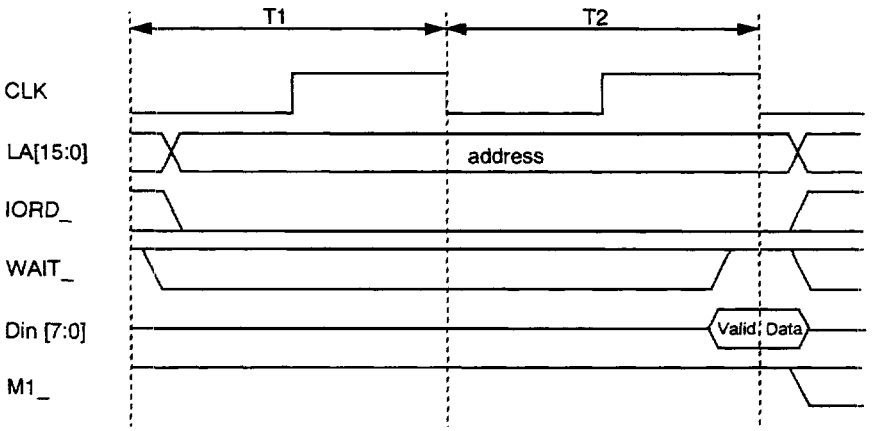


Figure 5-10. I/O read cycle (0 wait)

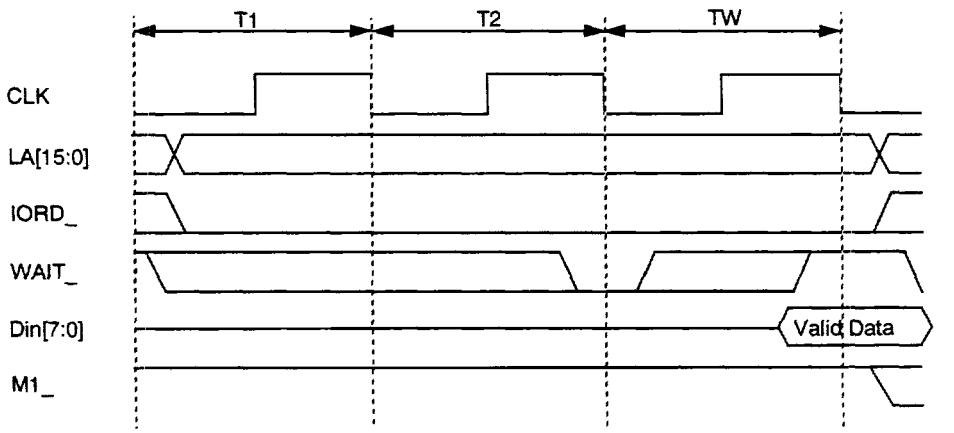


Figure 5-11. I/O read cycle (1 wait)

Internal bus I/O write cycle

This cycle is minimum of 2 clock cycles.

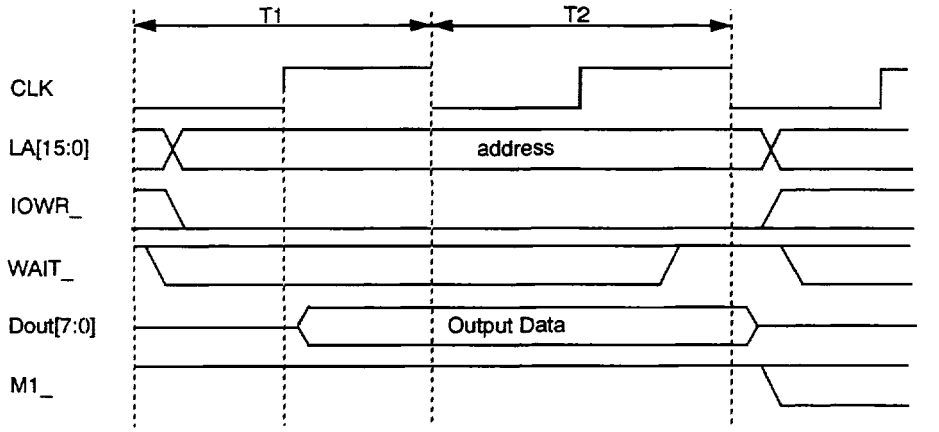


Figure 5-12. I/O write cycle (0 wait)

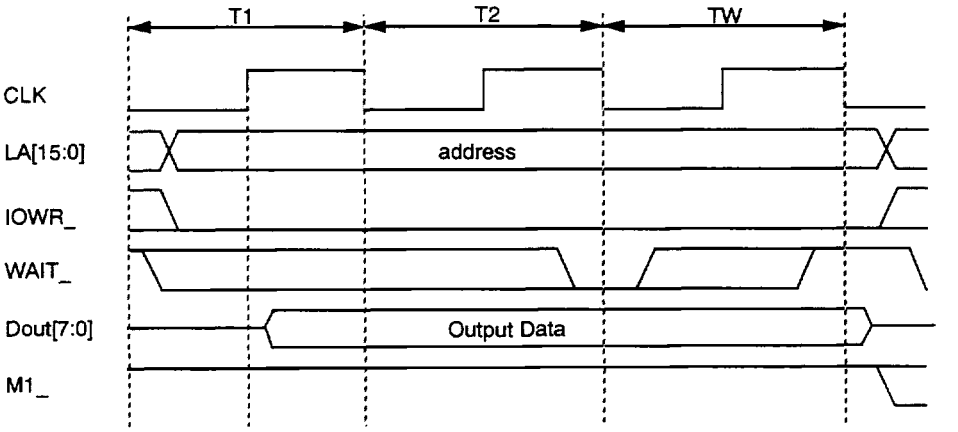


Figure 5-13. I/O write cycle (1 wait)

Instruction Prefetch cycle

The KC82 has the prefetch cycle at the end of instruction execution cycle all the time. Figure5-15 shows that KC82 executes the instruction sequence indicated in Figure5-14 as an example.

◁—▷ (referred as "arrow" hereinafter) in the Figure 5-15 denotes the prefetch cycle. The arrow ① part prefetches the instruction at n+3 (ADD A,D), the arrow ② part is for prefetching instruction at n+4 (opcode of JP instruction, C3h), the arrow ③ part is for prefetching 77h at 1000h (LD [HL], A instruction). As shown, these cycles appear at the end of execution of current instruction.

address	mnemonic	code
n	LD A, [1234h]	3A 34 12
n+3	ADD A, D	82
n+4	JP 1000h	C3 00 10
.	.	
.	.	
1000h	LD [HL], A	77
.	.	
1234h		5A

Figure 5-14. Source code for Figure5-15

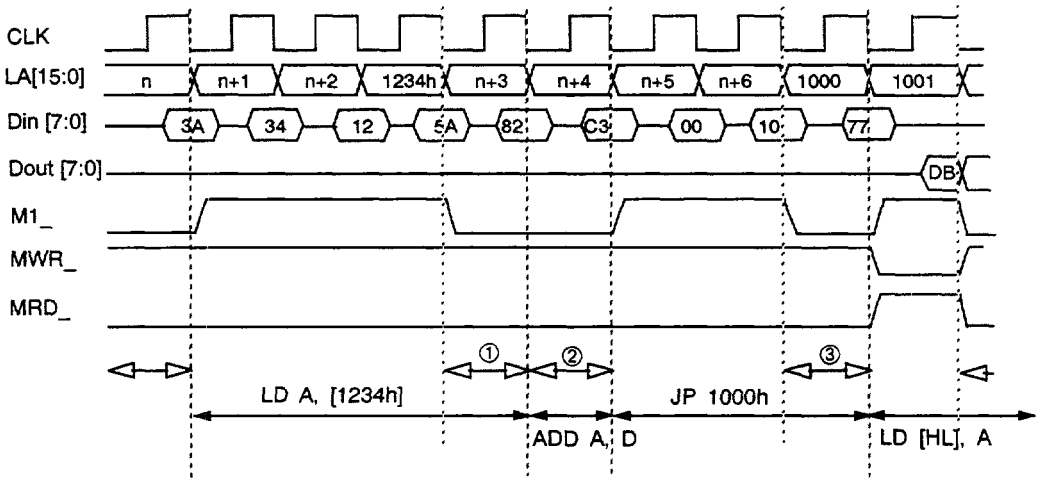


Figure 5-15. Prefetch cycle (◁—▷ denotes the prefetch cycle. Assumes memory location 1000h has 77h, memory location 1234h has 5Ah, and D register has 81H.)

Special case for prefetch. (Discard of the prefetched instruction)

For example, under normal condition, the prefetched instruction, at address 1000h, data 77h at ③ in Figure 5-15 on execution of the instruction, JP 1000h is held and execute correctly. However, on interrupt the prefetched instruction will be discarded and refetch on return from an interrupt service routine. On interrupt recognition, PC is decremented by one before saved onto stack. On RETI or RETN instruction the PC will be loaded with the address of the discarded instruction. Figure 5-19 has the timing diagram for the case of receiving interrupt in mode 1 when executing LDIR instruction. In the figure, the data "EDh" prefetched at ⑥, is discarded and then go to interrupt service. On return, the CPU resume from the prefetch of this "EDh". For the bus request cycle, the data "EDh" prefetched at ⑥ prior to the bus request will be kept, and on bus release the CPU starts to prefetch from the next address to the "EDh" as shown in the Figure 5-18.

5.6.2 Bus release (Bus request/acknowledge cycle)

Under normal operation, the CPU holds the control of address bus, control bus (MRD_, MWR_, IORD_ and IOWR_). However, if there is an external bus request (BREQ_="L"), address bus goes to high-impedance state, all the interrupts are disabled, BACK_ turns to "L", and the CPU releases the bus control to the external device. By using this feature, data transfer without CPU intervention can be done. Figure 5-17 shows basic timing for the bus request cycle. This cycle continues while BREQ_ stays "L".

Note that bus request is accepted only at the end of instructions except the instructions which have iteration, such as block transfer instructions, block search instructions and iterating IN/OUT instructions. In a special case, for the block move instructions the acceptance of a bus request occurs on every iteration of an execution cycle, not the end of the instruction. As shown in the Figure 5-18, BREQ_ is asserted at point ④ during the execution, the bus is released at the last execution cycle of an instruction ⑤.

5.6.3 Interrupt and timing

The KC82 can handle the following two kinds of interrupts.

1)maskable interrupt on INT_

2)Non-maskable interrupt on NMI_

2) has the higher priority over 1). If both request made simultaneously, 2) will be accepted over 1).

Maskable interrupt

The EI instruction enables interrupts, and the DI instruction disables interrupts. The control of interrupt is implemented using two flip-flops (IFF1 and IFF2). Figure 5-16 has the state table for these flip-flops.

events	IFF1	IFF2	
reset	0	0	
DI instruction	0	0	
EI instruction	1	1	
INT accepted	0	0	
NMI accepted	0	-	
RETN instruction accepted	IFF2	-	IFF2 is copied into IFF1
LD A, I instruction	-	-	IFF2 is copied into P/V
LD A, R instruction	-	-	IFF2 is copied into P/V

Figure 5-16. State diagram for IFF1 and IFF2
("-" denotes remain unchanged.)

The maskable interrupt will be accepted if all of the following conditions are met:

1)Both IFF1 and IFF2 are set. (The EI instruction and RETN instructions change the status of these flags after the execution of the following instruction. So if there are EI instruction followed by DI instruction, the interrupt request will not be accepted.)

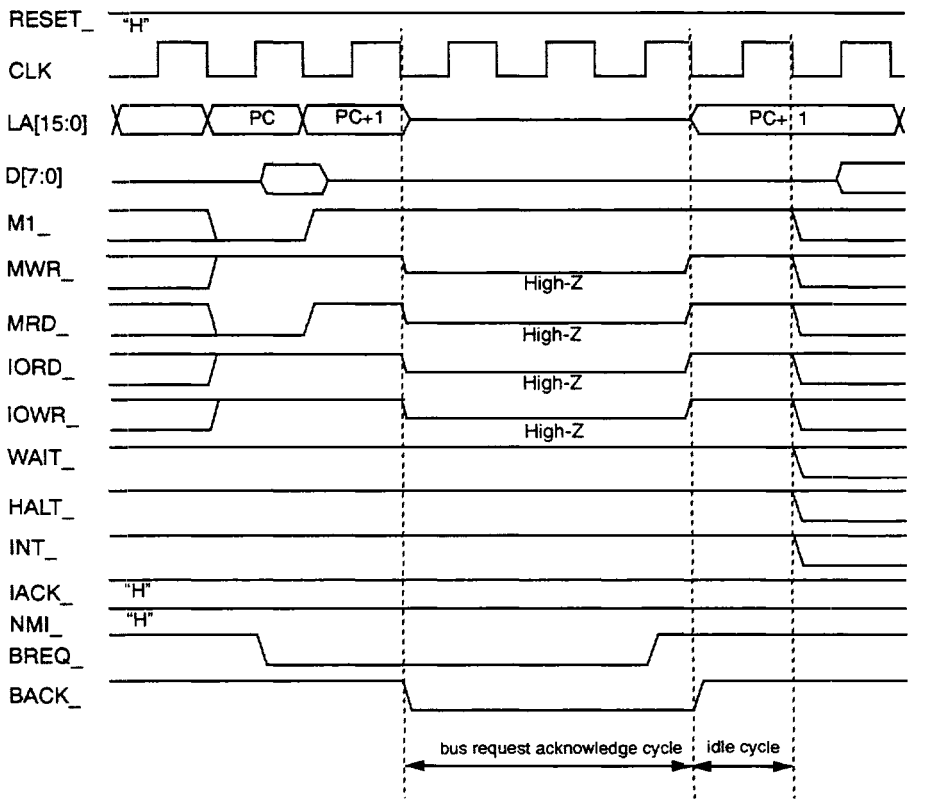


Figure 5-17. Bus request timing

2)BREQ_ is inactive ("H"). (When BREQ_ goes inactive, interrupt will be accepted after execution of an instruction.)

3)No NMI_ falling edge has been detected.

The KC82 has the dedicated signal, IACK_ for the external devices to acknowledged that the interrupt has been accepted, while the Z80 CPU signals using M1_ and IORQ_ combination for that purpose.

Maskable Interrupt modes

The KC82 has the following three different interrupt handling modes, and interrupt sequence is different by each modes. The following descriptions are the operations in each mode. Use the mode 2 usually.

1)Mode 0

This is the default mode and set to this on reset automatically.

Also, executing "IM 0" instruction sets the CPU to this mode. In this mode, the CPU executes the instruction read during interrupt acknowledge cycle.

Usually the instruction to be used in this mode is RST instructions, or CALL instruction.

Figure 5-20 and 5-21 show each timing for RST and CALL instructions. For the all interrupt modes, if INT_ goes active during execution of a mode change instruction (IM 0~2), the new mode become effective right after these instructions.

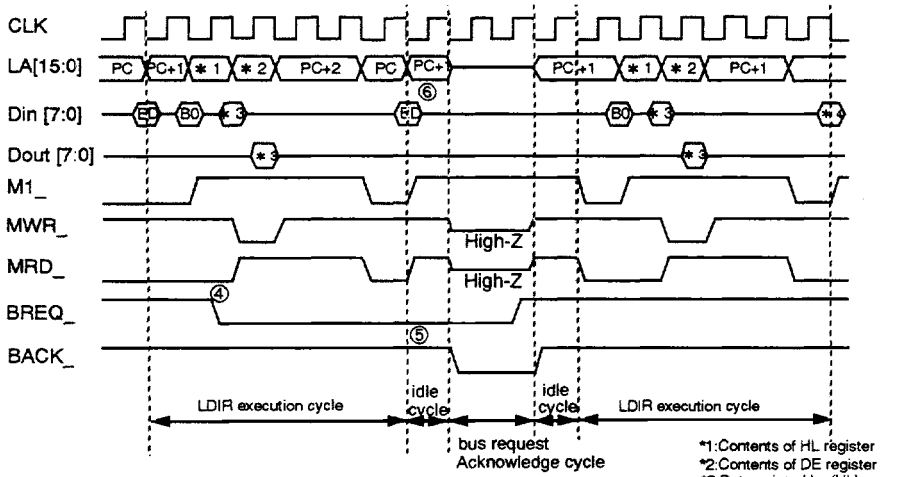


Figure 5-18. Bus request accepting timing

(The figure has the case for BREQ_ goes active at ④, accepting it at instruction boundary ⑤. Note that the data fetched at ⑥ will be kept.)

2)Mode 1

The "IM 1 instruction" sets the CPU in this mode. In this mode, the CPU saves the contents the PC (Program Counter) onto the stack, ignores the data read during the interrupt acknowledge cycle, and executes "RST 38H" instruction internally. Figure 5-22 has the timing for Mode 1 interrupt.

3)Mode 2

The "IM 2 instruction sets" the CPU to this mode. In this mode, as shown in the Figure 5-23-A, the 8-bit wide vector with 0 in LSB read during the interrupt acknowledge cycle as a lower byte, and the content of the I register as a upper byte form up a pointer which points an entry in a table of address for the interrupt service routines. After forming up the 16-bit address, the CPU fetches the table location and gets the start address, saves the return address onto the stack, then jumps to the service routine. Figure 5-23-B has the timing diagram for Mode 2 interrupt acknowledge cycle.

Timing on interrupt acceptance

The Figure 5-19 show the interrupt acknowledge timing. In this figure, an interrupt is accepted during block move instruction. ④ shows that when an interrupt request occurs, the interrupt will not be accepted until the prefetch cycle of the last instruction, ⑤. In this case, the data "EDh" fetched at ⑥ will not be kept by the CPU, and it will be refetched on the return from an interrupt.

Non-maskable interrupt

Non-maskable interrupt is an interrupt which can not be masked by software. The falling edge of the signal NMI_ is latched internally, and checked on the clock falling edge of the last cycle of every instruction. Non--maskable interrupt takes place if BREQ_ is inactive. If BREQ_ goes inactive, NMI_ will be accepted after execution of an instruction.

When a non-maskable interrupt is accepted, the CPU saves the contents of the PC onto the stack, and jumps to 0066h. Return from non-maskable service is done by "RETN" instruction. Figure 5-24 shows the timing for NMI_ acknowledge cycle.

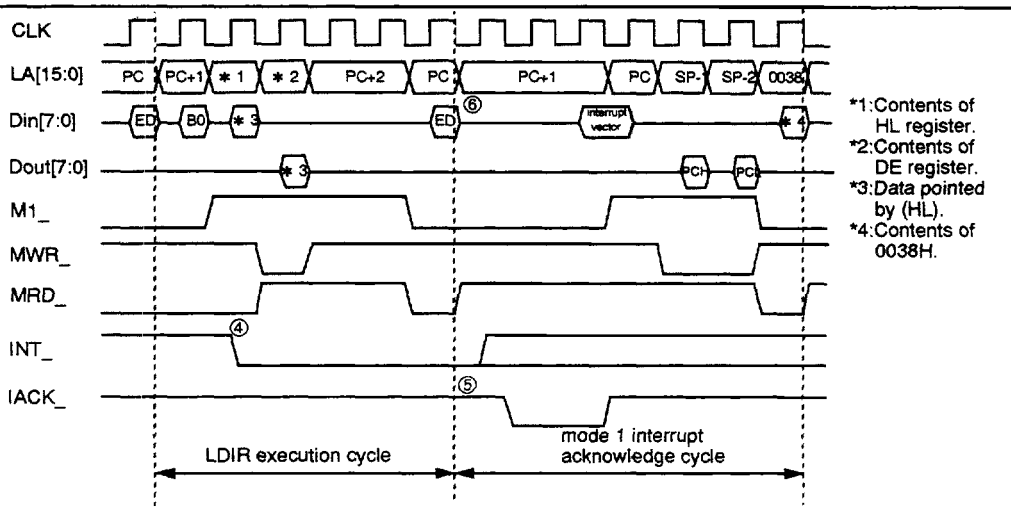


Figure 5-19 .Interrupt acknowledge cycle in mode 1

(The figure shows the case when the interrupt occurred at ④,accepting it at the instruction boundary ⑤. *This case restarts from the fetch of ED at ⑥ on the return from an interrupt.*)

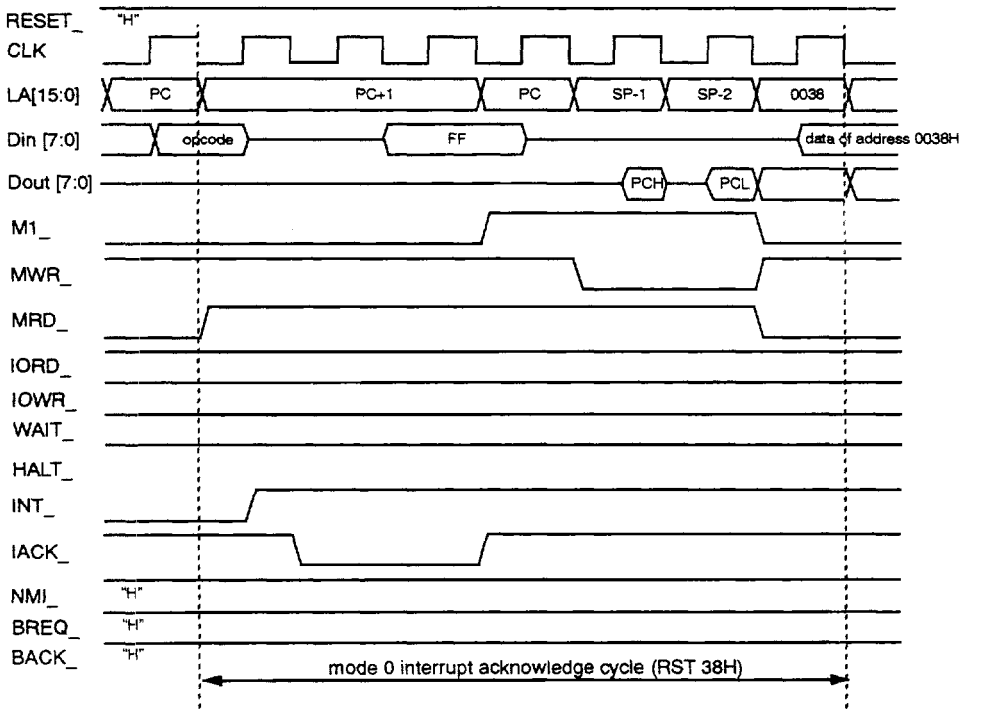


Figure 5-20. Mode 0 timing

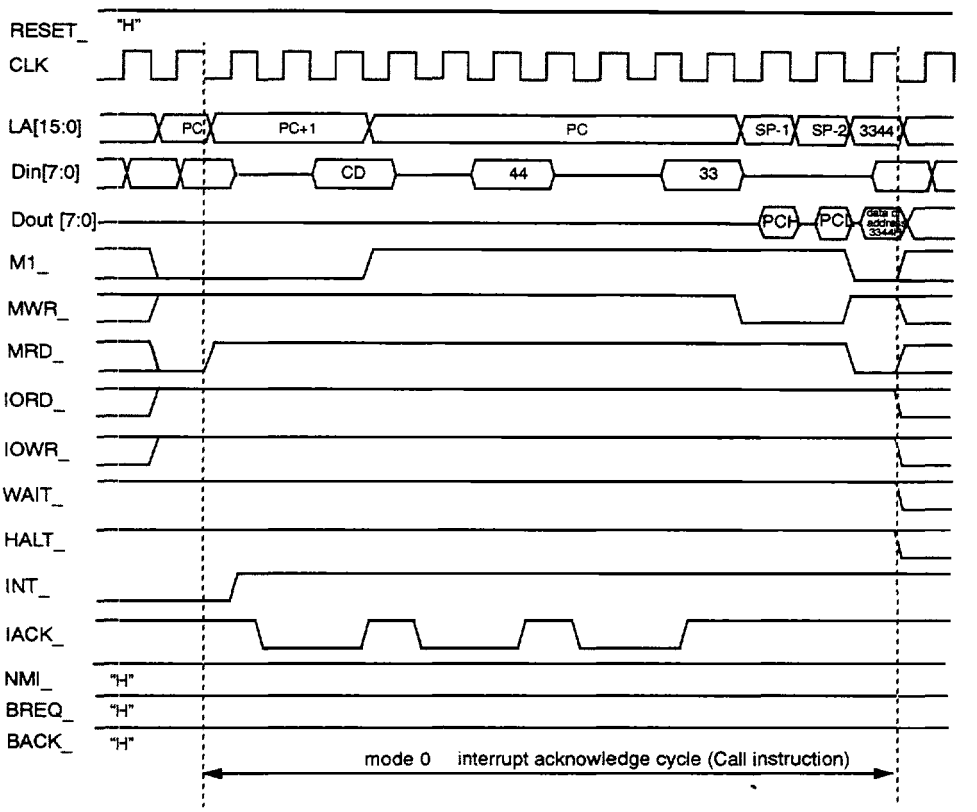


Figure 5-21. Mode 0 timing

Enable/disable of interrupts

Enable/disable of an interrupt is controlled by two flip-flops - IFF1 and IFF2. Refer to Figure 5-16 on the state diagram for these flip-flops. As shown, maskable interrupt will be accepted when both IFF1 and IFF2 set to 1. After reset and execution of DI instruction maskable interrupt is disabled, because both of these flip-flops are reset. An interrupt can be accepted while both IFF1 and IFF2 are set to 1 by execution of an EI instruction. The reason there are two flip-flops is to memorize the status of EI/DI condition with IFF2 on acceptance a Non-maskable interrupt. For example, if both IFF1 and IFF2 are '1' (maskable interrupt is enabled) on acceptance a non-maskable interrupt, in a non-maskable interrupt acknowledge cycle IFF1 is cleared to 0 and maskable interrupts are disabled. At this stage, IFF2 still holds "1" and the content of IFF2 is copied back into IFF1 on RETN instruction so that maskable interrupts could be enabled automatically again.

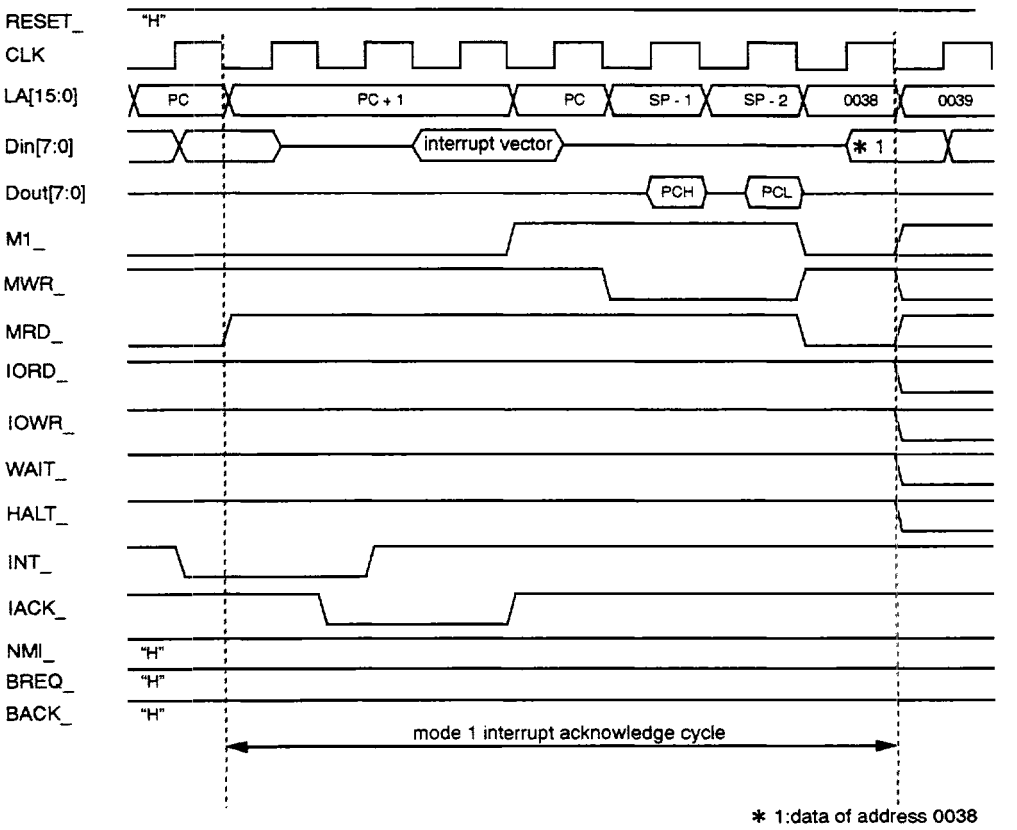


Figure 5-22. Mode 1 timing

5.6.4 Halt enter and exit

By executing the HALT instruction, the CPU enters into HALT cycle. In this cycle, the CPU continues to execute NOP instruction internally. The CPU exits from this cycle by either reset, or interrupts (Non-maskable interrupt or interrupt with IFF set). Figure 5-25 shows timing for the case of acceptance an interrupt in mode 2 during HALT cycle. Then the CPU exits from the HALT cycle. HALT cycle made up with two separate bus cycles; the first cycle is idle cycle and the second cycle is opcode fetch cycle. The address lines holds the next address to the HALT instruction during this cycle. The instruction fetched during the second opcode fetch cycle will not be read by the KC82. The INT_ input is sampled on the falling CLK edge of the second cycle, and forces HALT_ back to "H".

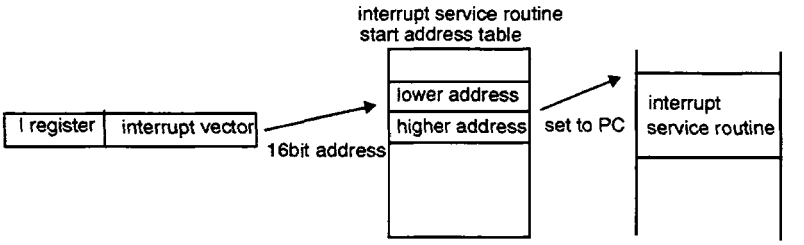


Figure 5-23-A. Mode 2 interrupt

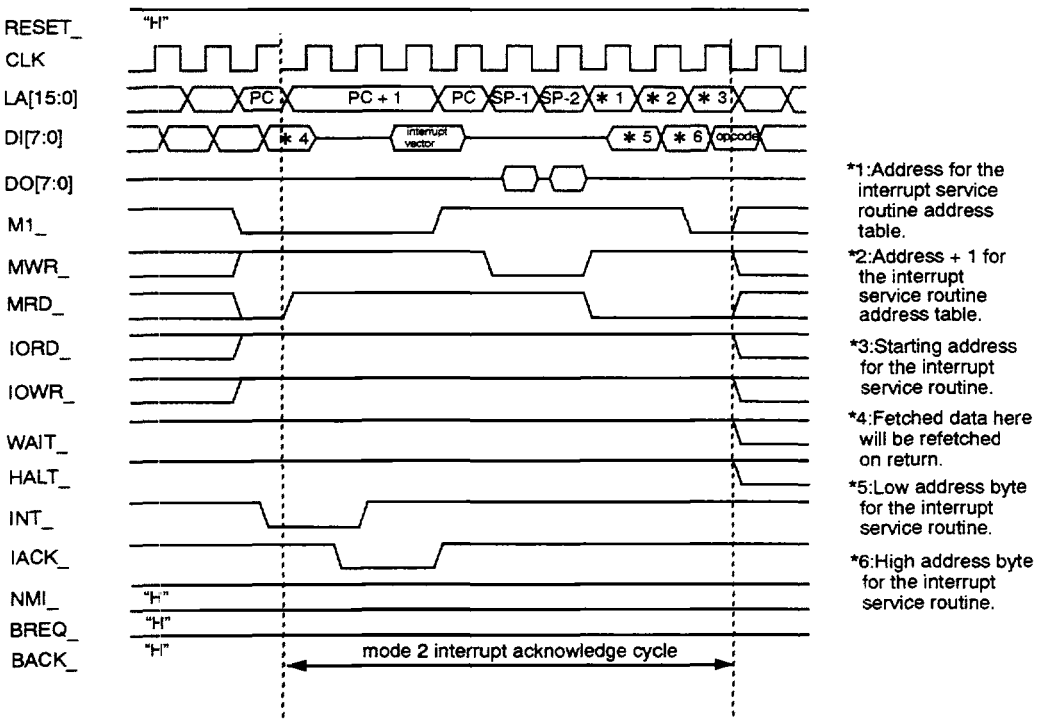


Figure 5-23-B. Mode 2 timing

5.6.5 Reset timing

By keeping RESET_ input "L" for the minimum 3 clock cycles, the KC82 is reset. During reset cycle, AEN_ goes "H". When RESET_ goes inactive ("H"), opcode fetch cycle will be initiated on the falling edge of 3rd clock, and starts from 0000h. The interrupt mode is set to Mode 0. IFF1, IFF2 flags and I, R registers are reset.

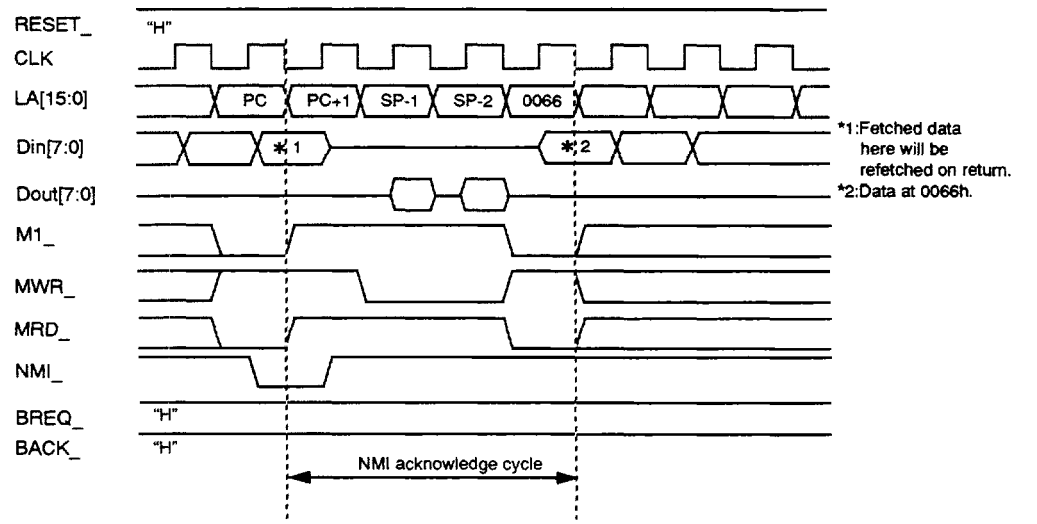


Figure 5-24. Non-maskable interrupt timing

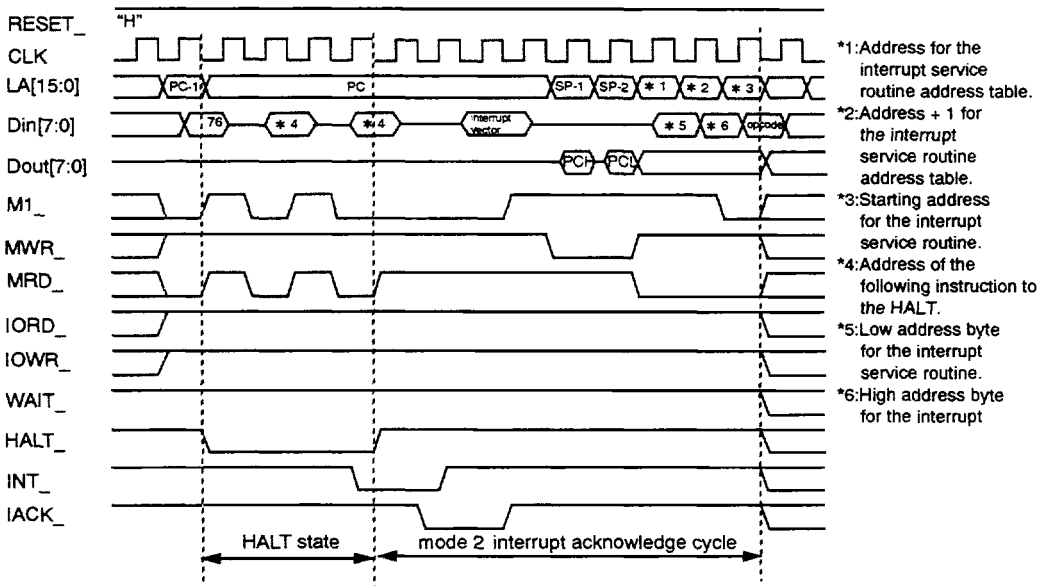


Figure 5-25. HALT Exit (Mode 2)

5.7 MMU

5.7.1 General description

The MMU block is the circuit which converts the KC82 16 bit logical address, LA[15:0], to 20 bit physical address, A[19:0]. However, the MMU converts addresses only in case of memory access. It does not affect addresses in case of I/O access. The MMU is consisted of the following registers and a physical address calculation unit. The operation of the physical address calculation unit is described in the later section.

Table 5-1. Register Architecture

Name		Number of Bits	Read / Write
Boundary/Base Register 1 (BBR1)		8bit	R/W
〃	2 (BBR2)	〃	R/W
〃	3 (BBR3)	〃	R/W
〃	4 (BBR4)	8bit*	R/W
Base Register	1 (BR1)	8bit	R/W
〃	2 (BR2)	〃	R/W
〃	3 (BR3)	〃	R/W
〃	4 (BR4)	〃	Read Only

*Upper 2 bit is Read Only

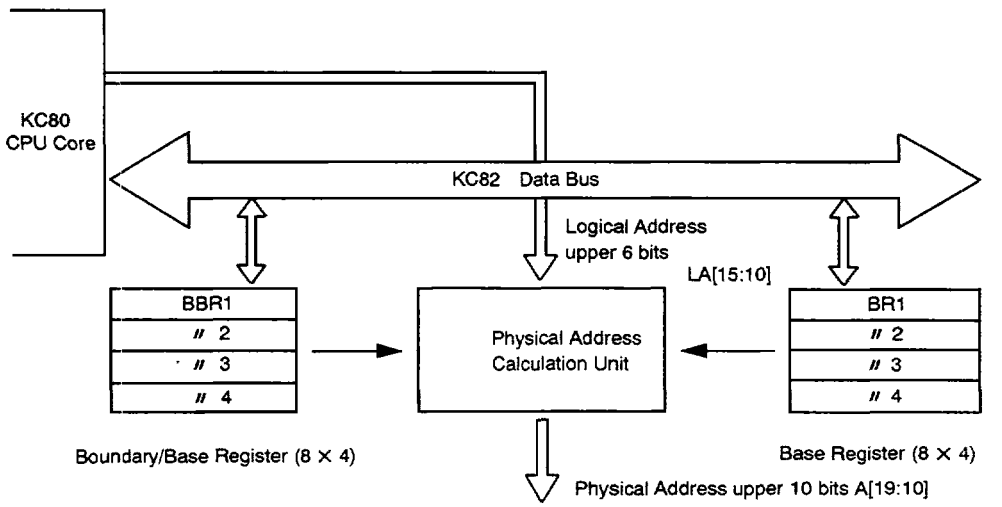


Figure 5-26. Block diagram

5.7.2 Description of registers

Eight 8 bit registers are included in the MMU block. These 8 registers hold four 10 bit data and four 6 bit data which are necessary for the MMU block.

Boundary/base registers (BBR1 ~ BBR4)

The upper 2 bits of these registers (A1<1:0> ~ A4<1:0>) are used to hold base addresses in the physical address space with the 8 bits of base registers. The lower 6 bits are logical boundary addresses (B1<5:0> ~ B4<5:0>). They are used as the boundary addresses to divide the logical space into five regions. The upper 2 bits of BBR4 are fixed, and the contents don't change when data is written on it.

Base registers(BR1 ~ BR4)

These registers compose 10 bit physical address bases (A1<9:0> ~ A4<9:0>) with the upper 2 bits of boundary/base resisters. They are used to hold base addresses in the physical address space. The content of BR4 is fixed to F0H.

Logical boundary address and physical address base

The MMU block divides the logical address space into five regions, and each region is mapped to the physical address space. The MMU block needs five logical boundary addresses (B0 ~ B4) and five physical address bases (A0 ~ A4). The content of A0 is fixed to 000H, and that of B0 is fixed to 00H. The other data are assigned in boundary/base registers and base registers as shown in Figure 5-27.

I/O Address	Register Name	bit7	6	5	0
00H	BBR1	A1<1:0>		B1<5:0>	
01H	BR1	A1<9:2>			
02H	BBR2	A2<1:0>		B2<5:0>	
03H	BR2	A2<9:2>			
04H	BBR3	A3<1:0>		B3<5:0>	
05H	BR3	A3<9:2>			
06H	BBR4	A4<1:0>	00B Fixed	B4<5:0>	
07H	BR4	A4<9:2>	F0H Fixed		

Figure 5-27. MMU Register Mapping

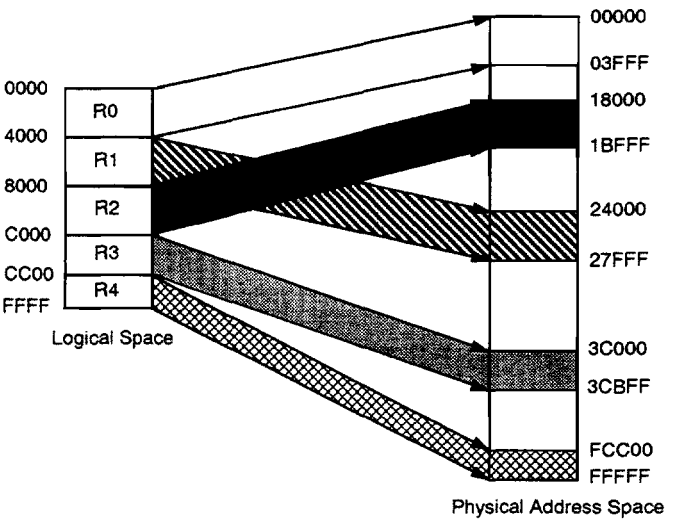
5.7.3 Calculation of physical address by MMU

The logical address space is divided into five regions (R0 ~ R4) as shown in Figure 5-28. The five regions are decided with logical boundary address (B0<5:0> ~ B4<5:0>). The lower limit address of Rn region is (Bn+1)*400H, and the higher limit address of Rn region is (B(n+1)+1)*400H-1. (The R0's lower limit address is always 00000H and the R4's higher limit address is always FFFFFH.) These regions in the physical space are created by adding the physical address base (10 bits) of divided each region and the upper 6 bits of logical address. (See Figure 5-29.) The division of logical space is decided by comparison of the upper 6 bits of logical address and the logical boundary address (B0 ~ B4). Setting of 1K byte unit is possible. The lowest value of the R1 region's lower limit address is 0400H and the content of A0 is fixed to 000H. Therefore, the lowest 1K byte of logical space (0000H ~ 03FFH), which is belong to R0 region, is always mapped to 00000H ~ 003FFH in the physical space.

An example is shown in Figure 5-28. For example, bit data are set as following;

B0 = 00H (Fixed)
B1 = 0FH
B2 = 1FH
B3 = 2FH
B4 = 32H
A0 = 000H (Fixed)
A1 = 080H
A2 = 040H
A3 = 0C0H
A4 = 3C0H (Fixed)

The relationship of five logical regions and physical regions is followed.



Region R0	Logical Address 0000H ~ 3FFFH	Physical Address 00000H ~ 03FFFH
Region R1	Logical Address 4000H ~ 7FFFH	Physical Address 24000H ~ 27FFFH
Region R2	Logical Address 8000H ~ BFFFH	Physical Address 18000H ~ 1BFFFH
Region R3	Logical Address C000H ~ CBFFH	Physical Address 3C000H ~ 3CBFFH
Region R4	Logical Address CC00H ~ FFFFH	Physical Address FCC00H ~ FFFFFH

Figure 5-28. Example of Relationship of Logical Address and Physical Address

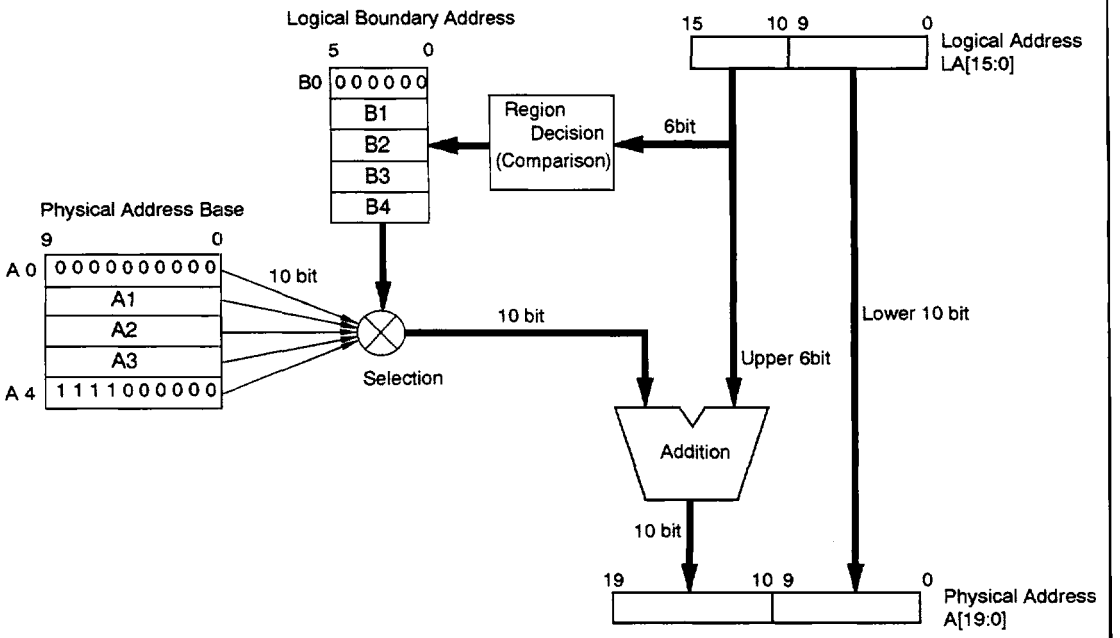


Figure 5-29. Physical Address Calculation

5.7.4 MMU mechanism

(1) Memory space

The MMU converts the address when the CPU accesses memory space. This is divided into following cases.

1.Instruction fetch
2.Read or write to memory space by instruction
3.Instruction fetch to interrupt restart address
4.Start address table in mode 2 interrupt

(2) I/O space

When the CPU accesses I/O space, the MMU does not convert the address and outputs the logical address. In this case, 0H is put on upper 4 bits of the address.

(3) DMA controller

When a DMA controller (KP37 etc.) is on the chip, the MMU does not convert the address which the DMA outputs. In case of KP37, a special register is necessary for each channel to assign the upper 4 bits of the physical address.

5.7.5 Reset

The registers are initialized when they are reset as following.

B0 = 00H (Fixed)
B1 = 3FH
B2 = 3FH
B3 = 3FH
B4 = 3FH
A0 = 000H (Fixed)
A1 = 000H
A2 = 000H
A3 = 000H
A4 = 3C0H (Fixed)

When they are reset, there is only R0 region in the logical address space. 64K byte of the logical address space is mapped to the lowest 64K byte in the physical address space.

5.7.6 Precautions on the use of the MMU

(1)When the logical boundary address is set as B0 < B1 < B2 < B3 < B4, all regions are valid. When the logical boundary addresses are set in opposite order or equally, the region whose region number is bigger is valid. The region whose region number is smaller is disappears. For example, if B1 >= B2 is set, R1 region disappears.

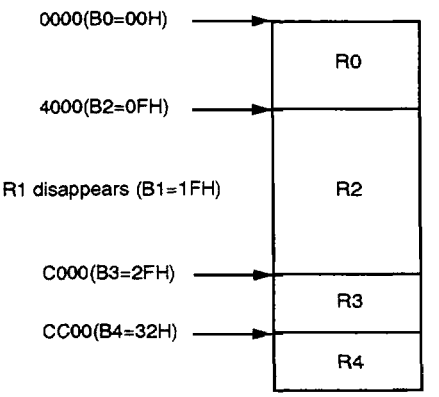


Figure 5-30. Example when B1 ≧ B2

(2)When data is written in MMU registers (boundary/ base registers, base registers), the set value is valid from the next bus cycle of I/O write cycle in which data is written in MMU registers.

(3)There is no hardware restriction for the use of each region(R0 ~ R4). However, the following assignments are recommended.

R0: resident program region (common program, interrupt vector, etc.)
R1: program bank window
R2: data bank window 1 (source)
R3: data bank window 2 (destination)
R4: resident data region (stack etc.)

(4)This MMU circuit occupies the following I/O addresses. Note that these I/O addresses cannot be assigned for user's I/O.

Table 5-2 I/O. Mapping MMU circuit uses.

I/O Address	Register Name
00H	BBR1
01H	BR1
02H	BBR2
03H	BR2
04H	BBR3
05H	BR3
06H	BBR4
07H	BR4
08H ~ 0FH	Reserved For Kawasaki Steel corp.

(5)The logical boundary address of R1 is decided by the R1 logical boundary address (B1). Because the lower limit address of the R1 region is (B1+1)*400H, the lowest value of the R1 region lower limit address is 0400H. On the other hand, the lower limit address of the R0 region is 0000H. Therefore, the lowest 1K byte of logical address is always belong to the R0 region. This lowest 1K byte is always mapped to the fixed location, 00000H ~ 003FFH.

(6)Set a value under 3FH to B1 ~ B4. 3FH to B1 ~ B4 is an invalid value and the corresponding region disappears.

6. Interrupt Controller

6.1 General description

The KL5C8080A1212 includes a KP69 macro cell as interrupt controller. The KP69 is a small-sized interrupt controller developed exclusively for microcontroller based on our CPUs (KC80 or KC82). The KP69 can support 16 levels of interrupt request inputs for the KC80 or KC82's Mode 2 interrupt. Each interrupt request can be set to the HIGH or LOW priority group, and within each group a higher number of interrupt request input level has higher priority. The edge/level operation and mask condition of each interrupt request input, higher 3 bits of interrupt vector can be programmed.

When the KP69 receives an interrupt request, it determines its mask condition and priority and sends the INT_ signal to the CPU. When the IACK_ signal returns from the CPU, it sends a programmed interrupt vector to the data bus. It also recognizes the end of interrupt service routine with EOI_ = "L" from the CPU. This EOI_ signal goes low while executing the RETI instruction. Therefore, if the RETI instruction is placed at the end of interrupt service routine, the KC69 automatically recognizes the RETI instruction and the end of interrupt service routine. The KC69 has also an illegal interrupt detection capability.

Features

- Support KC80/KC82 mode 2 interrupt.
- The priorities of 16 level interrupt request inputs can be programmed.
- Each interrupt request maskable.
- Nested interrupt operation possible.
- The edge/level mode of interrupt request input can be selected.
- Illegal interrupt detection capability
- Automatically detects the execution of the CPU's RETI instruction and the end of interrupt service routine.

6.2 Block diagram

The following is the block diagram of KC69. The KC69's IACK_ input, EOI_ input and INT_ output are internally connected to the CPU's IACK_ output, EOI_ output and INT_ input respectively. The signals shown in the table are connected to IR[15:0].

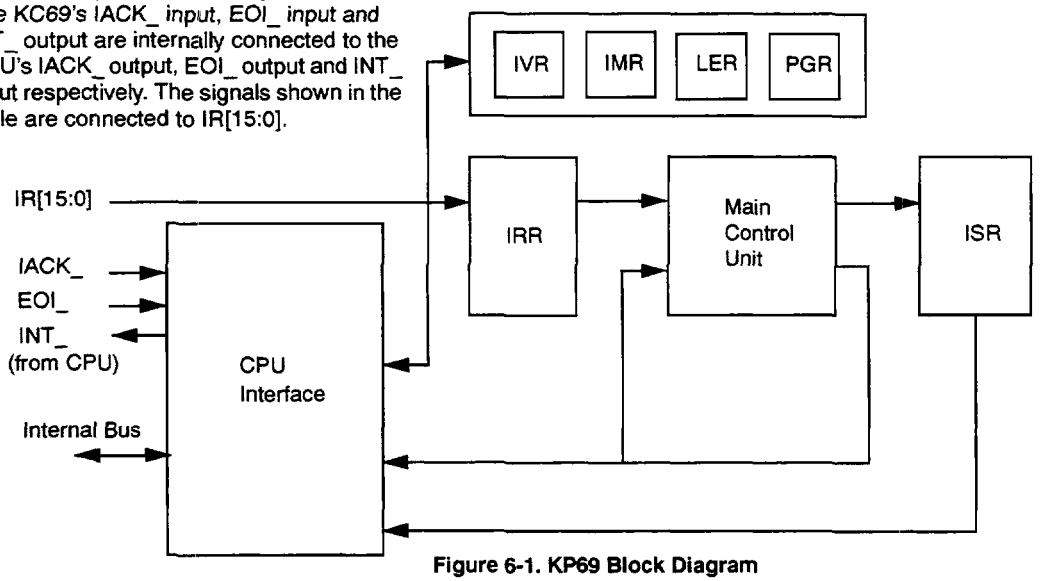


Figure 6-1. KP69 Block Diagram

Table 6-1. KL5C80A12 Interrupts

	Source of interrupt request
IR[15]	Timer/Counter B channel 2 OUTS output
IR[14]	Timer/Counter B channel 1 OUTS output
IR[13]	Timer/Counter B channel 0 OUTS output
IR[12]	Timer/Counter A channel 1 OUT output
IR[11]	Timer/Counter A channel 0 OUT output
IR[10]	USART TXEMPTY output
IR[9]	USART RXRDY output
IR[8]	USART TXRDYPIN output
IR[7]	External input P07/IR7
IR[6]	External input P06/IR6
IR[5]	External input P05/IR5
IR[4]	External input P04/IR4
IR[3]	External input P03/IR3
IR[2]	External input P02/IR2
IR[1]	External input P01/IR1
IR[0]	External input P00/IR0

6.3 Register architecture and I/O mapping

KP69 contains the following registers.

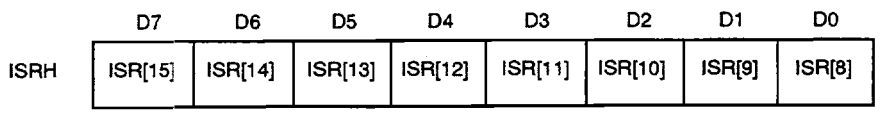
Table 6-2. I/O Mapping

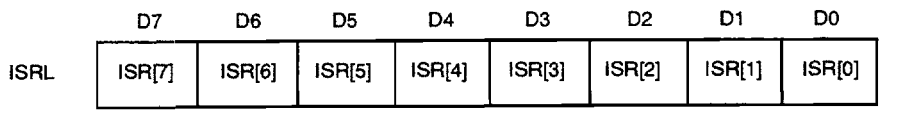
I/O address	Block	Write cycle	Read cycle
34H	Interrupt controller	LERL/PGRL	ISRL
35H		LERH/PGRH	ISRH
36H		IMRL	IMRL
37H		IVR / IMRH	IMRH

IRR (Interrupt Request Register)
(not readable, not writable)
The corresponding bits are set when an interrupt request is generated. When the interrupt request is accepted in the level mode, this register is reset by disappearance of the request. When the interrupt request is accepted in the edge mode, it is reset by the start of interrupt service. The edge detection circuits of all interrupt levels are reset right after the reset.

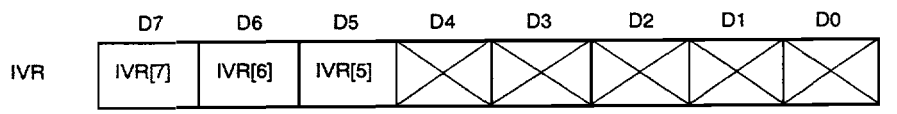
ISR (In Service Register)
(read only)
When the IACK_ signal is returned, the bits corresponding to the accepted interrupt request are set. They are reset at the end of request service. All bits are reset right after the reset. Each 8 bits can be read.

IVR (Interrupt Vector Register)
(write only)
This register specifies higher 3 bits of interrupt vector of the KC82 Mode 2 interrupt. Writable registers change when data is written into this register.





ISR[n]	description
0	Out of service
1	In service



LER (Level / Edge Register)
(write only)
This register controls the level/edge mode of each interrupt request input (hereinafter referred to as "IR input"). The mode can be set for each IR input. All bits are set to the level mode right after the reset. Be sure to write data to this register before setting IVR. This register is write only.

Note: To use an interrupt of timer/counter, the corresponding IR input should be set to the edge mode.

PGR (Priority Group Register)
(write only)
This register controls priority groups of IR inputs. The priority group can be set for each IR input. There are two kinds of priority groups; "HIGH" and "LOW". All bits are set to LOW right after the reset. Be sure to write data to this register after setting IVR. This register is write only.

	D7	D6	D5	D4	D3	D2	D1	D0
LERH	LER[15]	LER[14]	LER[13]	LER[12]	LER[11]	LER[10]	LER[9]	LER[8]

	DI7	DI6	DI5	DI4	DI3	DI2	DI1	DI0
LERL	LER[7]	LER[6]	LER[5]	LER[4]	LER[3]	LER[2]	LER[1]	LER[0]

LER[n]	description
0	LEVEL mode
1	EDGE mode

	D7	D6	D5	D4	D3	D2	D1	D0
PGRH	PGR[15]	PGR[14]	PGR[13]	PGR[12]	PGR[11]	PGR[10]	PGR[9]	PGR[8]

	D7	D6	D5	D4	D3	D2	D1	D0
PGRL	PGR[7]	PGR[6]	PGR[5]	PGR[4]	PGR[3]	PGR[2]	PGR[1]	PGR[0]

PGR[n]	description
0	"LOW" group
1	"HIGH" group

IMR (Interrupt Mask Register)
(readable, writable)
This register mask IR inputs. The mask state can be set for each IR input. All bits are set to "1" right after the reset. Be sure to write data to this register after setting IVR. This register is write/read enable.

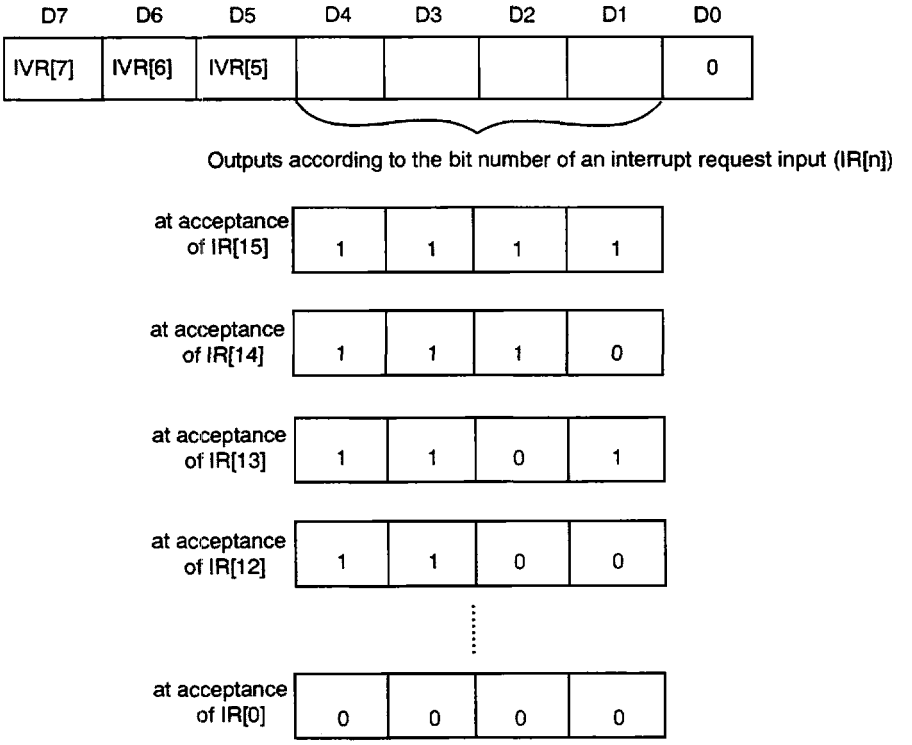
	D7	D6	D5	D4	D3	D2	D1	D0
IMRH	IMR[15]	IMR[14]	IMR[13]	IMR[12]	IMR[11]	IMR[10]	IMR[9]	IMR[8]

	D7	D6	D5	D4	D3	D2	D1	D0
IMRL	IMR[7]	IMR[6]	IMR[5]	IMR[4]	IMR[3]	IMR[2]	IMR[1]	IMR[0]

IMR[n]	description
0	non-masked state
1	masked state

6.4 Interrupt vector output

The following is the output format of interrupt vector.
The Figure 6-2 shows the interrupt vector output as response to IACK_.



bits 7 to 5: data set in IVR
bits 4 to 1: binary code of accepted interrupt level
bit 0 : always 0

Figure 6-2. Interrupt Vector

6.5 Priorities of interrupt requests

The priorities of interrupt request inputs are set in descending order of bit numbers right after the reset. The priorities can be changed in terms of priority group. There are two kinds of priority groups ("LOW" and "HIGH") and the priority group can beset for each IR. The IRs of "HIGH" group have higher priority over those of "LOW" group. Within each group, an IR with higher bit number has higher priority. The following is an example of IR priorities.

Interrupt request input	Group
IR[15]	LOW
IR[14]	LOW
IR[13]	LOW
IR[12]	HIGH
IR[11]	LOW
IR[10]	HIGH
IR[9]	LOW
IR[8]	HIGH
IR[7]	HIGH
IR[6]	LOW
IR[5]	LOW
IR[4]	HIGH
IR[3]	LOW
IR[2]	HIGH
IR[1]	HIGH
IR[0]	HIGH

⇒

Interrupt request input	Group	Priority
IR[12]	HIGH	Highest
IR[10]	HIGH	↓
IR[8]	HIGH	↓
IR[7]	HIGH	↓
IR[4]	HIGH	↓
IR[2]	HIGH	↓
IR[1]	HIGH	↓
IR[0]	HIGH	↓
IR[15]	LOW	↓
IR[14]	LOW	↓
IR[13]	LOW	↓
IR[11]	LOW	↓
IR[9]	LOW	↓
IR[6]	LOW	↓
IR[5]	LOW	↓
IR[3]	LOW	Lowest

Figure 6-3. Interrupt Request Priority

6.6 Register setting sequence

I/O addresses are assigned to the internal registers of KC69 as shown in Table 6-2. So initialization of registers after clearing the reset is performed in the following order: LER, IVR and IMR (or PGR). It should be noted that only IMR or PGR can be set after setting IVR.

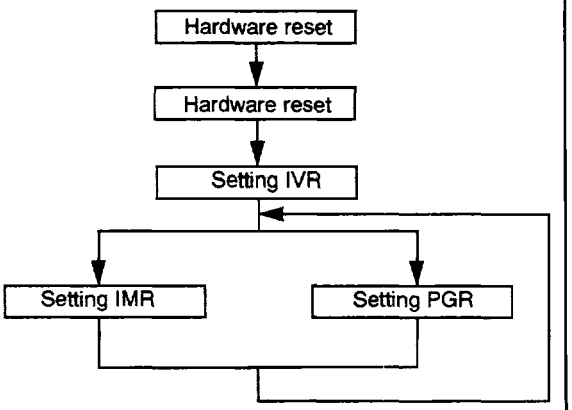


Figure 6-4. Register Setting Sequence

6.7 Readout of registers

Of the KP69 registers, ISR and IMR are always readable. Readout of ISR is required to determine an illegal interrupt.

6.8 Acceptance of interrupt request

There are two modes (level and edge) for the acceptance of interrupt request and they are set by LER.

Level mode

In level mode, "H" level of IR input is recognized as interrupt request.

Edge mode

In edge mode, the rising edge of IR input is recognized as interrupt request. In this case, the interrupt request is held until accepted.

Operation sequence

The following is the operations when an interrupt request is generated at IR[n] pin.

When IR input goes "H" to generate an interrupt request, the corresponding bits of IRR are set. This interrupt request becomes an INT_ signal through the mask conditions determined by IMR and the priority decision by ISR and PGR. The KC82 receives the signal and makes the IACK_ signal "L". When the KP69 receives this IACK_ signal, it outputs an interrupt vector corresponding to the IR[n] where the interrupt request has been generated to set the corresponding bit of ISR and (in edge mode) reset the corresponding bits of IRR. Then the CPU enters the interrupt service routine. When ISR is set, acceptance of an interrupt request in edge mode is enabled.

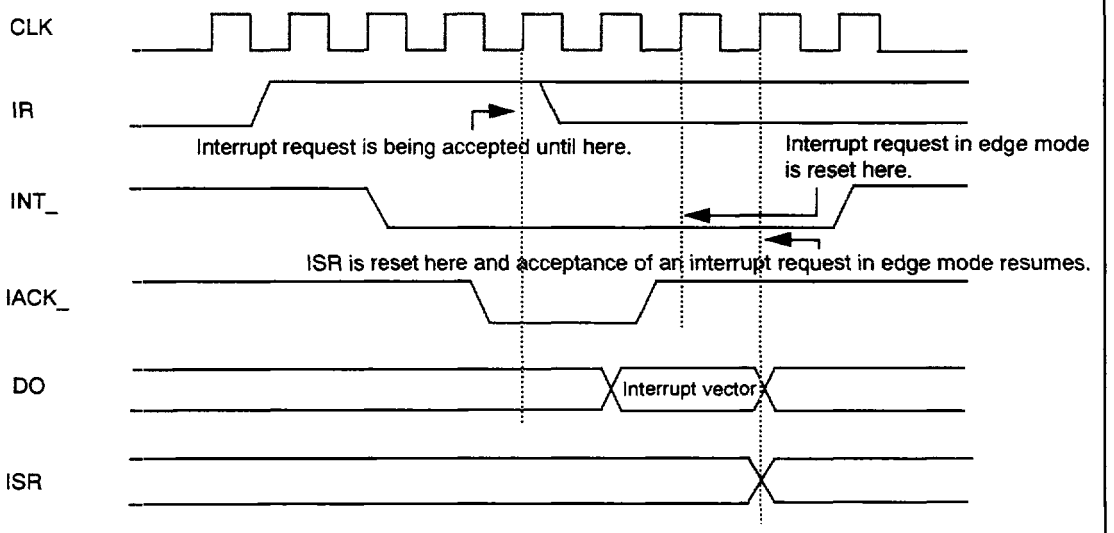


Figure 6-5. Interrupt Request Accepting Timing

6.9 End of Interrupt

When the EOI_ signal from the KC82 goes "L", the KP69 resets the ISR bits corresponding to the highest priority level of the interrupt being serviced. This terminates the interrupt service of that level.

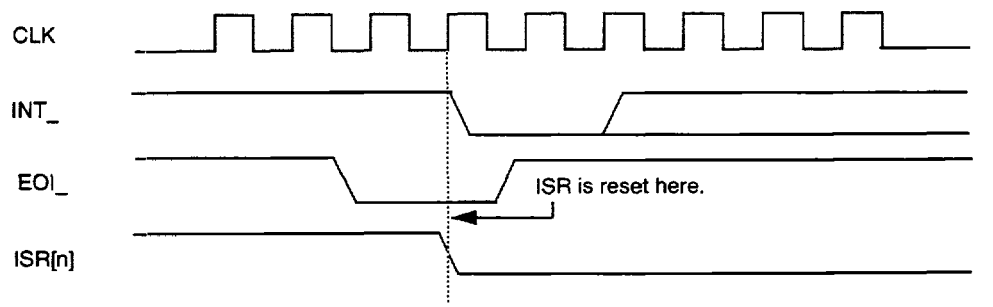


Figure 6-6. Interrupt Terminating Timing

6.10 Nested interrupt operation

The KP69 compares the interrupt level of a new interrupt request with that of the interrupt request being serviced in terms of the predetermined priority. When the interrupt level of a new interrupt request has higher priority than that of the interrupt request being serviced, the KP69 accepts a new interrupt request, otherwise the KP69 rejects it or keeps it wait. This implements a nested interrupt operation. The state of nested interrupts can be obtained by reading ISR.

6.11 Illegal interrupt operation

An illegal interrupt indicates the following state: when an interrupt request input set to the level mode made INT_ signal "L" and the KC82 returned IACK_ as response, the interrupt request has already disappeared and there is no higher interrupt request input than the interrupt request being serviced with the highest priority. In this state the KP69 does not set ISR and outputs an interrupt vector corresponding to IR[0], entering an illegal interrupt operation state which lasts until the EOI_ from the KC82. ISR is not reset by this EOI_. Any interrupt request (legal or illegal) would not be accepted in the illegal interrupt operation state. Therefore, during the IR[0] interrupt service it is necessary to distinguish a normal interrupt request or an illegal interrupt request based on ISR[0]. When the KP69 goes to an illegal interrupt request state during the IR[0] interrupt service, the readout of ISR gives ISR[0]=0. This indicates that ISR[0]=1 represents a legal interrupt request and ISR[0]=0 represents an illegal interrupt request.

6.12 Reset

When the RESET_ goes "L", the following operations are performed.

(1) IMR is set to "FFFFH". (All levels in masked state)
(2) IRR, ISR, LER and PGR are reset to "0000H".
(3) Illegal interrupt operation state is disabled.
(4) IVR is placed in non-set state. (Initialization is required.)

6.13 Precautions

(1) Be sure to use Mode 2 for CPU interrupt mode.
(2) Be sure to place RETI instruction (source code: ED 4D) at the end of interrupt service routine.
(3) To use an interrupt of timer/counter, the corresponding IR input should be set to the edge mode.

6.14 Advanced Applications

The external outputs IR7 to IR0 are always connected to the input buffer of pins 80 to 83 and 85 to 88. When the ports are set to the input direction, the conditions of interrupt request on IR7 to IR0 can be checked by readout of the ports. This is very useful for checking the presence of nested interrupt requests. When the ports are set to the output direction, an external interrupt can be generated through the output ports by software. Read the following precautions about how to use these pins.

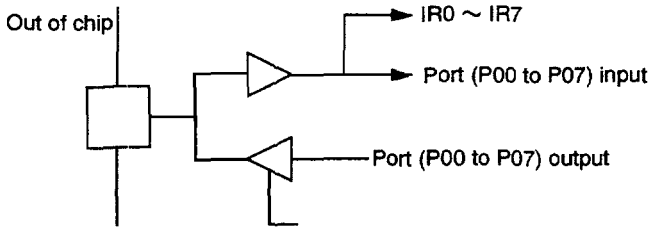


Figure 6-7. Structure of Pins 80 to 83 and 85 to 88

Precautions:

(1) when placing an external input to IR7 to IR0
Set the parallel ports to the input direction.

(2) when setting the parallel ports to the output direction
The values to be output from the parallel ports are input to IR7 to IR0. Mask the corresponding external request level when not using IR7 to IR0.

7. USART

7.1 General description

The USART in the chip is KP51 macro cell. KP51 is a programmable serial communication macro cell. Once initialized, it transmits and receives data according to the mode. This macro cell is compatible with popular 8251, but much faster than it.

Synchronous mode and asynchronous mode
KP51 provides the following two modes.

(1) Synchronous (sync) mode
- character length: 5 to 8 bits
- number of synchronous characters: 1 to 2 characters
- parity bit odd/even/non parity bit, internal synchronization detection/external synchronization detection

(2) Asynchronous (async) mode
- character length: 5 to 8 bits
- stop bit length: 1, 1.5, or 2 bits
- parity bit odd/even/non parity bit
- baud rate: x 1, x 6, x 64

Error detection during data reception
The following errors can be detected as status information while receiving data.
- parity error (when parity is enabled)
- overrun error
- framing error (async mode only)

Sending and detecting a break signal

Transmit enable and receive enable flags
When the transmit enable flag is set to '1', it is enabled to send data. When it is set to '0', transmit data stored in the buffer is sent and the transmit data pin goes '1' (marking state).

When the receive enable flag is set to '0', in async mode all receive operations (reception of data and change of flag) are disabled, and in sync mode the reception of data is not disabled and RXRDY and an overrun flag are reset. When it is set to '1', normal operations are performed.

Enter hunt feature
When detecting sync characters in sync mode, an enter hunt command enables sync characters to be detected at each bit boundary.

7.2 KP51 internal pin description

pin name	I/O	description
TXRDYPIN	O	Transmitter ready signal output When transmit data can be written and transmit is enabled (CTS_="H" and txen= '1'), it goes "H". When transmit data is written, it goes "L" and data transfer starts if in transmit enable state. As the KP51 has double internal buffer structure, this signal goes "H" to inform that a next transmit data can be written when data transfer starts. This signal differs from the bit 0 of status register (TXRDY) in the following points. Status register bit 0 - "H" when write of transmit data is enabled TXRDYPIN signal - "H" when transmit data can be written and transmit is enabled (CTS_= "L" and txen= "1").
TXEMPTY	O	Transmitter empty output It outputs "H" when there is no data to be sent. When transmit data is written, it outputs "L". This signal remains "H" in transmit disable state (CTS_= "H" and txen= '0'). In sync mode, when data is written and transmitted, and there is no data to be sent, it keeps automatically outputting sync characters. Even at this time TXEMPTY outputs "H".
RXRDY	O	Receiver ready signal output It outputs "H" when one character data is received and readout of data is enabled. It is reset to "L" when data is read out. The execution of a receive disable command (rxen= '0': setting the command register bit 2 to '0') also resets this signal to "L". In sync mode it is reset to "L" by issuing the enter hunter command (setting the command register bit 7 to '1').

Refer to Chapter 3 for the KP51's external I/O pins (CTS_, RXC,DSR_, RXD, TXD, TXC, DTR_, RTS_, SYNDBD, SYDTIN).

7.3 Block diagram

The following is the KP51 block diagram. The KP51 internal pins (TXRDYPIN, TXEMPTY, RXRDY) are connected to the interrupt controller.

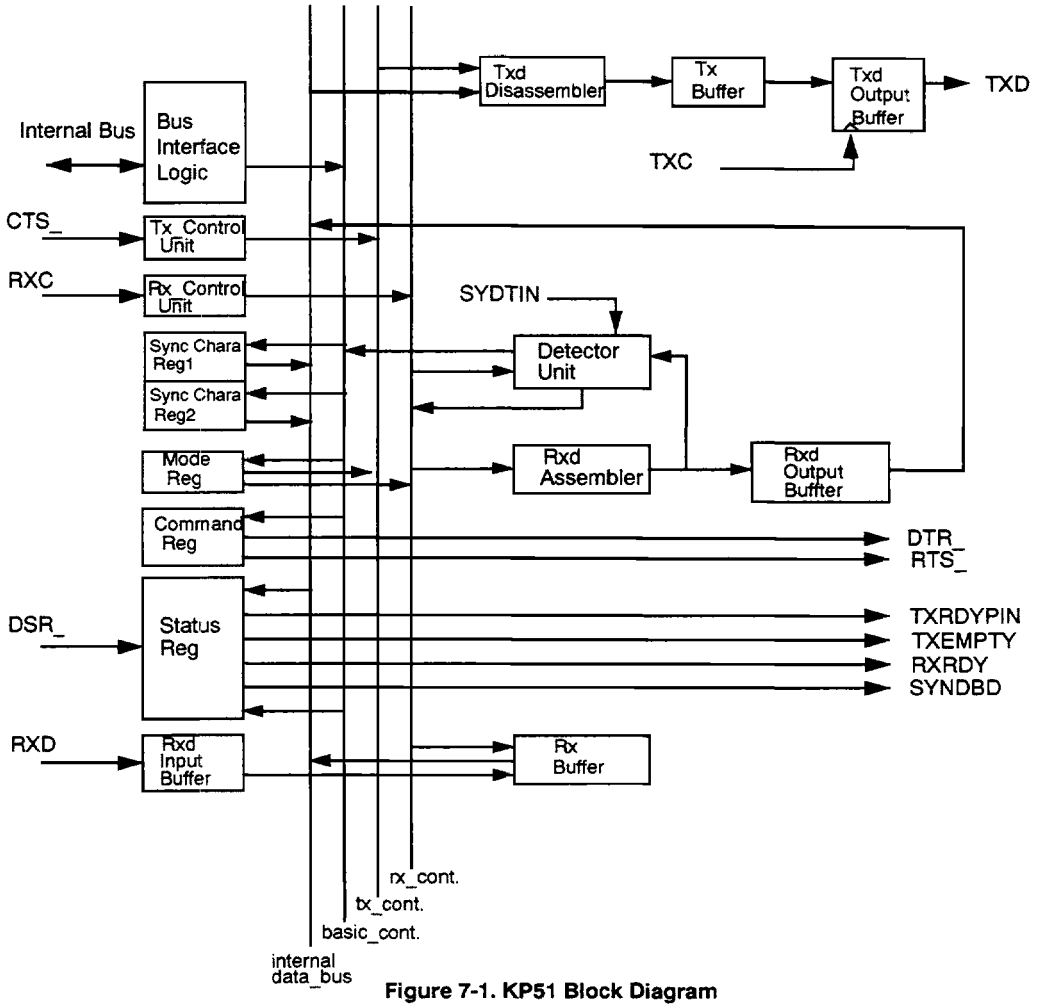


Figure 7-1. KP51 Block Diagram

I/O Register Mapping

I/O address	Block	Write cycle	Read cycle
38H	USART	Transmit data	Receive data
39H		Mode/command/sync character	Status

7.4 Initialization

To start transmitting or receiving data, the setting mode and command entry should be made after reset as shown in Figure 7-2.

Setting mode

When writing to I/O address 39H is made after reset, data is written to the mode register. The KP51 always waits for mode set state after reset (external RESET input or software reset command entry).

In this state, the sync or async mode is selected, and baud rate, character length and others are set. Set the mode according to the format described in the following pages.

Setting sync characters

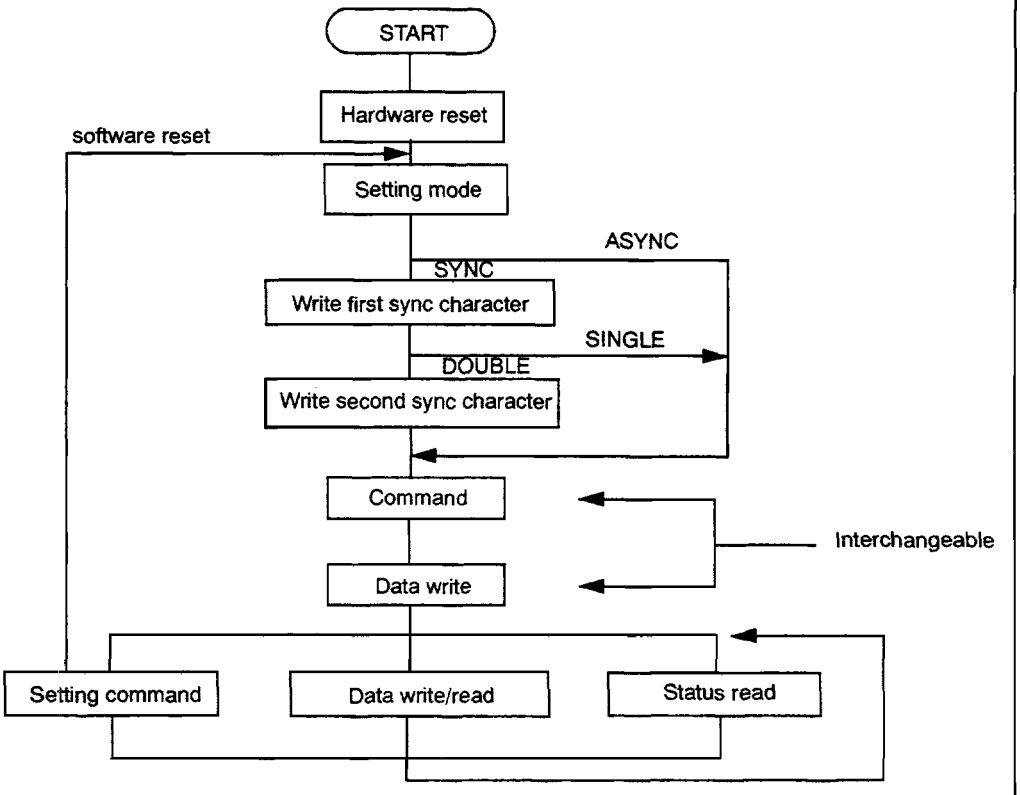
When the sync mode is selected, the KP51 then waits for the sync character. Write one sync character for single mode or two sync characters for double(bi-sync) mode. When the async mode is selected, the KP51 automatically skips this step and proceeds to the next command waiting state.

Setting command

When the mode and sync character are set, all data writes at I/O address = 39H are regarded as a command.

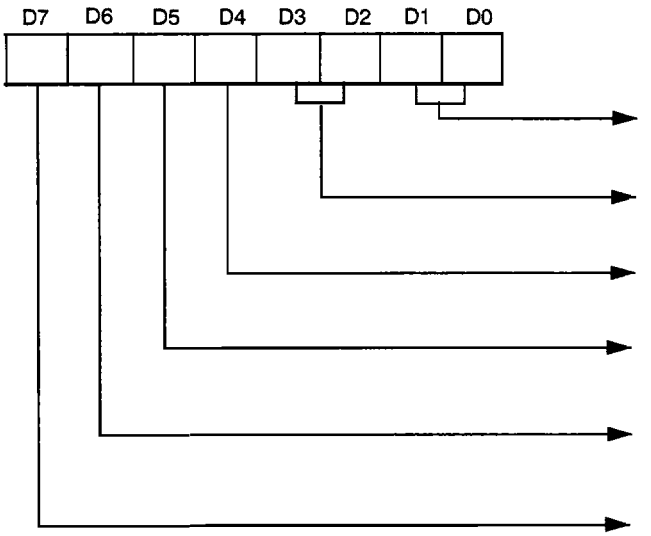
The command enables data transmission. About commands, refer to P.7-6.

Figure 7-2. Initialization Sequence Flowchart



Mode register

(1) sync mode



Both bits set to '0'

Character length	5	6	7	8
D2	0	1	0	1
D3	0	0	1	1

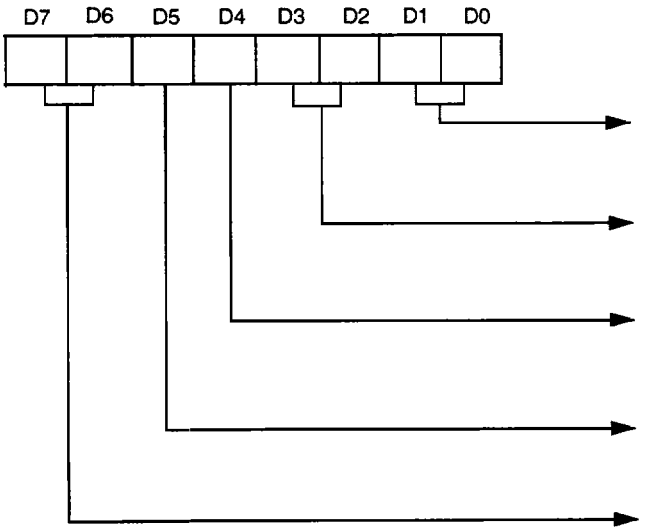
Parity enable
1 = enable
0 = disable

Parity check
1 = even
0 = odd

Sync detection
1 = external sync detection
0 = internal sync detection

Sync character
1 = single
0 = double

(2)async mode



Baud rate	x1	x16	x64
D0	1	0	1
D1	0	1	1

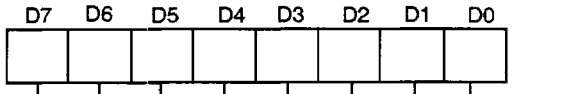
Character length	5	6	7	8
D2	0	1	0	1
D3	0	0	1	1

Parity enable
1 = enable
0 = disable

Parity check
1 = even
0 = odd

Stop bit	invalid	1	1.5	2
D6	0	1	0	1
D7	0	0	1	1

Command register



txen Transmit Enable
1 = enable
0 = disable

dtr Data Terminal Ready (DTR_ output)
1 = DTR_ "L"
0 = DTR_ "H"

rxen Receive Enable
1 = enable
0 = disable

sbrk Send Break Signal
1 = transmission break signal (TXD= "L" output)
0 = normal operation

er Error Flag Reset
1 = clear all error flags;
PE (parity error),
OE (overrun error), FE (framing error)

rts Request To Send (RTS_ output)
1 = RTS_ "L"
0 = RTS_ "H"

ir Software Reset
1 = reset (internal initialization)

eh Enter Hunt Mode (sync detection)
1 = enters hunt mode
(detection of sync character)

* D7, D6 and D4 of the command register are single operations so that clearing each bit is not necessary.

7.5 Operation modes

The operations in each mode are described below.

Data transmission in async mode

When data is written from the CPU, character data is disassembled to the transmit format with start bit, parity bit and stop bit based on the conditions specified by the mode register.
When a transmission break command (sbrk= '1') is issued, TXD outputs "L".
When a transmission break command (sbrk= '1') is not issued and transmit is enabled (CTS_= "L" or txen= '1'), data is transmitted. Before transmission the TXD signal remains "H" (marking state) until transmit data is written. When there is no data to be sent after transmitting 1 character, it returns to the marking state.
Transmit data is sent in synchronization with the rising edge of the TXC signal at 1, 1/16 or 1/64 times of the TXC according to the set baud rate.

Data reception in async mode

When the KP51 detects RXD input "H" at receive enable state (rxen= '1'), it waits until the RXD goes "L". When the RXD goes "L", a start bit becomes effective. If the KP51 samples a "L" in the center of a bit, it identifies it as start bit and starts receiving data. It detects a parity error, framing error and overrun error during data reception. Each of these errors can be checked by the readout of status register described later. These errors do not affect the receive operation itself.
Framing error is recognized not by the number of stop bits, but by the first stop bit being "L".

Data transmission in sync mode

Even at transmit enable state (CTS_= "L" and txen= '1') the TXD signal remains "H" (marking state) until a first data is written. When there is data to be sent, data is transmitted in synchronization with the rising edge of TXC. Next transmit data should be provided before there is no data to be sent (TXEMPTY goes "H"). When data is provided late, sync characters are automatically sent repeatedly.
When transmitting sync characters, the TXEMPTY signal is "H" until the next data is provided, and it goes "L" when data is provided.
Sync characters are not sent unless the first transmit data is written.

Data reception in sync mode

There are two kinds of sync detection; internal sync detection and external sync detection. In sync mode, an enter hunt command should be issued (bit 7 of the command register set to '1') before start receiving data. The enter hunt command enables async character to be detected.
In internal sync detection, a sync character is searched for a match at each rising edge of RXC. In bi-sync mode, the second sync character is also compared after the first sync character match (two contiguous sync characters should be received).
When the detection of sync characters is completed, SYNDBD pin goes "H" and the KP51 exits the hunt mode. The SYNDBD pin is usually set in the center of the final bit of data. But if data contains a parity bit, it is set in the center of the parity bit.
The detection of sync characters is performed at each specified character boundary even after the end of the hunt mode.
In external sync detection, it is recognized as completion of sync detection that the SYDTIN pin is "H" for more than one cycle of RXC, and then the KP51 exits the hunt mode. At this time the SYNDBD pin is set to "H" as in internal sync detection.
In this external sync detection, the internal sync detection is not performed.
In sync mode, data reception is performed regardless of whether transmit is enabled or disabled. As in async mode, detection of parity error is performed except in the hunt mode.

7.6 Status register

This 8-bit register stores the status information on data transmission and reception. Readout of this register allows you to make polling operation.

The following is the bit assignment of the status register. It should be noted that the set conditions may be different from those of external pins in some status.

D7	D6	D5	D4	D3	D2	D1	D0

D0 TXRDY
Set conditions are different from those of TXRDYPIN.
Status = '1' when the internal Data Buffer is empty
TXRDYPIN = "H" when the internal Data Buffer is empty, CTS_= "L" and Command Register bit 0 (txen)= '1'

D1 RXRDY
The same function as the RXRDY pin.

D2 TXEMPTY
The same function as the TXEMPTY pin.

D3 PE
Parity Error
It goes '1' when a parity error is detected in received data.
It changes regardless of whether receive is enabled (Command Register bit 2 (rxen)= '1') or disabled (rxen= '0').
In sync mode, it does not function in the hunt mode.
It is cleared to '0' by Error Clear command (Command Register bit4 (er)= '1').

D4 OE
Overrun Error
It is set to '1' when readout of data is delayed after its reception and the next data has already been received in receive enable state (rxen= '1').
It is cleared to '0' by an Error Clear command or Receive Disable command.

D5 FE
Framing Error
It is set to '1' when '0' is detected at stop bits in async mode.
Sampling is made only in the center of the first stop bit regardless of stop bit length.
It is cleared to '0' by an Error Clear command.

D6 SYBD
Sync character and Break signal Detect
It outputs the same value as the SYNDBD pin except that it does not change during status readout.
In sync mode, it is set to '1' when a sync character is detected and it is cleared to '0' by status read out regardless of whether external or internal sync detection.
In async mode, it is set to '1' when more than two characters of'0' are received as break signal (RXD input = "L") after start bit has been detected, and it is cleared to '0' by detection of RXD= "H" or the reset.

D7 DSR
Data Set Ready
It is set to '0' when DSR_= "H" and set to '1' when DSR_= "L".
It is read in synchronization with CLK.

* The content of Status Register is not updated during readout.

7.7 Precautions

Initialization

When performing initialization with a software reset command (setting the Command Register bit 6), be sure to write '00H' three times to the command register and then execute the software reset command.

Execution of Enter Hunt command

When setting the Receive Enable flag (setting the Command Register bit 2) and issuing an Enter Hunt command (setting the Command Register bit 7), be sure to execute an Error Clear command (set the Command Register bit 4) at the same time.

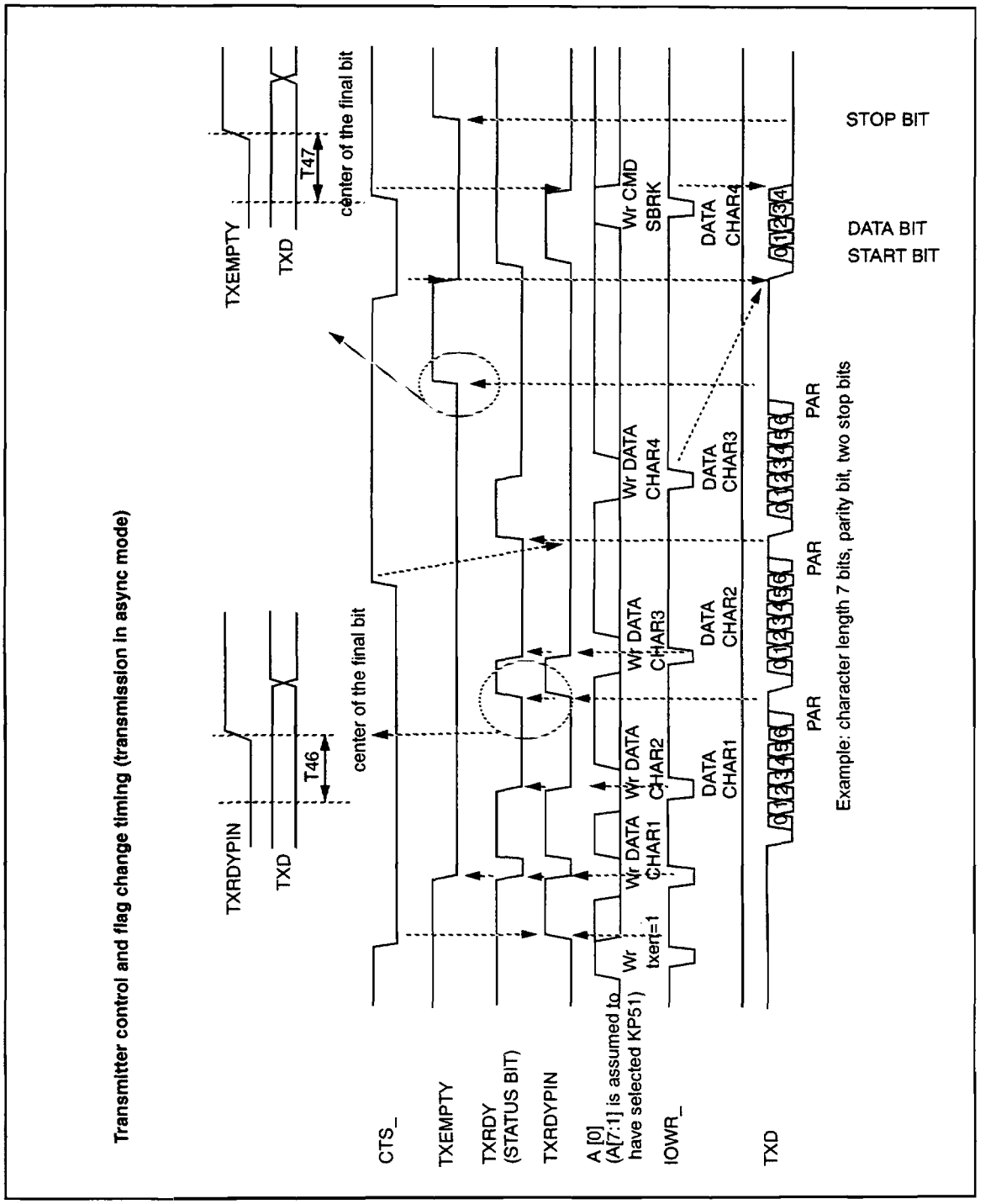
Data received after framing error

Data received after framing error is not guaranteed.

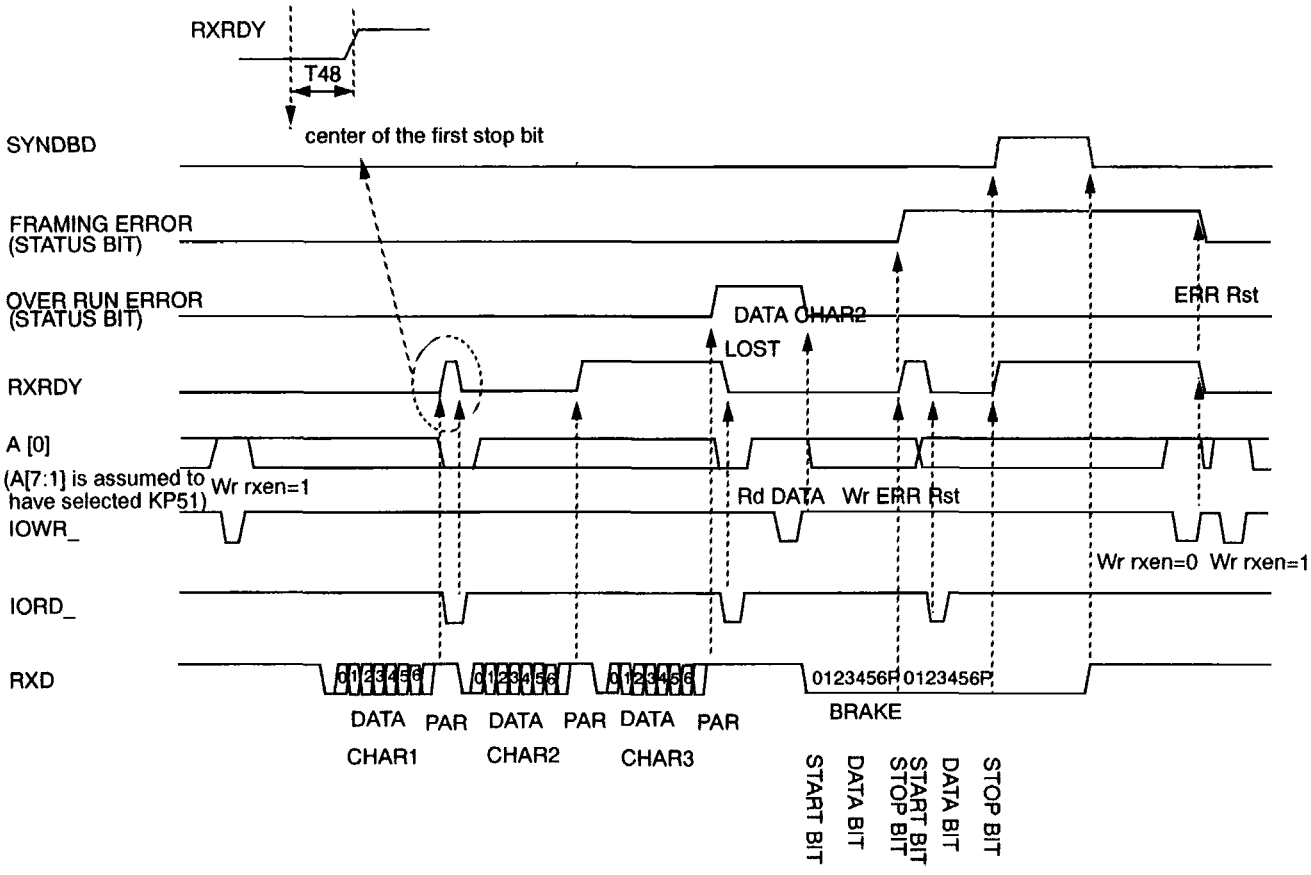
External sync detection

In external sync detection, SYDTIN input is always recognized when RXC= "H". Therefore, when SYNDBD signal and sync detect status (Status Register bit 6) are cleared by status read, SYNDBD signal and sync detect status are reset if SYDTIN signal is "H" at RXC= "H".

Transmitter control and flag change timing (transmission in async mode)

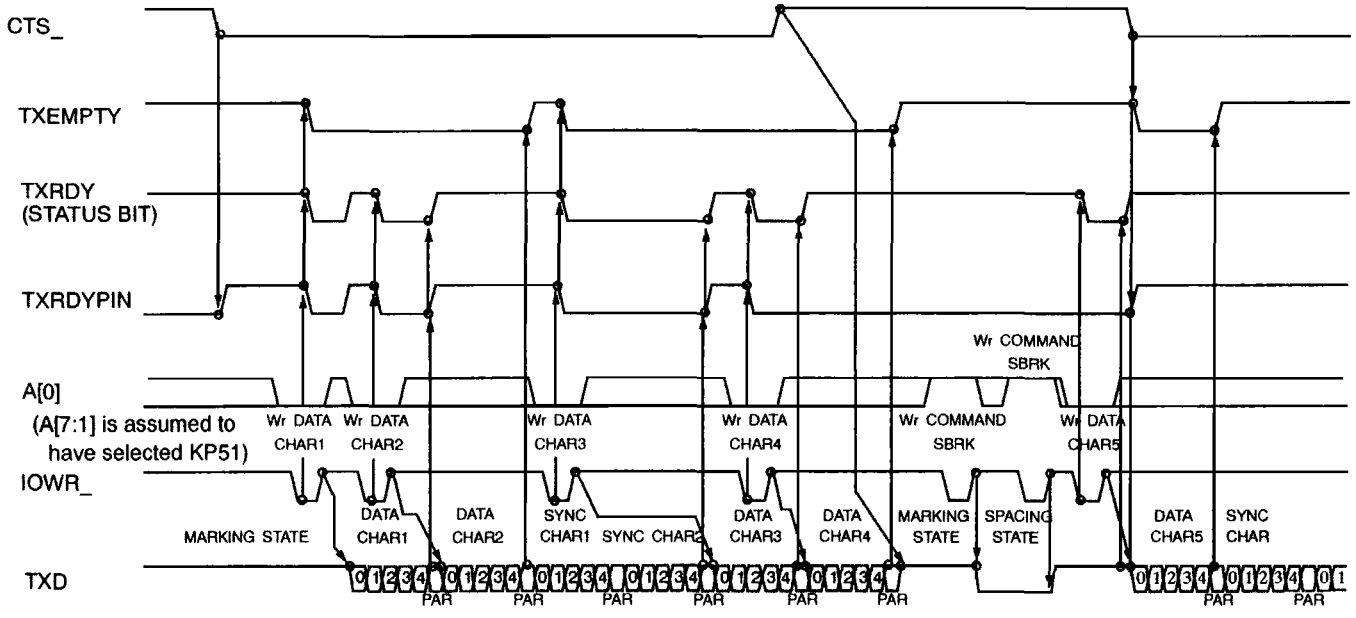


Receiver control and flag change timing (reception in async mode)



Example: character length 7 bits, parity bit, two stop bits

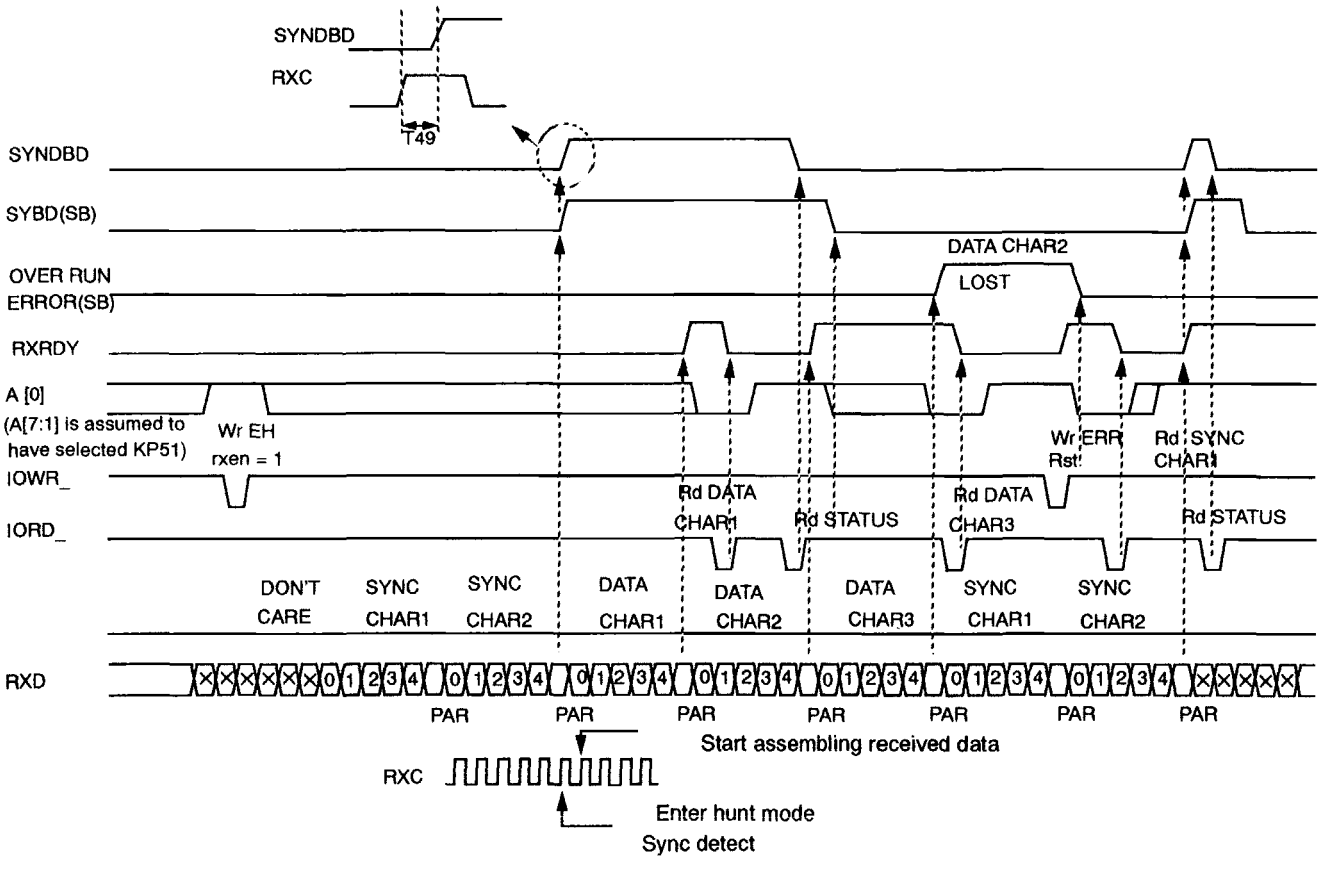
Transmitter control and flag change timing (transmission in sync mode)



Example: character length 5 bits, parity bit, two sync characters

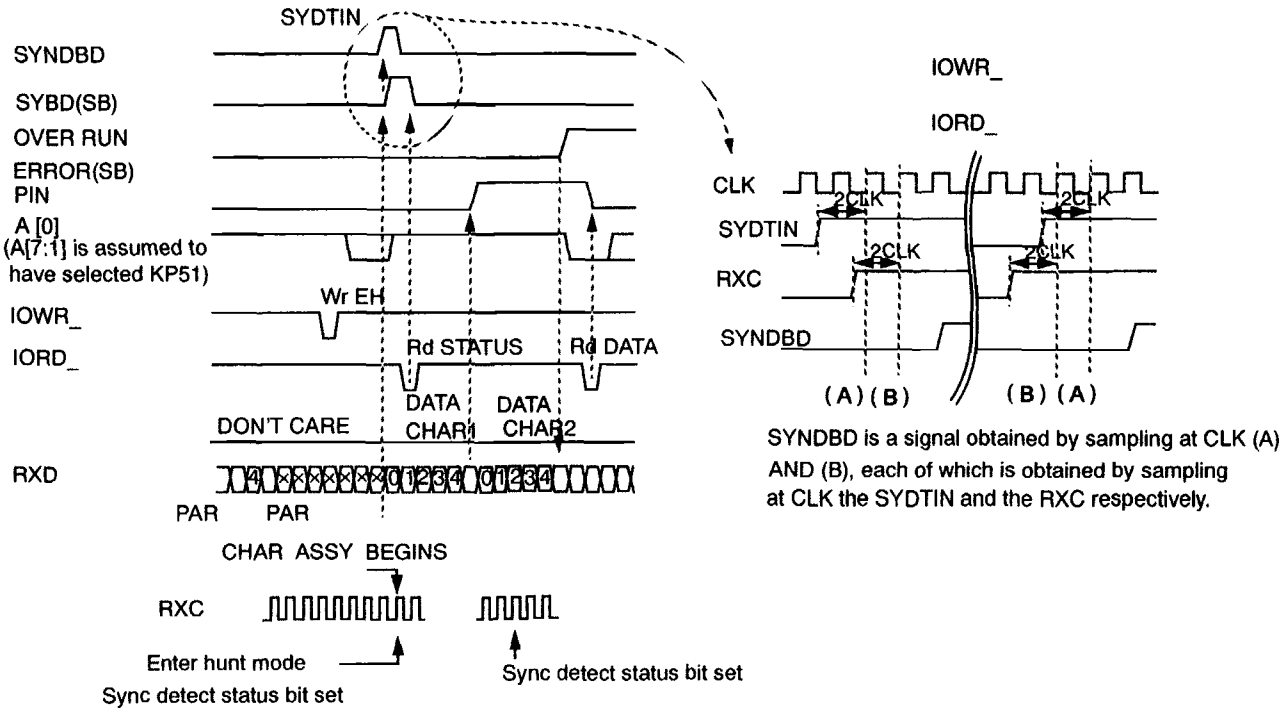
KAWATETSU

Receiver control and flag change timing (reception and internal sync detection in sync mode)



Example: character length 5 bits, parity bit, double sync characters

Receiver control and flag change timing (reception and external sync detection in sync mode)



SYNDBD is a signal obtained by sampling at CLK (A) AND (B), each of which is obtained by sampling at CLK the SYDTIN and the RXC respectively.

Example: character length 5 bits, parity bit

KAWATETSU

8. Timer/Counter A

8.1 General description

Timer/Counter A is a timer/counter block composed of two timer/counter macro cell KP64s. Two channels have the same structure, so one circuit is described in this chapter. This KP64 is a synchronous 16-bit programmable binary timer/counter designed for microcomputer. This macro cell consists of a down counter with four operation modes (frequency divide mode, pulse width modulation (PWM) mode, pulse generation mode, pulse width/frequency measurement mode), and its operation mode can be set by a mode control instruction from the CPU. A counter latch command enables count values to be held in internal register and later be read out. A status read command allows you to read out the set mode or status of OUT outputs.

Features

- Each channel contains a 16-bit down counter/timer, allowing four operation modes to be set and run.
- Stable counter readout with no affects on the count operation
- In addition to external clock (XCLK), system clock can be selected as counter clock.
- Possible to read out the set mode and status of OUT outputs.
- Two cascaded channels can function as a 32-bit counter (see Chapter 12).

8.2 Pin description

pin name	I/O	description
XCLK	I	external counter clock input (Determines the count rate of a counter except when the system clock is selected as counter clock at mode setting.) Count operation is performed at the falling edge of the XCLK.
GATE	I	gate input It specifies the count operation enable/disable for the counter, or gives a trigger signal. It inputs signals to be measured in the pulse width/frequency measurement mode.
OUT	O	OUT output It provides frequency divided output, one shot output, strobe output, measurement complete output depending on the set mode. A reset signal makes this output "L" asynchronously with the counter clock.

Note: The pins XCLK, GATE and OUT of channel 1 and channel 0 are referred to as XCLK1, XCLK0, GATEA1, GATEA0, OUTA1 and OUTA0 respectively in external pins.

8.3 Block diagram

The following are the block diagram of Timer/Counter A and I/O register mapping. OUTA1 and OUTA0 are connected to the built-in interrupt controller as well as external pins.

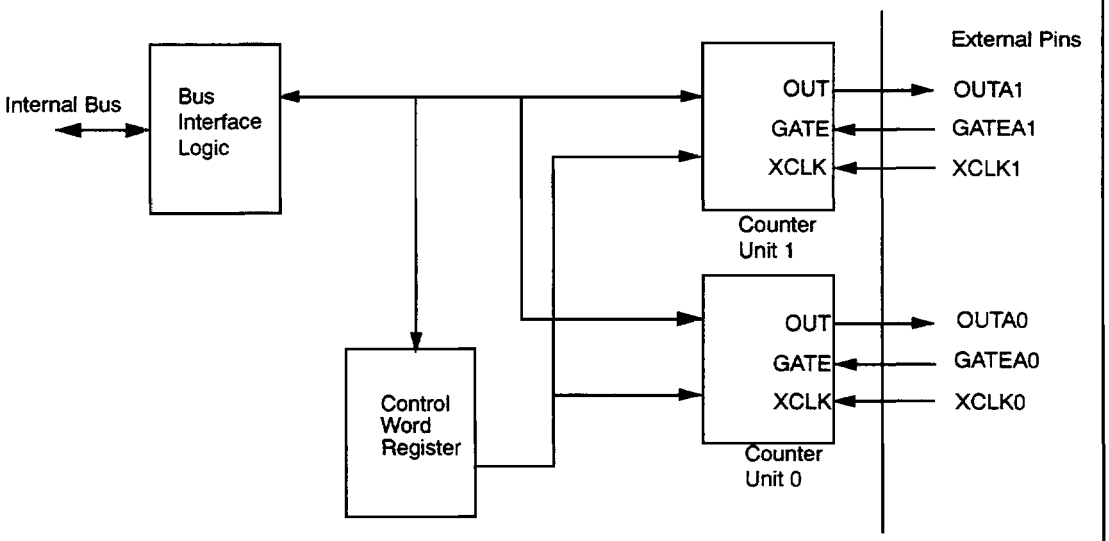


Figure 8-1. Timer/Counter A Block Diagram

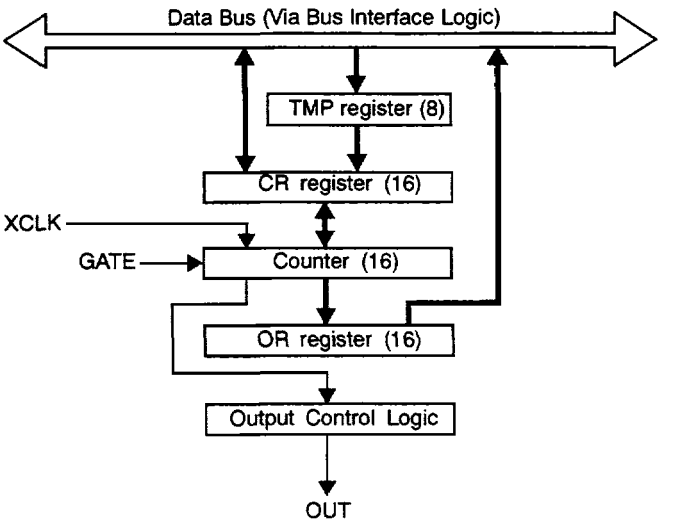


Figure 8-2. Block Diagram of Each Channel Counter Unit

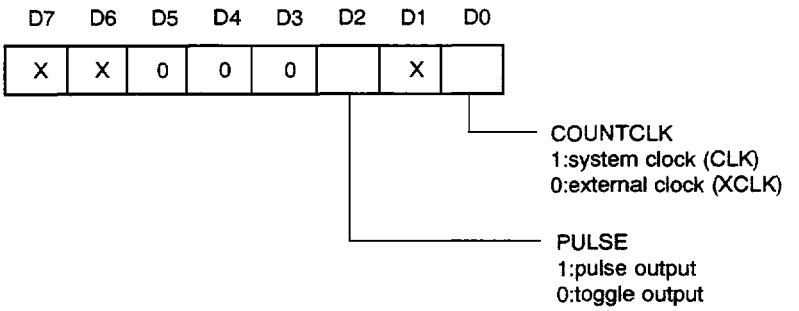
Table 8-1. I/O Register Mapping

I/O address	Block	Write cycle	Read cycle
28H	Timer/Counter A	channel 0 counter	channel 0 counter
29H		channel 0 control word	channel 0 status
2AH		channel 1 counter	channel 1 counter
2BH		channel 1 control word	channel 1 status

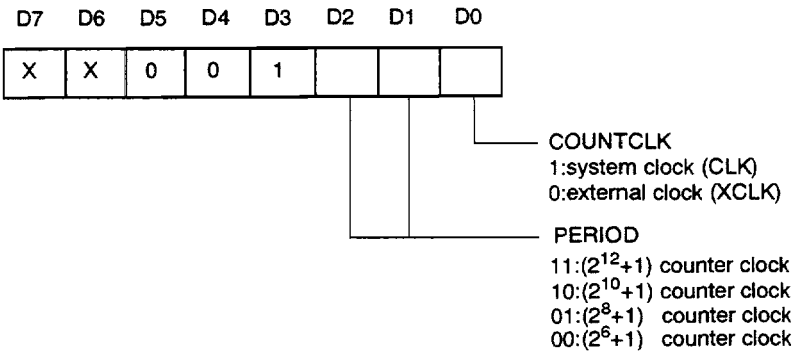
8.4 Setting mode

Mode is set by writing a control word into each channel.

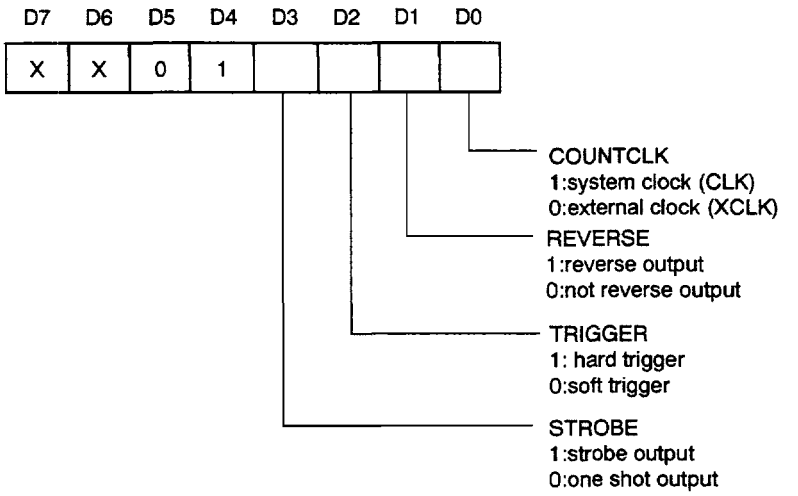
Frequency divide mode



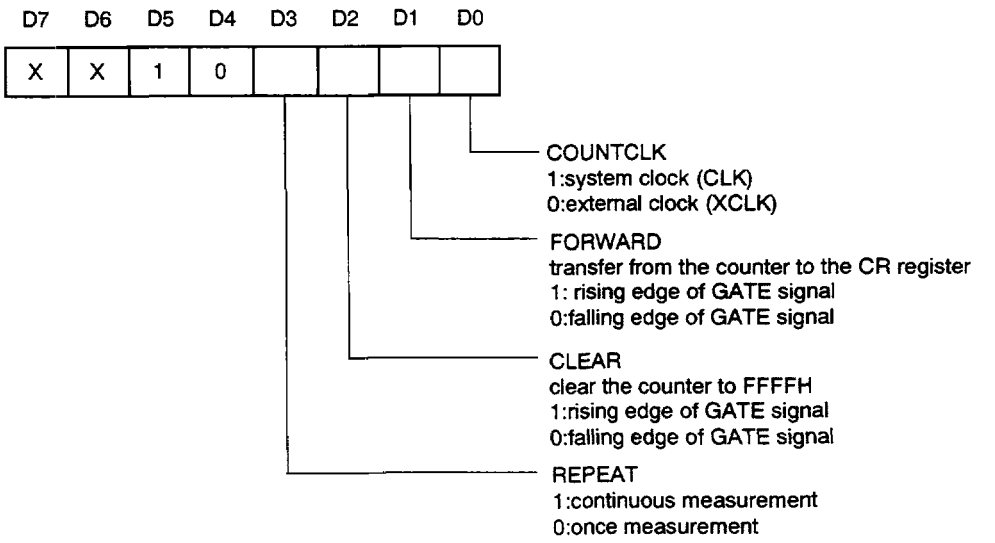
Pulse width modulation (PWM) mode



Pulse mode



Pulse width/frequency measurement mode



8.5 Commands

Counter latch command

A counter latch command is executed by setting a control word to D[5:3]= "111B" and D[1:0]= "00B", and holds count value of the counter in OR register. This count value is held until a next counter latch command is executed. Stored count value can be checked by executing two continuous read operations at A=28H or 2AH.

D7	D6	D5	D4	D3	D2	D1	D0
X	X	1	1	1	X	0	0

Read/write sequence clear command

Two write operations or two read operations are made to set the initial count value to the counter or read out data from the counter. But when for some reason you don't know whether higher bits or lower bits are read out, a read/write sequence clear command is useful. When a read/write sequence command is executed by setting a control word to D[5:3]= "111B" and a sequence clear bit to D[1]=1, a read sequence to read out data from the register within the counter and a write sequence to write count value to the counter can be cleared.

Flag clear command

When a flag clear command is executed by setting a control word to D[5:3]= "111B" and a flag clear bit to D[0]=1, a trigger accept flag in the pulse mode and a measurement complete flag (status D7) in the pulse width/frequency measurement mode can be cleared.

D7	D6	D5	D4	D3	D2	D1	D0
X	X	1	1	1	X		

FLAG
1:clear flag
0:not clear flag

R/W SEQUENCE
1:clear sequence
0: not clear sequence

Note: When D[1:0]=00B, it is recognized as a counter latch command.

8.6 Setting an initial count value to the counter

Although the counter is a 16-bit down counter, a bus to set data is 8-bit wide, so data should be written twice for lower byte and higher byte. The continuous two write operations are valid right after the reset, mode setting or read/write sequence command.

First write operation:

D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

(Numbers indicate data bits.)

Second write operation:

D7	D6	D5	D4	D3	D2	D1	D0
15	14	13	12	11	10	9	8

(Numbers indicate data bits.)

8.7 Counter readout

Counter readout can be performed only by the continuous two write operations right after a counter latch command. Counter values are once held in a register within the Timer/Counter A by a counter latch command, and then read out. Once held count values are kept until a next counter latch command is executed or readout is completed (see a readout example below).

First readout operation:

D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

(Numbers indicate data bits.)

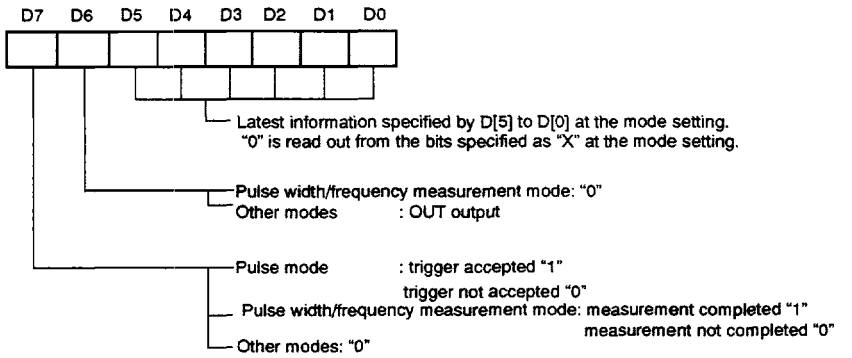
Second readout operation:

D7	D6	D5	D4	D3	D2	D1	D0
15	14	13	12	11	10	9	8

(Numbers indicate data bits.)

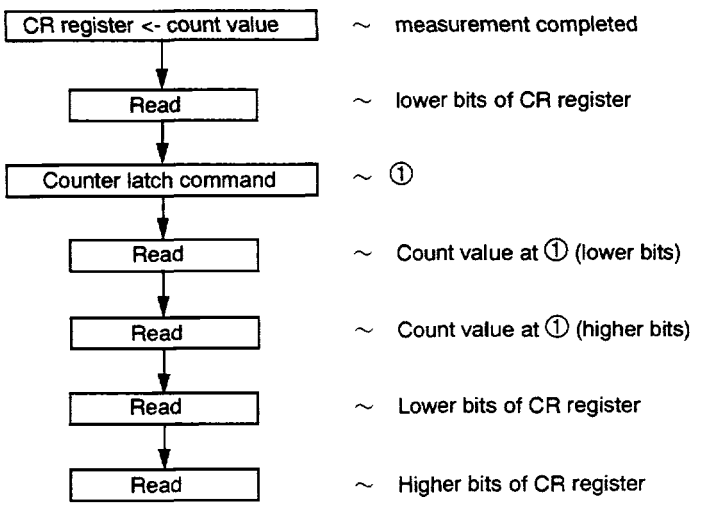
8.8 Status readout

Status information is obtained by reading out the status register. The format of the status register is as follows.



Readout example

Pulse width/frequency measurement mode



8.9 Operation modes

Frequency divide mode

In the frequency divide mode, the initial count value is loaded and decremented. When the counter reaches "0", the OUT output changes and the reload and decrement of the initial count value is repeated. When the mode is set, the CR register (a register for writing the initial count value) and the counter value are set to "FFFFH", and the OUT pin outputs "L". A toggle or pulse can be selected for its output, which can be used for interrupt.

The count operation is started by loading the initial count value, and has a GATE function (GATE input level "H"= count enable, GATE input level "L"= count disable).

For OUT output, whether a toggle output whose signal is reversed whenever the counter reaches "0" or a pulse output which outputs a pulse of one counter clock width whenever the counter reaches"0" can be selected at the mode setting. Reload is performed not by writing the initial value during count, but only when the counter reaches "0".

Be sure that the GATE input pulse width of both "H" and "L" is more than two cycles of system clock.

Pulse width modulation (PWM) mode

In the pulse width modulation mode, a repetitive pulse, which has the width (duration of "H") set by the initial count value and the frequency selected from four frequencies, is generated.

Selectable pulse frequencies are (2^n+1: n=6, 8, 10, 12) times the counter clock, and the pulse width is a time from when the initial count value is loaded and decremented to when the counter reaches "0", i.e. (initial count value + 1).

When the mode is set, CR register is set to "FFFFH", a value determined by the selected mode (2^n-1: n=6, 8, 10, 12) is read to the counter, and the OUT signal goes "L".

The count operation is started by loading the initial count value, and the following sequence is repeated; load of initial count value, its decrement, load of the initial count value's complement, its decrement, and load of initial count value. It has a GATE function as in the frequency divide mode. Reload is performed not by writing the initial value during count, but only when the counter reaches "0".

Be sure that the GATE input pulse width of both "H" and "L" is more than two cycles of system clock.

Pulse mode

In the pulse mode, an initialized count value is loaded and decremented. When the counter reaches "0", the OUT output changes and the count operation is stopped until a retrigger is generated.

As for count trigger, either a soft trigger to use the setting of initial count value as a trigger for count start, or a hard trigger to use the rising edge of signal input to the GATE pin as a trigger for count start, can be selected at the mode setting. As for OUT output, a one shot output whose signal is "L" during load and count, and goes "H" when the counter reaches "0", a strobe output which outputs a pulse of one counter clock width when the counter reaches "0", or a mode in which the polarity of OUT output is reversed, can be selected.

When the mode is set, the CR register and counter value are set to "FFFFH", and the OUT signal goes "H" or "L" when the output reverse is disabled or enabled respectively. The count operation is started by acceptance of a trigger and sets the trigger acceptance flag to "1". When the counter reaches "0", the OUT output changes, and the counter is cleared to "FFFFH" and stopped. When a new initial count value is written, the CR register is immediately set to the new value.

When the soft trigger is selected, a new value is written to the counter at the next counter clock and the

count operation continues (retrigger is possible during count). At this time the GATE functions as count enable. When the hard trigger is selected, a new value is written to the counter at the first counter clock after a retrigger is made, and the count operation continues (retrigger is possible during counting).
When a retrigger is generated, wait for more than one cycle of counter clock after a previous trigger. When the soft trigger is selected, be sure that the GATE input pulse width of both "H" and "L" is more than two cycles of system clock.

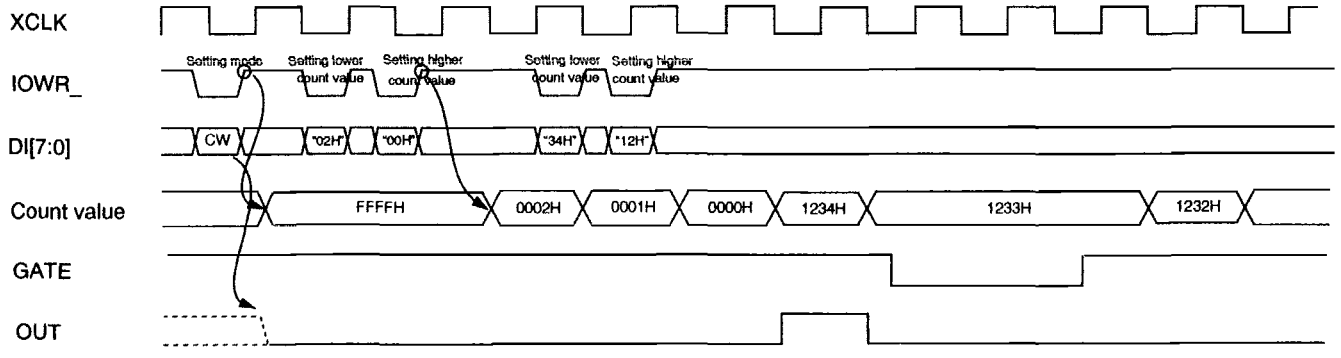
Pulse width/frequency measurement mode

In this mode, the pulse width and frequency of GATE input signal is measured, and after measurement a pulse of one system clock cycle is output to the OUT pin.
As signals to be measured, the four patterns of GATE input signal can be set; from the rising edge to the rising edge or to the falling edge, and from the falling edge to the rising edge or to the falling edge. Once measurement or continuous measurement can be selected at the mode setting. When the mode is set, the CR register and counter value are set to "FFFFH".
The continuous measurement from the rising edge to the rising edge of input signal is described below.
When a rising edge of input signal is recognized at the GATE pin, the count value's complement is sent to the CR register, the counter is cleared to "FFFFH", and then the count operation starts at the next counter clock. However, a pulse is not output to the OUT pin at the first transfer to the CR register.
When a rising edge of the next input signal is recognized, the count value's complement is sent again to the CR register, the counter is cleared to "FFFFH", and a "H" pulse of one system clock cycle is output to the OUT pin to set the measurement complete flag to "1". The count operation continues. Therefore, the pulse frequency can be checked by readout of the value in the CR register.

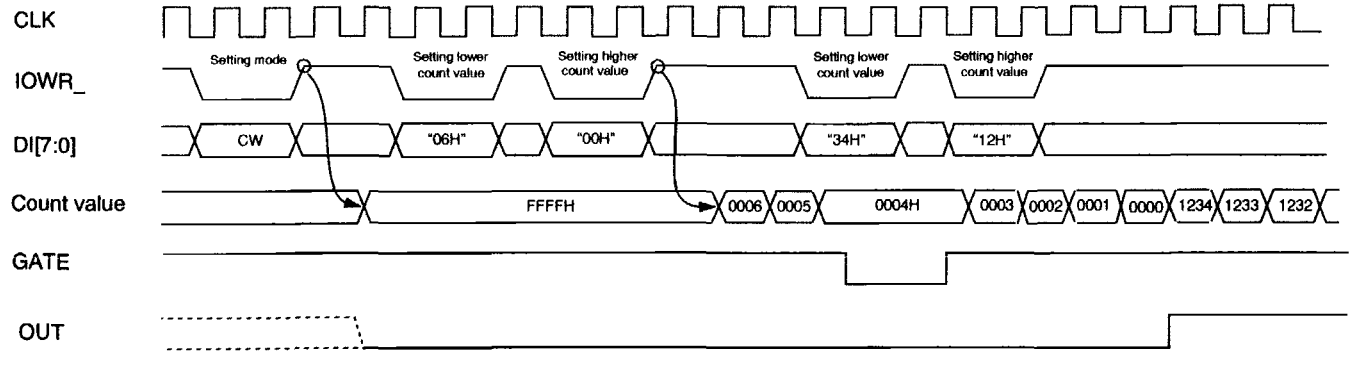
In the pulse width/frequency measurement mode, the CR register functions as buffer register. In the case of once measurement, the measurement operations from start to stop are the same as those in the continuous measurement, but after measurement the counter holds the last count value and stops.
Be sure that the period from the rising edge to the falling edge or from the falling edge to the rising edge of the GATE input signal is more than two cycles of system clock.

Operation modes

Frequency divide mode (external clock, pulse output)

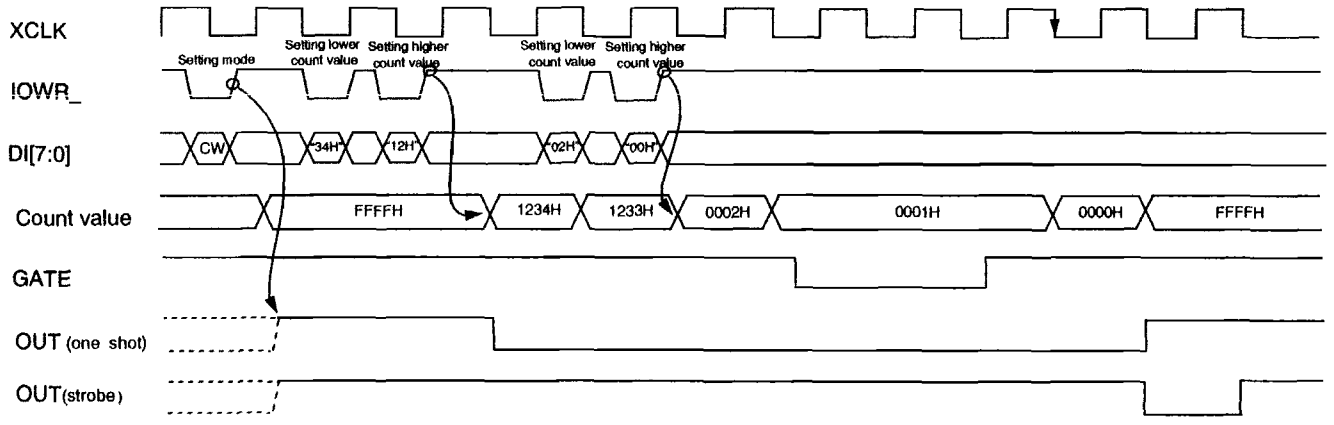


Frequency divide mode (system clock, toggle output)

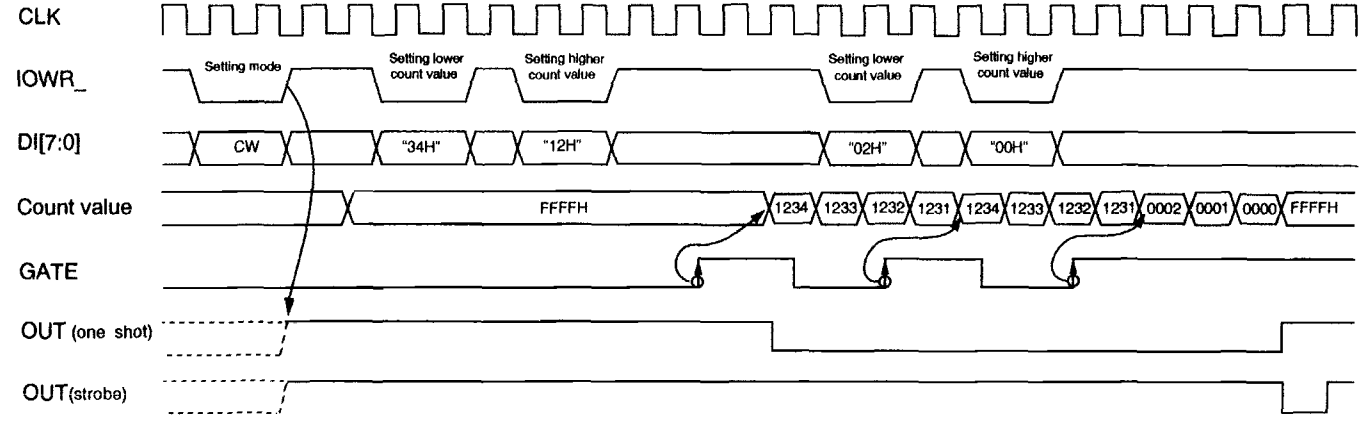


KAWATETSU

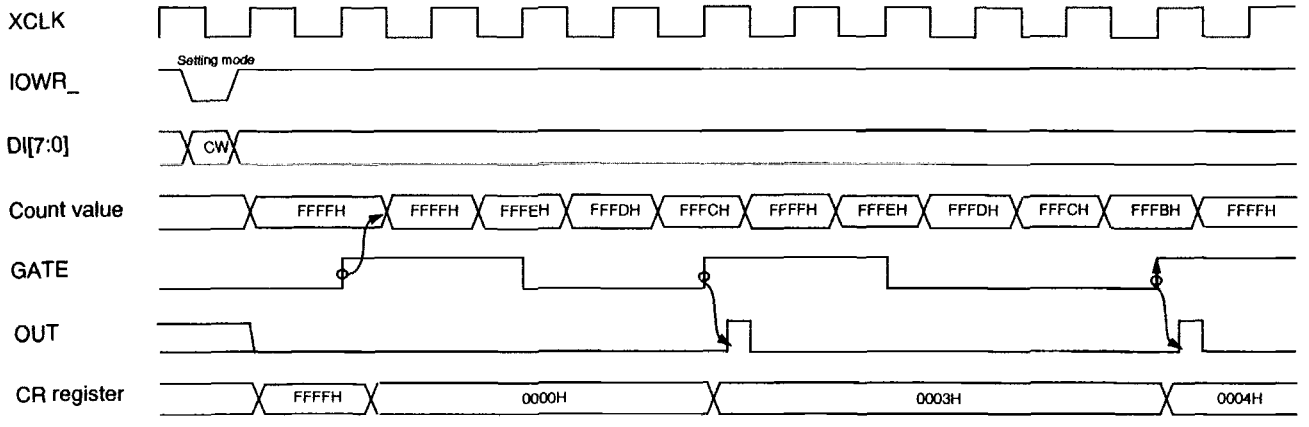
Pulse mode (external clock, soft trigger, REVERSE = 0)



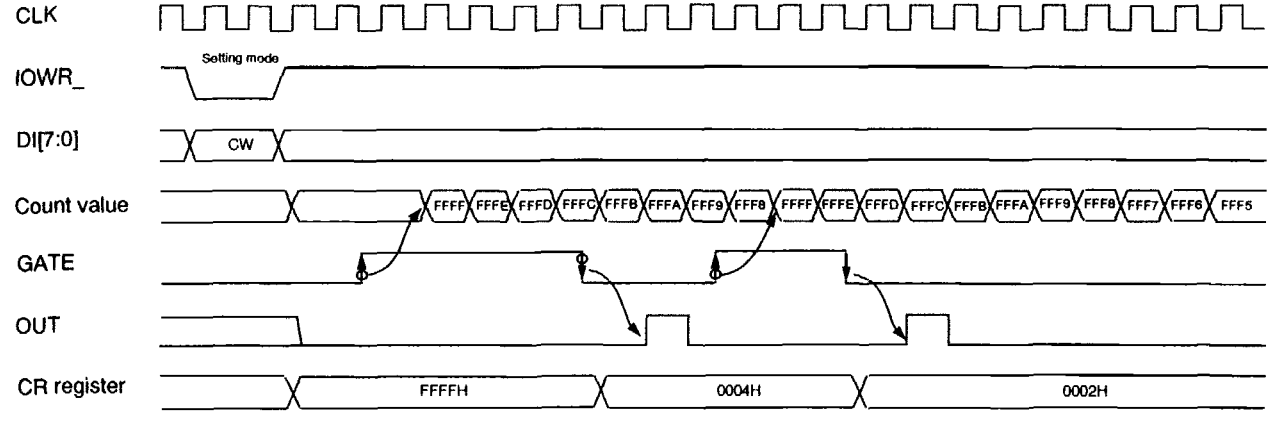
Pulse mode (system clock, hard trigger, REVERSE = 0)



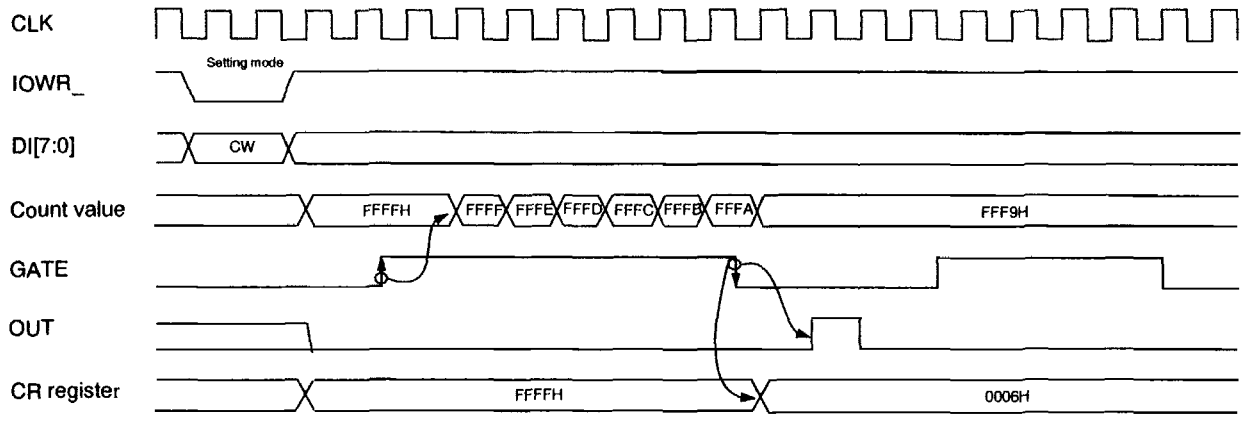
Pulse width measurement mode (external clock, continuous measurement) GATE signal up to up



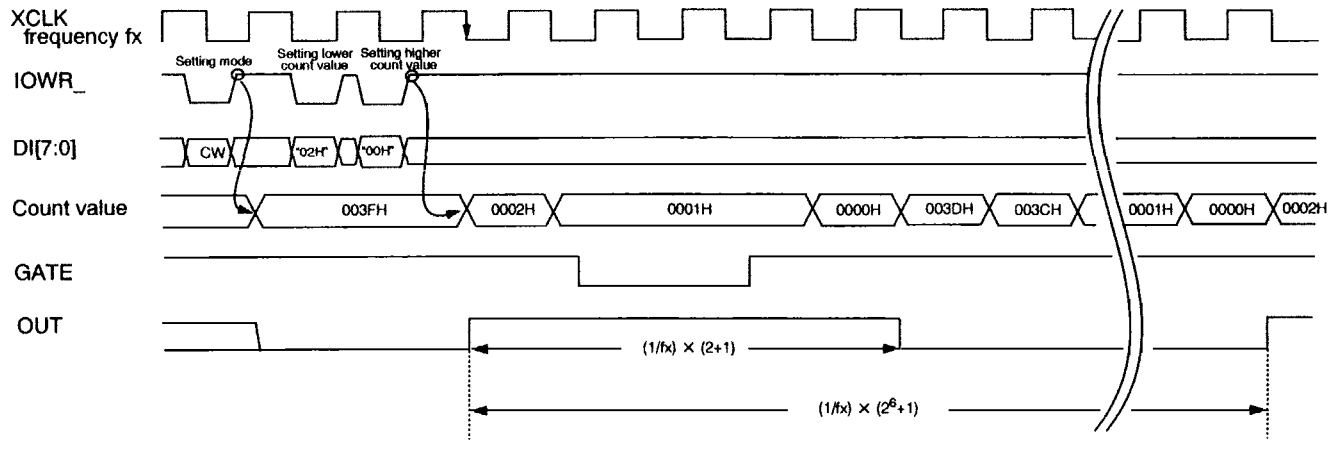
Pulse width measurement mode (system clock, continuous measurement) GATE signal up to down



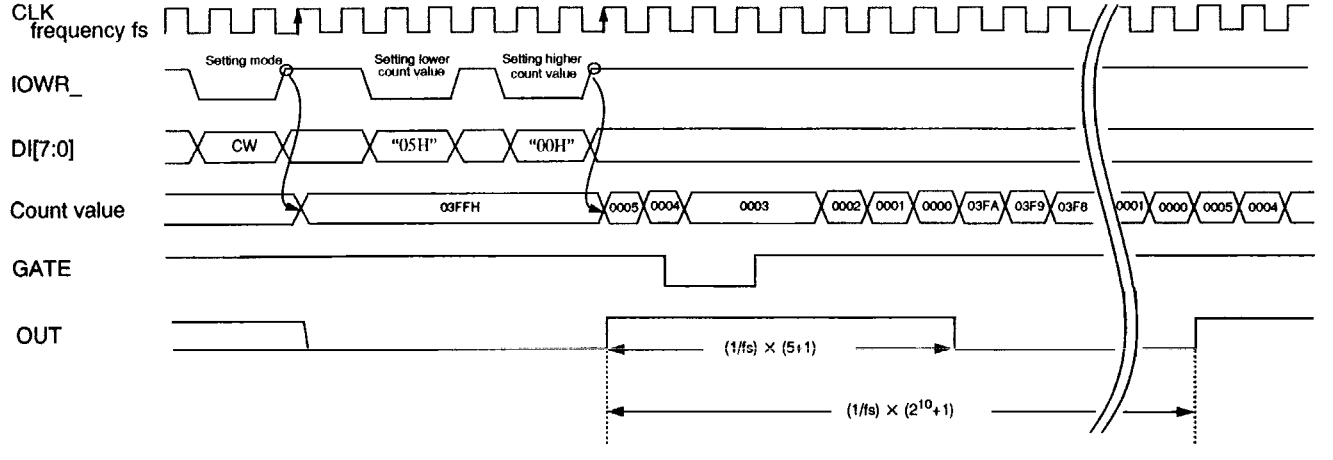
Pulse width measurement mode (system clock, once measurement) GATE signal up to down



PWM mode (external clock, (2^6+1) times frequency)



PWM mode (system clock, (2^{10}+1)times frequency)



8.10 Counter operation by GATE input in each mode

Mode	GATE input/counter operation		
	"L" or falling edge	Rising edge	"H"
Frequency divide	Count stop (hold)	-----	Count
Pulse width modulation (PWM)	Count stop (hold)	-----	Count
Pulse: soft trigger	Count stop (hold)	-----	Count
: hard trigger	Count wait (continue)	Count start	-----
Pulse width/frequency measurement	Input of waveform to be measured (see "Pulse width/ frequency measurement mode")		

8.11 Reset

Reset is performed by RESET_= "L" and affects the following features.

(1)Stops the operation of counter. The counter is kept in stop state even after clearing the reset.

(2)Clears a read sequence in readout of register within the counter and a write sequence in write of initial count value.

(3)Clears the CR register within the counter and the counter register to "FFFFH".

(4)Clears the register for readout of count value to "FFFFH".

(5)Enters the frequency divide mode (system clock synchronous, toggle output), sets the OUT output to "L", and holds the value after clearing the reset until the mode is set.

8.12 Precautions

Maximum/minimum value of initial count value

The following is the maximum/minimum values of initial count value.

Mode	Minimum value	Maximum value
Frequency divide	0001H	FFFFH
Pulse width modulation (PWM)*	0001H	0FFFH
Pulse	0001H	FFFFH
Pulse width/frequency measurement	Unable to set	Unable to set

* Valid values are dependent on the PERIOD of mode setting, and a valid bit width is any of lower 6, 8, 10 or 12.

Definitions of terms

Hard trigger: the rising edge of GATE input is recognized as trigger.

Soft trigger: the completion of initial count value write (two byte write) is recognized as trigger.

Timer/Counter B

9.1 General description

The KL5C80A12's Timer/Counter B is a three-channel version of KP63(16-bit high-performance timer/counter macro cell). The KP63 is asynchronous 16-bit programmable binary timer/counter with 8-bit prescaler designed for KC80 microcomputer. This macro cell consists of three down counters with three operation modes (pulse width modulation (PWM) mode, pulse generation mode and watchdog timer mode), and its operation can be set by a mode control word from the CPU. Count values can be read out from the bus. A status read command allows you to read out the set mode or status of OUT outputs.

Note The KP63 is essentially composed of four channels, and the KL5C80A12's Timer/Counter B is its three channel version. The four-channel structure is described in this chapter, but be sure not to use the channel 3.

Features

- Built-in 8-bit prescaler
- Built-in three 16-bit down counters, each of which enables three operation modes to be set and run
- Stable counter readout with no affects on count operation
- Set mode and status of OUT outputs can be read out.

9.2 Pin description

pin name	I/O	description
GATE[3:0]	I	GATE input It specifies the count operation enable/disable for the counter of each channel.
OUTP[3:0]	O	pulse OUT output It provides frequency divided output or PWM output depending on the set mode. A reset signal makes this output "L" asynchronously with system clock.
OUTS[3:0]	O	strobe OUT output It outputs a strobe signal of four system clock width.
SYNC	O	Sync signal output It outputs a signal of divide-by-4 system clock frequency. GATE[3:0] are sampled at the rising edge of this signal.

Note: The pins GATE, OUTP and OUTS of each channel are referred to as GATEB2, GATEB1, GATEB0, OUTP2, OUTP1, OUTP0, OUTS2, OUTS1 and OUTP0 respectively in external pins. Be sure to check external pin names in block diagram before using them.

9.3 Block diagram

The following are the block diagram of Timer/Counter B and I/O register mapping.
Each of OUTS outputs is connected to the built in interrupt controller as well as external pins.

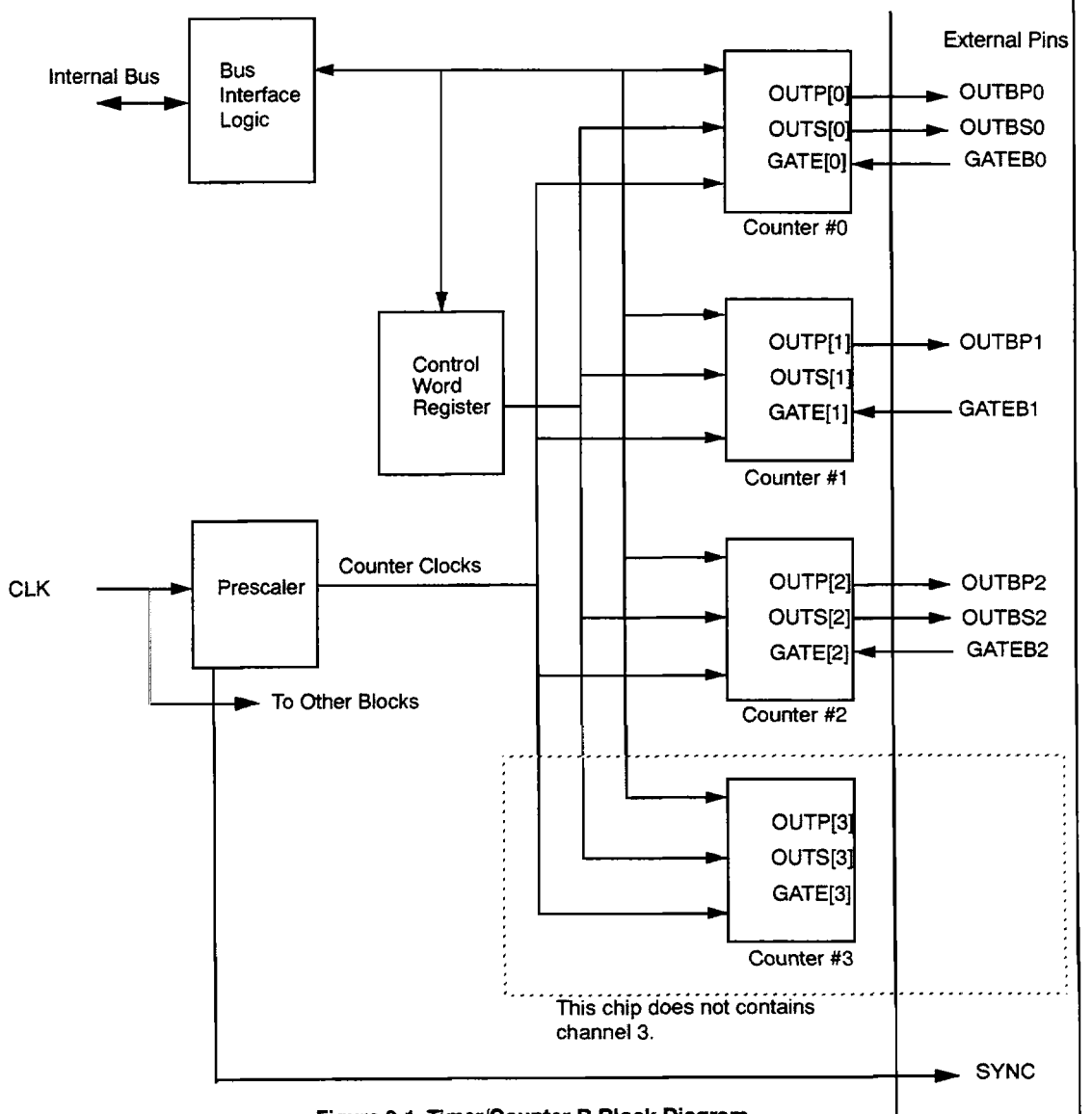


Figure 9-1. Timer/Counter B Block Diagram

Table 9-1. I/O Register Mapping

I/O address	Block	Write cycle	Read cycle
20H	Timer/Counter B	channel 0 counter	channel 0 counter
21H		channel 0 control word	channel 0 status
22H		channel 1 counter	channel 1 counter
23H		channel 1 control word	channel 1 status
24H		channel 2 counter	channel 2 counter
25H		channel 2 control word	channel 2 status
26H		reserved for Kawasaki Steel Corp.	reserved for Kawasaki Steel Corp.
27H		reserved for Kawasaki Steel Corp.	reserved for Kawasaki Steel Corp.

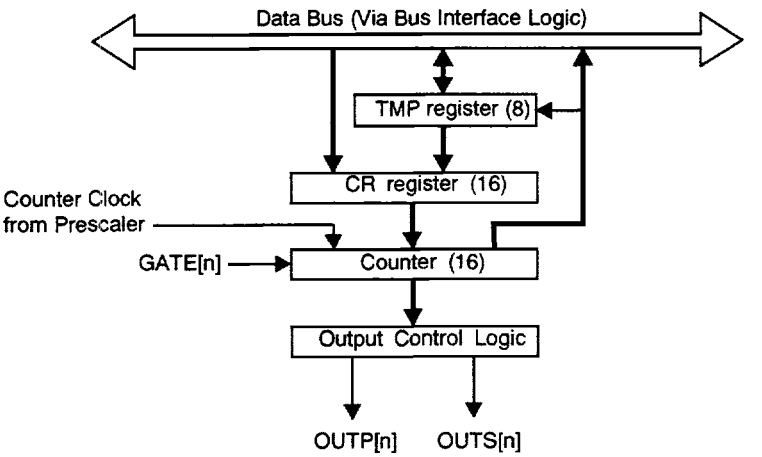
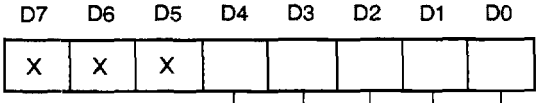


Figure 9-2. Block Diagram of Each Channel Counter Unit

9.4 Setting mode

Mode is set by writing a control word into each channel.



COUNTCLK
Sets the frequency division rate of prescaler.
11:1/4 system clock frequency (GATE effective)
10:1/4 system clock frequency (GATE neglected)
01:1/16 system clock frequency (GATE neglected)
00:1/256 system clock frequency (GATE neglected)

COUNT MODE
11:PWM mode
10:WDT mode
01:continuous count mode
00:one shot count mode

TOGGLE
1:Set the initial OUTP output to "H".
0:Set the initial OUTP output to "L".

The prescaler is shared by all channels, but its frequency division rate can be set for each channel.

9.5 Setting an initial count value to the counter

Although the counter is a 16-bit down counter, a data bus is 8 bit wide, so data is written twice for lower byte and higher byte. However, the TMP register is shared for write and read operations, so written data is destroyed if the counter is readout before data write has not been completed.

Lower byte:

D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

(Numbers indicate data bits.)

Higher byte:

D7	D6	D5	D4	D3	D2	D1	D0
15	14	13	12	11	10	9	8

(Numbers indicate data bits.)

9.6 Counter readout

Counter readout is performed by accessing the address of each counter. Data is read out twice by 8 bits in the order of lower byte and higher byte. A higher byte is read out via the TMP register. As the content of TMP register is copied from the higher byte of the counter when the lower byte is read out, the value in the TMP register would not change if a value in the counter changes during two read cycles. However, the TMP register is shared for write and read operations, so read data is destroyed if the counter is written before data read has not been completed.

Lower byte:

D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

(Numbers indicate data bits.)

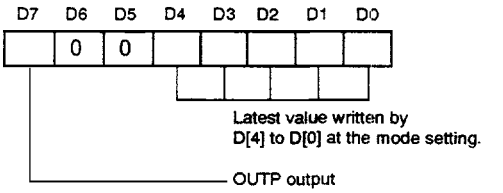
Higher byte:

D7	D6	D5	D4	D3	D2	D1	D0
15	14	13	12	11	10	9	8

(Numbers indicate data bits.)

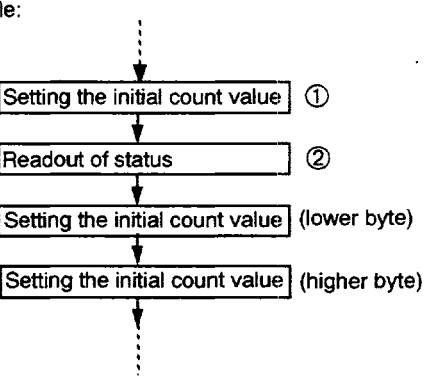
9.7 Status readout

Status register is provided for each channel. The format of the status register is as follows.



Readout of status clears a read/write sequence (see the example below). Therefore, reading out status before setting the initial count value or reading out the counter prevents writing value erroneously.

Example:



① may be higher or lower byte.
② clears a read/write sequence.

9.8 Operation modes

The following is the description for each operation mode with TOGGLE bit "0".

Continuous count mode

In the continuous count mode, the initial count value is loaded and decremented. When the counter reaches "0", the OUTP and OUTS outputs change and the reload and decrement of the initial count value is repeated.
When the mode is set, the OUTP and OUTS pins output "L". The count operation is started by loading the initial count value, and the OUTP pin outputs "H". Whenever the counter reaches"0", the OUTP output toggles and the OUTS pin outputs "H" of four system clock width.
Reload is performed not by writing the initial value during count, but only when the counter reaches "0".

One shot count mode

In the one shot mode, the initial count value is loaded and decremented. When the counter reaches "0", the OUTP and OUTS outputs change and the count operation stops until the initial count value is set again.
When the mode is set, the OUTP and OUTS pins output "L". The count operation is started by setting the initial count value, and the OUTP pin outputs "H" until the counter reaches "0". When the counter reaches "0", the OUTS pin outputs "H" of four system clock width.
Reload is performed by setting the initial count value.

Watch dog timer (WDT) mode

In the watch dog timer mode, the initial count value cannot be written to the CR register. Write operation to the CR register is recognized as a count start or retrigger.
When the mode is set, the counter stops its operation and the OUTP and OUTS pins output "L". The count operation is started by the write operation to the CR register (regardless of the data value at the write operation; one write operation) which causes data to be loaded from the CR register, and the OUTP pin outputs "H".
The write operation to the CR register during count causes data to be loaded from the CR register again, and the countdown is performed.
When the counter reaches "0", the OUTP pin outputs "L" and the OUTS pin outputs "H" of four system clock width, and then the count operation stops until the write operation to the CR register is performed.
In mode setting, the CR register is not initialized. Perform the following procedures to set the CR register to the initial count value. Set to the one shot count mode, set the CR register to the initial count value, and then return to the WDT mode. Perform the write operation to the CR register, which causes the value set in the CR register to be loaded and starts countdown.
The write operation to the CR register during count causes data to be loaded from the CR register again, and the countdown is performed.
These procedures allows any initial count value to be set in the WDT mode. When this timer/counter is used as watch dog timer, it is required not only to set this mode but also to connect the OUTS or OUTP output to the KC82 NMI_ input externally.

Pulse width modulation (PWM) mode

In the pulse width modulation mode, a repetitive pulse with the pulse width and frequency set by the initial count value is output at the OUTP pin.
The pulse width is set by the higher byte of CR register, and the pulse frequency set by its lower byte. In the first place, how to set pulse frequency is described.
The pulse frequency is a time from the lower byte of CR register loaded in the counter is decremented independently of the higher byte until its count value becomes "0". That is, it denotes the time of (set value

by the first write to the CR register + 1).
On the other hand, the pulse width is a time from the higher byte of CR register loaded in the counter is decremented independently of the lower byte until its count value becomes "0". That is, it denotes the time of (set value by the second write to the CR register + 1).
As both higher and lower bytes are simultaneously counted down independently, any desired pulse can be obtained from the OUTP pin.
When the mode is set, the counter stops and the OUTP and OUTS pins output "L". When both higher and lower bytes are set to initial count values and they are loaded, the count operation starts and the countdowns of higher and lower bytes are performed independently. But if the pulse width is greater than or equal to the pulse frequency, the OUTP is always "H".
Reload is performed not by writing the initial value during count, but only when the lower byte becomes "0".
(Ex. divide-by-4 frequency, initial count value=0308H)
Pulse width: 4(= 3 + 1) x 4 = 16 system clocks
Pulse frequency: 9(= 8 + 1) x 4 = 36 system clocks

9.9 OUTP and OUTS outputs

The following is the description for each operation mode with TOGGLE bit "0".
When the count value becomes "0" (the lower byte becomes "0" at PWM mode), the OUTS pin outputs "H" of four system clock width. So the OUTS output can be used for an interrupt to the CPU.
The OUTP output is a signal which can be reversed by the mode setting. In the continuous mode, it toggles whenever the count value becomes "0". In the one shot mode and the WDT mode, it always outputs "H" during count, and it outputs "L" when the count value becomes "0". In the PWM mode, it outputs signals with desired pulse width and frequency. Therefore, the OUTP output can be used for obtaining various pulse signals.

9.10 Operations

In this section, the operations of Timer/Counter B are described referring to the figures.

Countdown timing and SYNC signal (Figure 9-3-A)

Timer/Counter B outputs SYNC signal as synchronizing signal for countdown timing. Figure 9-3-A shows the case in which all channels are set to divide-by-4 frequency without GATE function.
As shown in the Figure, the countdown of the counter channel 1(hereinafter referred to as CH1) is made at the falling edge of the system clock by one system clock behind CH0. Similarly, CH2 and CH3 are delayed respectively by two and three system clocks behind CH0.
SYNC signal is fundamentally a signal of divide-by-4 system clock frequency, and its relationship with the system clock is shown in Figure 9-3-A.

Countdown timing at each selected counter clock (Figure 9-3-B)

Figure 9-3-B shows the case in which CH0 and CH1 are set to counter clock of divide-by-4 frequency without GATE function, CH2 to that of divide-by-16 frequency, and CH3 to that of divide-by-256 frequency.
In CH0 and CH1, the times from 1 to 1' and from 2 to 2' are four system clocks, the time from 3 to 3' is 16 system clocks, and the time from 4 to 4' is 256 system clocks.

Sampling timing of GATE signal (Figure 9-3-C)

Figure 9-3-C shows the case in which all channels are set to divide-by-4 frequency with GATE function. GATE signal sampling is made at every four system clocks (at the upper arrows of CLK in Figure; the rising edge of SYNC signal) simultaneously for all channels. Therefore, the polarity of the GATE signal sampled at this time determines whether the countdown is made or not at the next timing. GATE function can be used

only at the divide-by-4 frequency rate.

Countdown timing and control word write cycle (Figures 9-4 to 9-7)

Figures 9-4, 9-5, 9-6 and 9-7 show the relationships between countdown timings and control word write cycles for CH0, CH1, CH2 and CH3 respectively.
The left half of each figure shows the longest time until the mode setting by control word write cycle is reflected, i.e., the longest time until the OUTP and OUTS signals are initialized.
The right half of each figure shows the shortest time.

Countdown timing and count value write cycle (Figures 9-8 to 9-11)

Figures 9-8, 9-9, 9-10 and 9-11 show the relationships between countdown timings and count value write cycles for CH0, CH1, CH2 and CH3 respectively in the continuous count mode and PWM mode.
The left half of each figure shows the longest time until the higher count value setting is reflected in the counter, while the right half shows the shortest time.

Operations in the continuous count mode (Figure 9-12)

The Figure 9-12 shows CH0 in the continuous count mode in which divide-by-4 frequency without GATE function and with TOGGLE bit "0".
When the mode is set, the OUTP and OUTS signals are initialized at the next countdown timing and go "L".
When the lower count value "02H" and the higher count value "00H" are written, 0002H is loaded to the counter at the next countdown timing, causing the countdown to start and the OUTP output to toggle.
If the initial count value (F0ABH) is reset during count, i.e.between 1 and 2 in the Figure, the count value is set to F0ABH at the next countdown timing after the counter reaches "0", i.e. at the point 2 in the Figure.
At the same time the OUTP output toggles and a pulse of four system clock width is output at the OUTS pin.

Operations in the one shot count mode (Figure 9-13)

The Figure 9-13 shows CH0 in the one shot count mode in which divide-by-4 frequency without GATE function and with TOGGLE bit "0".
When the mode is set, the OUTP and OUTS signals are initialized at the next countdown timing and go "L".
When the lower count value "02H" and the higher count value "00H" are written, 0002H is loaded to the counter at the next countdown timing, causing the countdown to start and the OUTP output to go "H".
If the initial count value (0003H) is reset during count, i.e.between 1 and 2 in the Figure, a retrigger is generated at the next countdown timing (at 2 in the Figure) and the countdown continues from this initial count value. When the initial count value is not reset between 2 and 3, and the counter reaches "0"(at 3 in the Figure), it is cleared to "FFFFH" and the countdown stops. The OUTP output goes "L" and the OUTS pin outputs "H" of four system clock width.
When the reset of initial count value (ABCDH) is recognized at the timing from the count value "0" to "FFFFH" as at the point 1,it is recognized as a retrigger and both OUTP and OUTS signals do not change.

Operations in the watch dog timer (WDT) mode (Figure 9-14)

The Figure 9-14 shows CH0 in the watch dog timer (WDT) mode in which divide-by-4 frequency without GATE function and with TOGGLE bit "1".
In the WDT mode, the write operation of initial count value (write to the CR register) is recognized as a retrigger and the value of the initial count value register does not change. To set the initial count value register to a desired value, set the one shot count mode to write a desired value to the CR register, and then set the WDT mode.
The basic operations of the WDT mode are similar to those of the one shot count mode. Please refer to the Figure 9-14. At first, when the one shot count mode is

set, the OUTP output goes "H" and the OUTS output goes "L" at the next countdown timing. When the lower count value "03H" and the higher count value "00H" are written, 0003H is loaded to the counter at the next countdown timing, causing the countdown to start and the OUTP output to go "L".
Then, if the WDT mode is set (at 1 in the Figure), the OUTP output goes "H" at the next countdown timing and the counter has stopped at this moment. When the write operation of count value (any data) is made once,"0003H" in the CR register is loaded to the counter at the next countdown timing, causing the countdown to start for the first time in the WDT mode and the OUTP output to go "L" (at 2 in the Figure).
When the count write operation (any data) is made during count, i.e. between 2 and 3 in the Figure, it is recognized as a retrigger and "0003H" is reloaded at the next countdown timing (at 3 in the Figure). Then the countdown starts again. When the count value becomes "0", the OUTS pin outputs "H" of four system clock width, the OUTP pin outputs "H", the count value is cleared to "FFFFH" and the counter stops (at 4 in the Figure).
Then, if the write operation of count value (any data) is made once, "0003H" is loaded to the counter at the next countdown timing, causing the countdown to restart.
Although the OUTP output makes unnecessary change for the WDT mode when the one shot count mode is set, the OUTS pin does not output pulses until the count value becomes "0".
For more details refer to the separate Application Note.

Operations in the pulse width modulation (PWM) mode (Figure 9-15)

The Figure 9-15 shows CH0 in the pulse width modulation (PWM)mode in which divide-by-4 frequency without GATE function and with TOGGLE bit "0".
The basic operations of the PWM mode are similar to those of the continuous count mode. When the mode is set, the OUTP and OUTS signals are initialized at the next countdown timing and go "L". When the lower count value (pulse frequency data) "03H" and the higher count value (pulse width data) "01H" are written, 0103H is loaded to the counter at the next countdown timing, causing the countdown to start and the OUTP output to toggle.
As shown in the Figure, the countdowns of the higher and lower bytes are made independent and simultaneously. When the higher byte count becomes "0", the OUTP output toggles. One countdown operation is from 1 to 2, i.e., from the load of initial count value to when the lower count value becomes "0".
Between 1 and 2 the OUTP pin outputs a pulse with (1+1)/fc=2/fc of width and (3+1)/fc=4/fc of frequency (fc: countdown frequency). The OUTS pin outputs a pulse of four system clock width at 2.
If the initial count value (0204H) is reset during count, "0204H" is loaded to the counter at the next countdown timing after the lower count value becomes "0", i.e. at 2 in the Figure.
If the higher byte is greater than or equal to the lower byte, the OUTP pin always outputs "H" as shown between 3 and 4. At the next countdown timing after the lower byte becomes "0" (at 4 in the Figure), the initial count value "0505H" is loaded to the counter and the countdown is performed continuously again.
As described above, any frequency can be selected in the PWM mode of the Timer/Counter B.

Figure 9-3-A. Countdown timing (no GATE function, divide-by-4 frequency) and SYNC signal

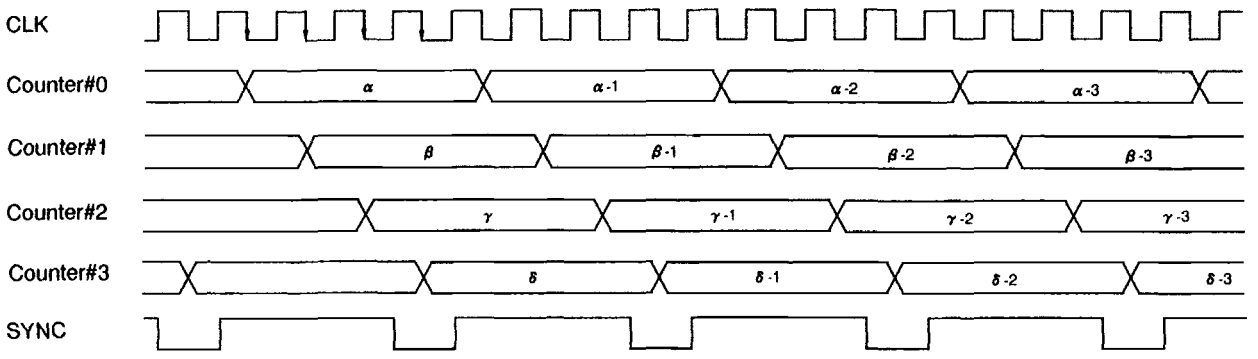


Figure 9-3-B. Countdown timing
(#0,1: no GATE function, divide-by-4 frequency, #2: divide-by-16 frequency,#3: divide-by-256 frequency)

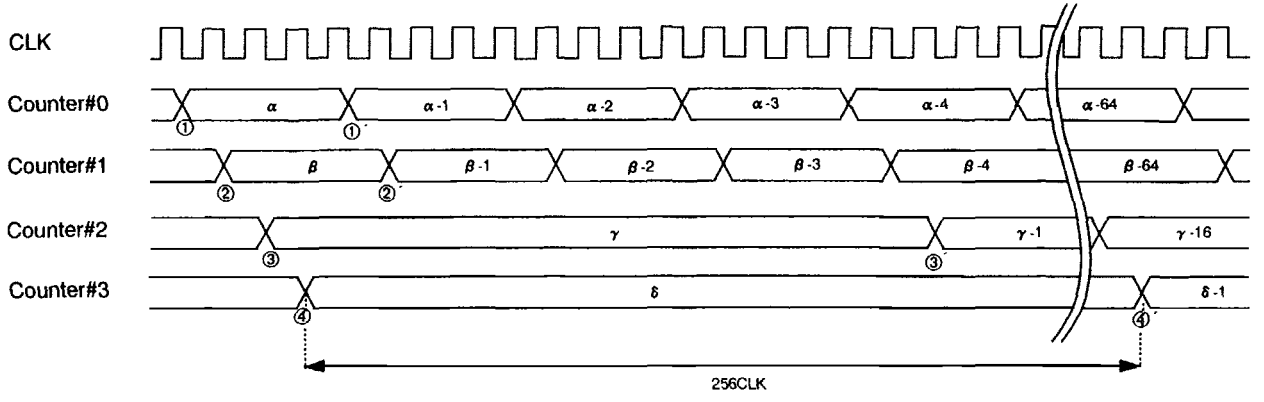


Figure 9-3-C. Sampling timing of GATE signal (GATE function, divide-by-4 frequency)

Figure 9-4. Countdown timing and control word write cycle (#0)

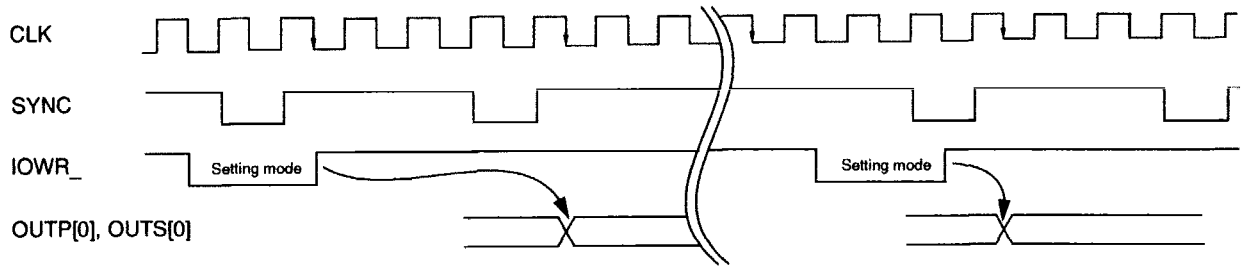


Figure 9-5. Countdown timing and control word write cycle (#1)

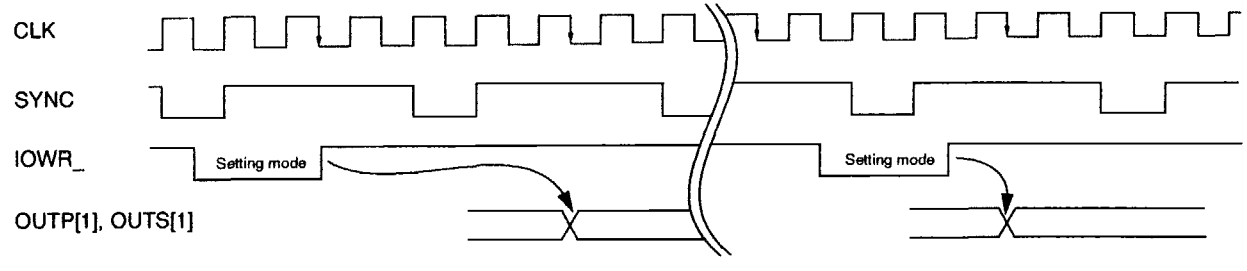


Figure 9-6. Countdown timing and control word write cycle (#2)

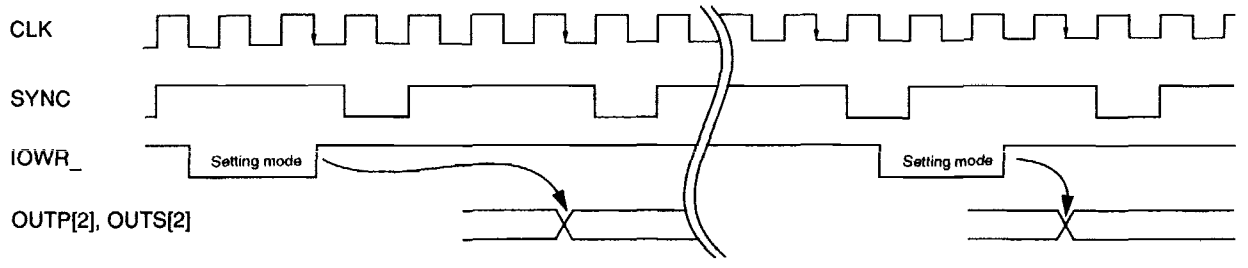


Figure 9-7. Countdown timing and control word write cycle (#3)

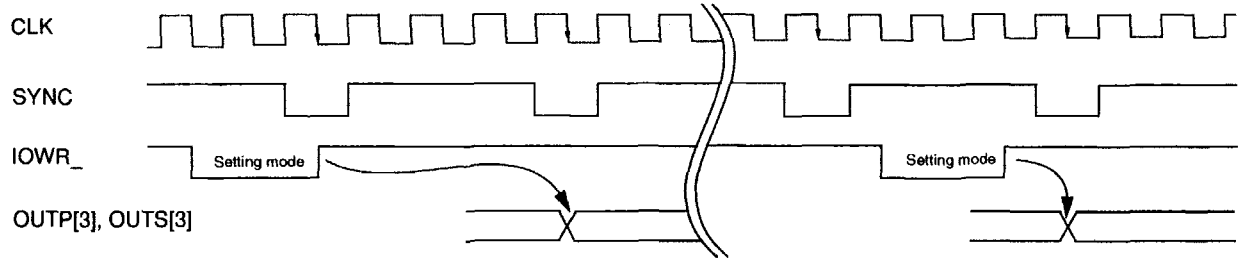


Figure 9-8. Countdown timing and count value write cycle (#0)

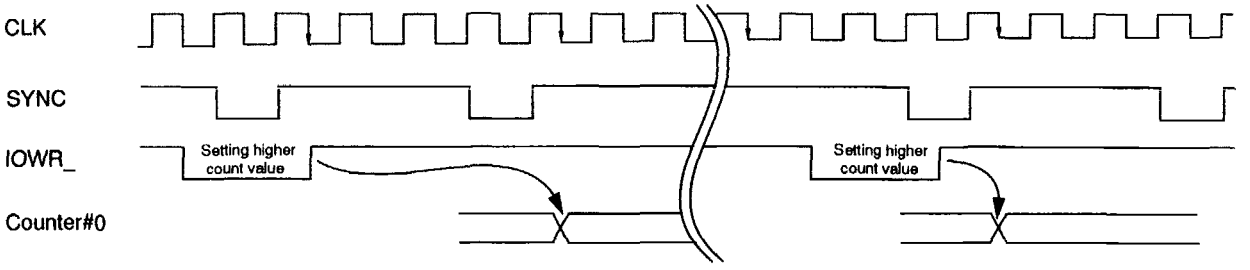


Figure 9-9. Countdown timing and count value write cycle (#1)

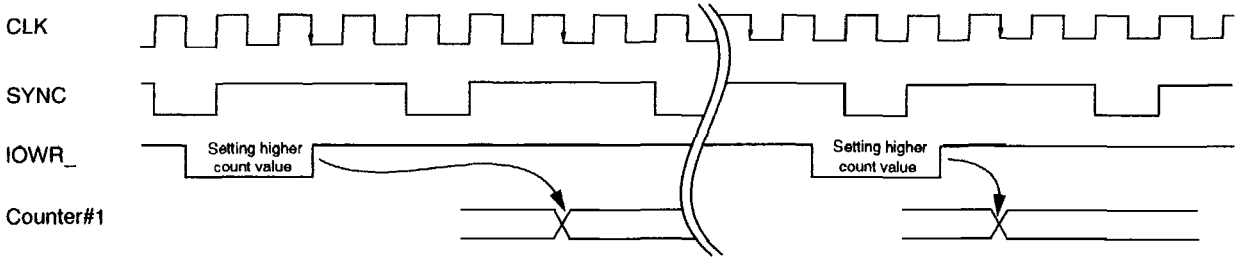


Figure 9-10. Countdown timing and count value write cycle (#2)

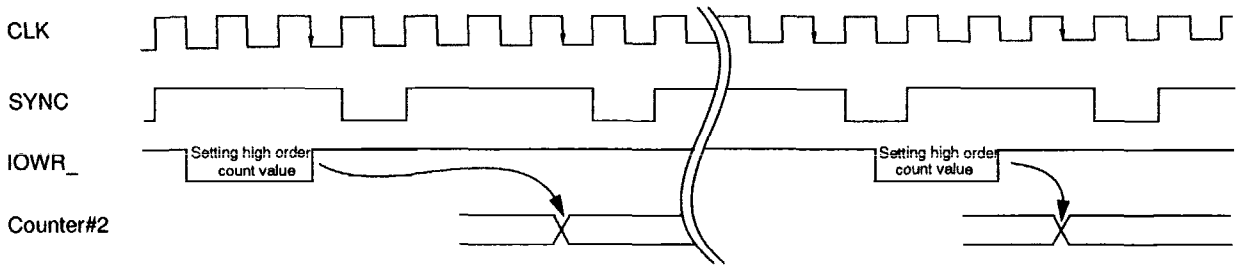


Figure 9-11. Countdown timing and count value write cycle (#3)

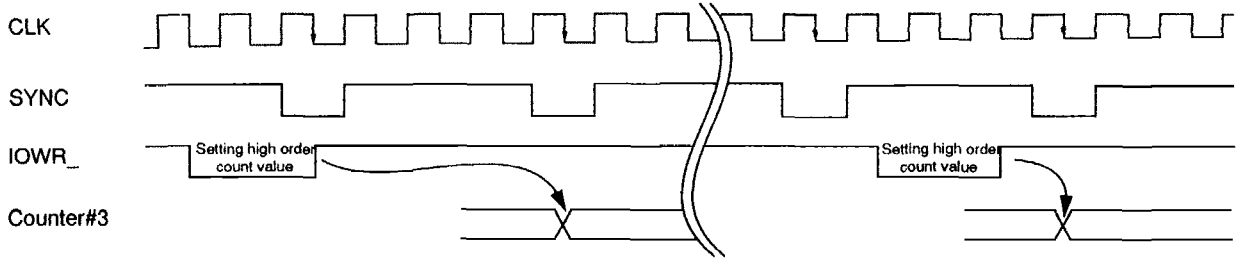


Figure 9-12. Continuous count mode (no GATE function, divide-by-4 frequency, TOGGLE bit "0")

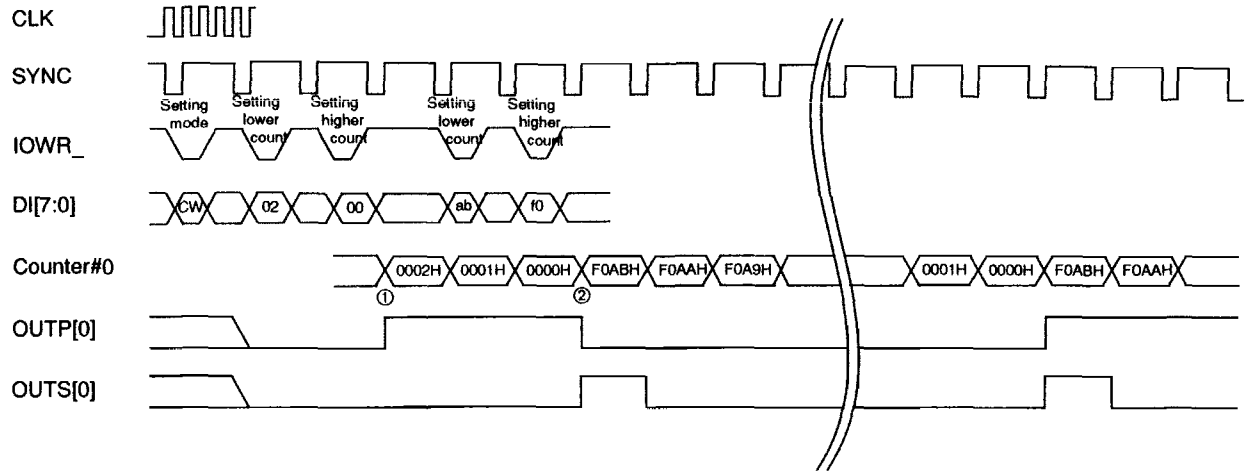


Figure 9-13. One shot count mode (no GATE function, divide-by-4 frequency, TOGGLE bit "0")

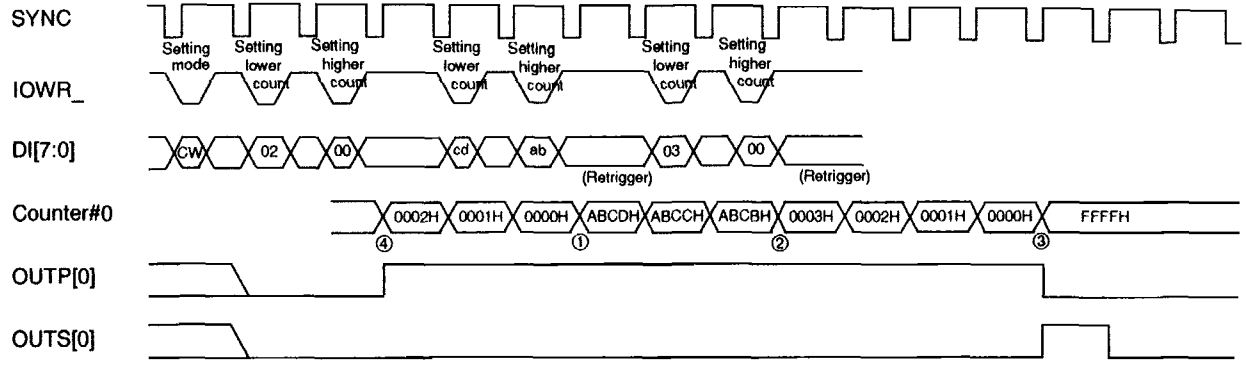


Figure 9-14. WDT mode (no GATE function, divide-by-4 frequency, TOGGLE bit "1")

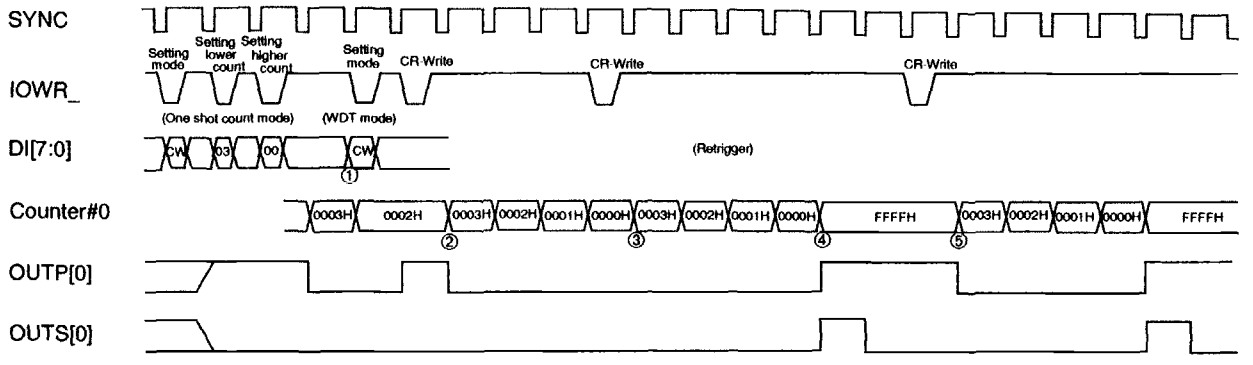
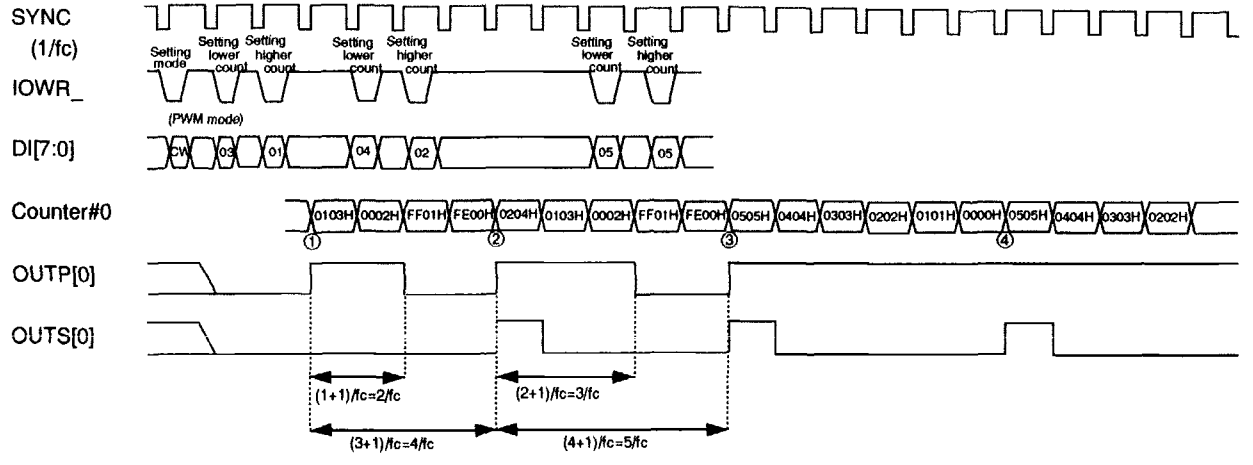


Figure 9-15. PWM mode (no GATE function, divide-by-4 frequency, TOGGLE bit "0")



9.11 Reset

Reset is performed by RESET_= "L" and affects the following features.

(1)Stops the operation of counter. The counter is kept in stop state even after clearing the reset.

(2)Clears a read sequence in readout of register within the counter and a write sequence in write of initial count value.

(3)Clears the CR register within the counter and the counter register to "FFFFH".

(4)Enters the one shot count mode (divide-by-256 frequency counter clock, TOGGLE bit "0"), sets the OUTP and OUTS outputs to "L", and holds the value after clearing the reset until the mode is set.

9.12 Precautions

Maximum/minimum value of initial count value

The following is the maximum/minimum values of initial count value.

Mode	Minimum value	Maximum value
Continuous count	0001H	FFFFH
One shot count	0001H	FFFFH
Pulse width modulation (PWM)	High order: 01H Low order: 01H	High order: FFH Low order: FFH
Watch dog timer (WDT)	Unable to set*	Unable to set*

*When it is set again and used in the WDT mode after it has been set to other mode, it operates according to the conditions of the mode in which the initial count value was set.

10. Parallel Port A

The parallel port A in the chip is KP65 macro cell. KP65 is the parallel port macro cell for KC80/KC82 ASIC microcontroller. It has 16 ports. Each port can be set as input or output one by one by program. Parallel port B in the chip is KP66 parallel port macro cell. See chapter 11 for parallel port B. The difference between parallel A and B is below here.

Features

KP65

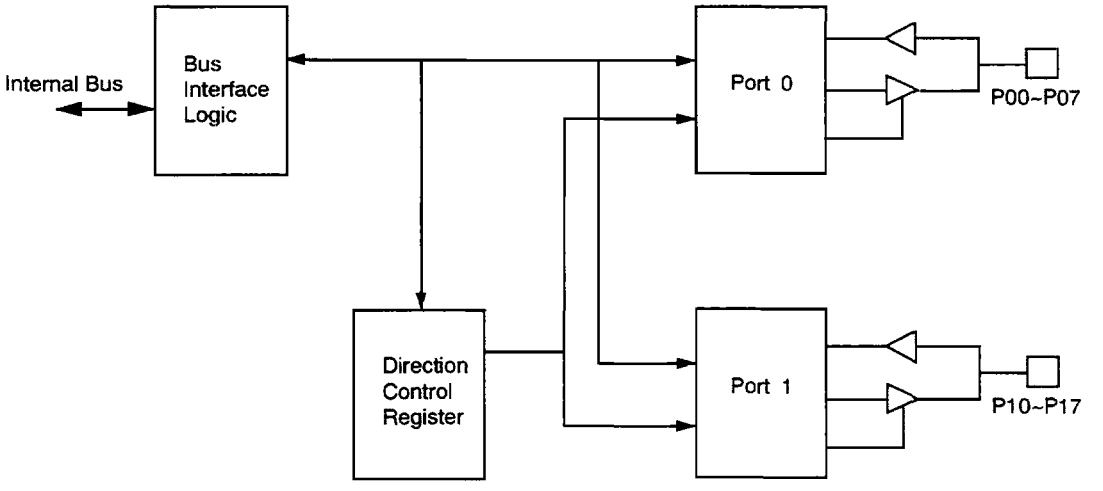
- The number of ports: 16
- Input/output setting: by each port

KP66

- The number of ports: 24
- Input/output setting: by each four ports
- Set/reset command available

10.1 Block Diagram

The KP65 block diagram and I/O register mapping is shown below.



I/O Register Mapping

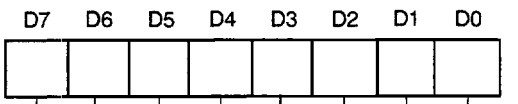
I/O address	Block	Write cycle	Read cycle
2CH	Parallel port A	Port 0	Port 0
2DH		Port 0 Direction Control Register	Port 0 Direction Control Register
2EH		Port 1	Port 1
2FH		Port 1 Direction Control Register	Port 1 Direction Control Register

10.2 Input/output Settings of Ports

Input/output settings for ports is done by I/O write to Port Direction Control Register.

Port Direction Control Registers

KP65 has two eight bit Port Direction Control Registers (I/O address = 2DH, 2FH). Each bit of these registers controls input/output direction of each port.



A[7:0]= 2DH: Port 0 Direction Control Register
2FH: Port 1 Direction Control Register

PnEN[0]
1: Set bit 0 of port n as output
0: Set bit 0 of port n as input

PnEN[1]
1: Set bit 1 of port n as output
0: Set bit 1 of port n as input

PnEN[2]
1: Set bit 2 of port n as output
0: Set bit 2 of port n as input

PnEN[3]
1: Set bit 3 of port n as output
0: Set bit 3 of port n as input

PnEN[4]
1: Set bit 4 of port n as output
0: Set bit 4 of port n as input

PnEN[5]
1: Set bit 5 of port n as output
0: Set bit 5 of port n as input

PnEN[6]
1: Set bit 6 of port n as output
0: Set bit 6 of port n as input

PnEN[7]
1: Set bit 7 of port n as output
0: Set bit 7 of port n as input

n=0 (at A[1]=0)
n=1 (at A[1]=1)

10.3 Input and Output Operation

Input Operation
The data input from the ports set as input ports are read from the data bus.

Output Operation
The data written into the port register are output from the ports set as outputs.

10.4 Reset

On reset, the registers below here are initialized.

Port 0	register	reset
Port 1	register	reset
Port 0	Direction Control Register	reset
Port 1	Direction Control Register	reset

10.5 Precaution

The port multiplexed with other signal must be set as an input port when the pin is not used as the port.

11. Parallel Port B

The parallel port B is KP66 macro cell. KP66 can be used as the parallel port macro cell for KC80/KC82 ASIC microcontroller. It has 24 ports. Each upper or lower 4 ports can be set as input or output as a group by software. Parallel port A is KP65 parallel port macro cell. See chapter 10 for parallel port A. The difference between parallel A and B is below here.

Features:

KP66

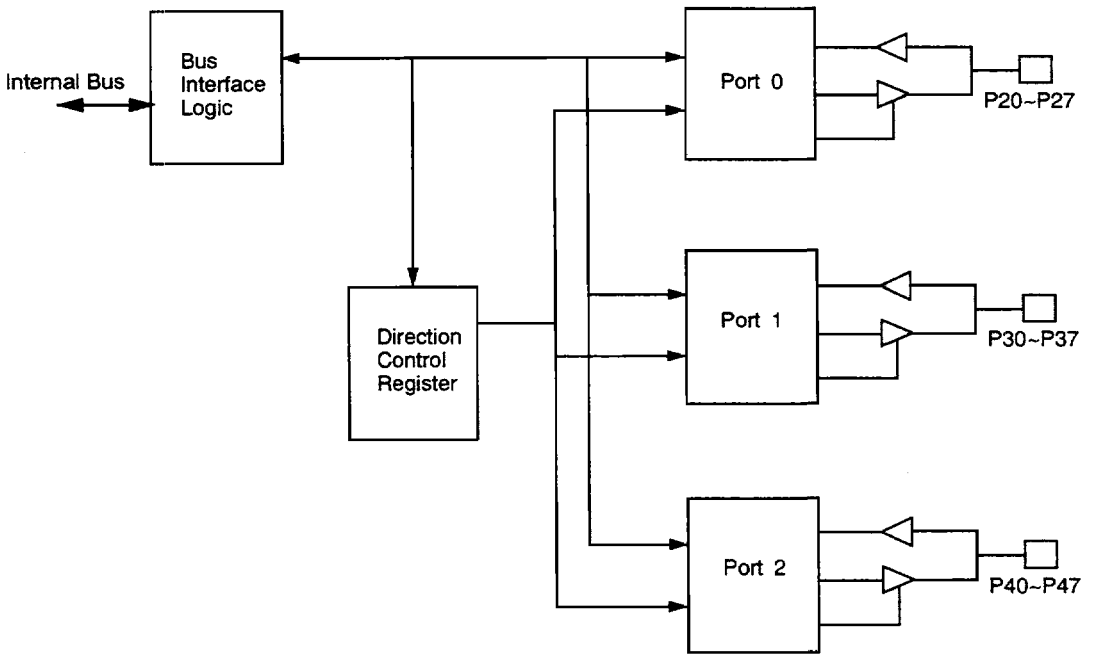
- The number of ports: 24
- Input/output setting unit: by each four ports
- Set/reset command available

KP65

- The number of ports: 16
- Input/output setting unit: by each port

11.1 Block Diagram

The KP66 block diagram and I/O register mapping is shown below.



I/O Register Mapping

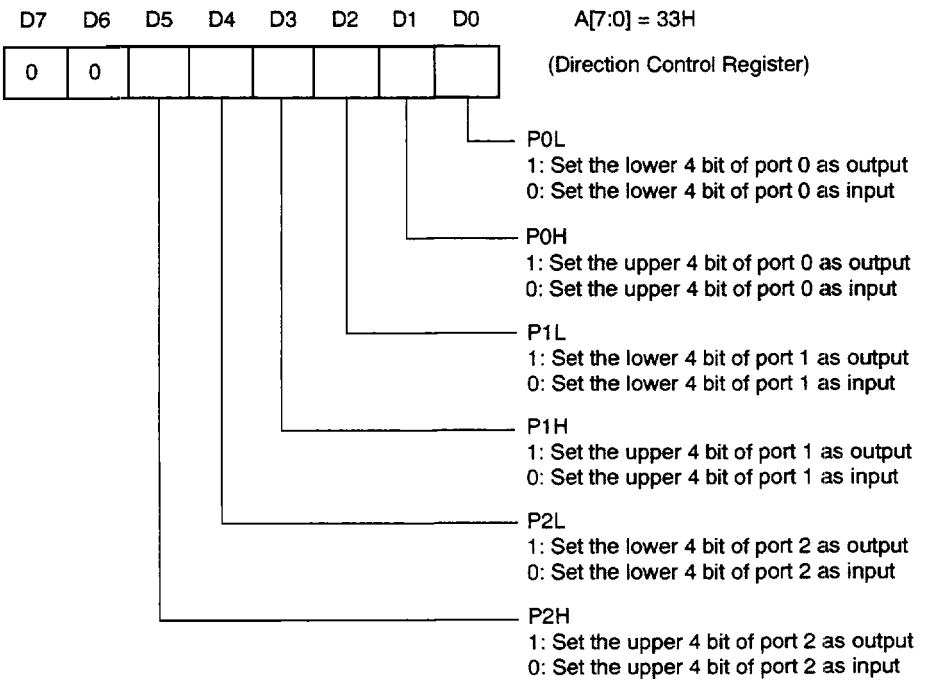
I/O address	Block	Write cycle	Read cycle
30H	Parallel port B	Port 0	Port 0
31H		Port 1	Port 1
32H		Port 2	Port 2
33H		Control Commands	Direction Control Register

11.2 Control Commands

Input/output direction of each port is set by control commands (I/O address = 33H). The upper 2 bits determine which command to be issued.

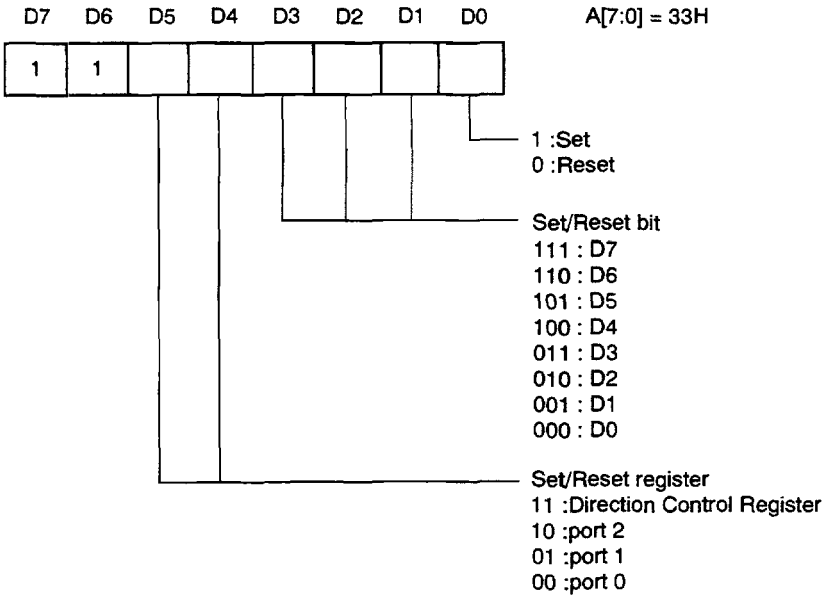
Port Direction Control Command (Direction Control Register)

In KP66, input/output directions of 24 ports are set by Direction Control Register. The input/output directions of each 4 ports is controlled by 1 bit of Direction Control Register. This command writes Direction Control Register directly.



Bit Control Command

Bit Control Command can set or reset only one bit of 24 ports or Direction Control Command.



11.3 Input and Output Operation

Input Operation
The data input from the ports set as input ports are read from the data bus.

Output Operation
The data written into the port register are output from the ports set as output ports.

11.4 Reset

On reset, the registers are initialized as follows.

Port 0	register	reset
Port 1	register	reset
Port 2	register	reset
Direction Control Register		reset

11.5 Precaution

The port multiplexed with other signal must be set as an input port when the pin is not used as the port.

12. Setting Operation Mode

The operation mode of this chip is specified by the external input pins MODE[1:0] and the system control registers.

12.1 Setting operation mode with the input pins

MODE [1:0] = 00 : disable
01 : Bug Finder Boot-on-RAM mode
10 : maximum mode (Outputs A18 and A17 from pins 68 and 69.)
11 : normal mode

Normal mode
In the normal mode, M1CS_ and M0CS_ are output from the pins 68 and 69. These two signals can be used as chip select signal for external memory. Up to 256KB of external memory can be connected.

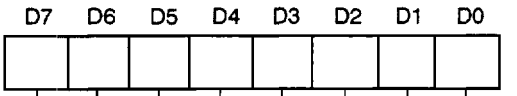
Maximum mode
In the maximum mode, A[18] and A[17] are output from the pins 68 and 69. Up to 512KB of external memory can be connected.

Bug Finder Boot-on-RAM mode
In the Bug Finder Boot-on-RAM mode, the connection of a Bug Finder adapter to the BFSIO pin allows the Bug Finder to start in external RAM. M1CS_ and M0CS_ are output from the pins 68 and 69. For more information see the Bug Finder manual. The operation of this chip is same as that in normal mode. The external memory which can be connected the maximum 256 Kbyte.

In either the maximum mode or normal mode, there is 1MB of internal address space and the internal RAM is mapped to the address space FFE00H - FFFFFH (512 byte).

12.2 System control registers

SCR0 (I/O address = 3AH)



Serial port TXC, RXC input
1 : Input from pins 73 and 74 is used.
0 : OUTBP output of Timer B channel 1 is connected internaly. Pins 73 and 74 function as P14 and P15.

Functions of pins 90, 91, 94 and 95
1 : Function as DSR_, CTS_, DTR_ and RTS_.
0 : Function as P41, P42, P45 and P46. "L" is input to DSR_ and CTS_.

Functions of pins 89 and 93
1 : Function as SYNDBD and SYDTIN.
0 : Function as P43 and P47. "L" is input to SYDTIN.

Functions of pins 1, 4, 5 and 6
1 : Function as OUTBS0, OUTBP1, OUTBP2 and SYNC.
0 : Function as P30 to 33.

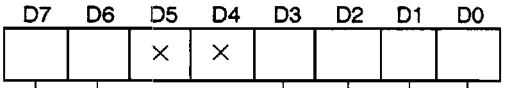
Functions of pins 97 to 100
1 : Function as OUTBS2, OUTBP0, OUTA1 and OUTA0.
0 : Function as P34 to 37.

Timer A channel 0 GATE input (GATEA0)
1 : Input from pin 79.
0 : "H" is always input. Pin 79 functions as P10.

Timer A channel 1 GATE input (GATEA1)
1 : Input from pin 76.
0 : "H" is always input. Pin 76 functions as P12.

Operations of Timer A
1 : The channels 1 and 0 of Timer A are cascaded and function as 32-bit counter.
0 : The channels 1 and 0 of Timer A function independently.

SCR1 (I/O address = 3BH)



Functions of pin 72
1 : Outputs KC82's M1_.
0 : Functions as P16.

Functions of pin 71
1 : Outputs KC82's HALT_.
0 : Functions as P17.

Functions of pins 92 and 96
1 : Function as KC82's BREQ_ and BACK_.
0 : Function as P40 and P44.
"H" is always input to KC82's BREQ_.

Functions of pin 85
1 : Function as KC82's NMI_ input.
0 : Function as P03 or IR3.
"H" is always input to KC82's NMI_.

External bus wait control

	external memory (00000 - 7FFFF)	external memory (80000 - FFDFF)	external I/O
00	1 wait state	1 wait state	2 wait states
01	1 〃 *	1 〃 *	2 〃
10	1 〃	0 〃	1 〃
11	0 〃	0 〃	1 〃

* wide write strobe option

Note: To use the external interrupt inputs IR0 to IR7 of the interrupt controller, set the corresponding parallel ports to input direction. See Chapter 6 "Interrupt Controller" for how to use the interrupt controller.

How to use pins 75 and 78

The I/O directions of pins 75 and 78 is determined only by the direction control register of the parallel port A regardless of the system control register.

Set P11 and P13 to input direction to use the pins 75 and 78 as XCLK0 and 1.

To use these pins as parallel port, set the corresponding Timer/Counter A channels to the mode in which the system clock is used as counter clock.

How to use pins 80, 81, 82 and 83

The I/O directions of pins 80, 81, 82 and 83 is determined only by the direction control register of the parallel port A regardless of the system control register. Set the P04, P05, P06 and P07 of the parallel port A to input direction to use external input from these pins as IR 4, 5, 6and 7.

On the contrary, to use these pins as P04, P05, P06 and P07 of the parallel port A, IR 4, 5, 6 and 7 should be masked by the setting of the interrupt controller.

How to use pin 85

The I/O direction of pin 85 is determined only by the direction control register of the parallel port A when pin 85 is used as P03 or IR3 by the system control register (D3 in SCR1= '0'). Set the P03 of the parallel port A to input direction to use external input from pin 85 as IR3.

On the contrary, to use this pin as P03 of the parallel port A, IR3 should be masked by the setting of the interrupt controller.

How to use pins 86, 87 and 88

The I/O directions of pins 86, 87 and 88 is determined only by the direction control register of the parallel port A regardless of the system control register. Set the P00, P01 and P02 of the parallel port A to input direction to use external input from these pins as IR 1, 2 and 3 or GATEB0, 1 and 2.

In this setting, external signal is always input in both IR 1, 2, 3 and GATEB 0, 1, 2. Any of the functions can be disabled by the mode setting of each block. For example, to use the pin 88 as GATEB0, it is required to set P00 to input direction by the direction control register of the parallel port A, and mask IR0 by the setting of the interrupt controller.

To use these pins as parallel port, it is recommended that the corresponding external interrupt inputs are masked and the counter channels of corresponding GATE inputs (GATEB 0, 1 and 2)are used without GATE capability.

How to use pins 89, 90, 91 and 92

When one or more pins 89, 90, 91 and 92 are used as control output signals (SYNDBD, RTS_, DTR_, BACK_), all of the four pins are used only as output pins. In this case, even when they are used as the parallel ports (P47, P46, P45 and P44) setting of the output direction is necessary because they are used only as the output ports.

The controls of these pins are programmed by the system control registers SCR1 (D2), SCR0(D1, D2), and the direction control register of the port B (D5, see chapter 11).

How to use pins 93, 94, 95 and 96

When one or more pins 93, 94, 95 and 96 are used as control input signals (SYDTIN, CTS_, DSR_, BREQ_), all of the four pins are used only as input pins. In this case, even when they are used as the parallel ports (P43, P42, P41 and P40) setting of the input direction is necessary because they are used only as the input ports.

The controls of these pins are programmed by the system control registers SCR1 (D2), SCR0(D1, D2), and the direction control register of the port B (D4, see chapter 11).

12.3 Reset

Both SCR0 and SCR1 are reset to "00H".

13. Address Mapping

Internal I/O mapping

Table 13-1. Internal I/O Mapping

I/O address	Block	Write cycle	Read cycle
00H	KC82(MMU)	BBR1 (boundary/base register 1)	BBR1 (boundary/base register 1)
01H		BR1 (base register 1)	BR1 (base register 1)
02H		BBR2 (boundary/base register 2)	BBR2 (boundary/base register 2)
03H		BR2 (base register 2)	BR2 (base register 2)
04H		BBR3 (boundary/base register 3)	BBR3 (boundary/base register 3)
05H		BR3 (base register 3)	BR3 (base register 3)
06H		BBR4 (boundary/base register 4)	BBR4 (boundary/base register 4)
07H		BR4 (base register 4)	BR4 (base register 4)
08H~0FH	reserved for Kawasaki Steel Corp.		
20H	Timer/Counter B	channel 0 counter	channel 0 counter
21H		channel 0 control word	channel 0 status
22H		channel 1 counter	channel 1 counter
23H		channel 1 control word	channel 1 status
24H		channel 2 counter	channel 2 counter
25H		channel 2 control word	channel 2 status
26H		reserved for Kawasaki Steel Corp.	reserved for Kawasaki Steel Corp.
27H		reserved for Kawasaki Steel Corp.	reserved for Kawasaki Steel Corp.
28H	Timer/Counter A	channel 0 counter	channel 0 counter
29H		channel 0 control word	channel 0 status
2AH		channel 1 counter	channel 1 counter
2BH		channel 1 control word	channel 1 status
2CH	Parallel port A	Port 0	Port 0
2DH		Port 0 direction control register	Port 0 direction control register
2EH		Port 1	Port 1
2FH		Port 1 direction control register	Port 1 direction control register
30H	Parallel port B	Port 0	Port 0
31H		Port 1	Port 1
32H		Port 2	Port 2
33H		Control command	Direction control register
34H	Interrupt controller	LERL/PGRL	ISRL
35H		LERH/PGRH	ISRH
36H		IMRL	IMRL
37H		IVR/IMRH	IMRH
38H	Serial port	Transmit data	Receive data
39H		Setting command/mode	Status
3AH	System control register	SCR0	SCR0
3BH		SCR1	SCR1
3CH~3FH	recerved for KSC		

NOTE: I/O address decode for those internal I/O is eight bit decode. The upper 8 bits, A15~A8 are neglected.

Memory mapping

Normal mode:

Internal RAM area	Physical address space	FFE00H to FFFFFH (512 byte)
External memory area 1	Physical address space (M1CS_)	E0000H to FFDFFH (128 K - 512 byte)
External memory area 0	Physical address space (M0CS_)	00000H to IFFFFH (128 Kbyte)

Maximum mode:

Internal RAM area	Physical address space	FFE00H to FFFFFH (512 byte)
External memory area	Physical address space	00000H to 7FFFFH(512 Kbyte)

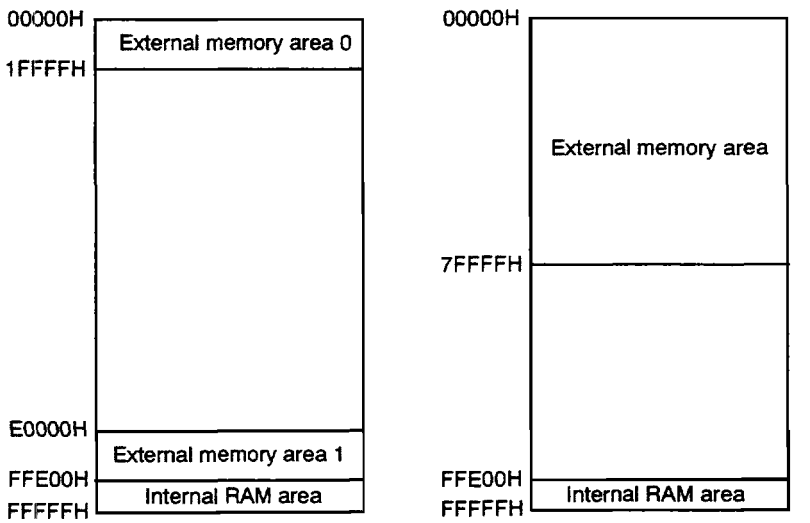


Figure 13-1. Memory Mapping

14. Oscillation Circuit

14.1 General description

The KL5C80A12 contains an oscillation buffer to generate system clock. The system clock in the chip is a signal which has a divide-by-2 frequency of the signal generated by this oscillation circuit.

14.2 Circuit structure

To generate system clock, the KL5C80A12 can be connected with a crystal oscillator (or ceramic oscillator), a feedback resistor and a condenser as external parts to form an oscillation circuit as shown in the right figure. The constants of external parts are dependent on oscillator, substrate boards and so on to be used. Use the values recommended by oscillator manufacturers for optimum values of external parts constants. The system clock in the chip is a signal which has a divide-by-2 frequency of the signal generated by this oscillation circuit. A frequency divide circuit is built in the chip.

Table 14-1. Oscillation Frequency

Oscillation frequency	Operation power supply voltage
2~20 MHz	5V ± 5%, 5V ± 10%

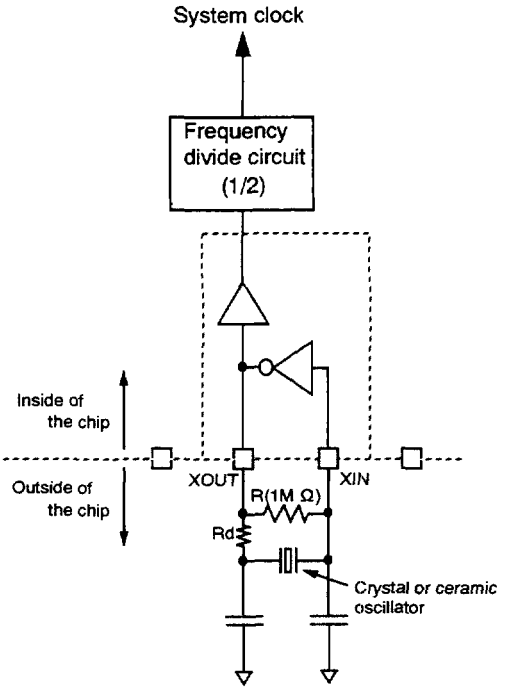


Figure 14-1. Oscillation Circuit

Notes

Be sure to use the CLK pin when the system clock is output. Be sure not to output its signal directly from XIN and XOUT.

Be sure to input a signal with twice the system clock frequency from XIN when the system clock is input from the outside. In this case, make a parasitic capacity of XOUT as low as possible.

Table 14-2. Recommended Range of External Circuit Constants

	Rd	CI, CO
Crystal	100 ~ 800 Ω	5 ~ 30pF
Ceramic	30 ~ 300 Ω	5 ~ 100pF

15. Connecting Debugging Tool

There are two software development tools for this LSI; ICE and Bug Finder. There are several ICEs for KL5C80A12. Please co80A12ntact our sales person. Bug Finder is a simplified debugging tool developed by Kawasaki Steel Corporation. For more details of this tool, please contact our sales person.

16. Electrical Characteristics

16.1 Absolute Maximum Ratings

Table 16-1. Absolute Maximum Ratings (with respect to GND)

Item	Symbol	Rating	Unit
Power supply voltage	V_{DD}	-0.6 ~ +7.0	V
Input voltage	V_{IN}	-0.6 ~ VDD+0.6	V
Storage temperature	T_{STG}	-40 ~ +125	℃

16.2 D.C. Characteristics (5V +- 10%)

Table 16-2. Recommended Operation Conditions

Item	Symbol	Rating	Unit
Power supply voltage	V_{DD}	4.5 ~ 5.5	V
Ambient temperature	T_A	0 ~ +70	℃

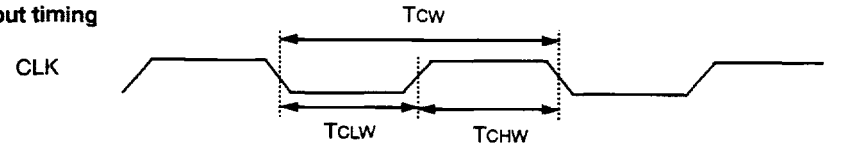
Table 16-3. Electrical Characteristics (under recommended operation conditions)

Item	Symbol	Rating			Unit	Test condition
		Min.	Typ.	Max.		
Input voltage (all input pins except RESET_)	V_{IH}	3.6	—	VDD	V	
	V_{IL}	GND	—	1.4	V	
RESET_ input pin (Schmitt trigger input)	V+	2.4	—	4.0	V	
	V-	0.9	—	2.3	V	
	Vh	0.9	—	—	V	
Output voltage	V_{OH}	3.5	—	—	V	I_{OH}=-4mA
	V_{OL}	—	—	0.4	V	I_{OL}=4mA
Output current	I_{OUT}			± 4	mA	
Input leakage current	I_{IL}	-10	—	—	μ A	V_{IN}=GND
	I_H	—	—	10	μ A	V_{IN}=V_{DD}
Output leakage current	I_{OZ}	-10	—	10	μ A	at high impedance output
Pull-up current	I_{PU}	20	95	250	μ A	V_{IN}=GND
Power supply current	I_{DDOP}		25*		mA	f(CLK)=10MHz
Standby supply current	I_{DDS}		1.0*	100	μ A	CLK stop, all pins=H

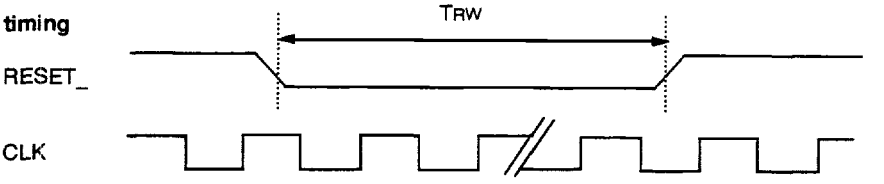
* at T_A=25 ℃

16.3 A.C. Characteristics

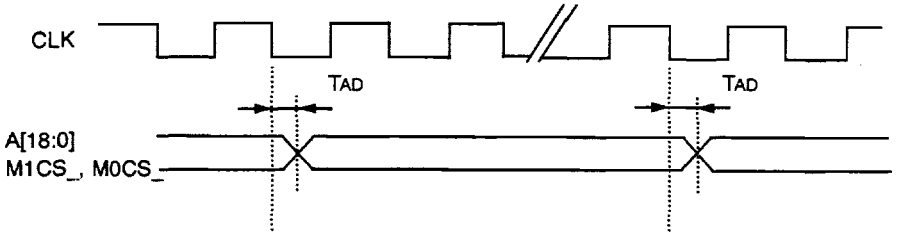
Clock output timing



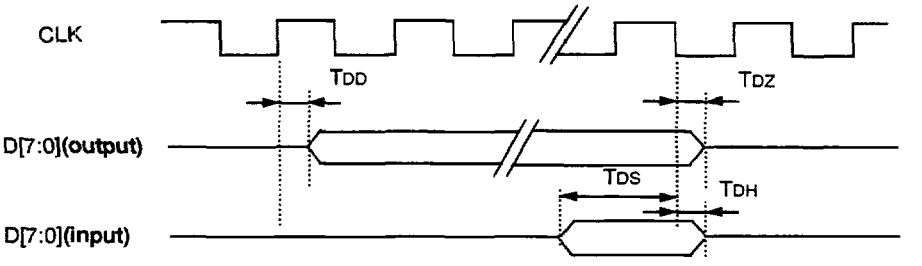
Reset input timing



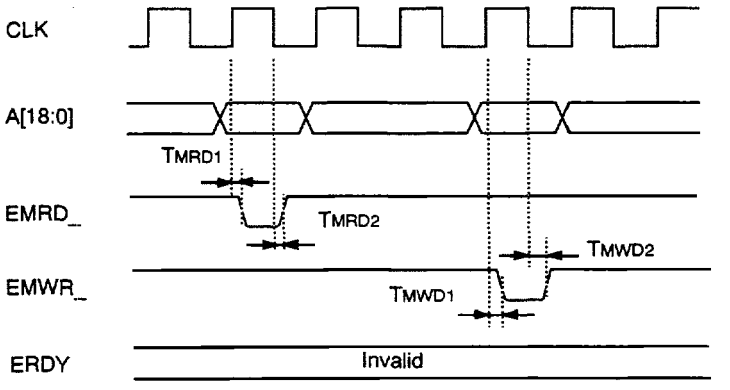
Address output timing



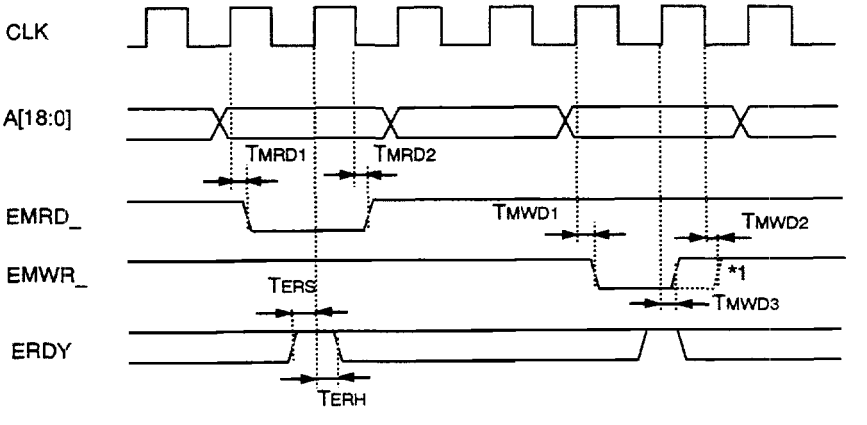
Data I/O timing



External memory access cycle (0 wait state)

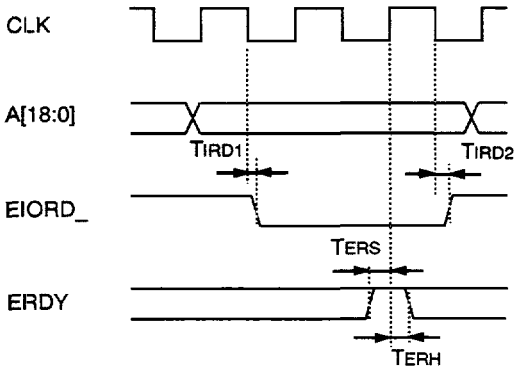


External memory access cycle (1 wait state)

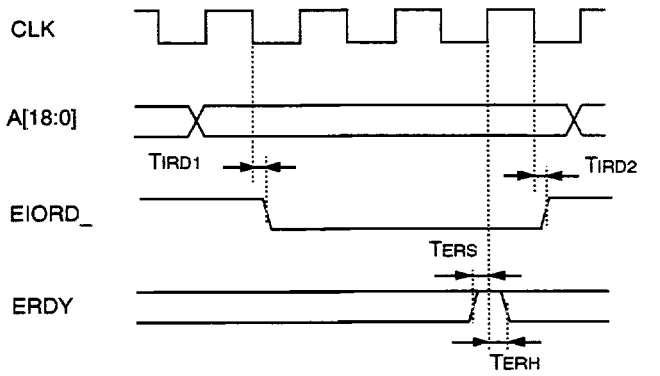


*1 timing of EMWR_ signal with the wide write strobe option selected

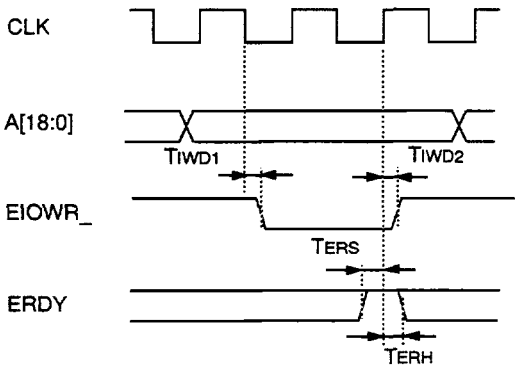
External I/O read cycle (1 wait state)



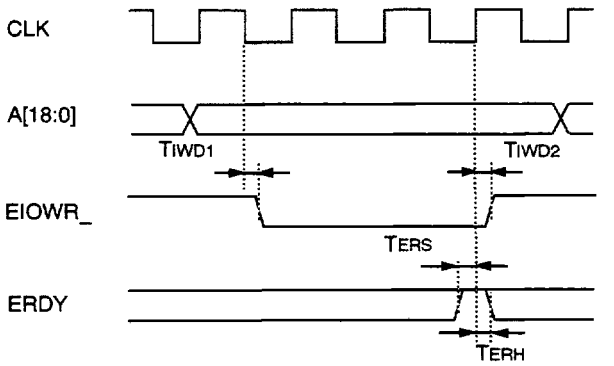
External I/O read cycle (2 wait states)



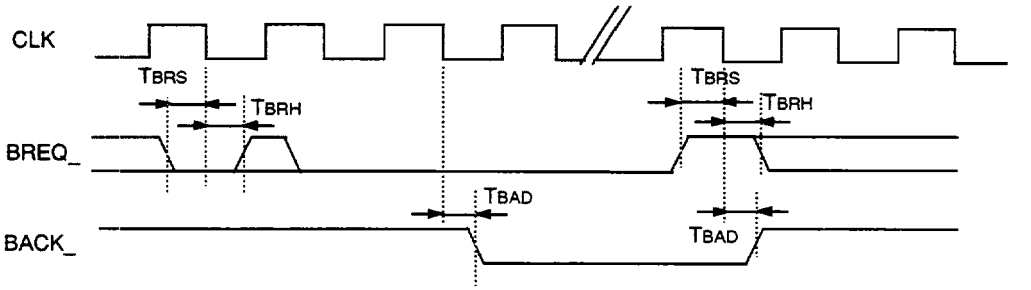
External I/O write cycle (1 wait state)



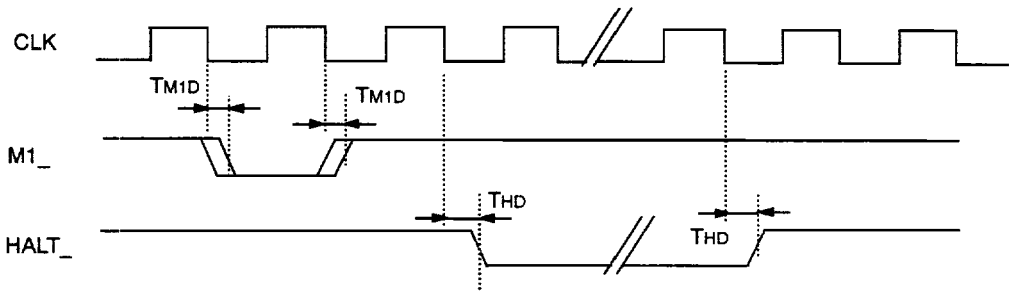
External I/O write cycle (2 wait states)



Bus request input timing

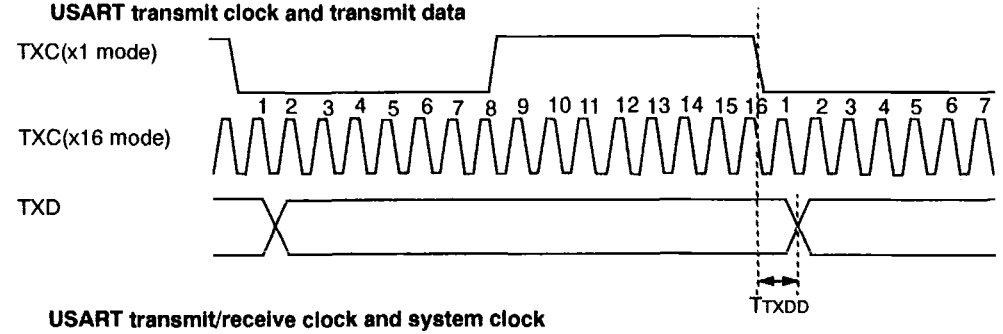


CPU control output timing

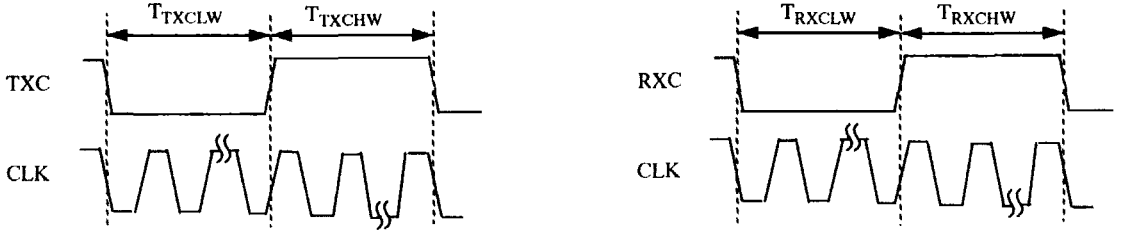


Be sure to sample M1_ signal at the rising edge of CLK to use the signal.

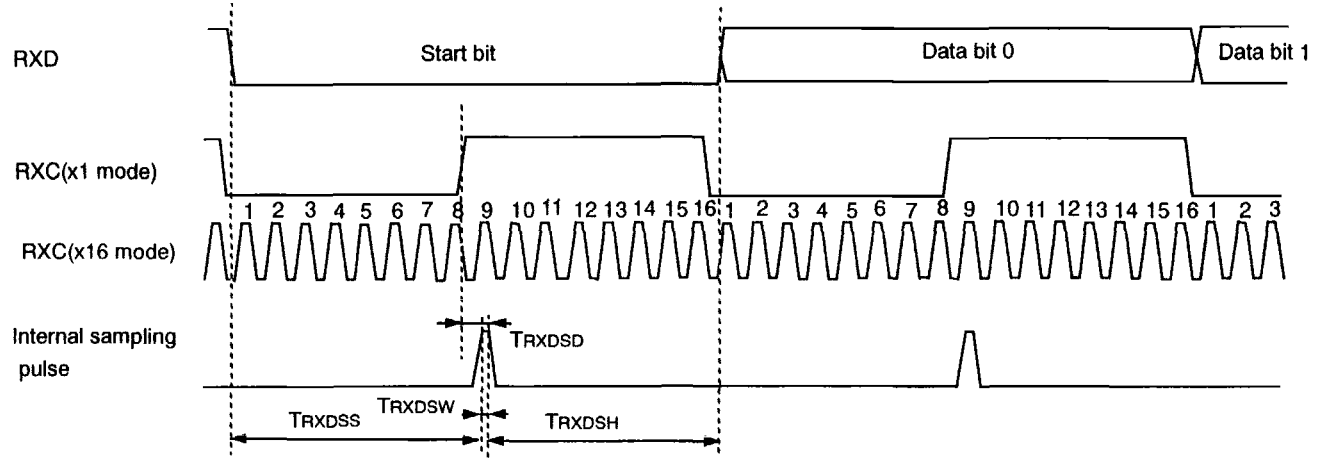
USART transmit clock and transmit data



USART transmit/receive clock and system clock

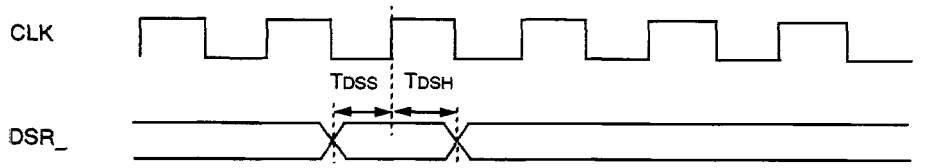


USART receive clock and receive data

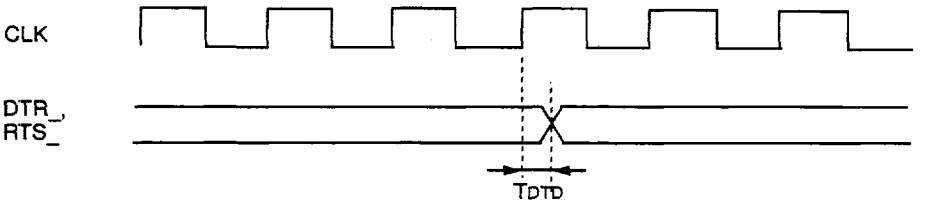


KAWATETSU

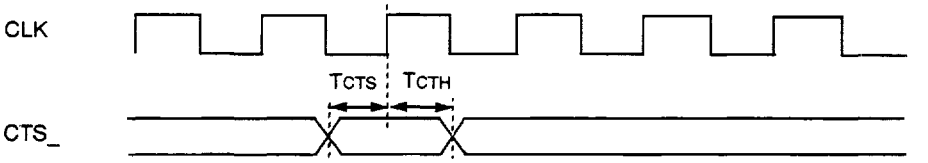
DSR_ input timing



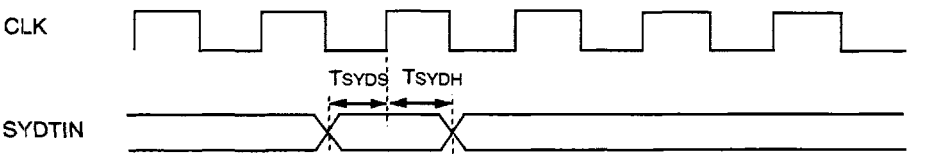
DTR_, RTS_ output timing



CTS_ input timing

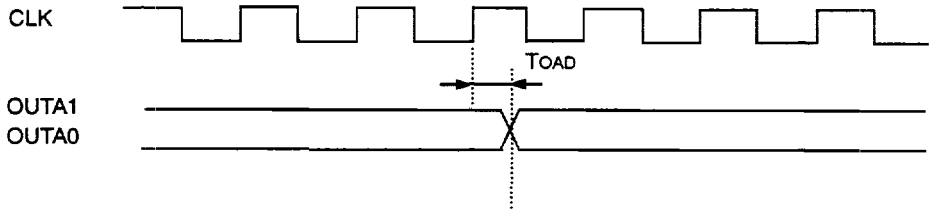


SYDTIN input timing

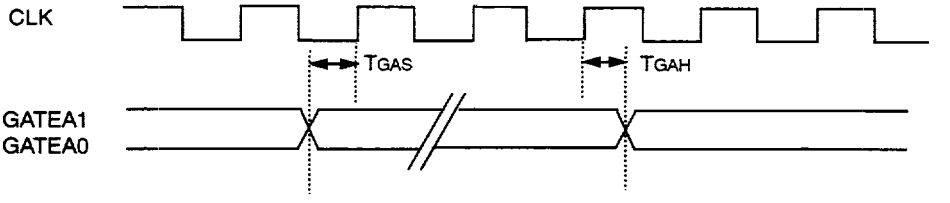


Timer/Counter A

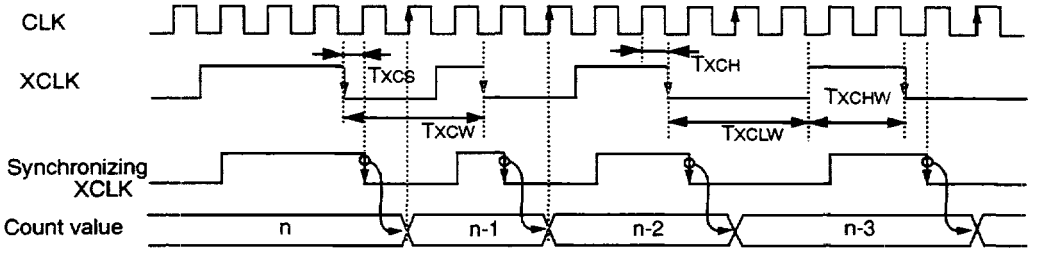
Counter output timing



Counter gate timing



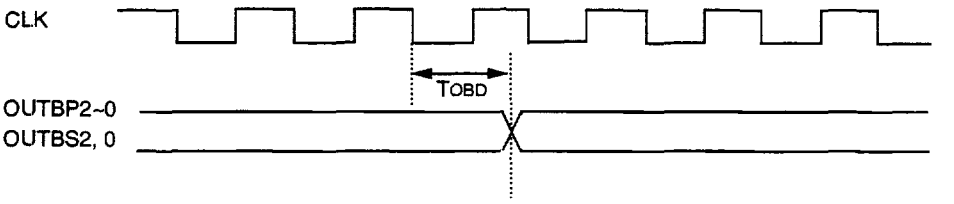
External counter clock input timing



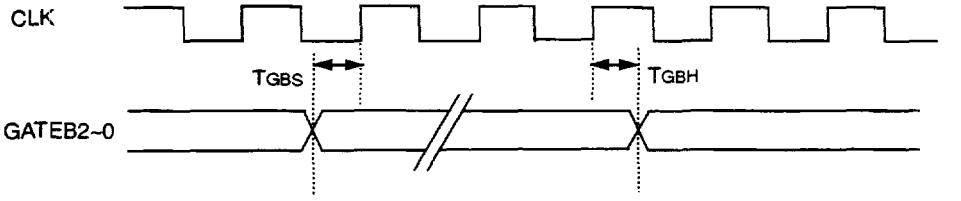
※ Be sure to sample external clock at the rising edge of system clock.

Timer/Counter B

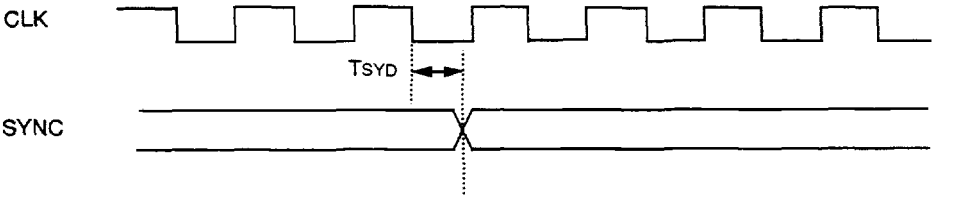
Counter output timing



Counter gate timing

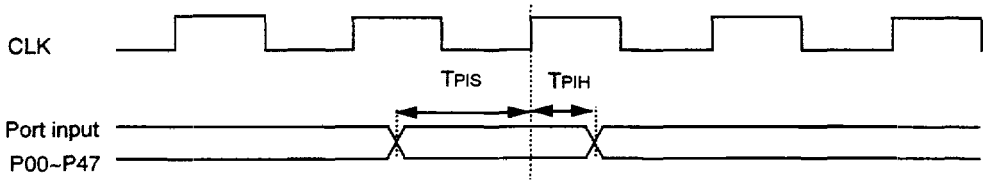


SYNC output timing

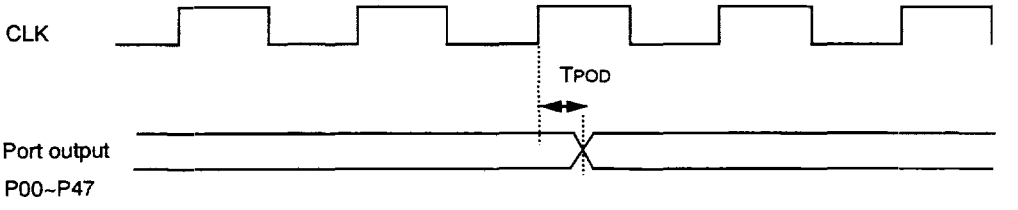


Parallel PortA/B

Port input timing

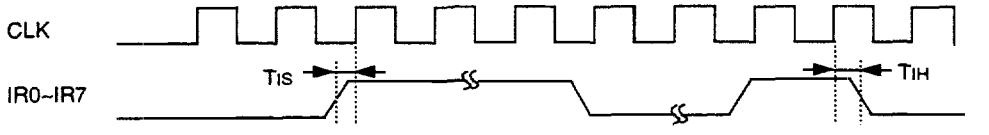


Port output timing



Interrupt Controller

External interrupt timing



KL5C80A12 AC Characteristics (5V ± 10%)

No.	Item	MIN	TYP	MAX	Unit
TCYC	XIN cycle time	50.0			ns
TCW	CLK cycle time	100.0			ns
TCLW	CLK "L" pulse width		50.0		ns
TCHW	CLK "H" pulse width		50.0		ns
TRW	RESET_ pulse width	3			clk
TAD	address delay	8.0		50.0	ns
TDD	CLK→data output delay			37.0	ns
TDZ	CLK→data output off delay	6.0			ns
TDS	data input set up time	7.0			ns
TDH	data input hold time	0.5			ns
TMRD1	CLK rising edge → EMRD_ "L" delay			22.0	ns
TMRD2	CLK falling edge →EMRD_ "H" delay	5.0		25.0	ns
TMWD1	CLK rising edge →EMWR_ "L" delay			22.0	ns
TMWD2	CLK falling edge →EMWR_ "H" delay	5.0		25.0	ns
TMWD3	CLK rising edge →EMWR_ "H" delay			23.0	ns
TERS	ERDY set up time	5.0			ns
TERH	ERDY hold time	0.5			ns
TIRD1	CLK falling edge →EIORD_ "L" delay			22.0	ns
TIRD2	CLK falling edge →EIORD_ "H" delay	5.0		23.0	ns
TIWD1	CLK falling edge →EIOWR_ "L" delay			22.0	ns
TIWD2	CLK rising edge →EIOWR_ "H" delay			23.0	ns
TBRS	BREQ_ set up time	5.0			ns
TBRH	BREQ_ hold time	1.6			ns
TBAD	BACK_ delay			43.0	ns
TM1D	M1_ delay			44.0	ns
THD	HALT_ delay			45.0	ns
TTXDD	TXD delay (from external TXC input)			21.0	ns
TTXCLW	TXC "L" pulse width	2.5			clk
TTXCHW	TXC "H" pulse width	2.5			clk

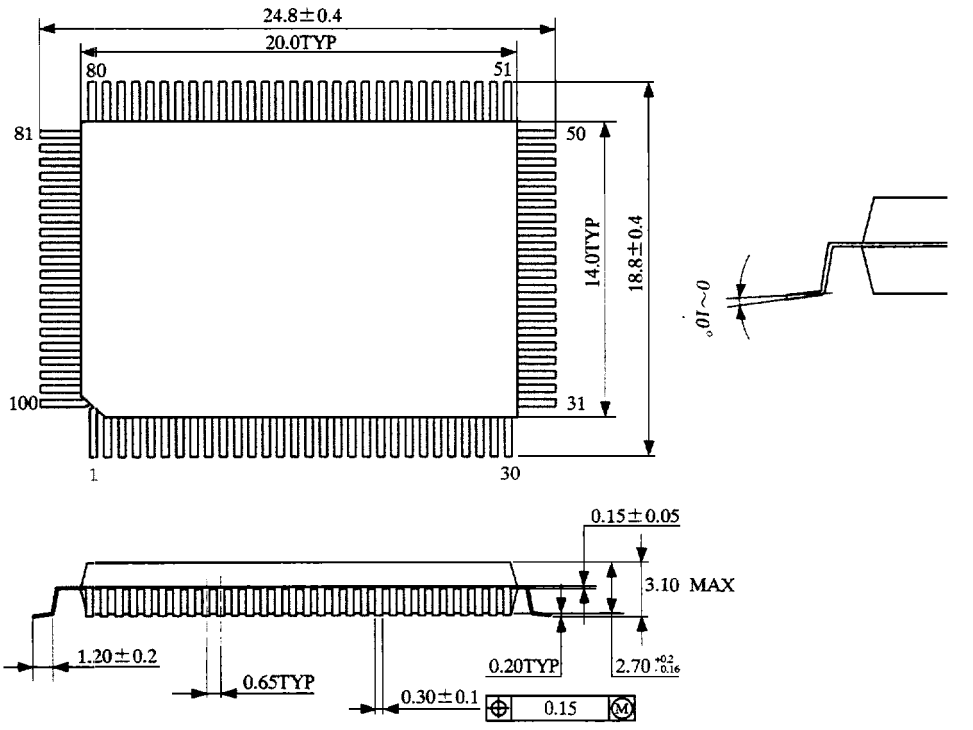
No.	Item	MIN	TYP	MAX	Unit
TRXCLW	RXC "L" pulse width	2.5			clk
TRXCHW	RXC "H" pulse width	2.5			clk
TRXDSD	RXD sampling pulse delay	2		3	clk
TRXDSW	RXD sampling pulse delay		1		clk
TRXDSS	RXD set up time		3		clk
TRXDSH	RXD hold time		0		clk
TDSS	DSR_ set up time	5.0			ns
TDSH	DSR_ hold time	2.8			ns
TDTD	DTR_, RTS_ delay			25.0	ns
TCTS	CTS_ set up time	5.0			ns
TCTH	CTS_ hold time	2.6			ns
TSYDS	SYDTIN set up time	5.0			ns
TSYDH	SYDTIN hold time	2.9			ns
TOAD	Timer/counter A output delay			22.0	ns
TGAS	GATEA input set up time	5.0			ns
TGAH	GATEA input hold time	0.5			ns
TXCS	XCLK input set up time	5.0			ns
TXCH	XCLK input hold time	3.0			ns
TXCW	XCLK period	2			clk
TXCLW	XCLK "L" pulse width	1			clk
TXCHW	XCLK "H" pulse width	1			clk
TOBD	Timer/counter B output delay			42.0	ns
TGBS	GATEB input set up time	5.0			ns
TGBH	GATEB input hold time	3.7			ns
TSYD	SYNC output delay			19.0	ns
TPIS	Port input set up time	5.0			ns
TPIH	Port input hold time	0.5			ns
TPOD	Port output delay			30.0	ns
TIS	External interrupt input set up time	5.0			ns
TIH	External interrupt input hold time	2.3			ns

Note 1) Output load CL is 70pf.

2) "clk" in unit is numbers of the system clock.

17. Physical Dimensions

The KL5C80A12's package is a plastic QFP100. The following is the physical dimensions of QFP100.



Note: All dimensions are in mm.