DATA SHEET

NEC

MOS INTEGRATED CIRCUIT

μPD70741

V821™
32-/16-BIT MICROPROCESSOR

The μPD70741 (V821) is a 32/16-bit RISC microprocessor that uses, as its processor core, the high-performance 32-bit microprocessor μPD70732 (V810™) designed for built-in control applications. It incorporates peripheral functions such as a DRAM/ROM controller, 2-channel DMA controller, real-time pulse unit, serial interface, and interrupt controller.

The V821, which offers quick real-time response, high-speed integer instructions, bit string instructions, and floating-point instructions, is ideally suited to use in OA equipment such as printers and facsimiles, image processing devices such as those used in navigation units, portable devices, and other devices demanding excellent cost performance.

The functions are described in detail in the following User's Manuals, which should be read before starting design work.

- **V821 User's Manual Hardware : U10077E**
- **V810 Family™ User's Manual Architecture : U10082E**

FEATURES

○ The V810 32-bit microprocessor is used as the CPU core
- Separate address/data bus
  Address bus : 24 bits
  Data bus : 16 bits
- Built-in 1-Kbyte instruction cache memory
- Pipeline structure of 1-clock pitch
- Internal 4-Gbyte linear address space
- 32-bit general-purpose registers: 32

○ Instructions ideal for various application fields
- Floating-point operation instructions and bit string instructions

○ Interrupts controller
- Nonmaskable : 1 external input
- Maskable : 8 external inputs and 11 types of internal sources
- Priorities can be specified in units of four groups.

○ Wait control unit
- Capable of CS control over four blocks in both memory and I/O spaces.
- Linear address space of each block: 16M bytes

○ Memory access control functions
- Supports DRAM high-speed page mode.
- Supports page-ROM page mode.

○ DMA controller (DMAC): 2 channels
- Maximum transfer count: 65 536
- Two transfer types (fly-by (1-cycle) transfer and 2-cycle transfer)
- Three transfer modes (single transfer, single-step transfer, and block transfer)

○ Serial interfaces : 2 channels
- Asynchronous serial interface (UART): 1 channel
- Synchronous serial interface (CSI): 1 channel

○ Real-time pulse unit
- 16-bit timer/event counter : 1 channel
- 16-bit interval timer : 1 channel

○ Watchdog timer functions

○ Clock generator functions

○ Standby functions (HALT, IDLE, and STOP modes)

The information in this document is subject to change without notice.

Document No. U11678EJ4V0DS00 (4th edition)
Date Published June 1998 J CP(K)
Printed in Japan

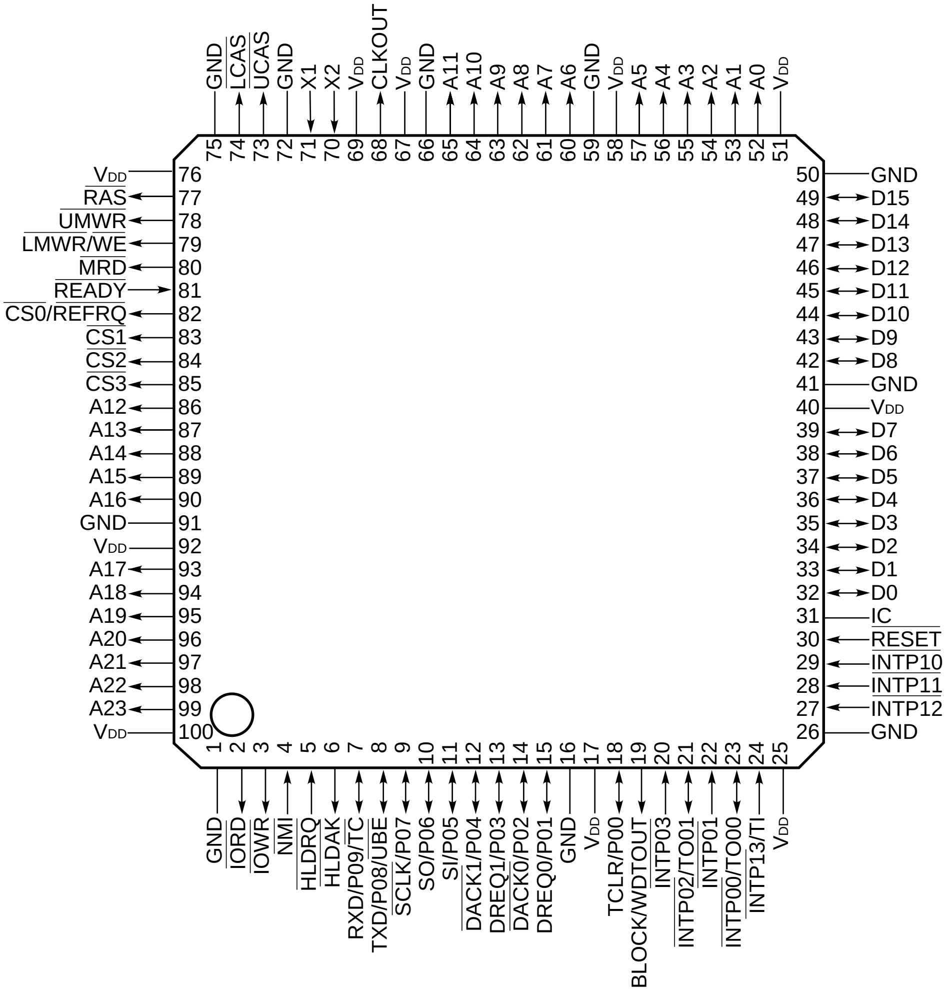
The mark ★ shows major revised points.

★ ORDERING INFORMATION

Part number	Package
μPD70741GC-25-8EU	100-pin plastic LQFP (fine pitch) (14 × 14 × 1.40 mm)

★ PIN CONFIGURATION (TOP VIEW)

100-pin plastic LQFP (fine pitch) (14 × 14 mm)
μPD70741GC-25-8EU

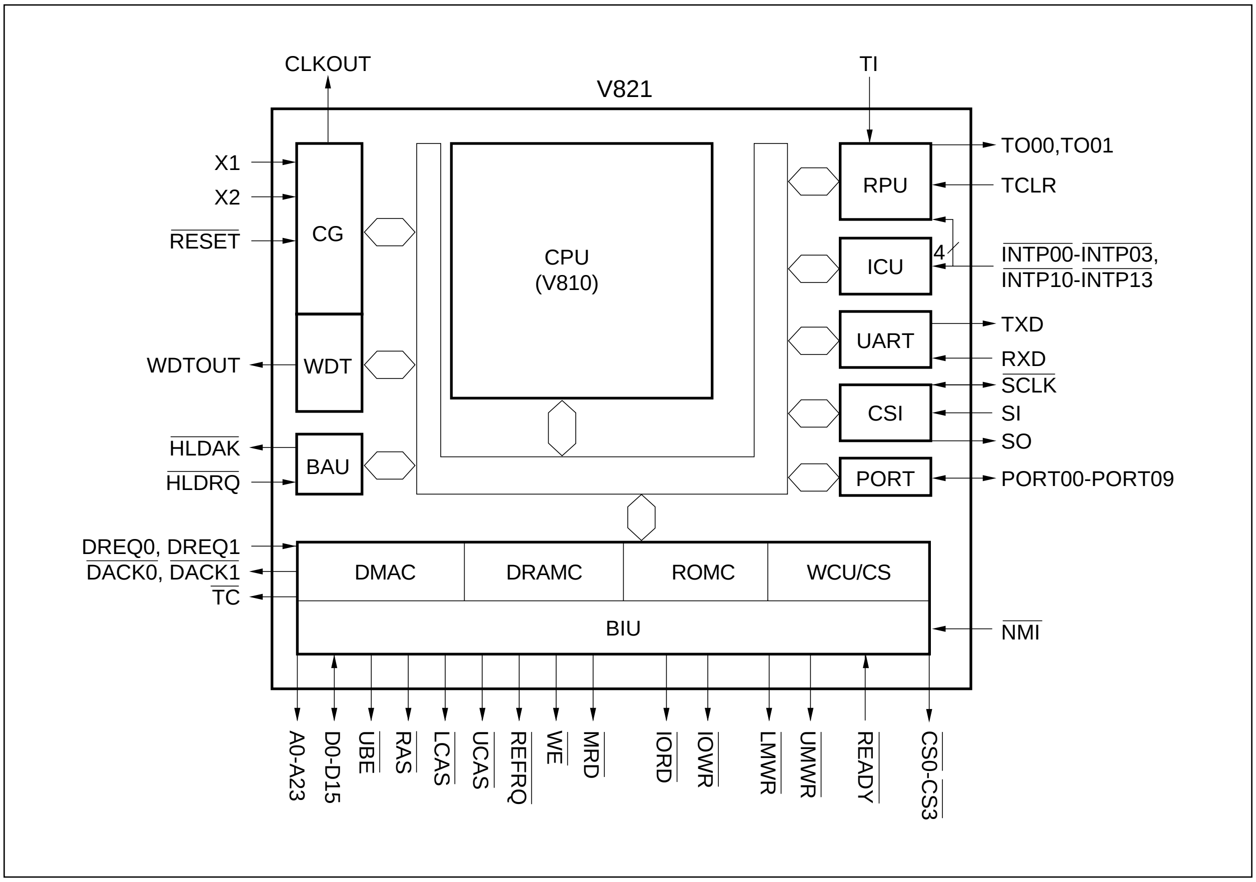


Caution Connect the IC pin to GND through a resistor.

PIN NAMES

A0-A23	:	Address Bus
BLOCK	:	Bus Lock
CLKOUT	:	System Clock Out
$\overline{\text{CS0}}$-$\overline{\text{CS3}}$	:	Chip Select
D0-D15	:	Data Bus
$\overline{\text{DACK0}}$, $\overline{\text{DACK1}}$	:	DMA Acknowledge
DREQ0, DREQ1	:	DMA Request
$\overline{\text{HLDAK}}$	:	Hold Acknowledge
$\overline{\text{HLDRQ}}$	:	Hold Request
$\overline{\text{INTP00}}$-$\overline{\text{INTP03}}$, $\overline{\text{INTP10}}$-$\overline{\text{INTP13}}$	:	Interrupt Request
$\overline{\text{IORD}}$	:	I/O Read
$\overline{\text{IOWR}}$	:	I/O Write
$\overline{\text{LCAS}}$	:	Lower Column Address Strobe
$\overline{\text{LMWR}}$	:	Lower Memory Write
$\overline{\text{MRD}}$	:	Memory Read
$\overline{\text{NMI}}$	:	Non-maskable Interrupt Request
P00-P09	:	Port
$\overline{\text{RAS}}$	:	Row Address Strobe
$\overline{\text{READY}}$	:	Ready
$\overline{\text{REFRQ}}$	:	Refresh Request
$\overline{\text{RESET}}$	:	Reset
RXD	:	Receive Data
$\overline{\text{SCLK}}$	:	Serial Clock
SI	:	Serial Input
SO	:	Serial Output
$\overline{\text{TC}}$	:	Terminal Count
TCLR	:	Timer Clear
TI	:	Timer Input
TO00, TO01	:	Timer Output
TXD	:	Transmit Data
$\overline{\text{UBE}}$	:	Upper Byte Enable
$\overline{\text{UCAS}}$	:	Upper Column Address Strobe
$\overline{\text{UMWR}}$	:	Upper Memory Write
WDTOUT	:	Watchdog Timer Output
$\overline{\text{WE}}$	:	Write Enable
X1, X2	:	Crystal Oscillator

★ INTERNAL BLOCK DIAGRAM



CONTENTS

1. PIN FUNCTIONS

1.1 Port Pins

Pin name	Input/output	Function	Dual-function pin
P00	Input/output	Port 0	TCLR
P01		10-bit input/output port	DREQ0
P02		Can be set for input/output bit.	$\overline{\text{DACK0}}$
P03			DREQ1
P04			$\overline{\text{DACK1}}$
P05			SI
P06			SO
P07			$\overline{\text{SCLK}}$
P08			TXD/$\overline{\text{UBE}}$
P09			RXD/$\overline{\text{TC}}$

Remark After a reset is released, each port pin is set as an input port pin.

1.2 Non-Port Pins

(1/2)

Pin name	Input/output	Function	Dual-function pin
A0-A23	Tristate output	Address bus signal	-
D0-D15	Tristate input/output	Bidirectional data bus signal	-
$\overline{\text{READY}}$	Input	Bus cycle termination permit signal	-
$\overline{\text{HLDRQ}}$	Input	Bus mastership request signal	-
$\overline{\text{HLDAK}}$	Output	Bus mastership permit signal	-
BLOCK	Output	Bus mastership prohibit signal	WDTOUT
$\overline{\text{MRD}}$	Tristate output	Read strobe signal to memory	-
$\overline{\text{LMWR}}$	Tristate output	Write strobe signal to lower data in memory	$\overline{\text{WE}}$
$\overline{\text{UMWR}}$	Tristate output	Write strobe signal to upper data in memory	-
$\overline{\text{IORD}}$	Tristate output	Read strobe signal to I/O data	-
$\overline{\text{IOWR}}$	Tristate output	Write strobe signal to I/O data	-
$\overline{\text{UBE}}$	Tristate output	Data bus upper data enable signal	TXD/P08
$\overline{\text{RESET}}$	Input	System reset input	-
X1, X2	Input	Crystal connection/external clock input	-

(2/2)

Pin name	Input/output	Function	Dual-function pin
CLKOUT	Output	System clock output	-
$\overline{\text{CS0}}$	Tristate output	Chip select signal	$\overline{\text{REFRQ}}$
$\overline{\text{CS1}}$			-
$\overline{\text{CS2}}$			-
$\overline{\text{CS3}}$			-
$\overline{\text{INTP00}}$	Input	Interrupt request input	TO00
$\overline{\text{INTP01}}$			-
$\overline{\text{INTP02}}$			TO01
$\overline{\text{INTP03}}$			-
$\overline{\text{INTP10}}$			-
$\overline{\text{INTP11}}$			-
$\overline{\text{INTP12}}$			-
$\overline{\text{INTP13}}$			TI
$\overline{\text{NMI}}$	Input	Nonmaskable interrupt request input	-
$\overline{\text{REFRQ}}$	Tristate output	Refresh request signal to DRAM	$\overline{\text{CS0}}$
$\overline{\text{RAS}}$	Tristate output	Row address strobe signal to DRAM	-
$\overline{\text{LCAS}}$	Tristate output	Column address strobe signal to lower data in DRAM	-
$\overline{\text{UCAS}}$	Tristate output	Column address strobe signal to upper data in DRAM	-
$\overline{\text{WE}}$	Tristate output	Write strobe signal to DRAM	$\overline{\text{LMWR}}$
DREQ0	Input	DMA request signal (channel 0)	P01
DREQ1	Input	DMA request signal (channel 1)	P03
$\overline{\text{DACK0}}$	Output	DMA permit signal (channel 0)	P02
$\overline{\text{DACK1}}$	Output	DMA permit signal (channel 1)	P04
$\overline{\text{TC}}$	Output	DMA end signal	RXD/P09
TO00	Output	RPU pulse output	$\overline{\text{INTP00}}$
TO01			$\overline{\text{INTP02}}$
TCLR	Input	External clear or start signal input to timer 0	P00
TI	Input	External count clock input to timer 0	$\overline{\text{INTP13}}$
TXD	Output	UART serial data output	$\overline{\text{UBE}}$/P08
RXD	Input	UART serial data input	$\overline{\text{TC}}$/P09
$\overline{\text{SCLK}}$	Input/output	CSI serial clock input/output	P07
SO	Output	CSI serial data output	P06
SI	Input	CSI serial data input	P05
WDTOUT	Output	WDT overflow signal	BLOCK
IC	-	Internal connection (must be connected to GND through a resistor)	-
V_{DD}	-	Supplies positive power.	-
GND	-	Ground potential	-

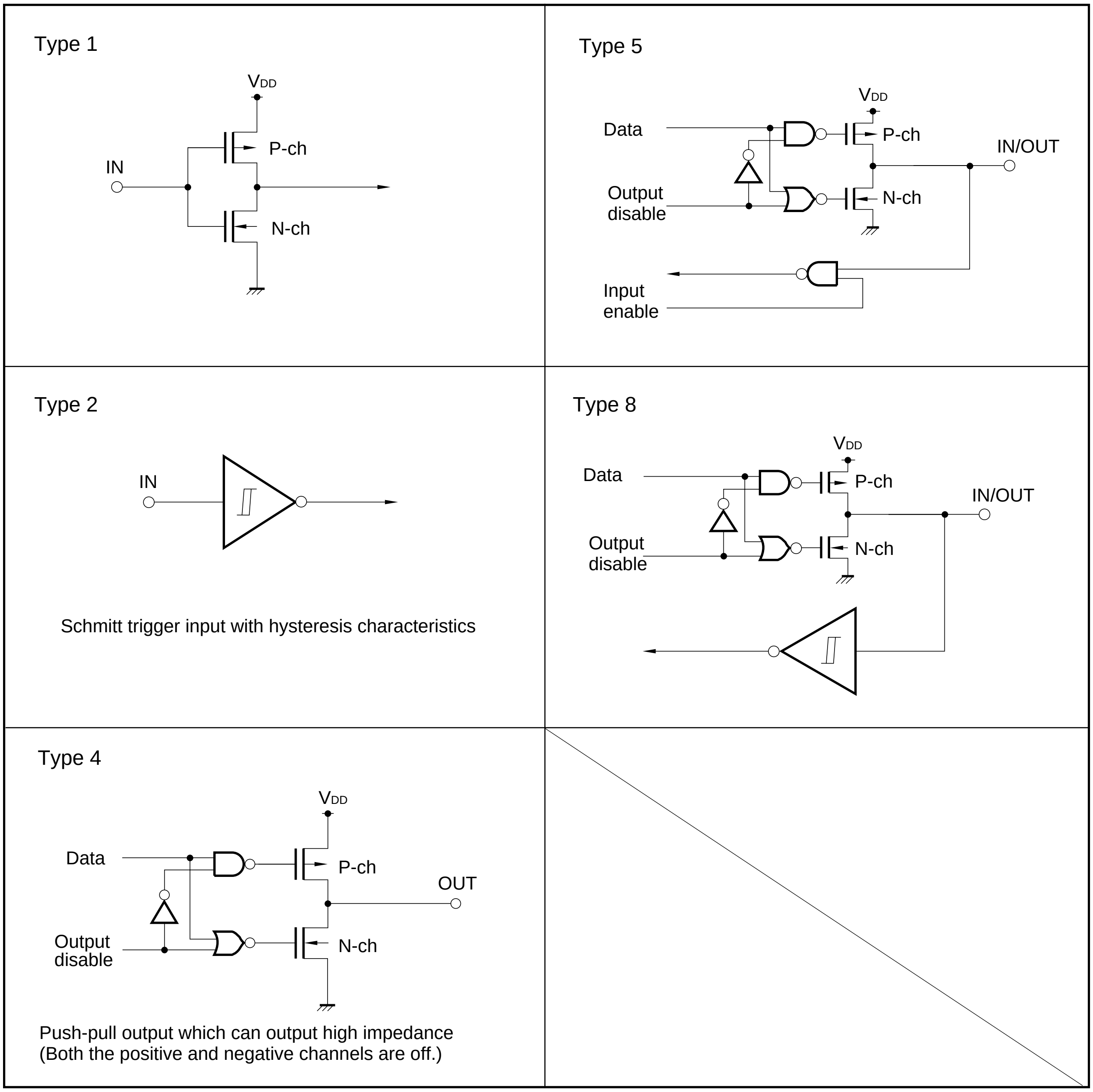
1.3 Pin I/O Circuits and Processing of Unused Pins

Table 1-1 shows the I/O circuit type of each pin and the processing for unused pins. Figure 1-1 shows the I/O circuit of each type.

Table 1-1. I/O Circuits Type of Each Pin and Recommended Connection of Unused Pins

Pin	I/O circuit type	Recommended connection
P00/TCLR	5	Input status: Individually connected to V_{DD} or GND through a resistor. Output status: Open
P01/DREQ0		
P02/$\overline{\text{DACK0}}$		
P03/DREQ1		
P04/$\overline{\text{DACK1}}$		
P05/SI		
P06/SO		
P07/$\overline{\text{SCLK}}$		
P08/TXD/$\overline{\text{UBE}}$		
P09/RXD/$\overline{\text{TC}}$		
D0-D15	5	Open
A0-A7, A16-A18		
A8-A15, A19-A23	4	Open
$\overline{\text{READY}}$	1	Connected to GND through a resistor.
$\overline{\text{HLDRQ}}$		Connected to V_{DD} through a resistor.
$\overline{\text{HLDAK}}$	4	Open
BLOCK/WDTOUT		
$\overline{\text{MRD}}$		
$\overline{\text{LMWR}}$/$\overline{\text{WE}}$		
$\overline{\text{UMWR}}$		
$\overline{\text{IORD}}$		
$\overline{\text{IOWR}}$		
CLKOUT		
$\overline{\text{CS0}}$/$\overline{\text{REFRQ}}$		
$\overline{\text{CS1}}$-$\overline{\text{CS3}}$		
$\overline{\text{INTP00}}$/TO00	8	Connected to V_{DD} through a resistor.
$\overline{\text{INTP01}}$	2	Connected to V_{DD} through a resistor.
$\overline{\text{INTP02}}$/TO01	8	Connected to V_{DD} through a resistor.
$\overline{\text{INTP03}}$	2	Connected to V_{DD} through a resistor.
$\overline{\text{INTP10}}$-$\overline{\text{INTP12}}$		
$\overline{\text{INTP13}}$/TI		
$\overline{\text{NMI}}$		
$\overline{\text{RESET}}$		
$\overline{\text{RAS}}$	4	Open
$\overline{\text{LCAS}}$		
$\overline{\text{UCAS}}$		
X2	-	
IC	-	Connected to GND through a resistor.

Figure 1-1. Pin I/O Circuits



Schmitt trigger input with hysteresis characteristics

Push-pull output which can output high impedance
(Both the positive and negative channels are off.)

2. INTERNAL UNITS

2.1 Bus Interface Unit (BIU)

Controls the pins of the address bus, data bus, and control bus. A bus cycle activated by the CPU or DMAC is controlled via the WCU, DRAMC, and ROMC.

2.2 Wait Control Unit (WCU)

Manages the four blocks corresponding to four chip select signals ($\overline{\text{CS0}}$-$\overline{\text{CS3}}$).

This block generates chip select signals, performs wait control, and selects a bus cycle type.

2.3 DRAM Controller (DRAMC)

Generates the $\overline{\text{RAS}}$, $\overline{\text{UCAS}}$, and $\overline{\text{LCAS}}$ signals (2CAS control) and controls access to DRAM.

This block supports DRAM high-speed page mode. Access to DRAM can be of either of two types, each having a different cycle, normal access (off-page) or high-speed page access (on-page).

2.4 ROM Controller (ROMC)

Supports access to ROM supporting a page access function.

Performs address comparison relative to the previous bus cycle and performs wait control for normal access (off-page)/page access (on-page). It supports page widths of 8-64 bytes.

2.5 Interrupt Controller

Handles maskable interrupt requests ($\overline{\text{INTP00}}$-$\overline{\text{INTP03}}$, $\overline{\text{INTP10}}$-$\overline{\text{INTP13}}$) from both the built-in and external peripheral hardware. Priorities can be specified for these interrupt requests, in units of four groups. It can apply multiple handling control to the interrupt sources.

2.6 DMA Controller (DMAC)

Transfers data between memory and I/O, as instructed by the CPU.

There are two address modes, fly-by (1-cycle) transfer and 2-cycle transfer. There are three bus modes, single transfer, single-step transfer, and block transfer.

2.7 Serial Interfaces (UART/CSI)

As serial interfaces, the V821 features an asynchronous serial interface (UART) and a synchronous serial interface (CSI), one channel being assigned to each.

The UART transfers data via pins TXD and RXD.

The CSI transfers data via pins SO, SI, and $\overline{\text{SCLK}}$.

Either the baud rate generator or the system clock can be selected as the serial clock source.

2.8 Real-Time Pulse Unit (RPU)

This block incorporates a 16-bit timer/event counter and a 16-bit interval timer. It can calculate pulse intervals and frequencies and output programmable pulses.

2.9 Watchdog Timer (WDT)

This block incorporates an 8-bit watchdog timer to detect a program hanging up or system errors. If the watchdog timer overflows, the WDTOUT pin becomes active.

2.10 Clock Generator (CG)

Supplies clock pulses at a frequency five times greater than that of the oscillator connected to pins X1 and X2 (when the built-in PLL is being used) or at half the frequency (when the built-in PLL is not being used) of the operating clock pulses for the CPU. Also, instead of connecting an oscillator, external clock pulses can be input.

2.11 Bus Arbitration Unit (BAU)

Arbitrates any contention over bus mastership between the bus masters (CPU, DRAMC, DMAC, external bus master). Bus mastership can be switched in each bus cycle and also in the idle state.

2.12 Port

Port 0 provides a total of ten input/output port pins. The pins can be used as either port or control pins.

3. CPU FUNCTIONS

The CPU has functions equivalent to those of the V810 microprocessor, designed for built-in control. It offers bit string instructions, floating-point instructions, and quick real-time response.

3.1 Features

The features of the CPU are:

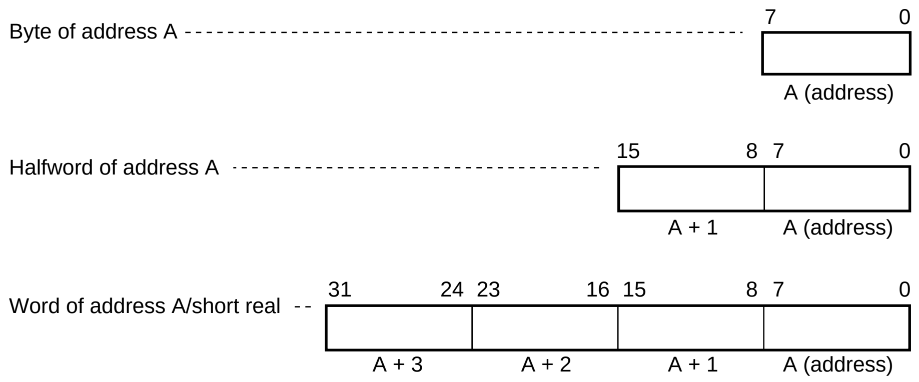
- High-performance 32-bit RISC microprocessor
  - Built-in 1-Kbyte cache memory
  - Pipeline structure of 1-clock pitch
  - 16-bit data bus
  - 32-bit general-purpose registers: 32
  - 4-Gbyte linear address space
- Instructions ideal for various application fields
  - Floating-point operation instructions (conforming to the IEEE754 data format)
  - Bit string instructions
- High-speed interrupt response
- Debug support functions

3.2 Address Space

The V821 supports internal memory and I/O spaces of 4G bytes each. The V821 outputs 24-bit addresses to memory and I/O, such that the addresses range from 0 to 2^{24} - 1.

In byte data, bit 0 is defined as the LSB (Least Significant Bit) and bit 7 as the MSB (Most Significant Bit). In multiple-byte data, bit 0 of the byte data in the lower address is defined as the LSB and bit 7 of the byte data in the upper address as the MSB, unless noted otherwise.

In the case of the V821, 2-byte data is referred to as halfword data, and 4-byte data as word data. In this data sheet, in representations of multiple-byte memory and I/O data, the right address corresponds to the lower address and the left address to the upper address, as shown below.



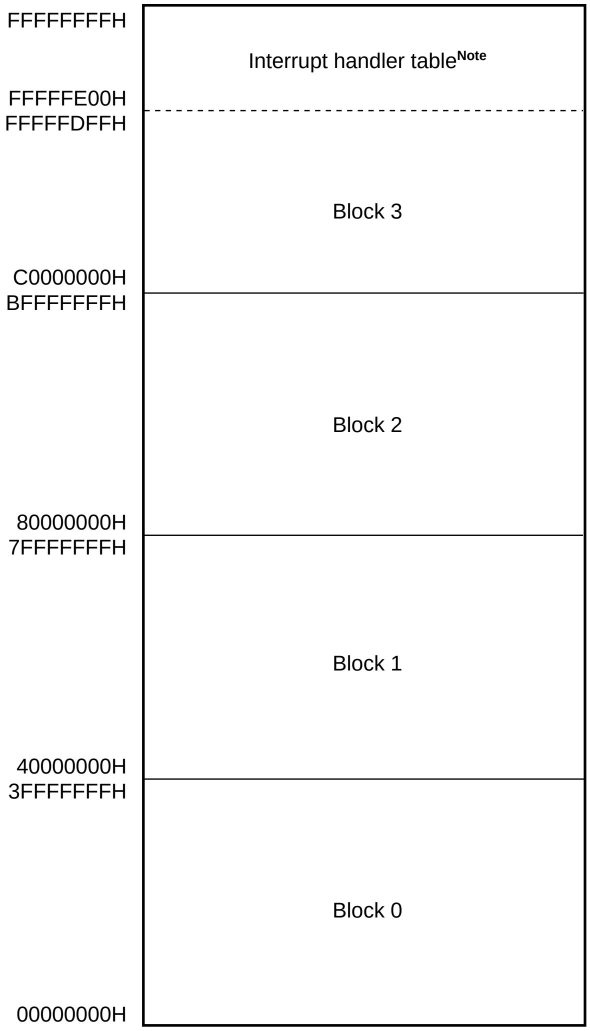
3.2.1 Memory map

Figure 3-1 shows the memory map of the V821.

The internal 4-Gbyte memory space is divided into blocks of 1G byte each.

Each block has a linear address space of 16M bytes. (The lower 24 bits of a 32-bit address are output.)

Figure 3-1. Memory Map



Note See **Table 4-1** for details.

3.2.2 I/O map

Figure 3-2 shows the I/O map of the V821.

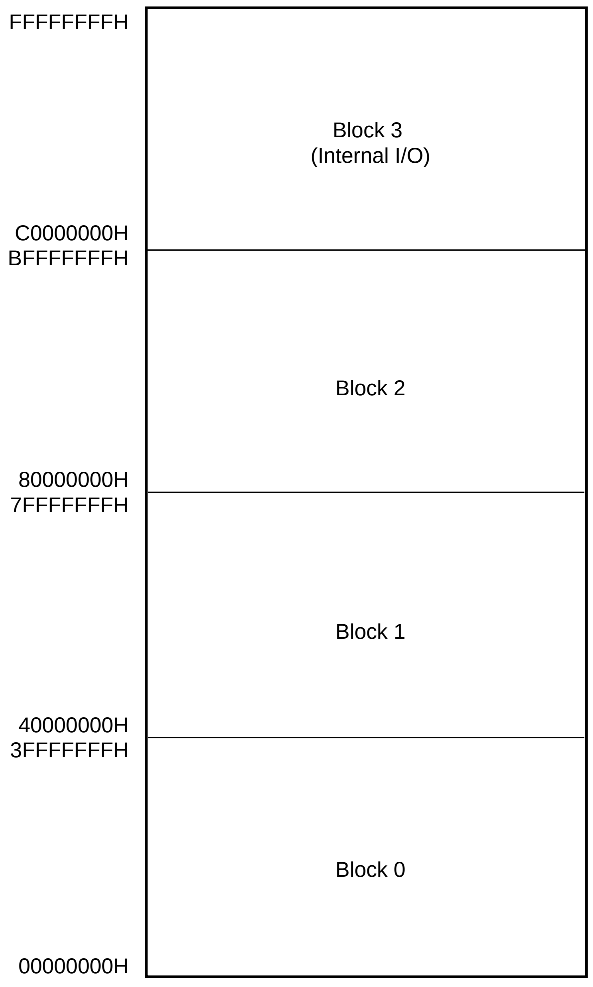
The internal 4-Gbyte memory space is divided into blocks of 1G byte each.

Each block has a linear address space of 16M bytes. (The lower 24 bits of a 32-bit address are output.)

The V821 reserves I/O addresses C0000000H-FFFFFFFFH (I/O block 3) as an internal I/O space. Each unit is mapped to this internal I/O space.

See **Section 3.4** for details of the configuration of the internal I/O space.

Figure 3-2. I/O Map



3.3 CPU Register Set

The registers of the V821 belong to one of two sets, the general-purpose program register set and the dedicated system register set. All registers are 32 bits in wide.

Program register set

31	0
r0	Zero Register
r1	Reserved for Address Generation
r2	Handler Stack Pointer (hp)
r3	Stack Pointer (sp)
r4	Global Pointer (gp)
r5	Text Pointer (tp)
r6	
r7	
r8	
r9	
r10	
r11	
r12	
r13	
r14	
r15	
r16	
r17	
r18	
r19	
r20	
r21	
r22	
r23	
r24	
r25	
r26	String Destination Bit Offset
r27	String Source Bit Offset
r28	String Length
r29	String Destination
r30	String Source
r31	Link Pointer (lp)

31	0
PC	Program Counter

System register set

31	0
EIPC	Exception/Interrupt PC
EIPSW	Exception/Interrupt PSW

31	0
FEPC	Fatal Error PC
FEPSW	Fatal Error PSW

31	0
ECR	Exception Cause Register

31	0
PSW	Program Status Word

31	0
PIR	Processor ID Register

31	0
TKCW	Task Control Word

31	0
CHCW	Cache Control Word

31	0
ADTRE	Address Trap Register

3.3.1 Program register set

The program register set includes general-purpose registers and a program counter.

(1) General-purpose registers

The V821 has 32 general-purpose registers, r0-r31. These registers can be used for data or address variables. Registers r0 and r26-r30 are used implicitly with instructions. Caution is therefore necessary when using these registers. Registers r1-r5 and r31 are used implicitly by the assembler and the C compiler. Before using these registers, therefore, the contents of the registers must be saved to prevent their being destroyed. After using the registers, their contents must be restored.

Table 3-1. Program Registers

Name	Use	Explanation
r0	Zero register	Always stores zeros.
r1	Assembler-reserved register	Used as a working register to create 32-bit immediate.
r2	Handler stack pointer	Used as a stack pointer for the handler.
r3	Stack pointer	Used to create a stack frame at a function call.
r4	Global pointer	Used to access a global variable in a data area.
r5	Text pointer	Points to the top of a text area
r6-r25	-	Register for an address/data variable
r26	String destination start bit offset	Used to execute a bit string instruction.
r27	String source start bit offset	
r28	String length register	
r29	String destination start address register	
r30	String start address register	
r31	Link pointer	Stores a return point address according to the execution of a JAL instruction.

(2) Program counter

Stores the address of an instruction while a program is running. Bit 0 of the program counter (PC) is fixed to 0, thus preventing a branch to an odd address. It is initialized to FFFFFFF0H at reset.

3.3.2 System register set

System registers are used to control the state of the CPU and store interrupt information.

Table 3-2. System Register Numbers

No.	Register name	Use	Explanation
0	EIPC	Registers for saving the current status upon the occurrence of an exception or interrupt	Retain the contents of PC and PSW if an exception or interrupt occurs. Note, however, that there is only one pair of these registers. When multiple interrupts are allowed, therefore, the contents of the registers must be saved by the program.
1	EIPSW		
2	FEPC	Registers for saving the current status upon the occurrence of an NMI or double exception	Retain the contents of PC and PSW if an NMI or double exception occurs.
3	FEPSW		
4	ECR	Exception source register	Stores the source of an exception, maskable interrupt, or NMI. The upper 16 bits of this register are called "FECC" and set to the exception code of an NMI/double exception. The lower 16 bits are called "EICC" and set to the exception code of an exception/interrupt.
5	PSW	Program status word	The program status word is a set of flags indicating the state of the program (result of executing an instruction) and the state of the CPU.
6	PIR	Processor ID register	Used to identify a CPU type number.
7	TKCW	Task control word	Used to control a floating-point operation.
8-23	Reserved		
24	CHCW	Cache control word	Used to control the built-in instruction cache.
25	ADTRE	Address trap register	Stores the address used to detect an address match with PC, and to generate an address trap.
26-31	Reserved		

Read and write operations made to these system registers can be performed using the system register load/store instructions (LDSR and STSR) with the system register numbers specified.

3.4 Built-in Peripheral I/O Registers

The built-in peripheral I/O registers are allocated to the 256-byte area between C0000000H and C00000FFH in the 1-Gbyte space between C0000000H and FFFFFFFFH. Starting from address C0000100H, 256-byte images are created every 256 bytes.

The least significant bit of an address is not decoded. Thus, when byte access is attempted to a register at an odd address (2n+1), a register at an even address (2n) is actually performed.

When 16-bit access is attempted to an 8-bit I/O register, the upper eight bits are ignored for write, and become undefined for read.

Table 3-3 lists the built-in peripheral I/O registers.

Table 3-3. Built-in Peripheral I/O Registers (1/2)

Address	Function register name	Abbreviation	Manipulatable bits		Initial value
			8-bits	16-bits	
C0000010	Port mode control register 0	PMC0		o	0000H
C0000012	Port mode register 0	PM0		o	03FFH
C0000014	Port register 0	P0		o	Not defined
C0000020	Bus cycle type control register	BCTC	o		01H
C0000022	Programmable wait control register 0	PWC0	o		77H
C0000024	Programmable wait control register 1	PWC1	o		77H
C0000026	Programmable wait control register 2	PWC2	o		77H
C0000028	DRAM configuration register	DRC	o		81H
C000002A	Refresh control register	RFC	o		80H
C000002C	Page-ROM configuration register	PRC	o		80H
C0000040	DMA source address register 0H	DSA0H		o	Not defined
C0000042	DMA source address register 0L	DSA0L		o	Not defined
C0000044	DMA destination address register 0H	DDA0H		o	Not defined
C0000046	DMA destination address register 0L	DDA0L		o	Not defined
C0000048	DMA source address register 1H	DSA1H		o	Not defined
C000004A	DMA source address register 1L	DSA1L		o	Not defined
C000004C	DMA destination address register 1H	DDA1H		o	Not defined
C000004E	DMA destination address register 1L	DDA1L		o	Not defined
C0000050	DMA byte count register 0	DBC0		o	Not defined
C0000052	DMA byte count register 1	DBC1		o	Not defined
C0000054	DMA channel control register 0	DCHC0		o	0000H
C0000056	DMA channel control register 1	DCHC1		o	0000H
C0000060	Timer unit mode register 0	TUM0		o	0A00H
C0000062	Timer control register 0	TMC0	o		00H
C0000064	Timer control register 1	TMC1	o		00H
C0000066	Timer output control register 0	TOC0	o		03H
C0000068	Timer overflow status register	TOVS	o		00H
C0000070	Timer register 0	TM0		o	0000H
C0000072	Capture/compare register 00	CC00		o	Not defined
C0000074	Capture/compare register 01	CC01		o	Not defined
C0000076	Capture/compare register 02	CC02		o	Not defined
C0000078	Capture/compare register 03	CC03		o	Not defined
C000007C	Timer register 1	TM1		o	0000H
C000007E	Compare register 1	CM1		o	Not defined
C0000080	Asynchronous serial interface mode register	ASIM	o		00H
C0000082	Asynchronous serial interface status register	ASIS	o		00H

Table 3-3. Built-in Peripheral I/O Registers (2/2)

Address	Function register name	Abbreviation	Manipulatable bits		Initial value
			8-bits	16-bits	
C0000084	Reception buffer	RXB		o	Not defined
C0000086	Reception buffer L	RXBL	o		Not defined
C0000088	Transmission shift register	TXS		o	Not defined
C000008A	Transmission shift register L	TXSL	o		Not defined
C0000090	Synchronous serial interface mode register	CSIM	o		00H
C0000092	Serial I/O shift register	SIO	o		Not defined
C00000A0	Baud rate generator register	BRG	o		Not defined
C00000A2	Baud rate generator prescale mode register	BPRM	o		00H
C00000B0	Interrupt group priority register	IGP	o		E4H
C00000B2	Interrupt clear register	ICR		o	0000H
C00000B4	Interrupt request register	IRR		o	0000H
C00000B6	Interrupt request mask register	IMR		o	FFFFH
C00000B8	ICU mode register	IMOD		o	AAAAH
C00000C0	WDT mode register	WDTM	o		00H
C00000D0	Standby control register	STBC	o		00H
C00000E0	Clock control register	CGC	o		03H

3.5 Data Types

3.5.1 Data types

The data types supported by the V821 are as follows:

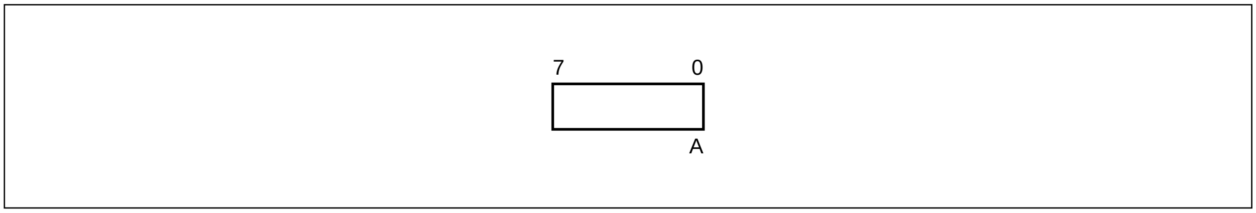
- Integer (8, 16, 32 bits)
- Unsigned integer (8, 16, 32 bits)
- Bit string
- Single-precision floating-point data (32 bits)

(1) Data type and addressing

The V821 uses the little-endian data addressing. In this addressing, if fixed-length data is located in a memory area, the data must be either of the data types shown below.

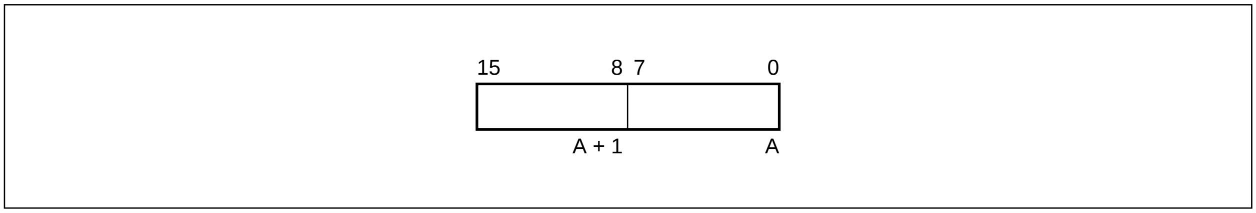
(a) Byte

A byte is consecutive 8-bit data whose first-bit address is aligned to a byte boundary. Each bit in a byte is numbered from 0 to 7: LSB (the least significant bit) is bit 0 and MSB (the most significant bit) is bit 7. To access a byte, specify address A. (See diagram below.)



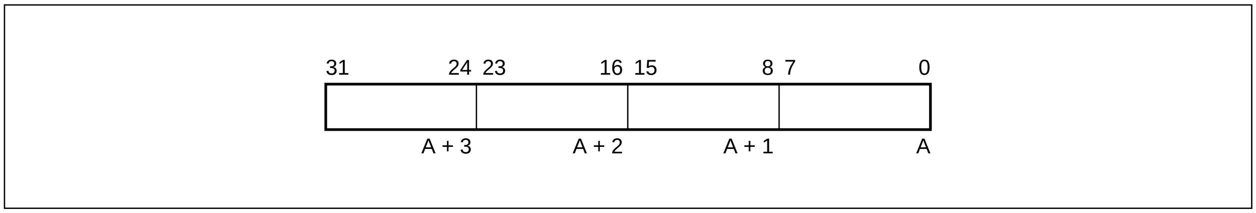
(b) Halfword

A halfword is consecutive 16-bit (= 2 bytes) data whose first-bit address is aligned to a halfword boundary. Each bit in a halfword is numbered from 0 to 15: LSB (the least significant bit) is bit 0 and MSB (the most significant bit) is bit 15. To access a halfword, specify the address A only (lowest bit must be 0).



(c) Word/short real

A word, also called short real, is consecutive 32-bit (= 4 bytes) data whose first-bit address is aligned to a word boundary. Each bit in a word is numbered from 0 to 31: LSB (the least significant bit) is bit 0 and MSB (the most significant bit) is bit 31. To access a word or short real, specify the address A only (lower two bits must be 0).



(2) Integer

In the V821, all integers are expressed in the two's-complement binary notation, and are composed of either 8 bits, 16 bits, or 32 bits. Regardless of the data length, bit 0 is the least significant bit, and higher-numbered bits express higher digits of the integer with the highest bit expressing its sign.

Data length		Range
Byte	8 bits	-128 to +127
Halfword	16 bits	-32 768 to +32 767
Word	32 bits	-2 147 483 648 to +2 147 483 647

(3) Unsigned integer

An unsigned integer is either zero or a positive integer unlike the integer explained in (2) which can be negative as well as zero and positive. Unsigned integers are expressed in the binary notation in the same way as integers, and are either 8 bits, 16 bits, or 32 bits long. Regardless of the data length, the bit assignments are the same as in the case of integers except that unsigned integers do not include a sign bit; the highest bit is also a part of the integer.

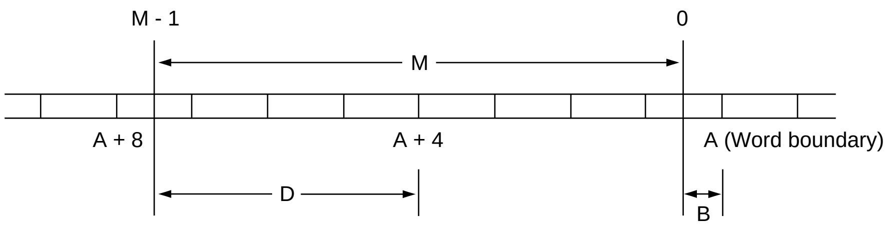
Data length		Range
Byte	8 bits	0 to 255
Halfword	16 bits	0 to 65 535
Word	32 bits	0 to 4 294 967 295

(4) Bit string

A bit string is a type of data whose bit length is variable from 0 to 2^{32} - 1. To specify a bit-string data, define the following three attributes.

- A : address of the string data's first word (lower two bits must be 0.)
- B : in-word bit offset in the string data (0 to 31)
- M : bit length of the string data (0 to 2^{32} - 1)

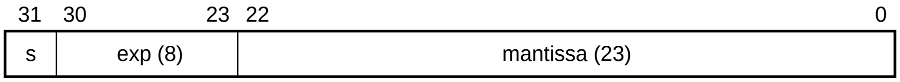
The above three attributes may vary depending on the bit-string data manipulation direction: upward or downward, as shown below. The former is the direction from lower addresses to higher addresses while the latter is the direction from higher to lower addresses.



Attribute	Upward	Downward
First-word address (0s in bits 1 and 0)	A	A + 4
In-word bit offset (0 to 31)	B	D
Bit length (0 to 2^{32} - 1)	M	M

(5) Single-precision floating-point data

This data type is 32 bits long and its bit allocation complies with the IEEE single format. A single-precision floating-point data consists of 1-bit mantissa sign bit, 8-bit exponent, and 23-bit mantissa. The exponent is offset-expressed from the bias value - 127, and the mantissa is binary-expressed with the integer part omitted.



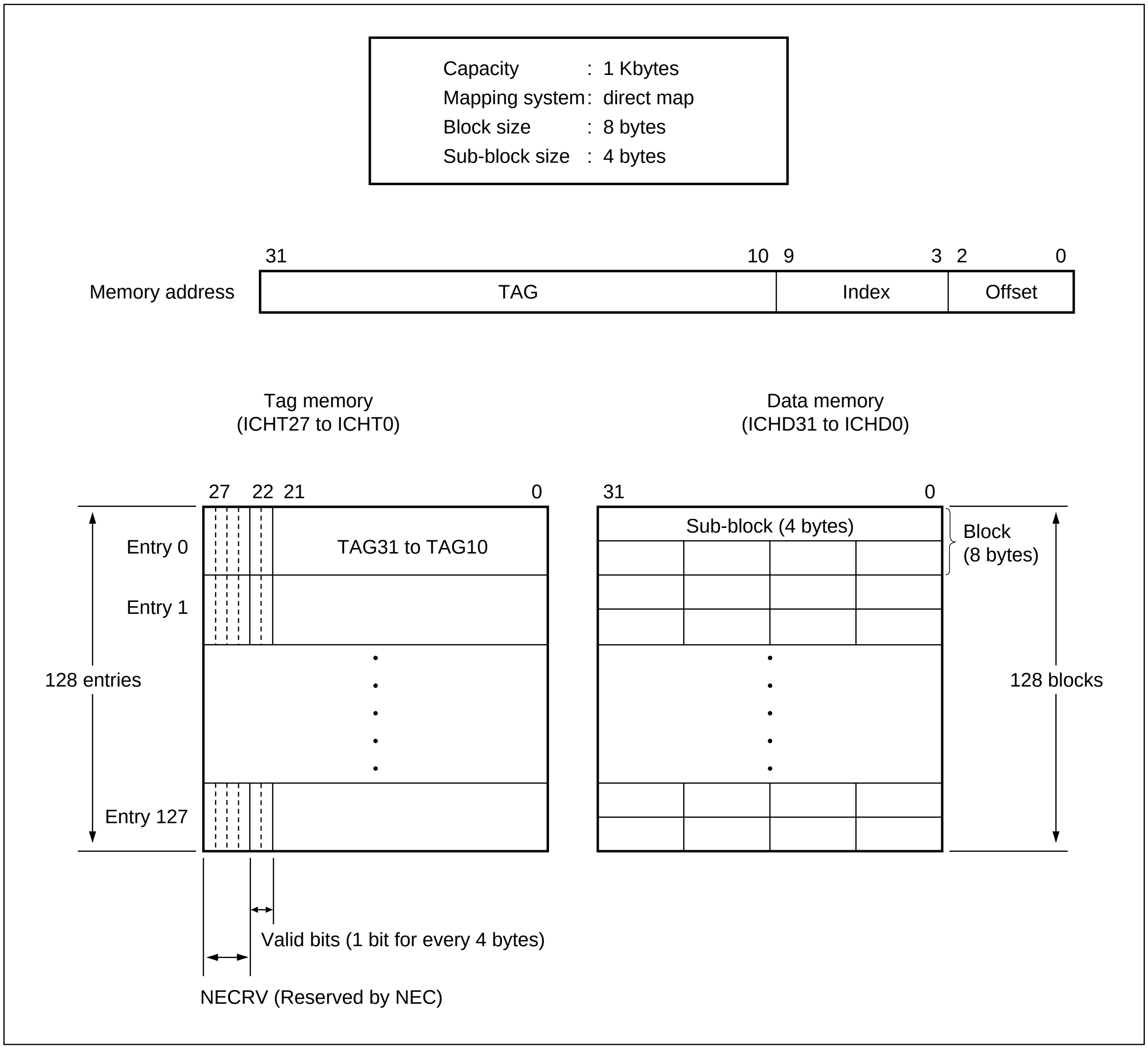
3.5.2 Data alignment

In the V821, a word data must be aligned to a word boundary (with the lowest two bits of the address fixed to 0s), and a halfword data to a halfword boundary (with the lowest bit of the address fixed to 0). If a data is not aligned as specified, the lowest two bits (in the case of word) or one bit (in the case of halfword) of its address will forcibly be masked with 0s when the data is accessed.

3.6 Cache

Figure 3-3 shows the instruction cache configuration provided to the V821.

Figure 3-3. Cache Configuration



Capacity	: 1 Kbytes
Mapping system	: direct map
Block size	: 8 bytes
Sub-block size	: 4 bytes

4. INTERRUPT/EXCEPTION HANDLING FUNCTIONS

The V821 features an interrupt controller (ICU) that is dedicated to interrupt handling. The V821 thus supports a powerful interrupt handling function capable of handling interrupt requests issued by up to 16 sources.

As referred to in this manual, an interrupt is an event which occurs independently of program execution while an exception is an event that depends on program execution. In general, an exception assumes a higher priority than an interrupt.

The V821 can handle interrupt requests issued by both built-in peripheral hardware and external devices. Exception handling can be triggered by executing an instruction (TRAP instruction) as well as by the occurrence of an exception (such as an address trap or invalid instruction code).

4.1 Features

○ Interrupts

- Nonmaskable interrupt : 1 source
- Maskable interrupt : 15 sources
- Programmable priority control with four groups
- Multiple interrupt handling control according to priority
- Mask specification for each maskable interrupt request
- Valid edge specification for external interrupt requests
- The noise eliminator introducing an analog delay (60 to 300 ns) is incorporated into the nonmaskable interrupt ($\overline{\text{NMI}}$) pin.

○ Exceptions

- Software exception : 32 sources
- Exception trap : 10 sources

Table 4-1 lists the interrupt and exception sources.

Table 4-1. Interrupts (1/2)

Type	Category	Group	Priority	Interrupt/exception source			Exception	Handler	Return
			in group	Name	Source	Unit	code	address	PC[Note 1]
Reset	Interrupt	-	-	RESET	Reset input	-	FFF0H	FFFFFFF0H	Undefined
Non-maskable	Interrupt	-	-	NMI	$\overline{\text{NMI}}$ input	-	FFD0H	FFFFFFD0H	Next PC[Note 2]
Software exception	Exception	-	-	TRAP1nH	trap instruction	-	FFBnH	FFFFFFB0H	Next PC
		-	-	TRAP0nH	trap instruction	-	FFAnH	FFFFFFA0H	Next PC
Exception trap	Exception	-	-	DP-EX	Double exception	-	Note 3	FFFFFFD0H	Current PC
		-	-	AD-TR	Address trap	-	FFC0H	FFFFFFC0H	
		-	-	I-OPC	Invalid instruction code	-	FF90H	FFFFFF90H	
		-	-	DIV0	Division by zero	-	FF80H	FFFFFF80H	
		-	-	FIZ	Invalid floating-point operation	-	FF70H	FFFFFF60H	
		-	-	FZD	Floating-point division by zero	-	FF68H	FFFFFF60H	
		-	-	FOV	Floating-point overflow	-	FF64H	FFFFFF60H	
		-	-	FUD	Floating-point underflow[Note 4]	-	FF62H	FFFFFF60H	
		-	-	FPR	Floating-point degraded precision[Note 4]	-	FF61H	FFFFFF60H	
		-	-	FRO	Floating-point reserved operand	-	FF60H	FFFFFF60H	

Remark n = 0H to FH

Notes 1. PC value saved in EIPC or FEPC at the beginning of interrupt/exception handling

2. Return PC = current PC if an interrupt occurred during the execution of an instruction which was stopped by an interrupt (DIV/DIVU, floating-point, and bit string instructions).

3. The exception code for the exception which occurred first is written into in the 16 low-order bits of ECR, while and that for the second exception is written into the 16 high-order bits.

4. The V821 is not subject to floating-point underflow or degraded precision exceptions.

Table 4-1. Interrupts (2/2)

Type	Category	Group	Priority	Interrupt/exception source			Exception	Handler	Return
			in group	Name	Source	Unit	code	address	PC[Note 1]
Maskable	Interrupt	GR3	3	RESERVED	Reserved	-	FEF0H	FFFFFEF0H	Next PC[Note 2]
			2	INTOV0	Timer 0 overflow	RPU	FEE0H	FFFFFEE0H	
			1	INTSER	UART reception error	UART	FED0H	FFFFFED0H	
			0	INTP13	$\overline{\text{INTP13}}$ pin input	External	FEC0H	FFFFFEC0H	
		GR2	3	INTSR	UART reception end	UART	FEB0H	FFFFFEB0H	
			2	INTST	UART transmission end	UART	FEA0H	FFFFFEA0H	
			1	INTCSI	CSI transmission/reception end	CSI	FE90H	FFFFFE90H	
			0	INTP12	$\overline{\text{INTP12}}$ pin input	External	FE80H	FFFFFE80H	
		GR1	3	INTDMA	DMA transfer end	DMAC	FE70H	FFFFFE70H	
			2	INTP00/ INTCC00	$\overline{\text{INTP00}}$ pin input/CC00 match	External/ RPU	FE60H	FFFFFE60H	
			1	INTP01/ INTCC01	$\overline{\text{INTP01}}$ pin input/CC01 match	External/ RPU	FE50H	FFFFFE50H	
			0	INTP11	$\overline{\text{INTP11}}$ pin input	External	FE40H	FFFFFE40H	
		GR0	3	INTCM1	CM1 match	RPU	FE30H	FFFFFE30H	
			2	INTP02/ INTCC02	$\overline{\text{INTP02}}$ pin input/CC02 match	External/ RPU	FE20H	FFFFFE20H	
			1	INTP03/ INTCC03	$\overline{\text{INTP03}}$ pin input/CC03 match	External/ RPU	FE10H	FFFFFE10H	
			0	INTP10	$\overline{\text{INTP10}}$ pin input	External	FE00H	FFFFFE00H	

Notes 1. PC value saved in EIPC or FEPC at the beginning of interrupt/exception handling

2. Return PC = current PC if an interrupt occurred during the execution of an instruction which was stopped by an interrupt (DIV/DIVU, floating-point, and bit string instructions).

Caution The exception code and handler address for a maskable interrupt assume the values existing when the default priority is specified.

5. WAIT CONTROL FUNCTIONS

The wait control unit (WCU) manages the four blocks corresponding to the four chip select signals, generates the chip select signals, performs wait control, and selects the bus cycle types.

5.1 Features

- Able to control up to four blocks in the memory and I/O spaces
- Linear address space of each block: 16 Mbytes
- Wait control
  - Automatic insertion of 0-7 waits per block
  - Insertion of waits using the $\overline{\text{READY}}$ pin
- Bus cycle selection function
  - Page-ROM cycle selectable (address block 3)
  - DRAM cycle selectable (address block 0)

Figure 5-1. Memory and I/O Maps

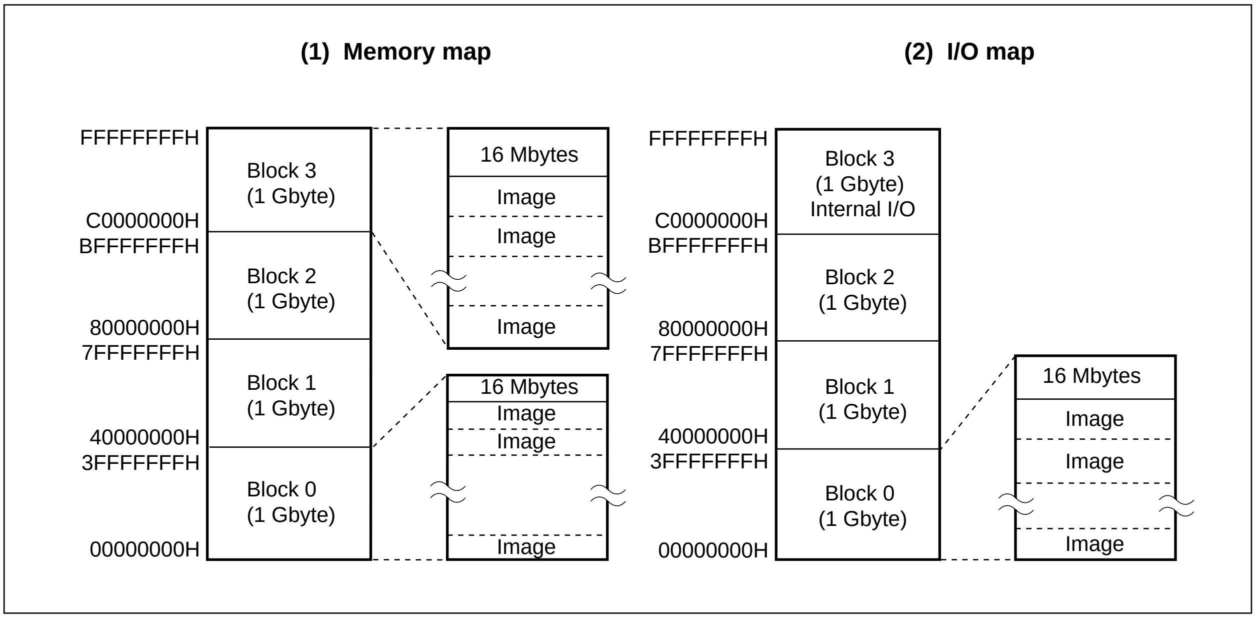


Table 5-1. Bus Cycles during Which the Wait Function Is Effective

Bus cycle		Programmable wait	Wait with the $\overline{\text{READY}}$ pin
SRAM (ROM) cycle (Blocks 0-3)		0-7 waits	o
DRAM cycle (Block 0)	off-page	2 or 3 waits	o
	on-page	0 or 1 wait	×
Page-ROM cycle (Block 3)	off-page	0-7 waits	×
	on-page	0 or 1 wait	×
External I/O cycle (Blocks 0-2)		0-7 waits	o
Internal I/O cycle (Block 3)		1 or 2 waits	×
CBR refresh cycle		Fixed (3 waits)	o
CBR self-refresh cycle		-	×
Fly-by DMA transfer			
SRAM (ROM) cycle (Blocks 0-3)		0-7 waits	o
DRAM cycle (Block 0)	off-page	2-7 waits	o
	on-page	0-7 waits	×
Page-ROM cycle (Block 3)	off-page	0-7 waits	×
	on-page	0-7 waits	×
Halt acknowledge cycle		Fixed (0 wait)	×
Machine fault cycle (I/O block 0 write)		0-7 wait	o

Remark o: Effective
×: Not effective

6. MEMORY ACCESS CONTROL FUNCTIONS

6.1 DRAM Controller (DRAMC)

The DRAM controller (DRAMC) generates the $\overline{\text{REFRQ}}$, $\overline{\text{RAS}}$, $\overline{\text{LCAS}}$, and $\overline{\text{UCAS}}$ signals, and controls access to DRAM. Access to DRAM is achieved by multiplexing the DRAM row and column addresses and outputting them from the address pins.

The microprocessor assumes the connected DRAM to be of × 4 bits or more, and that it supports high-speed page mode. There are two types of DRAM access cycles, on-page (2 or 3 clock pulses) and off-page (4 or 5 pulses).

Refresh uses the $\overline{\text{CAS}}$ before $\overline{\text{RAS}}$ method, allowing the user to set any refresh period. In IDLE and STOP modes, CBR self-refresh is performed.

6.1.1 Features

- Generates the $\overline{\text{REFRQ}}$, $\overline{\text{RAS}}$, $\overline{\text{LCAS}}$, and $\overline{\text{UCAS}}$ signals.
- Supports DRAM high-speed page mode.
- Address multiplexing function: 8, 9, 10, and 11 bits
- CBR refresh and CBR self-refresh functions

6.1.2 Address multiplexing function

In the DRAM cycle, row and column addresses are multiplexed according to the value of the DAW bits of the DRAM configuration register (DRC), then output, as shown in Figure 6-1. In Figure 6-1, a0-a23 are the addresses output from the CPU, while A0-A23 are the address pins of the V821. For example, if DAW = 11, row address a12-a22 and column address a1-a11 are output from address pins (A1-A11).

Table 6-1 lists the relationship between the connectable DRAMs and address multiplexing widths. Depending on the connected DRAM, the DRAM space can be between 128 Kbytes and 8 Mbytes.

Figure 6-1. Output of Row and Column Addresses

Address pin	A23 → A16	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
Row address																	
DAW = 11	a23 → a16	a15	a14	a13	a23	a22	a21	a20	a19	a18	a17	a16	a15	a14	a13	a12	a11
DAW = 10	a23 → a16	a15	a14	a23	a22	a21	a20	a19	a18	a17	a16	a15	a14	a13	a12	a11	a10
DAW = 01	a23 → a16	a15	a23	a22	a21	a20	a19	a18	a17	a16	a15	a14	a13	a12	a11	a10	a9
DAW = 00	a23 → a16	a23	a22	a21	a20	a19	a18	a17	a16	a15	a14	a13	a12	a11	a10	a9	a8
Column address	a23 → a0																

Table 6-1. Examples of DRAM and Address Multiplexing Width

Address multiplexing width	DRAM capacity (in bits) and configuration				DRAM space (in bytes)
	256 K	1 M	4 M	16 M	
8 bits	64 K × 4	-	-	-	128 K
9 bits	-	256 K × 4	256 K × 16	-	512 K
	-	-	512 K × 8	-	1 M
10 bits	-	-	1 M × 4	1 M × 16	2 M
	-	-	-	2 M × 8	4 M
11 bits	-	-	-	4 M × 4	8 M

6.1.3 Refresh function

DRAMC can automatically generate the distributed CBR refresh cycle needed to refresh external DRAM. Whether refresh should be enabled or disabled, and the refresh interval, are specified using the refresh control register (RFC).

While another bus master is occupying a bus, DRAMC cannot forcibly acquire the bus. In this case, in response to a refresh request issued from DRAMC, BAU makes the $\overline{\text{HLDAK}}$ pin inactive to post notification of the occurrence of a refresh request. In this state, by making the $\overline{\text{HLDRQ}}$ pin inactive, the refresh cycle is activated.

6.1.4 Self-refresh function

DRAMC generates the CBR self-refresh cycle in IDLE and STOP modes. The self-refresh cycle is activated by setting the SMD bit of the standby control register (STBC) to IDLE or STOP mode and executing the HALT instruction.

To enable DRAM to perform self-refresh, the standard $\overline{\text{RAS}}$ pulse width for DRAM (100 μs or greater) must be ensured.

Self-refresh is canceled using the $\overline{\text{RESET}}$ or $\overline{\text{NMI}}$ pin. The procedure for cancellation by $\overline{\text{RESET}}$ input is the same as that for normal reset.

6.2 ROM Controller (ROMC)

The ROM controller supports access to ROM having a page access function (page-ROM).

The ROM controller performs address comparison with the previous bus cycle and performs wait control for normal access (off-page)/page access (on-page). It supports page widths of 8-64 bytes.

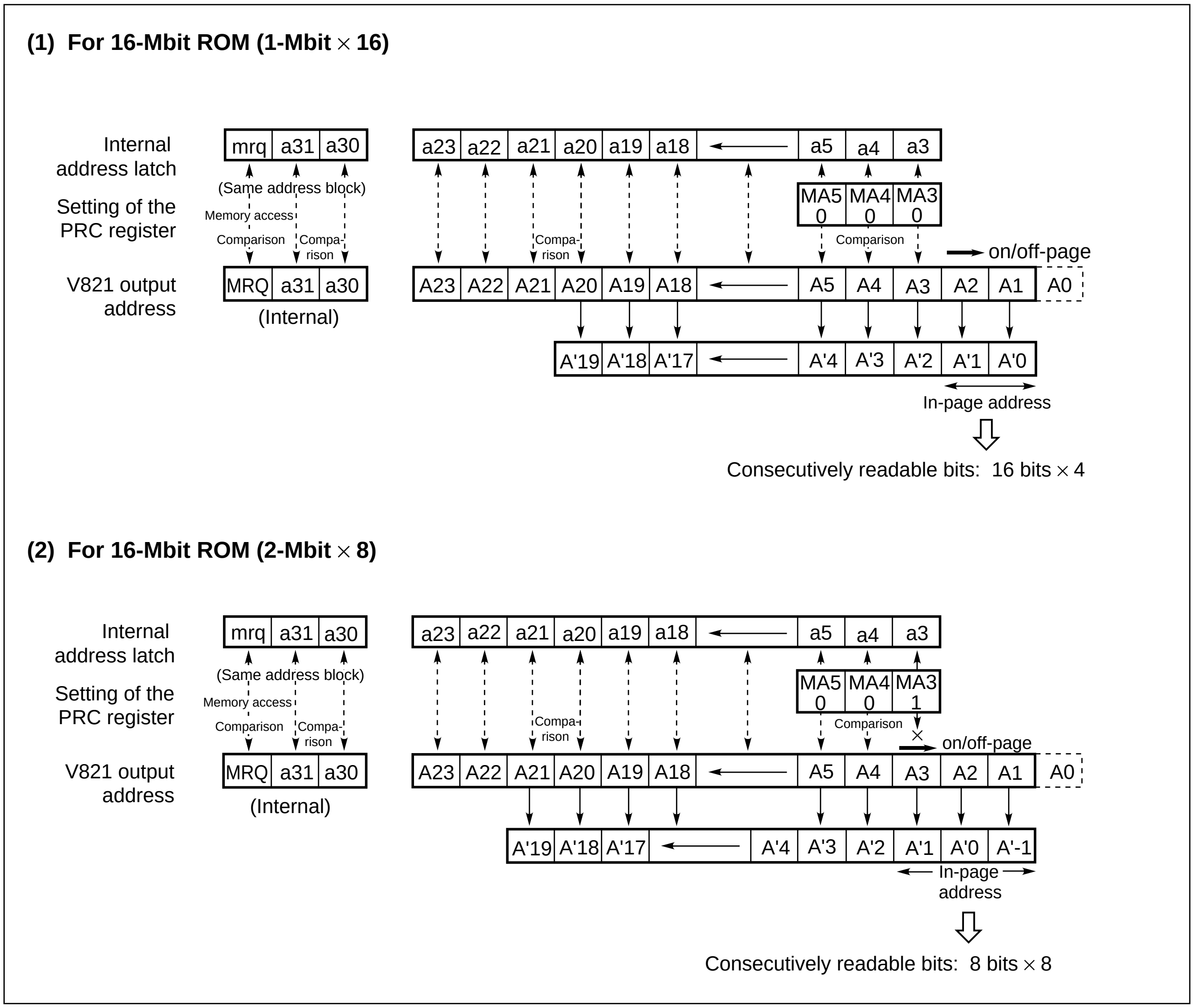
The page-ROM cycle is supported with address block 3.

6.2.1 on-page/off-page decision

Whether the page-ROM cycle is on-page or off-page is determined by latching the address during the previous cycle and comparing it with the address during the current cycle.

The address(es) (A3-A5) to be masked (not compared) is set using the page-ROM configuration register (PRC), according to the configuration of the connected page-ROM and the number of consecutively readable bits.

Figure 6-2. on-page/off-page Decision When ROM Having a Page Access Function Is Connected



(1) For 16-Mbit ROM (1-Mbit × 16)

(2) For 16-Mbit ROM (2-Mbit × 8)

7. DMA FUNCTIONS (DMA CONTROLLER)

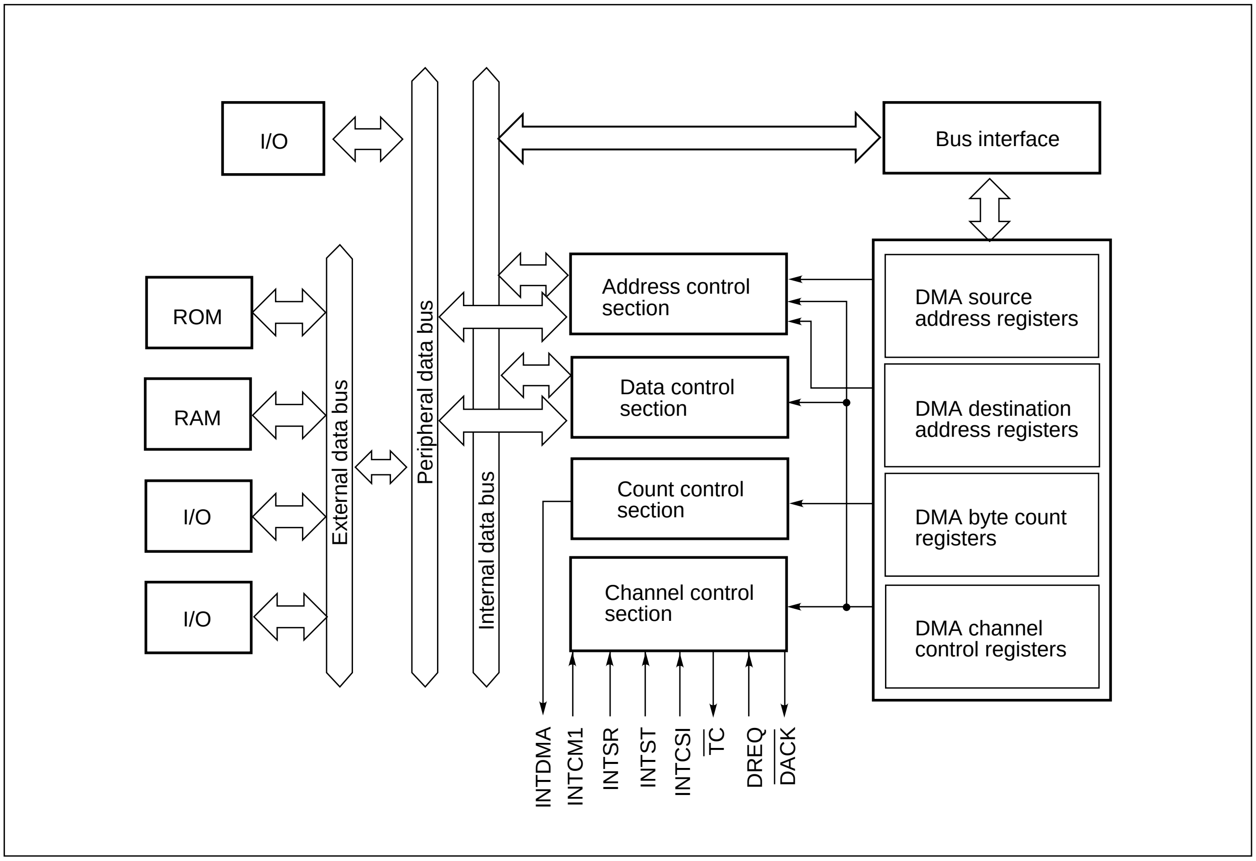
The V821 includes a DMA (Direct Memory Access) controller that executes and controls DMA transfer.

The DMAC (DMA controller) transfers data between memory and I/O, or within memory, based on DMA requests issued by the built-in peripheral hardware (serial interfaces and timer), external DREQ pins, or software triggers.

7.1 Features

- Two independent DMA channels
- Transfer units: 8/16 bits
- Maximum transfer count: 65 536 (2^{16})
- Two types of transfer
  - Fly-by (one-cycle) transfer
  - Two-cycle transfer
- Three transfer modes
  - Single transfer mode
  - Single-step transfer mode
  - Block transfer mode
- Transfer requests
  - External DREQ pin (× 2)
  - Requests from built-in peripheral hardware (serial interfaces and timer)
  - Requests from software
- Transfer objects
  - Memory to I/O and vice versa
  - Memory to memory and vice versa
- Programmable wait function
- DMA transfer end output signal ($\overline{\text{TC}}$)

Figure 7-1. Block Diagram of DMAC



8. SERIAL INTERFACE FUNCTION

8.1 Features

The V821 provides two transmission and reception channels as part of its serial interface function.

The two interface modes listed below are supported, one channel being provided for each mode. The two modes operate independently of each other.

(1) Asynchronous serial interface (UART)
(2) Synchronous serial interface (CSI)

In UART mode, one-byte serial data is transmitted or received after a start bit, and full-duplex communication is enabled.

In CSI mode, data is transferred using three signal lines (three-wire serial I/O): the serial clock ($\overline{\text{SCLK}}$), serial input (SI), and serial output (SO).

8.2 Asynchronous Serial Interface (UART)

8.2.1 Features

○ Transfer rate 110 bps to 38 400 bps (when BRG is used with ϕ = 25 MHz)
781 Kbps maximum (when ϕ/2 is used with ϕ = 25 MHz)

○ Full-duplex communication

○ Two-pin configuration TXD : Transmission data output pin
RXD : Reception data input pin

○ Reception error detection function

- Parity error
- Framing error
- Overrun error

○ Interrupt source (3 types)

- Reception error interrupt (INTSER)
- Reception completion interrupt (INTSR)
- Transmission completion interrupt (INTST)

○ The character length for transmission and reception data is specified upon ASIM reception.

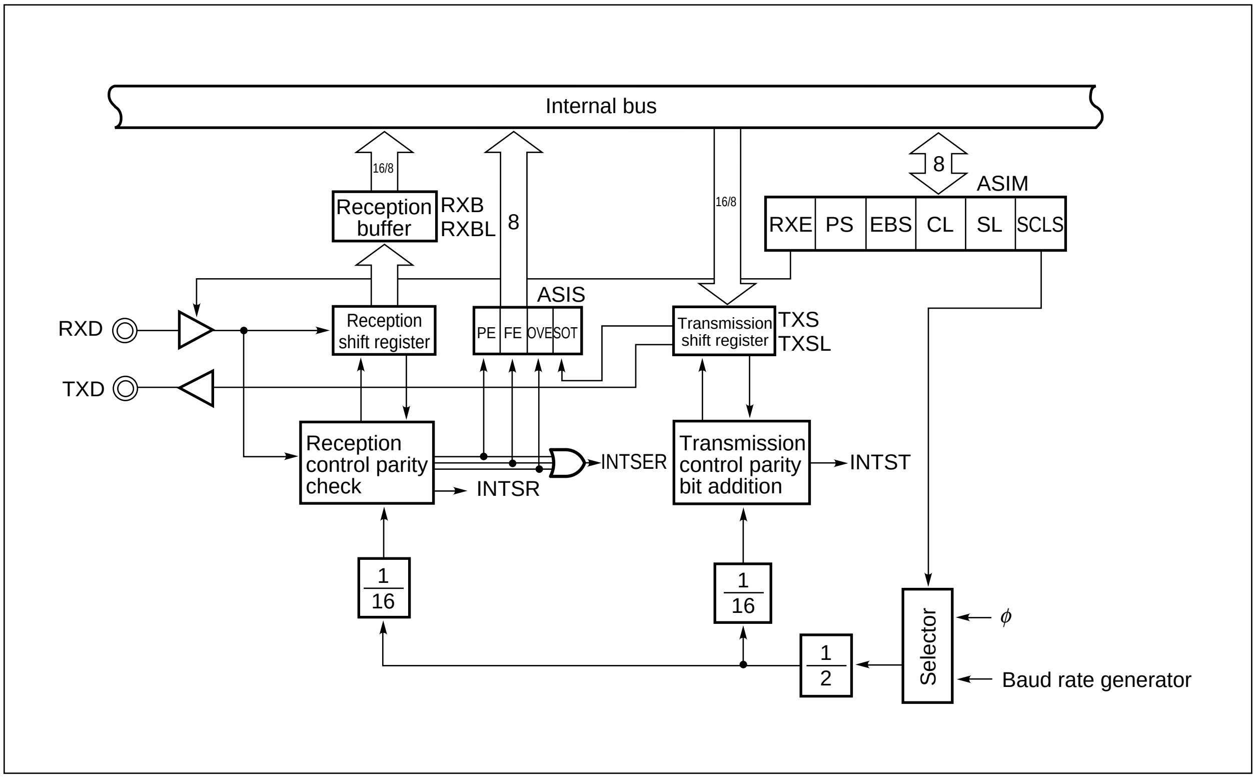
○ Character length : 7 or 8 bits
9 bits (when an extended bit is used)

○ Parity function: Odd parity, even parity, zero parity, without parity

○ Transmission stop bit: 1 or 2 bits

○ On-chip baud rate generator

Figure 8-1. Block Diagram of Asynchronous Serial Interface



8.3 Synchronous Serial Interface (CSI)

8.3.1 Features

○ High-speed transfer 6.25 Mbps maximum (when ϕ/2 is used with ϕ = 25 MHz)

○ Half-duplex communication

○ Character length: 8 bits

○ Switchable between the MSB and LSB to lead data transfer

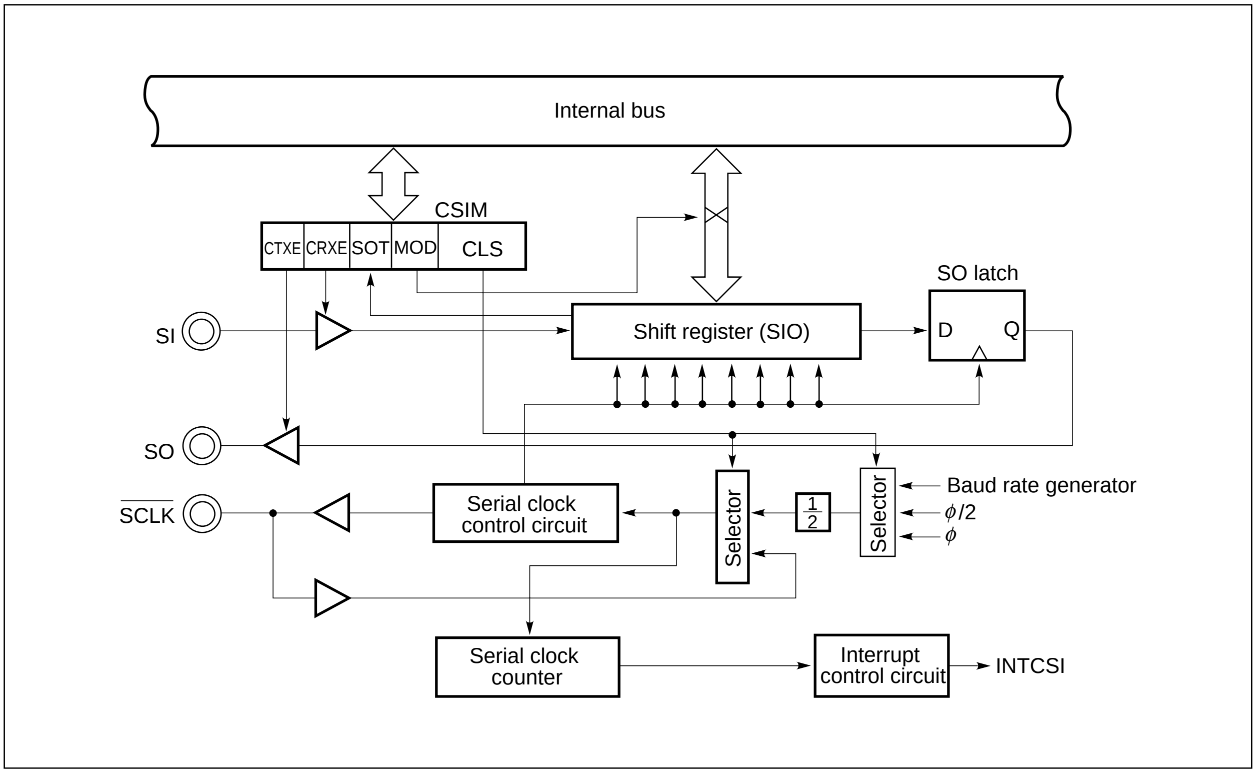
○ Allows selection between external serial clock input and internal serial clock output

○ Three-wire method SO : Serial data output
SI : Serial data input
$\overline{\text{SCLK}}$: Serial clock I/O pin

○ One interrupt source

- Interrupt request signal (INTCSI)

Figure 8-2. Block Diagram of Clock Synchronous Serial Interface



8.4 Baud Rate Generator (BRG)

8.4.1 Configuration and function

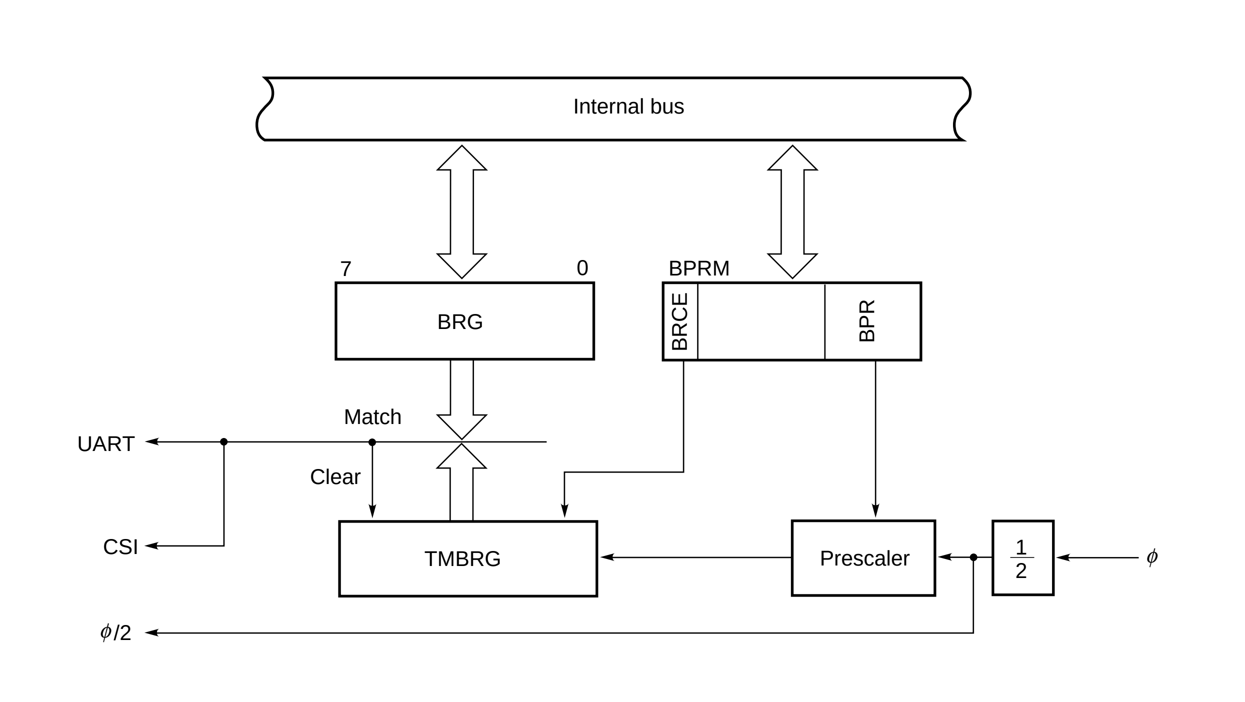
With the serial interface, a serial clock chosen from the baud rate generator output and clocks generated using the system clock (ϕ) can be used as a baud rate.

A serial clock source can be specified by using the SCLS bit of the ASIM register when the UART is used, or by using the CLS bit of the CSIM register when the CSI is used.

When baud rate generator output is specified, the baud rate generator is selected as the clock source.

The same serial clock is used for both transmission and reception on a channel, so that the same baud rate applies to transmission and reception.

Figure 8-3. Block Diagram



9. TIMER/COUNTER FUNCTIONS (REAL-TIME PULSE UNIT)

The real-time pulse unit (RPU) measures pulse intervals and frequencies, and outputs programmable pulses. It is capable of 16-bit measurement. It can also generate various types of pulses, such as interval pulse and one-shot pulse.

9.1 Features

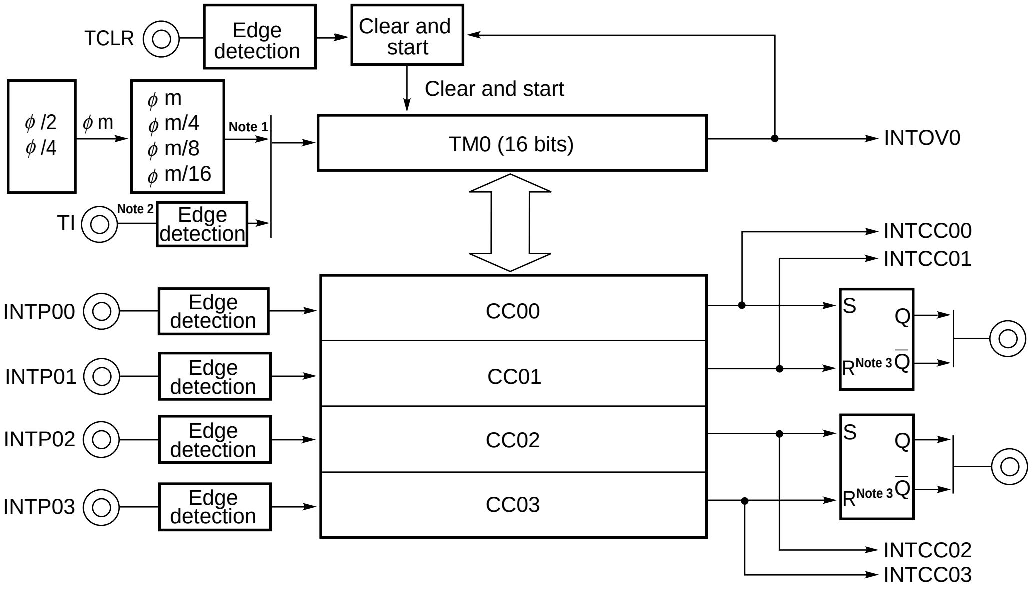
○ Timer 0 (TM0)

- 16-bit timer/event counter
- Two count clock sources (system clock frequency division selected or external pulse input)
- Four capture/compare registers
- Count clear pin (TCLR)
- Five interrupt sources
- Two external pulse outputs

○ Timer 1 (TM1)

- 16-bit interval timer
- Count clock generated by dividing the system clock frequency
- Compare register
- Interrupt source

Figure 9-1. Timer 0 (16-Bit Timer/Event Counter)



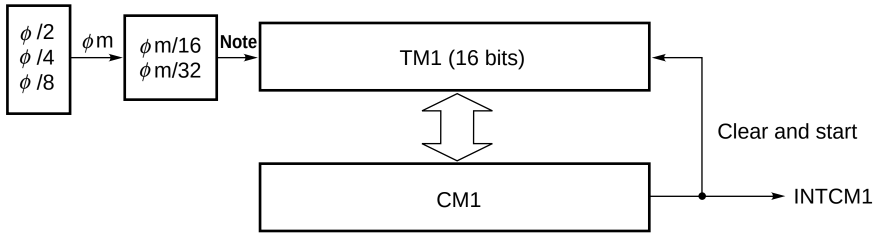
Notes 1. Internal count clock

2. External count clock

3. A reset takes precedence.

Remark ϕ: System clock

Figure 9-2. Timer 1 (16-Bit Interval Timer)



Note Internal count clock

Remark ϕ: System clock

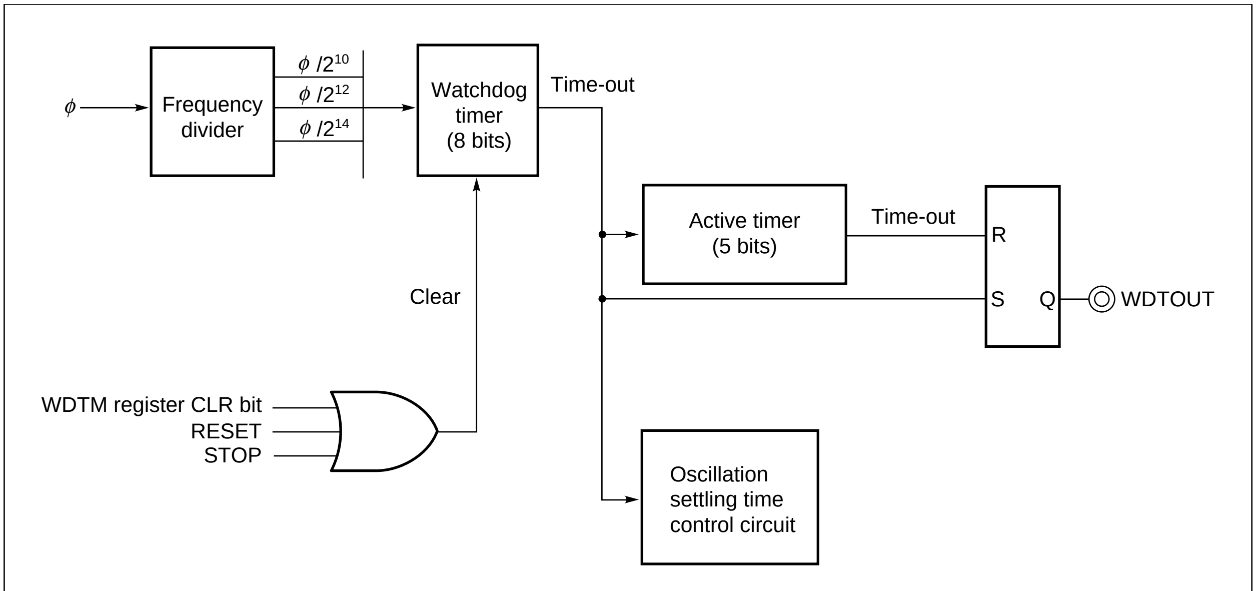
10. WATCHDOG TIMER FUNCTIONS

The watchdog timer is intended to prevent program crash and deadlock.

10.1 Features

- The following three different time-out time values can be specified: 10.5 ms, 41.9 ms, and 167.8 ms (when system clock ϕ = 25 MHz)
- Watchdog timer time-out output (WDTOUT)

Figure 10-1. Watchdog Timer Block Diagram



Remark ϕ: System clock

(1) Watchdog timer

One of the watchdog timer functions is to secure the oscillation settling time of the system clock. When the system is reset or placed in STOP mode, the timer is cleared to 00H.

The watchdog timer behaves in the standby modes as follows:

(a) STOP mode

The watchdog timer stops counting. When the system is released from STOP mode, the timer value is cleared.

The watchdog timer starts counting at 00H, and keeps counting until a time-out occurs. A time-out signal is supplied to the oscillation settling time control circuit, thus starting to supply the system clock pulse. At this point, the WDTOUT pin does not become active. If the system is released from STOP mode by the $\overline{\text{NMI}}$ pin, the timer continues counting.

(b) IDLE mode

The watchdog timer stops counting, but it holds the count value.

When the system is released from IDLE mode, the watchdog timer resumes counting by starting at the current count value.

(c) HALT mode

The watchdog timer continues counting.

(2) Active timer

The watchdog timer outputs the WDTOUT signal when it times out. The active timer retains this signal for 32 clock cycles.

When the watchdog timer times out, it can cause a system reset by connecting the WDTOUT and $\overline{\text{RESET}}$ pins through an external circuit.

10.2 Operation

The watchdog timer indicates that the program or system is running normally, by keeping the WDTOUT pin from becoming active.

To use the watchdog timer, it is necessary to specify the WDTM register so that the watchdog timer is cleared (restarted to count) at constant intervals during program execution or at the beginning of a subroutine. If the watchdog timer expires because it is not cleared within a specified period of time, the WDTOUT pin becomes active, indicating a program failure. In addition, the WDT time-out flag (OV) is set. This flag is cleared by clearing the WDT counter.

To use a watchdog timer time-out as an interrupt source, it is necessary to connect the WDTOUT pin to an external interrupt request pin ($\overline{\text{INTPn}}$ or $\overline{\text{NMI}}$) through an external circuit.

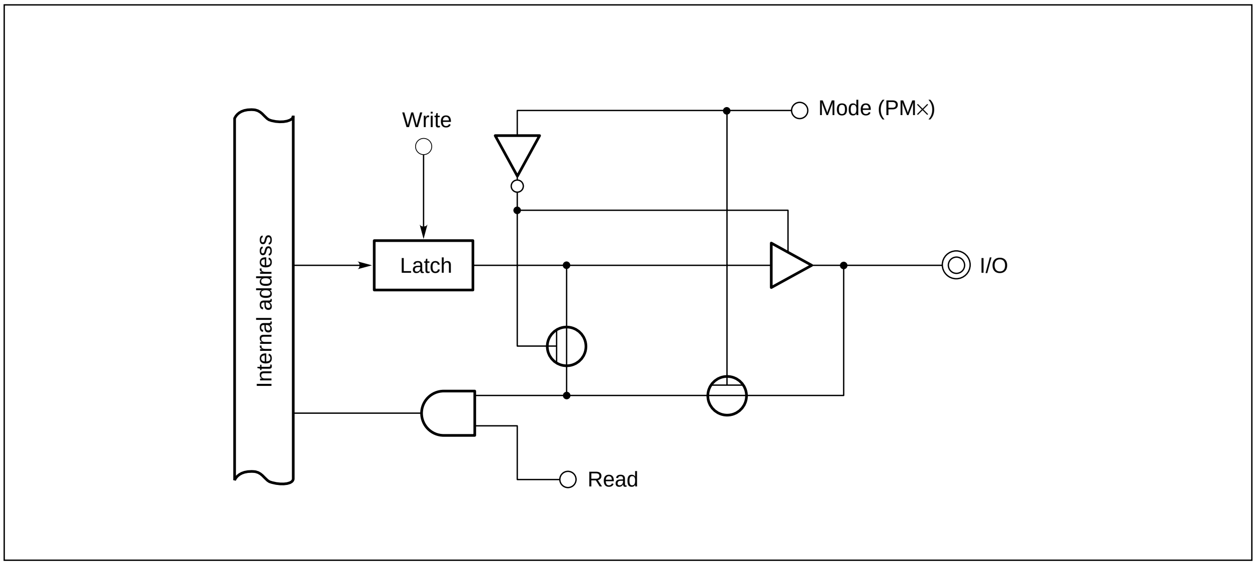
11. PORT FUNCTIONS

The V821 pins are dual-function pins that can function as both port and control pins. See **Chapter 1** for details of each pin.

11.1 Features

- 10 input/output ports (P00 to P09)

Figure 11-1. Configuration



12. CLOCK GENERATION FUNCTIONS

The clock generator generates and controls the internal clock pulse (ϕ) for the CPU and other built-in hardware units.

12.1 Features

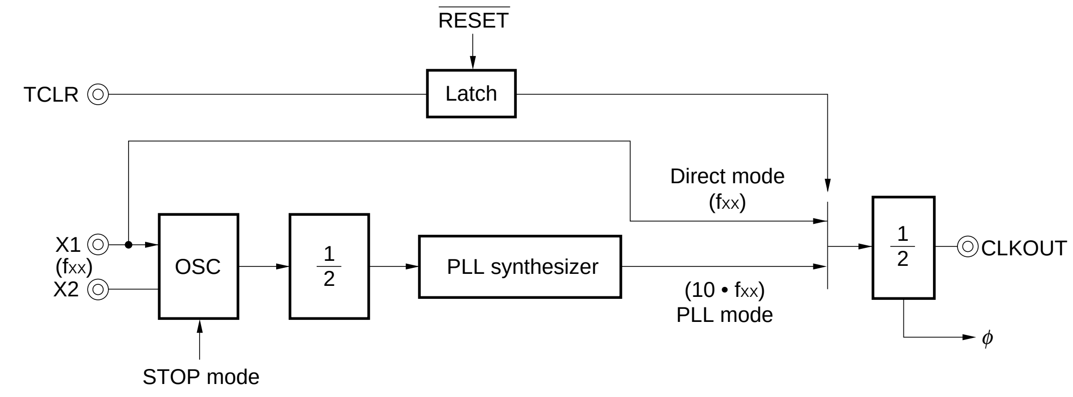
○ Frequency multiplication (5 times) using a PLL (phase locked loop) synthesizer

○ Clock sources

- Resonator-based oscillation: $f_{XX} = \phi/5$ (PLL mode)
- External clock : $f_{XX} = \phi/5$ (PLL mode)
- External clock : $f_{XX} = 2 \times \phi$ (direct mode)

○ Clock output control

★ **Figure 12-1. Configuration**



ϕ : Internal clock frequency (ϕ = 1/2•10•fxx: PLL mode)
Internal clock frequency (ϕ = 1/2•fxx: Direct mode)

OSC : Oscillator (only for the PLL mode)

13. STANDBY FUNCTIONS

The V821 supports three standby modes to suppress power dissipation. In these standby modes, the operation of the clock is controlled. The HALT instruction is used to select a standby mode. Mode switching is controlled using the standby control register.

13.1 Features

○ HALT mode (Only the CPU clock stops.)
○ IDLE mode (The CPU and peripheral operation clocks stop. The clock generator continues to operate.)
○ STOP mode (The entire system, including the clock generator, stops.)

13.2 Standby Mode

The standby modes of the V821 are detailed below.

(1) HALT mode

In this mode, the clock generator (oscillator and PLL synthesizer) continues operation, but the CPU clock stops. Clock supply to other built-in peripheral functions continues to allow them to keep running. Intermittent operation achieved using this standby mode in conjunction with the ordinary operation mode can reduce the total power dissipation of the system.

(2) IDLE mode

In this mode, the clock generator (oscillator and PLL synthesizer) continues operation, but internal system clock supply is stopped to bring the entire system to a stop.

When the system is released from IDLE mode, it is unnecessary to secure oscillation settling time for the oscillator, and therefore it is possible for the system to shift to the ordinary operation quickly.

For the oscillation settling time and current drain, IDLE mode lies in between STOP and HALT modes. IDLE mode is suitable for an application where it is necessary to cut the oscillation settling time using a low current drain mode.

(3) STOP mode

In this mode, the clock generator (oscillator and PLL synthesizer) is stopped to bring the entire system to a stop. This mode can generate an ultra-low power dissipation condition; only leak current occurs.

(a) PLL mode

In this mode, the PLL synthesizer clock output is stopped simultaneously with the oscillator. After the system is released from STOP mode, it is necessary to allow oscillation settling time for the oscillator. Some programs require a PLL lock-up time.

(b) Direct mode

In the direct mode, it is unnecessary to secure lock-up time.

Table 13-1 lists the operation of the clock generator in the ordinary, HALT, IDLE, and STOP modes. An effective low power dissipation system can be implemented by combining and switching these modes.

Table 13-1. Clock Generator Operation under Standby Control

Clock source		Standby mode	Oscillator (OSC)	PLL synthesizer	Clock supply to the peripheral I/O	Clock supply to the CPU
PLL mode	Resonator-based oscillation	Ordinary	o	o	o	o
		HALT	o	o	o	×
		IDLE	o	o	×	×
		STOP	×	×	×	×
	External clock	Ordinary	×	o	o	o
		HALT	×	o	o	×
		IDLE	×	o	×	×
		STOP	×	×	×	×
Direct mode		Ordinary	×	×	o	o
		HALT	×	×	o	×
		IDLE	×	×	×	×
		STOP	×	×	×	×

Remark o : Operating
× : Stopped

Table 13-2. Operation Status in HALT, IDLE, or STOP Mode

Function	HALT mode	IDLE mode	STOP mode
Clock generator	Operating		Stopped
Internal system clock	Operating	Stopped	
CPU	Stopped		
I/O line	Retained		
Peripheral function	Operating	Stopped	
Internal data	All internal data, such as in CPU registers is retained.		
A0-A23, $\overline{\text{UBE}}$	PC output[Note]		PC output
D0-D15	High impedance		
$\overline{\text{CS0}}$-$\overline{\text{CS3}}$	1[Note]		1
$\overline{\text{IORD}}$, $\overline{\text{IOWR}}$			
$\overline{\text{MWR}}$/$\overline{\text{LMWR}}$, $\overline{\text{UMWR}}$			
$\overline{\text{REFRQ}}$, $\overline{\text{RAS}}$, $\overline{\text{LCAS}}$, $\overline{\text{UCAS}}$	1 (other than during CBR refresh)[Note]	CBR self-refresh[Note]	CBR self-refresh
$\overline{\text{HLDRQ}}$	Operating[Note]		Stopped
CLKOUT	Clock output (when the clock output is not inhibited)		1

Note High impedance when $\overline{\text{HLDAK}}$ = 0

14. RESET FUNCTIONS

Inputting a low level to the $\overline{\text{RESET}}$ pin triggers a system reset, thus initializing the on-chip hardware.

When the $\overline{\text{RESET}}$ pin is driven from a low level to a high, the CPU starts program execution. The registers should be initialized in a program as required.

14.1 Features

○ The reset pin is provided with a noise suppressor circuit based on an analog delay (60 to 300 ns).

14.2 Pin Functions

Table 14-1 lists the state of the output from each pin during a system reset. The output state is retained during the entire reset period.

After the $\overline{\text{RESET}}$ pin is kept at a low level for 30 clock cycles, if the $\overline{\text{HLDRQ}}$ signal is inactive, a memory read cycle is started to fetch an instruction.

Even during a reset period (when the $\overline{\text{RESET}}$ pin is kept at a low level), activating the $\overline{\text{HLDRQ}}$ signal can place the bus on hold. The state of each pin with the bus put on hold during a reset is basically the same as that with the bus put on hold during a non-reset period.

The $\overline{\text{HLDRQ}}$ signal should be kept inactive during a power-on reset.

It is necessary to provide a pull-up or pull-down resistor to the pins that become high impedance during a reset. If no pull-up or pull-down resistor is provided to these pins, memory may be damaged when the pins are driven to high impedance.

The CLKOUT pin supplies clock pulses even during a reset.

Table 14-1. Output State of Each Pin during a Reset

Pin	Operation state	Pin	Operation state
A0-A23	Not defined	$\overline{\text{HLDAK}}$	High level
D0-D15	High impedance	$\overline{\text{MRD}}$	
P00/TCLR		$\overline{\text{LMWR}}$/$\overline{\text{WE}}$	
P01/DREQ0		$\overline{\text{UMWR}}$	
P02/$\overline{\text{DACK0}}$		$\overline{\text{IORD}}$	
P03/DREQ1		$\overline{\text{IOWR}}$	
P04/$\overline{\text{DACK1}}$		$\overline{\text{CS1}}$-$\overline{\text{CS3}}$	
P05/SI		$\overline{\text{RAS}}$	
P06/SO		$\overline{\text{LCAS}}$	
P07/$\overline{\text{SCLK}}$		$\overline{\text{UCAS}}$	
P08/TXD/$\overline{\text{UBE}}$		$\overline{\text{CS0}}$/$\overline{\text{REFRQ}}$	
P09/RXD/$\overline{\text{TC}}$		BLOCK/WDTOUT	Low level

15. INSTRUCTION SET

15.1 Instruction Format

The V821 instructions are formatted in either 16 bits or 32 bits. Examples of the 16-bit format instruction are binomial operation, control, and conditional branch; those for the 32-bit format are load/store, I/O manipulate, 16-bit immediate, jump & link, and extended operations.

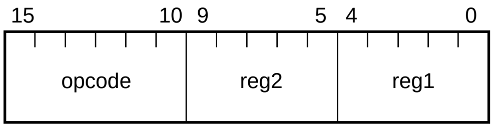
Some instructions have an unused field. However, do not write a program that uses this field because it is reserved for future use. This unused field must be set to zeros.

Instructions are stored in memory in the following manner.

- The lower half of an instruction, that is, the half which includes bit 0, is stored at the lower address.
- The higher half of an instruction, that is, the half which includes bit 15 or 31, is stored at the higher address.

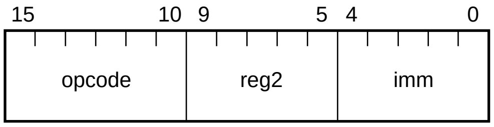
(1) reg-reg instruction format (Format I)

This format consists of one 6-bit field to hold an operation code and two 5-bit fields to specify general-purpose registers as instruction's operands. 16-bit instructions use this format.



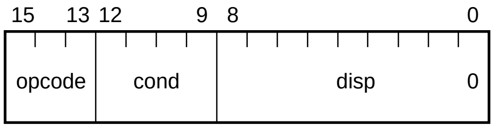
(2) imm-reg instruction format (Format II)

This format consists of one 6-bit field to hold an operation code, one 5-bit field to hold an immediate data, and one field to specify a general-purpose register as an operand. 16-bit instructions use this format.



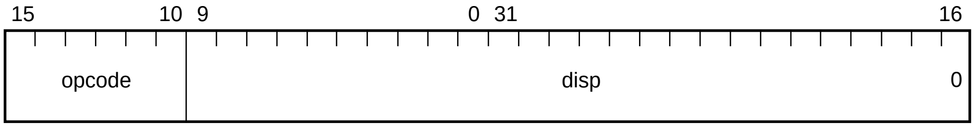
(3) Conditional branch instruction format (Format III)

This format consists of one 3-bit field to hold an operation code, one 4-bit field to hold a condition code, and one 9-bit field to hold a branch displacement (with its LSB masked to 0). 16-bit instructions use this format.



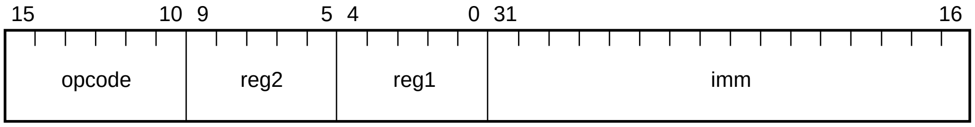
(4) Intermediate jump instruction format (Format IV)

This format consists of one 6-bit field to hold an operation code and one 26-bit field to hold a displacement (with its LSB masked to 0). 32-bit instructions use this format.



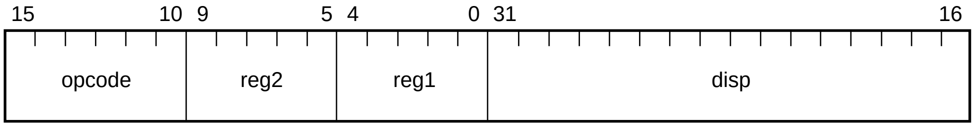
(5) 3-operand instruction format (Format V)

This format consists of one 6-bit field to hold an operation code, two fields to specify general-purpose registers as operands, and one 16-bit field to hold an immediate data. 32-bit instructions use this format.



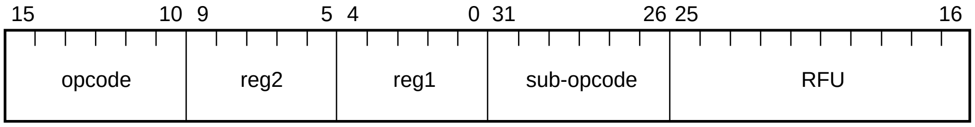
(6) Load/store instruction format (Format VI)

This format consists of one 6-bit field to hold an operation code, two fields to specify a general-purpose register, and one 16-bit field to hold a displacement. 32-bit instructions use this format.



(7) Extension instruction format (Format VII)

This format consists of one 6-bit field to hold an operation code, two 5-bit fields to specify general-purpose registers as operands, and one 6-bit field to hold an sub-operation code. 32-bit instructions use this format.



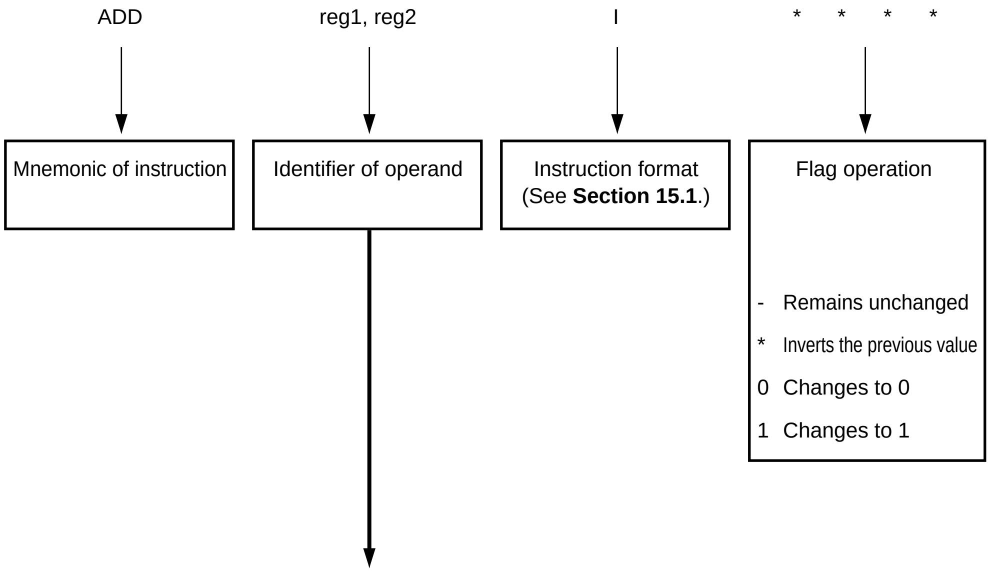
15.2 Instruction Mnemonic (In Alphabetical Order)

The list of mnemonics is shown below.

This section lists the instructions incorporated in the V821 along with their operations. The instructions are listed in the instruction mnemonic's alphabetical order to allow users to use this section as a quick reference or dictionary. The conventions used in the list are shown below.

Instruction mnemonic	Operand (s)	Format	CY	OV	S	Z	Instruction function

Legend



Identifier	Description
reg1	General-purpose register (Used as a source register)
reg2	General-purpose register (Used mainly as a destination register and occasionally as a source register)
imm5	5-bit immediate
imm16	16-bit immediate
disp9	9-bit displacement
disp16	16-bit displacement
disp26	26-bit displacement
regID	System register number
vector adr	Trap handler address that corresponds to a trap vector

Table 15-1. Instruction Mnemonics (In Alphabetical Order) (1/9)

Instruction mnemonic	Operand (s)	Format	CY	OV	S	Z	Instruction function
ADD	reg1, reg2	I	*	*	*	*	Addition: Adds the word data in the reg2-specified register and the word data in the reg1-specified register, then stores the result into the reg2-specified register.
ADD	imm5, reg2	II	*	*	*	*	Addition: Sign-extends the 5-bit immediate data to 32 bits, and adds the extended immediate data and the word data in the reg2-specified register, then stores the result into the reg2-specified register.
ADDF.S	reg1, reg2	VII	*	0	*	*	Floating-point addition: Adds the single-precision floating-point data in the reg2-specified register and the single-precision floating-point data in the reg1-specified register, then restores the result into the reg2-specified register while changing flags according to the result.
ADDI	imm16, reg1, reg2	V	*	*	*	*	Addition: Sign-extends the 16-bit immediate data to 32 bits, and adds the extended immediate data and the word data in the reg1-specified register, then stores the result into the reg2-specified register.
AND	reg1, reg2	I	-	0	*	*	AND: Performs the logical AND operation on the word data in the reg2-specified register and the word data in the reg1-specified register, then stores the result into the reg2-specified register.
ANDBSU	-	II	-	-	-	-	Transfer after ANDing bit strings: Performs a logical AND operation on a source bit string and a destination bit string, then transfers the result to the destination bit string.
ANDI	imm16, reg1, reg2	V	-	0	0	*	AND: Sign-extends the 16-bit immediate data to 32 bits, and performs a logical AND operation on the extended immediate data and the word data in the reg1-specified register, then stores the result into the reg2-specified register.
ANDNBSU	-	II	-	-	-	-	Transfer after NOTting a bit string then ANDing it with another bit string: Performs a logical AND operation on a destination bit string and the 1's complement of a source bit string, then transfers the result to the destination bit string.
BC	disp9	III	-	-	-	-	Conditional branch (if Carry): PC relative branch
BE	disp9	III	-	-	-	-	Conditional branch (if Equal): PC relative branch
BGE	disp9	III	-	-	-	-	Conditional branch (if Greater than or Equal): PC relative branch
BGT	disp9	III	-	-	-	-	Conditional branch (if Greater than): PC relative branch

Table 15-1. Instruction Mnemonics (In Alphabetical Order) (2/9)

Instruction mnemonic	Operand (s)	Format	CY	OV	S	Z	Instruction function
BH	disp9	III	-	-	-	-	Conditional branch (if Higher): PC relative branch
BL	disp9	III	-	-	-	-	Conditional branch (if Lower): PC relative branch
BLE	disp9	III	-	-	-	-	Conditional branch (if Less than or Equal): PC relative branch
BLT	disp9	III	-	-	-	-	Conditional branch (if Less than): PC relative branch
BN	disp9	III	-	-	-	-	Conditional branch (if Negative): PC relative branch
BNC	disp9	III	-	-	-	-	Conditional branch (if Not Carry): PC relative branch
BNE	disp9	III	-	-	-	-	Conditional branch (if Not Equal): PC relative branch
BNH	disp9	III	-	-	-	-	Conditional branch (if Not Higher): PC relative branch
BNL	disp9	III	-	-	-	-	Conditional branch (if Not Lower): PC relative branch
BNV	disp9	III	-	-	-	-	Conditional branch (if Not Overflow): PC relative branch
BNZ	disp9	III	-	-	-	-	Conditional branch (if Not Zero): PC relative branch
BP	disp9	III	-	-	-	-	Conditional branch (if Positive): PC relative branch
BR	disp9	III	-	-	-	-	Unconditional branch: PC relative branch
BV	disp9	III	-	-	-	-	Conditional branch (if Overflow): PC relative branch
BZ	disp9	III	-	-	-	-	Conditional branch (if Zero): PC relative branch
CAXI	disp16 [reg1], reg2	VI	*	*	*	*	Inter-processor synchronization in a multi-processor system
CMP	reg1, reg2	I	*	*	*	*	Comparison: Subtracts the word data in the reg1-specified register from that for reg2 for comparison, then changes flags according to the result.
CMP	imm5, reg2	II	*	*	*	*	Comparison: Sign-extends the 5-bit immediate data to 32 bits, and subtracts the extended immediate data from the word data in the reg2-specified register for comparison, then changes flags according to the result.
CMPF.S	reg1, reg2	VII	*	0	*	*	Floating-point comparison: Subtracts the single-precision floating-point data in the reg1-specified register from that for reg2 for comparison, then changes flags according to the result.

Table 15-1. Instruction Mnemonics (In Alphabetical Order) (3/9)

Instruction mnemonic	Operand (s)	Format	CY	OV	S	Z	Instruction function
CVT.SW	reg1, reg2	VII	-	0	*	*	Data conversion from floating-point to integer: Converts the single-precision floating-point data in the reg1-specified register into an integer data, then stores the result into the reg2-specified register while changing flags according to the result.
CVT.WS	reg1, reg2	VII	*	0	*	*	Data conversion from integer to floating-point: Converts the integer data in the reg1-specified register into a single-precision floating-point data, then stores the result into the reg2-specified register while changing flags according to the result.
DIV	reg1, reg2	I	-	*	*	*	Signed division: Divides the word data in the reg2-specified register by that for reg1 with their sign bits validated, then stores the quotient into the reg2-specified register and the remainder into r30. Division is performed so that the sign of the remainder matches that of the dividend.
DIVF.S	reg1, reg2	VII	*	0	*	*	Floating-point division: Divides the single-precision floating-point data in the reg2-specified register by that for reg1, then stores the result into the reg2-specified register while changing flags according to the result.
DIVU	reg1, reg2	I	-	0	*	*	Unsigned division: Divides the word data in the reg2-specified register by that for reg1 with their data handled as unsigned data, then stores the quotient into the reg2-specified register and the remainder into r30. Division is performed so that the sign of the remainder matches that of the dividend.
HALT	-	II	-	-	-	-	Processor stop
IN.B	disp16 [reg1], reg2	VI	-	-	-	-	Port input: Sign-extends the 16-bit displacement to 32 bits, and adds the extended displacement and the content of the reg1-specified register to generate a 32-bit unsigned port address, then reads the byte data located at the generated port address, zero-extends the byte data to 32 bits, and stores the result into the reg2-specified register.
IN.H	disp16 [reg1], reg2	VI	-	-	-	-	Port input: Sign-extends the 16-bit displacement to 32 bits, and adds the extended displacement and the content of the reg1-specified register to generate a 32-bit unsigned port address, then reads the halfword data located at the generated port address while masking the address's bit 0 to 0, zero-extends the halfword data to 32 bits, and stores the result into the reg2-specified register.

Table 15-1. Instruction Mnemonics (In Alphabetical Order) (4/9)

Instruction mnemonic	Operand (s)	Format	CY	OV	S	Z	Instruction function
IN.W	disp16 [reg1], reg2	VI	-	-	-	-	Port input: Sign-extends the 16-bit displacement to 32 bits, and adds the extended displacement and the content of the reg1-specified register to generate a 32-bit unsigned port address, then reads the word data located at the generated address while masking the address's bits 0 and 1 to 0, and stores the word into the reg2-specified register.
JAL	disp26	IV	-	-	-	-	Jump and link: Increments the current PC by 4, then saves it into r31, and sign-extends the 26-bit displacement to 32 bits while masking the displacement's bit 0 to 0, adds the extended displacement and the PC value, loads the PC with the addition result, so that the instruction stored at the PC-pointing address is executed next.
JMP	[reg1]	I	-	-	-	-	Register-indirect unconditional branch: Loads the PC with the jump address value in the reg1-specified register while masking the value's bit 0 to 0, so that the instruction stored at the address pointed by the reg1-specified register is executed next.
JR	disp26	IV	-	-	-	-	Unconditional branch: Sign-extends the 26-bit displacement to 32 bits while masking bit 0 to 0, adds the result with the current PC value, and loads the PC with the addition result so that the instruction stored at the PC-pointing address is executed next.
LD.B	disp16 [reg1], reg2	VI	-	-	-	-	Byte load: Sign-extends the 16-bit displacement to 32 bits, and adds the result with the content of the reg1-specified register to generate the 32-bit unsigned address, then reads the byte data located at the generated address, sign-extends the byte data to 32 bits, and stores the result into the reg2-specified register.
LD.H	disp16 [reg1], reg2	VI	-	-	-	-	Halfword load: Sign-extends the 16-bit displacement to 32 bits, and adds the result with the content of the reg1-specified register to generate a 32-bit unsigned address while masking its bit 0 to 0, then reads the halfword data located at the generated address, sign-extends the halfword data to 32 bits, and stores the result into the reg2-specified register.
LD.W	disp16 [reg1], reg2	VI	-	-	-	-	Word load: Sign-extends the 16-bit displacement to 32 bits and adds the result with the content of the reg1-specified register to generate a 32-bit unsigned address while masking bits 0 and 1 to 0, then reads the word data located at the generated address and stores the data into the reg2-specified register.

Table 15-1. Instruction Mnemonics (In Alphabetical Order) (5/9)

Instruction mnemonic	Operand (s)	Format	CY	OV	S	Z	Instruction function
LDSR	reg2, regID	II	*	*	*	*	Loading system register: Transfers the word data in the reg2-specified register to the system register specified with the system register number (regID).
MOV	reg1, reg2	I	-	-	-	-	Transferring data: Loads the reg2-specified register with the word data in of the reg1-specified register.
MOV	imm5, reg2	II	-	-	-	-	Transferring data: Sign-extends the 5-bit immediate data to 32 bits, then loads the reg2-specified register with the extended immediate data.
MOVBSU	-	II	-	-	-	-	Transferring bit strings: Loads the destination bit string with the source bit string.
MOVEA	imm16, reg1, reg2	V	-	-	-	-	Addition: Sign-extends the 16-bit immediate data to 32 bits, adds it with the word data in the reg1-specified register, then stores the addition result into reg2.
MOVHI	imm16, reg1, reg2	V	-	-	-	-	Addition: Appends 16-bit zeros below the 16-bit immediate data to form a 32-bit word data, then adds it with the word data in the reg1-specified register, and stores the result into the reg2-specified register.
MUL	reg1, reg2	I	-	*	*	*	Signed multiplication: Signed-multiplies the word data in the reg2-specified register by that for reg1, then separates the 64-bit (double-word) result into two 32-bit data, and stores the higher 32 bits into r30 and the lower 32 bits into the reg2-specified register.
MULF.S	reg1, reg2	VII	*	0	*	*	Floating-point multiplication: Multiplies the single-precision floating-point data in the reg2-specified register by that for reg1, then stores the result into the reg2-specified register while changing flags according to the result.
MULU	reg1, reg2	I	-	*	*	*	Unsigned multiplication: Multiplies the word data in the reg2-specified register by that for reg1 while handling these data as unsigned data, then separates the 64-bit (double-word) result into two 32-bit data, and stores the higher 32 bits into r30 and the lower 32 bits into the reg2-specified register.
NOP	-	III	-	-	-	-	No operation
NOT	reg1, reg2	I	-	0	*	*	Logical NOT: Obtains the 1’s complement (logical NOT) of the content of the reg1-specified register, then stores the result into the reg2-specified register.

Table 15-1. Instruction Mnemonics (In Alphabetical Order) (6/9)

Instruction mnemonic	Operand (s)	Format	CY	OV	S	Z	Instruction function
NOTBSU	-	II	-	-	-	-	Transfer after NOTting a bit string: Obtains the 1's complement (all bits inverted) of the source bit string, then transfers the result to the destination bit string.
OR	reg1, reg2	I	-	0	*	*	OR: Performs a logical OR operation on the word data in the reg2-specified register and that for reg1, then stores the result into the reg2-specified register.
ORBSU	-	II	-	-	-	-	Transfer after ORing bit strings: Performs a logical OR operation on the source and destination bit strings, then transfers the result to the destination bit string.
ORI	imm16, reg1, reg2	V	-	0	*	*	OR: Zero-extends the 16-bit immediate data to 32 bits, performs a logical OR operation on the extended data and the word data in the reg1-specified register, then stores the result into the reg2-specified register.
ORNBSU	-	II	-	-	-	-	Transfer after NOTting a bit string and ORing it with another bit string: Obtains the 1's complement (logical NOT) of the source bit string, performs a logical OR operation on the NOTted bit string and the destination bit string, then transfers the result to the destination bit string.
OUT.B	reg2, disp16 [reg1]	VI	-	-	-	-	Port output: Sign-extends the 16-bit displacement to 32 bits, adds the extended value and the content of the reg1-specified register to generate a 32-bit unsigned port address, then outputs the lowest 8 bits (= 1 byte) of the reg2-specified register onto the port pins corresponding to the generated port address.
OUT.H	reg2, disp16 [reg1]	VI	-	-	-	-	Port output: Sign-extends the 16-bit displacement to 32 bits, adds the extended value and the content of the reg1-specified register to generate a 32-bit unsigned port address with its bit 0 masked to 0, then outputs the lowest 16 bits (= 1 halfword) of the reg2-specified register onto the port pins corresponding to the generated port address.
OUT.W	reg2, disp16 [reg1]	VI	-	-	-	-	Port output: Sign-extends the 16-bit displacement to 32 bits, adds the extended value and the content of the reg1-specified register to generate a 32-bit unsigned port address with its bits 0 and 1 masked to 0, then outputs the 32 bits (= 1 word) of the reg2-specified register onto the port pins corresponding to the generated port address.

Table 15-1. Instruction Mnemonics (In Alphabetical Order) (7/9)

Instruction mnemonic	Operand (s)	Format	CY	OV	S	Z	Instruction function
RETI	-	II	*	*	*	*	Return from a trap or interrupt routine: Reads the restore PC and PSW from the system registers and loads them to the due places to return from a trap or interrupt routine to the original operation flow.
SAR	reg1, reg2	I	*	0	*	*	Arithmetic right shift: Shifts every bit of the word data in the reg2-specified register to the right by the number of times specified with the reg1-specified register's lowest 5 bits, then stores the result into the reg2-specified register. In arithmetic right shift operations, the MSB is loaded with the LSB value at each shift.
SAR	imm5, reg2	II	*	0	*	*	Arithmetic right shift: Zero-extends the 5-bit immediate data to 32 bits, shifts every bit of the word data in the reg2-specified register to the right by the number of times specified with the extended immediate data, then stores the result into the reg2-specified register.
SCH0BSU SCH0BSD	- -	II II	- -	- -	- -	* *	Searching 0s in a bit string: Searches "0" bits in the source bit string, and loads r30 and r27 with the address of the bit next to the first detected "0" bit, then r29 with the number of bits skipped until the first "0" bit is detected, and r28 with the value subtracted by the r29 value.
SCH1BSU SCH1BSD	- -	II II	- -	- -	- -	- -	Searching 1s in a bit string: Searches 1s in the source bit string, and loads r30 and r27 with the bit address next to the first detected "1" bit, then r29 with the number of bits skipped until the first "1" is detected, and r28 with the value subtracted by the r29 value.
SETF	imm5, reg2	II	-	-	-	-	Flag condition setting: Sets the reg2-specified register to 1 if the condition flag value matches the lowest 4 bits of the 5-bit immediate data, and sets the reg2-specified register to 0 when they do not match.
SHL	reg1, reg2	I	*	0	*	*	Logical left shift: Shifts every bit of the word data in the reg2-specified register to the left by the number of times specified with the reg1-specified register's lowest 5 bits, then stores the result into the reg2-specified register. In logical left shift operations, the LSB is loaded with 0 at each shift.

Table 15-1. Instruction Mnemonics (In Alphabetical Order) (8/9)

Instruction mnemonic	Operand (s)	Format	CY	OV	S	Z	Instruction function
SHL	imm5, reg2	II	*	0	*	*	Logical left shift: Zero-extends the 5-bit immediate data to 32 bits, shifts every bit of the word data in the reg2-specified register to the left by the number of times specified by the extended immediate data, then stores the result into the reg2-specified register.
SHR	reg1, reg2	I	*	0	*	*	Logical right shift: Shifts every bit of the word data in the reg2-specified register to the right by the number of times specified with the reg1-specified register's lowest 5 bits, then stores the result into the reg2-specified register. In logical right shift operations, the MSB is loaded with 0 at each shift.
SHR	imm5, reg2	II	*	0	*	*	Logical right shift: Zero-extends the 5-bit immediate data to 32 bits, shifts every bit of the word data in the reg2-specified register to the right by the number of times specified by the extended immediate data, then stores the result into the reg2-specified register.
ST.B	reg2, disp16 [reg1]	VI	-	-	-	-	Byte store: Sign-extends the 16-bit displacement to 32 bits and adds the 32-bit displacement and the content of the reg1-specified register to generate a 32-bit unsigned address, then transfers the reg2-specified register's lowest 8 bits to the generated address.
ST.H	reg2, disp16 [reg1]	VI	-	-	-	-	Halfword store: Sign-extends the 16-bit displacement to 32 bits with its bit 0 masked to 0, and adds the content of the reg1-specified register and the 32-bit displacement to generate a 32-bit unsigned address, then transfers the reg2-specified register's lower 16 bits to the generated address.
ST.W	reg2, disp16 [reg1]	VI	-	-	-	-	Word store: Sign-extends the 16-bit displacement to 32 bits with its bits 0 and 1 masked to 0, and adds the reg1-specified register and the 32-bit displacement to generate a 32-bit unsigned address, then transfers the word data of the reg2-specified register to the generated address.
STSR	regID, reg2	II	-	-	-	-	Storing system register contents: Loads the reg2-specified register with the content of the system register specified by the system register number (regID).
SUB	reg1, reg2	I	*	*	*	*	Subtraction: Subtracts the word data in the reg1-specified register from that in the reg2-specified register, then stores the result into the reg2-specified register.

Table 15-1. Instruction Mnemonics (In Alphabetical Order) (9/9)

Instruction mnemonic	Operand (s)	Format	CY	OV	S	Z	Instruction function
SUBF.S	reg1, reg2	VII	*	0	*	*	Floating-point subtraction: Subtracts the single-precision floating-point data in the reg1-specified register from that for reg2, then stores the result into the reg2-specified register while changing flags according to the result.
TRAP	vector	II	-	-	-	-	Software trap: Jumps to a trap handler address according to the vector-specified trap vector (from 0 to 31) to start an exception handling after completing all necessary saving and presetting procedures as follows: (1) Saving the restore PC and PSW into the FEPC and FEPSW system registers, respectively, if the PSW's EP flag = 1, or into the EIPC and EIPSW system registers, respectively, if EP = 0 (2) Setting an exception code into the ECR's FECC and FEPSW flags if the PSW's EP flag = 1, or into the ECR's EICC if EP = 0 (3) Setting the PSW's ID flag and clearing the PSW's AE flag (4) Setting the PSW's NP flag if the PSW's EP flag = 1, or setting the PSW's ID flag if EP = 0
TRNC.SW	reg1, reg2	VII	-	0	*	*	Conversion from floating-point data to integer: Converts the single-precision floating-point data in the reg1-specified register into an integer data, then stores the result into the reg2-specified register while changing flags according to the result.
XOR	reg1, reg2	I	-	0	*	*	Exclusive OR: Performs a logical exclusive-OR operation on the word data in the reg2-specified register and that for reg1, then stores the result into the reg2-specified register.
XORBSU	-	II	-	-	-	-	Transfer of exclusive ORed bit string: Performs a logical exclusive-OR operation on the source and destination bit strings, then transfers the result to the destination bit string.
XORI	imm16, reg1, reg2	V	-	0	*	*	Exclusive OR: Zero-extends the 16-bit immediate data to 32 bits and performs a logical exclusive-OR operation on the extended immediate data and the word data in the reg1-specified register, then stores the result into the reg2-specified register.
XORNBSU	-	II	-	-	-	-	Transfer after exclusive-ORing a NOTted bit string and another bit string: Obtains the 1's complement (NOT) of the source bit string, and exclusive-ORs it with the destination bit string, then transfers the result to the destination bit string.

16. ELECTRICAL SPECIFICATIONS

ABSOLUTE MAXIMUM RATINGS (T_A = 25 °C)

Parameter	Symbol	Conditions	Rating	Unit
Supply voltage	V_{DD}		-0.5 to +7.0	V
Input voltage	V_I	V_{DD} = +5.0 V ± 10 %	-0.5 to V_{DD} + 0.3	V
Clock input voltage	V_K	V_{DD} = +5.0 V ± 10 %	-0.5 to V_{DD} + 0.3	V
Output voltage	V_O	V_{DD} = +5.0 V ± 10 %	-0.5 to V_{DD} + 0.3	V
Operating ambient temperature	T_A		-40 to +85	°C
Storage temperature	T_{stg}		-65 to +150	°C

Cautions 1. Do not connect an output (or input/output) pin of an IC device directly to any other output (or input/output) pin of the same device, or directly to V_{DD} , V_{CC}, or GND. Open-drain pins and open-collector pins can, however, be connected directly to each other. Note, however, that these restrictions do not apply to those high-impedance pins that are provided with an external circuit for which timings have been designed such that no output contention occurs.

2. Absolute maximum ratings are rated values beyond which some physical damages may be caused to the product; if any of the parameters in the table above exceeds its rated value even for a moment, the quality of the product may deteriorate. Be sure to use the product with a moderate value within the rated range.

The standard values and conditions listed in the DC and AC characteristics tables indicate the ranges in which the normal operation and performance of the product can be guaranteed.

DC CHARACTERISTICS (T_A = -40 to +85 °C, V_{DD} = +5.0 V ± 10 %)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Low-level clock input voltage	V_{KL}		-0.5		+0.6	V
High-level clock input voltage	V_{KH}		4.0		V_{DD} + 0.3	V
Low-level input voltage	V_{IL1}	Other than $\overline{\text{RESET}}$, $\overline{\text{NMI}}$, and $\overline{\text{INTPn}}$	-0.5		+0.8	V
	V_{IL2}	$\overline{\text{RESET}}$, $\overline{\text{NMI}}$, and $\overline{\text{INTPn}}$	-0.5		+0.2V_{DD}	V
High-level input voltage	V_{IH1}	Other than $\overline{\text{RESET}}$, $\overline{\text{NMI}}$, and $\overline{\text{INTPn}}$	2.2		V_{DD} + 0.3	V
	V_{IH2}	$\overline{\text{RESET}}$, $\overline{\text{NMI}}$, and $\overline{\text{INTPn}}$	0.8V_{DD}		V_{DD} + 0.3	V
Schmitt-triggered input hysteresis width	V_{SH}	$\overline{\text{RESET}}$, $\overline{\text{NMI}}$, and $\overline{\text{INTPn}}$	0.5			V
Low-level output voltage	V_{OL}	I_{OL} = 2.5 mA			0.45	V
High-level output voltage	V_{OH}	I_{OH} = -2.5 mA	0.7V_{DD}			V
		I_{OH} = -100 μA	V_{DD} - 0.4			V
Low-level input leakage current	I_{LIL}	V_{IN} = 0 V			-10	μA
High-level input leakage current	I_{LIH}	V_{IN} = V_{DD}			10	μA
Low-level output leakage current	I_{LOL}	V_O = 0 V			-10	μA
High-level output leakage current	I_{LOH}	V_O = V_{DD}			10	μA
Supply current	I_{DD}	Operation (f = 25 MHz)		100	150	mA
		HALT (f = 25 MHz)		18	45	mA
		IDLE (f = 25 MHz)		4	35	mA
		STOP		5	20	μA

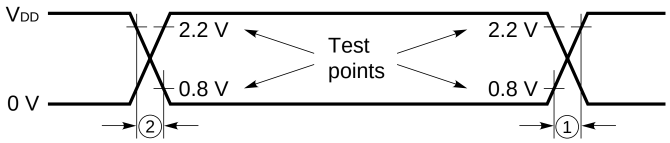
CAPACITANCE (T$_A$ = 25 °C, V$_{DD}$ = +5.0 V ± 10 %)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
Input capacitance	C_I	fc = 1 MHz		15	pF
Input/output capacitance	C_{IO}			15	pF

AC CHARACTERISTICS (T_A = -40 to +85 °C, V_{DD} = +5.0 V ± 10 %)

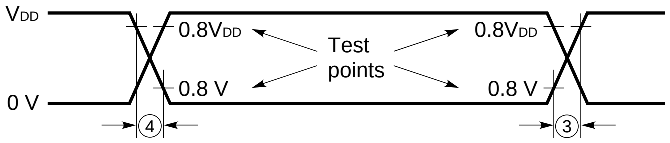
AC Test Input Waveform (Other than $\overline{\text{RESET}}$, $\overline{\text{NMI}}$, and $\overline{\text{INTPn}}$)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Input rise time	①	t_R			7	ns
Input fall time	②	t_F			7	ns

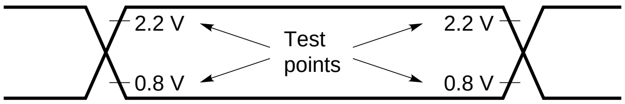


AC Test Input Waveform ($\overline{\text{RESET}}$, $\overline{\text{NMI}}$, and $\overline{\text{INTPn}}$)

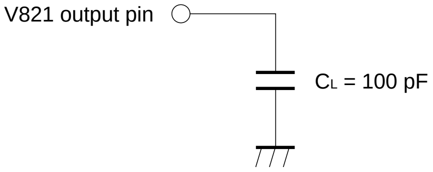
Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Schmitt-triggered input rise time	③	t_{RS}			10	ns
Schmitt-triggered input fall time	④	t_{FS}			10	ns



AC Test Output Waveform

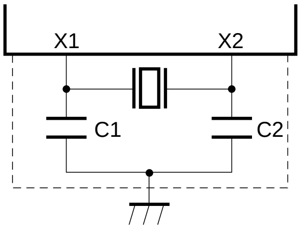


Load Condition



RECOMMENDED OSCILLATION CIRCUIT

(a) Connecting a ceramic resonator
(Murata Mfg. Co., Ltd.: T_A = -20 to +80 °C, TDK Corp.: T_A = -40 to +85 °C)



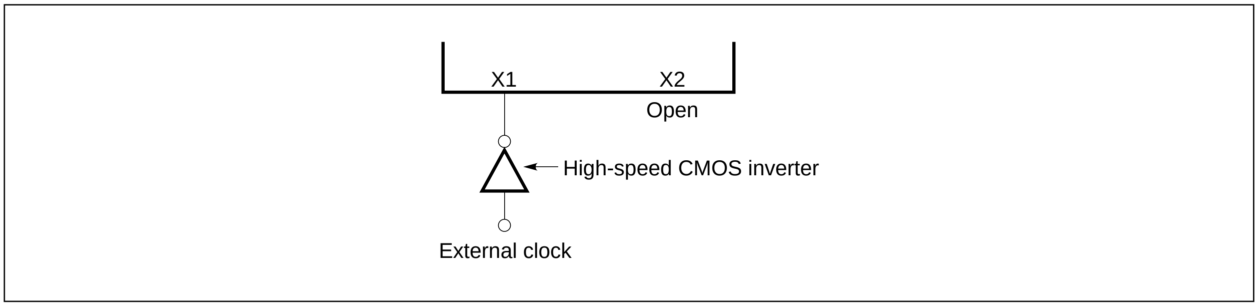
Cautions 1. The oscillation circuit should be placed as close to the X1 and X2 pins as possible.

2. Do not draw other signal lines in the area enclosed by broken lines.

3. Throughly evaluate the matching between the μPD70741 and the oscillation circuit.

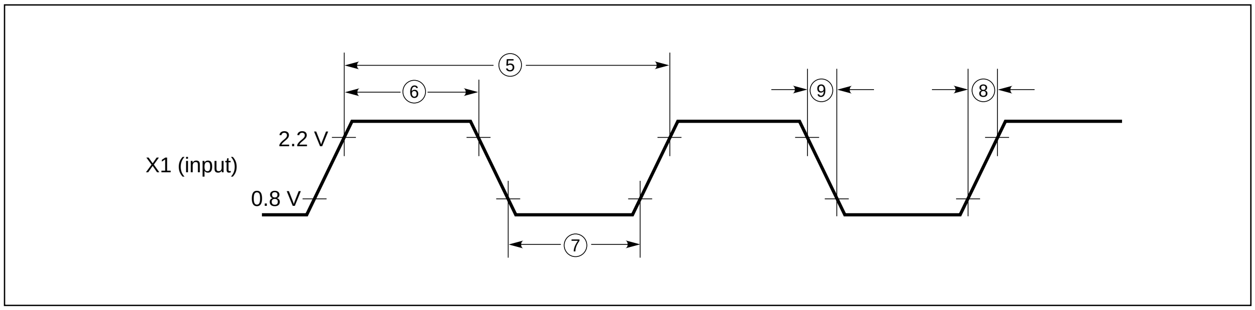
Manufacturer	Product name	Oscillation frequency fxx (MHz)	Recommended circuit constants		Oscillating voltage range		Oscillation settling time (MAX.) T_{OST} (ms)
			C1 (pF)	C2 (pF)	MIN. (V)	MAX. (V)	
Murata Mfg. Co., Ltd	CSA5.00MG	5.00	30	30	4.5	5.5	0.102
	CST5.00MGW	5.00	Built-in	Built-in	4.5	5.5	0.102
	CSA4.00MG	4.00	30	30	4.5	5.5	0.1
	CST4.00MGW	4.00	Built-in	Built-in	4.5	5.5	0.1
	CSA3.20MG	3.20	30	30	2.7	3.3	0.102
					4.5	5.5	0.102
	CST3.20MGW	3.20	Built-in	Built-in	2.7	3.3	0.102
					4.5	5.5	0.102
	CSA2.00MG040	2.00	100	100	2.7	3.3	0.498
					4.5	5.5	0.498
	CST2.00MG040	2.00	Built-in	Built-in	2.7	3.3	0.498
					4.5	5.5	0.498
TDK	CCR5.0MC3	5.00	Built-in	Built-in	4.5	5.5	0.28
	FCR5.0MC5	5.00	Built-in	Built-in	4.5	5.5	0.22
	CCR4.0MC3	4.00	Built-in	Built-in	4.5	5.5	0.3
	FCR4.0MC5	4.00	Built-in	Built-in	4.5	5.5	0.22
	CCR3.2MC3	3.20	Built-in	Built-in	2.7	5.5	0.38
	CCR2.0MC33	2.00	Built-in	Built-in	2.7	5.5	0.36

(b) External clock input



(1) Clock input timing

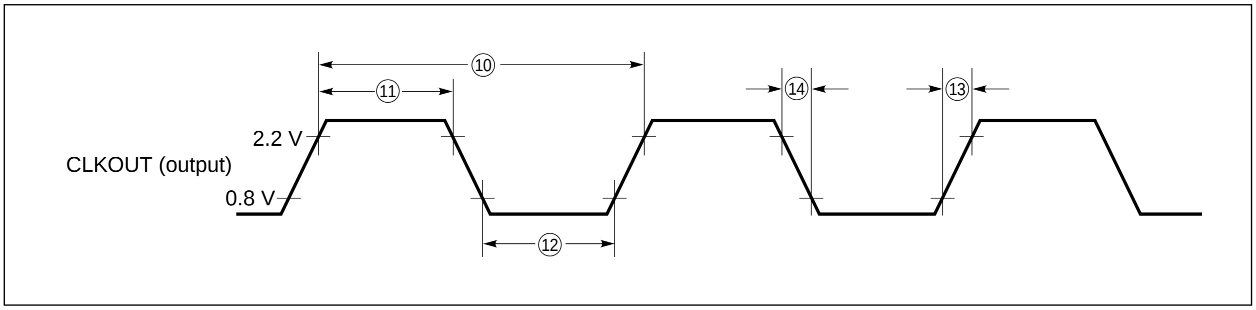
Parameter	Symbol		Conditions	MIN.	MAX.	Unit
External clock cycle	⑤	t_{CYX}	Direct mode	20		ns
			PLL mode	200	500	ns
External clock high-level width	⑥	t_{XKH}	Direct mode	7		ns
			PLL mode	85		ns
External clock low-level width	⑦	t_{XKL}	Direct mode	7		ns
			PLL mode	85		ns
External clock rise time	⑧	t_{XKR}	Direct mode		3	ns
			PLL mode		15	ns
External clock fall time	⑨	t_{XKF}	Direct mode		3	ns
			PLL mode		15	ns



(2) CLKOUT output timing

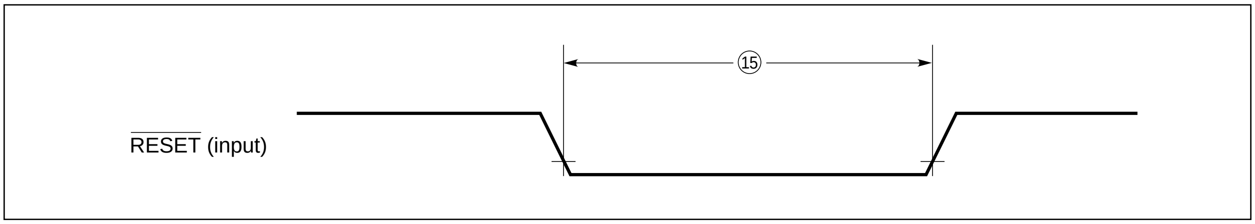
Parameter	Symbol		Conditions	MIN.	MAX.	Unit
CLKOUT cycle	⑩	t_{CYK}		40	100	ns
CLKOUT high-level width	⑪	t_{KKH}		0.5T - 3		ns
CLKOUT low-level width	⑫	t_{KKL}		0.5T - 3		ns
CLKOUT rise time (0.8 V → 2.2 V)	⑬	t_{KR}			5	ns
CLKOUT fall time (2.2 V → 0.8 V)	⑭	t_{KF}			5	ns

Remark T: t_{CYK}



(3) Reset input timing

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Reset input width	⑮	t_{WRL}	Power-on reset	10		ms
			STOP mode release	10		ms
			System reset	30		t_{CYK}



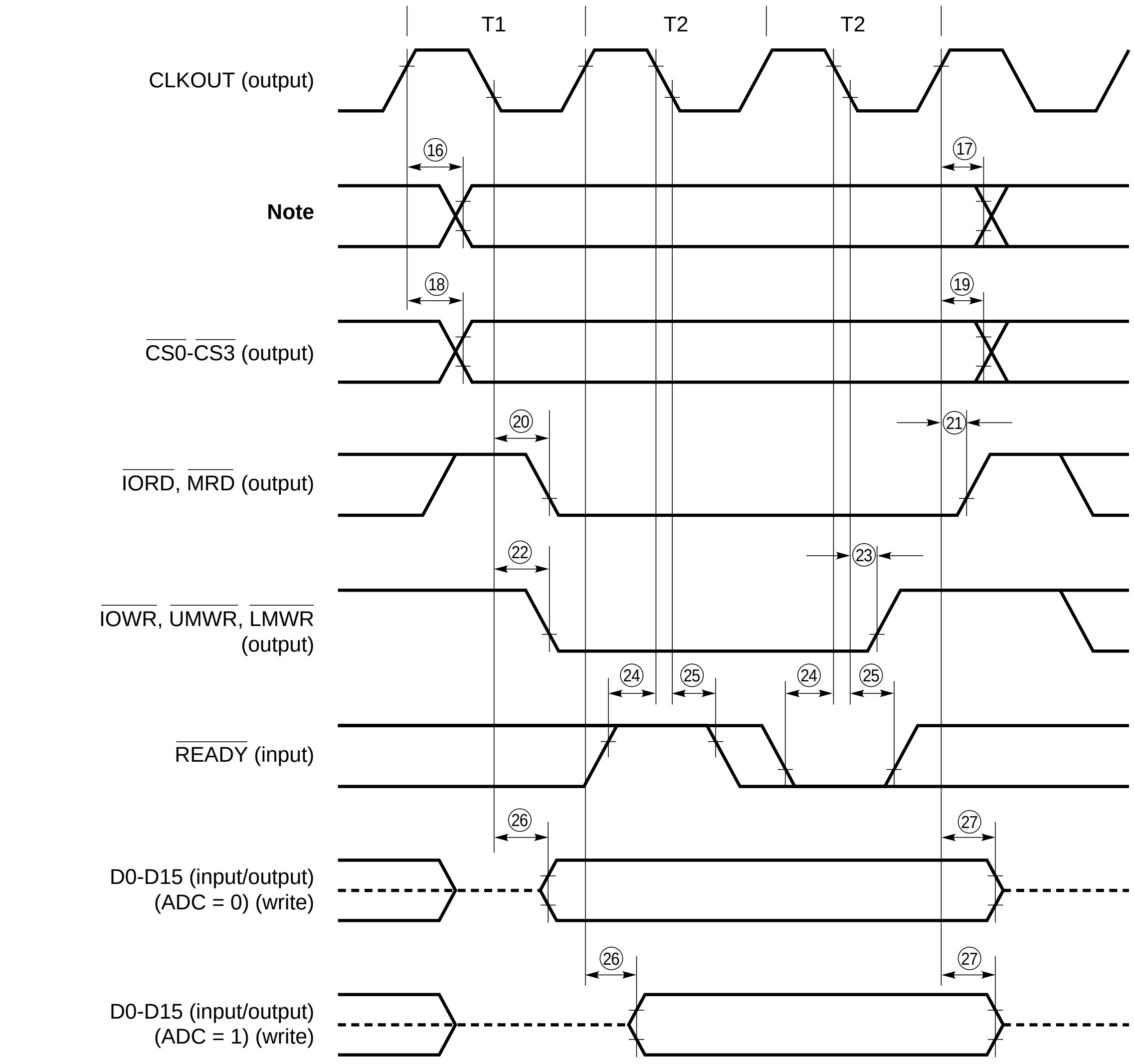
[MEMO]

(4) SRAM, ROM, and I/O access timing

(a) Access timing (1/2)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Address output delay (relative to CLKOUT↑)	⑯	t_{DKA}		2	15	ns
Address output hold time (relative to CLKOUT↑)	⑰	t_{HKA}		2	15	ns
$\overline{\text{CSn}}$ output delay (relative to CLKOUT↑)	⑱	t_{DKCS}		2	15	ns
$\overline{\text{CSn}}$ output hold time (relative to CLKOUT↑)	⑲	t_{HKCS}		2	15	ns
$\overline{\text{RD}}$ output delay (relative to CLKOUT↓)	⑳	t_{DKRD}		2	15	ns
$\overline{\text{RD}}$ output hold time (relative to CLKOUT↑)	㉑	t_{HKRD}		2	15	ns
$\overline{\text{WR}}$ output delay (relative to CLKOUT↓)	㉒	t_{DKWR}		1	12	ns
$\overline{\text{WR}}$ output hold time (relative to CLKOUT↓)	㉓	t_{HKWR}		1	12	ns
$\overline{\text{READY}}$ setup time (relative to CLKOUT↓)	㉔	t_{SRYK}		6		ns
$\overline{\text{READY}}$ hold time (relative to CLKOUT↓)	㉕	t_{HKRY}		6		ns
Data output delay (from float, relative to CLKOUT)	㉖	t_{LZKDT}		2	15	ns
Data output hold time (to float, relative to CLKOUT↑)	㉗	t_{HZKDT}		2	15	ns

(a) Access timing (2/2)



Note A0-A23 (output), UBE (output), BLOCK (output)

Remark Broken lines indicate high impedance.

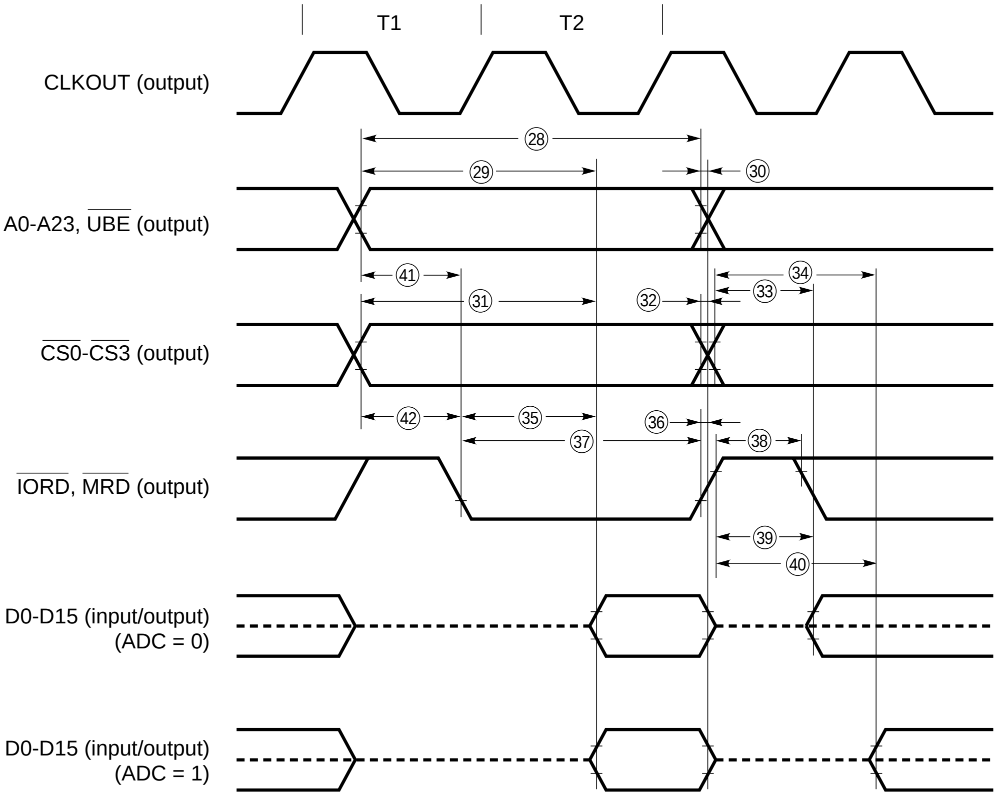
(b) Read timing (1/2)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Read cycle time	㉘	t_{RC}		(n + 2)T - 10		ns
Address access time	㉙	t_{AA}			(n + 2)T - 25	ns
Hold time from address to data input	㉚	t_{ADH}		0		ns
$\overline{\text{CSn}}$ access time	㉛	t_{CSA}			(n + 2)T - 25	ns
Hold time from $\overline{\text{CSn}}$ to data input	㉜	t_{CDH}		0		ns
Delay from $\overline{\text{CSn}}$↑ to write data output (ADC = 0)	㉝	t_{DCD0}		0.5T - 10		ns
Delay from $\overline{\text{CSn}}$↑ to write data output (ADC = 1)	㉞	t_{DCD1}		1T - 10		ns
$\overline{\text{RD}}$ access time	㉟	t_{RDA}			(n + 1.5)T - 25	ns
Hold time from $\overline{\text{RD}}$ to data input	㊱	t_{RDH}		0		ns
$\overline{\text{RD}}$ pulse width	㊲	t_{RDP}		(n + 1.5)T - 7		ns
$\overline{\text{RD}}$ high-level width	㊳	t_{RDRDH}		0.5T - 10		ns
Delay from $\overline{\text{RD}}$↑ to write data output (ADC = 0)	㊴	t_{DRD0}		0.5T - 10		ns
Delay from $\overline{\text{RD}}$↑ to write data output (ADC = 1)	㊵	t_{DRD1}		1T - 10		ns
Address valid time prior to $\overline{\text{RD}}$	㊶	t_{ARS}		0.5T - 7		ns
$\overline{\text{CSn}}$ valid time prior to $\overline{\text{RD}}$	㊷	t_{CRS}		0.5T - 7		ns

Remark T : t_{CYK}

n : Wait state count

(b) Read timing (2/2)


T1
T2
CLKOUT (output)
A0-A23, $\overline{\text{UBE}}$ (output)
$\overline{\text{CS0}}$-$\overline{\text{CS3}}$ (output)
$\overline{\text{IORD}}$, $\overline{\text{MRD}}$ (output)
D0-D15 (input/output) (ADC = 0)
D0-D15 (input/output) (ADC = 1)
28
29
30
41
31
32
34
33
42
35
36
37
38
39
40

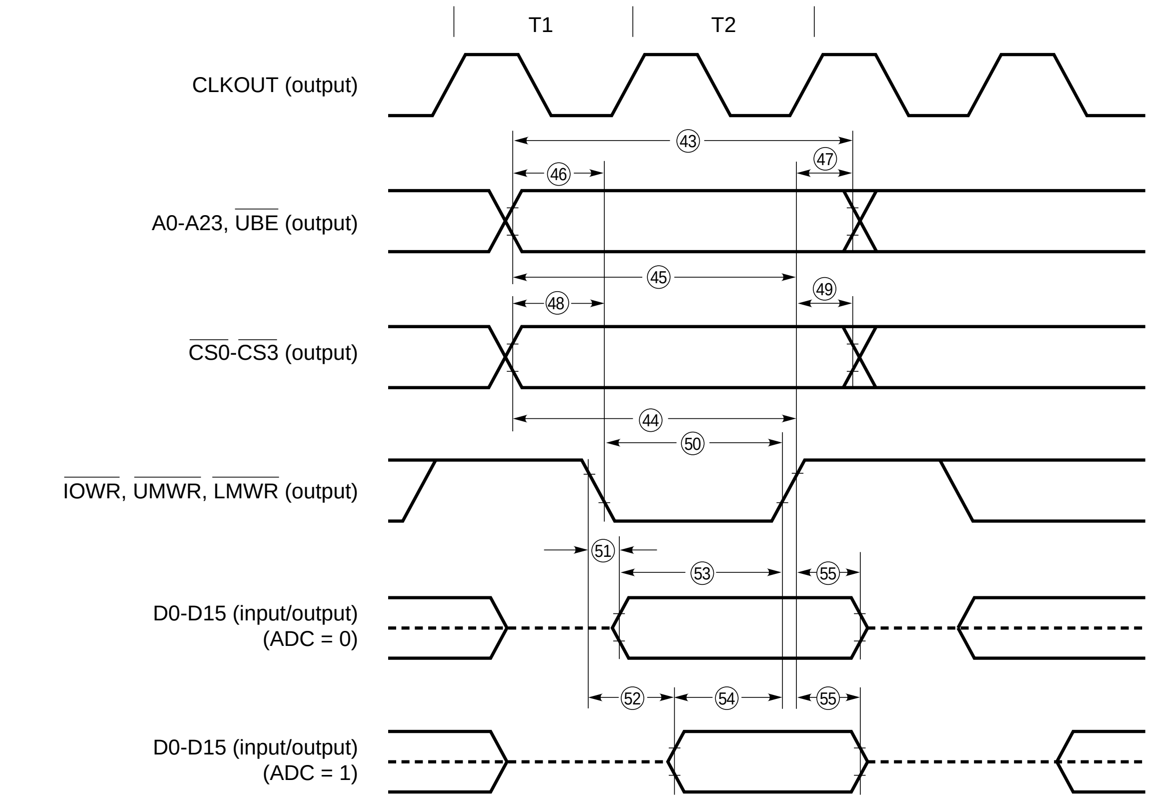
Remark Broken lines indicate high impedance.

(c) Write timing (1/2)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Write cycle time	㊸	t_{WC}		(n + 2)T - 10		ns
$\overline{\text{CSn}}$ setup time (relative to $\overline{\text{WR}}$↑)	㊹	t_{CW}		(n + 1.5)T - 10		ns
Address setup time (relative to $\overline{\text{WR}}$↑)	㊺	t_{AW}		(n + 1.5)T - 10		ns
Address valid time prior to $\overline{\text{WR}}$	㊻	t_{AWS}		0.5T - 7		ns
Address valid time after $\overline{\text{WR}}$	㊼	t_{AWH}		0.5T - 10		ns
$\overline{\text{CSn}}$ valid time prior to $\overline{\text{WR}}$	㊽	t_{CWS}		0.5T - 7		ns
$\overline{\text{CSn}}$ valid time after $\overline{\text{WR}}$	㊾	t_{CWH}		0.5T - 10		ns
$\overline{\text{WR}}$ pulse width	㊿	t_{WRP}		(n + 1)T - 7		ns
Delay from $\overline{\text{WR}}$↓ to data output (ADC = 0)	(51)	t_{WDS0}		-10		ns
Delay from $\overline{\text{WR}}$↓ to data output (ADC = 1)	(52)	t_{WDS1}		0.5T - 10		ns
Data output valid time prior to $\overline{\text{WR}}$ (ADC = 0)	(53)	t_{DWS0}		(n + 1)T - 7		ns
Data output valid time prior to $\overline{\text{WR}}$ (ADC = 1)	(54)	t_{DWS1}		(n + 0.5)T - 7		ns
Data output valid time after $\overline{\text{WR}}$	(55)	t_{DWH}		0.5T - 10		ns

Remark T : t_{CYK}
n : Wait state count

(c) Write timing (2/2)


T1
T2
CLKOUT (output)
A0-A23, $\overline{\text{UBE}}$ (output)
$\overline{\text{CS0}}$-$\overline{\text{CS3}}$ (output)
$\overline{\text{IOWR}}$, $\overline{\text{UMWR}}$, $\overline{\text{LMWR}}$ (output)
D0-D15 (input/output) (ADC = 0)
D0-D15 (input/output) (ADC = 1)
43
47
46
45
49
48
44
50
51
53
55
52
54
55

Remark Broken lines indicate high impedance.

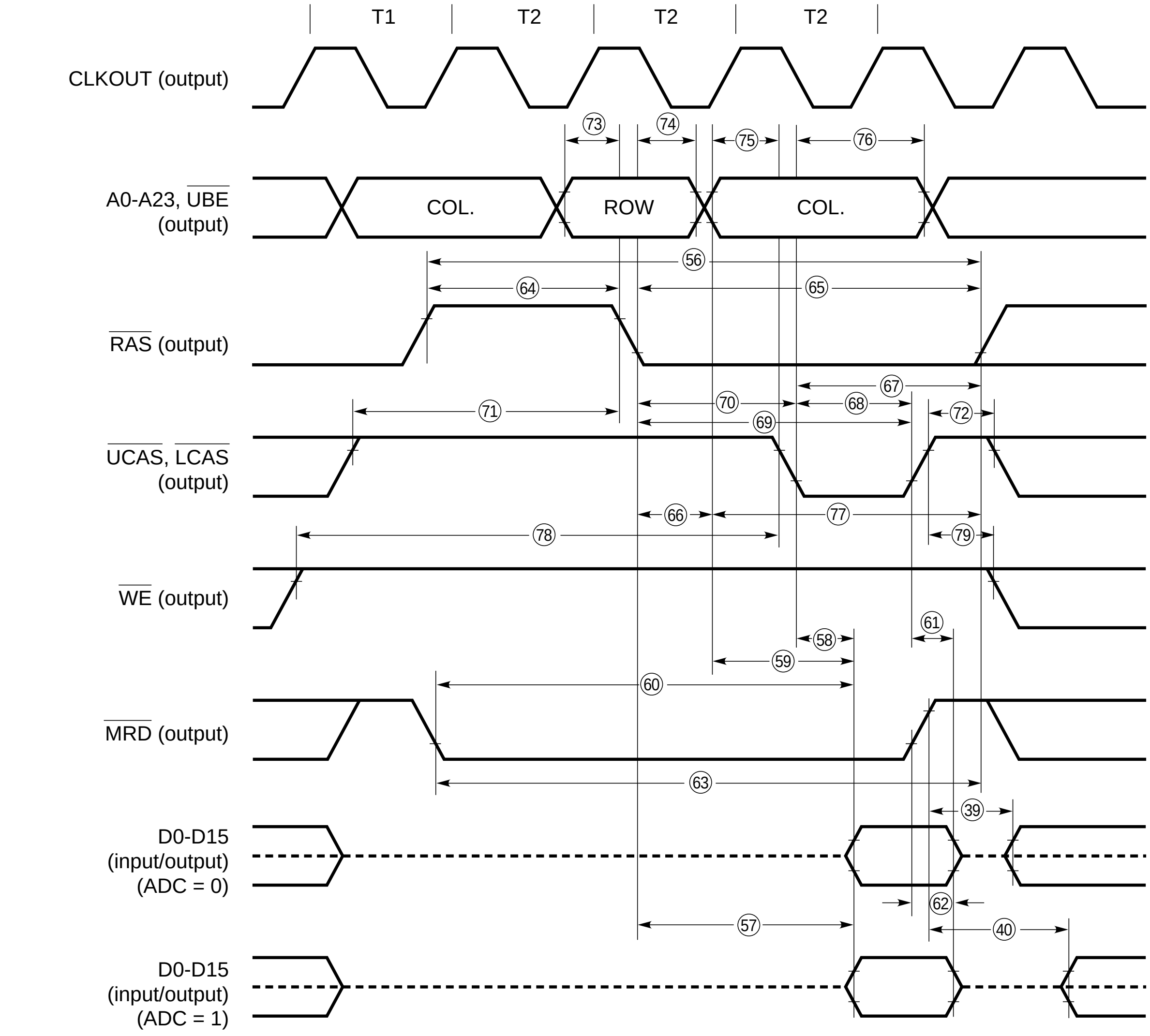
(5) DRAM access timing (when DRAM is directly connected)

(a) Read timing (normal access: off-page) (1/2)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Delay from $\overline{RD}$↑ to write data output (ADC = 0)	39	t_{DRD0}		0.5T - 10		ns
Delay from $\overline{RD}$↑ to write data output (ADC = 1)	40	t_{DRD1}		1T - 10		ns
Read/write cycle time	56	t_{RC}		(w + 4)T - 10		ns
$\overline{RAS}$ access time	57	t_{RAC}			(w + 2)T - 20	ns
$\overline{CAS}$ access time	58	t_{CAC}			(w + 1)T - 20	ns
Access time from column address	59	t_{AA}			(w + 1)T - 3	ns
Output enable access time	60	t_{OEA}			1.5T - 20	ns
Output buffer turn-off delay (relative to $\overline{CAS}$)	61	t_{OFF}		0		ns
Output buffer turn-off delay (relative to $\overline{MRD}$)	62	t_{OEZ}		0		ns
$\overline{RD}$ setup time (relative to $\overline{RAS}$↑)	63	t_{OES}		1.5T		ns
$\overline{RAS}$ precharge time	64	t_{RP}		1.5T - 10		ns
$\overline{RAS}$ pulse width	65	t_{RAS}		(w + 2.5)T - 20		ns
$\overline{RAS}$ column address delay	66	t_{RAD}		0.5T - 3	0.5T + 7	ns
$\overline{RAS}$ hold width (read)	67	t_{RSH}		(w + 1.5)T - 20		ns
$\overline{CAS}$ pulse width (read)	68	t_{CAS}		(w + 1)T - 15		ns
$\overline{CAS}$ hold width	69	t_{CSH}		(w + 2)T - 15		ns
$\overline{RAS}$-$\overline{CAS}$ delay (read)	70	t_{RCD}		1T - 15		ns
$\overline{RAS}$-$\overline{CAS}$ precharge time	71	t_{CRP}		1.5T		ns
$\overline{CAS}$ precharge time	72	t_{CP}		0.5T - 10		ns
Row address setup time	73	t_{ASR}		0.5T - 15		ns
Row address hold time	74	t_{RAH}		0.5T - 7		ns
Column address setup time (read)	75	t_{ASC}		0.5T - 15		ns
Column address hold time (read)	76	t_{CAH}		(w + 1)T - 15		ns
Column address read time relative to $\overline{RAS}$	77	t_{RAL}		(w + 1.5)T		ns
Read command setup time	78	t_{RCS}		0.5T		ns
Read command hold time	79	t_{RCH}		0.5T - 15		ns

Remark T : t_{CYK}
w : Wait state count - 2

(a) Read timing (normal access: off-page) (2/2)


T1
T2
T2
T2
CLKOUT (output)
73
74
75
76
A0-A23, $\overline{\text{UBE}}$ (output)
COL.
ROW
COL.
56
64
65
$\overline{\text{RAS}}$ (output)
67
70
68
71
72
69
$\overline{\text{UCAS}}$, $\overline{\text{LCAS}}$ (output)
66
77
78
79
$\overline{\text{WE}}$ (output)
61
58
59
60
$\overline{\text{MRD}}$ (output)
63
39
D0-D15 (input/output) (ADC = 0)
62
57
40
D0-D15 (input/output) (ADC = 1)

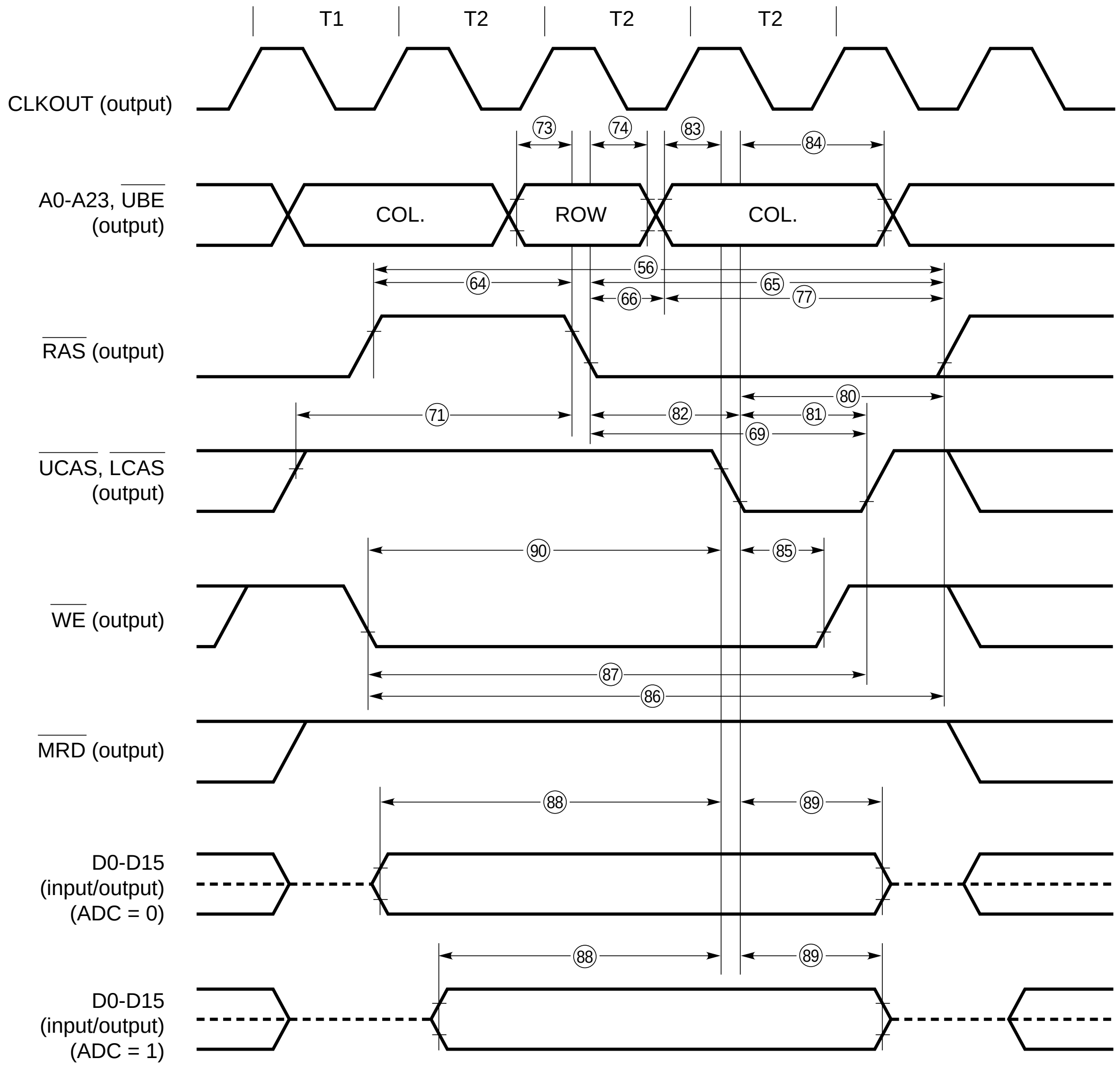
Remark Broken lines indicate high impedance.

(b) Write timing (normal access: off-page) (1/2)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Read/write cycle time	56	t_{RC}		(w + 4)T - 10		ns
$\overline{\text{RAS}}$ precharge time	64	t_{RP}		1.5T - 10		ns
$\overline{\text{RAS}}$ pulse width	65	t_{RAS}		(w + 2.5)T - 20		ns
$\overline{\text{RAS}}$ column address delay	66	t_{RAD}		0.5T - 3	0.5T + 7	ns
$\overline{\text{CAS}}$ hold width	69	t_{CSH}		(w + 2)T - 15		ns
$\overline{\text{RAS}}$-$\overline{\text{CAS}}$ precharge time	71	t_{CRP}		1.5T		ns
Row address setup time	73	t_{ASR}		0.5T - 15		ns
Row address hold time	74	t_{RAH}		0.5T - 7		ns
Column address read time relative to $\overline{\text{RAS}}$	77	t_{RAL}		(w + 1.5)T		ns
$\overline{\text{RAS}}$ hold width (write)	80	t_{RSH}		1.5T - 20		ns
$\overline{\text{CAS}}$ pulse width (write)	81	t_{CAS}		1T - 15		ns
$\overline{\text{RAS}}$-$\overline{\text{CAS}}$ delay (write)	82	t_{RCD}		(w + 1)T - 15		ns
Column address setup time (write)	83	t_{ASC}		(w + 0.5)T - 15		ns
Column address hold time (write)	84	t_{CAH}		1T - 15		ns
Write command hold time	85	t_{WCH}		0.5T - 10		ns
Write command read time relative to $\overline{\text{RAS}}$	86	t_{RWL}		1.5T		ns
Write command read time relative to $\overline{\text{CAS}}$	87	t_{CWL}		1T		ns
Data setup time (relative to $\overline{\text{CAS}}$↓)	88	t_{DS}		0.5T - 15		ns
Data hold time (relative to $\overline{\text{CAS}}$↓)	89	t_{DH}		1T - 20	1T + 10	ns
Write command setup time	90	t_{WCS}		0.5T - 15		ns

Remark T : t_{CYK}
w : Wait state count - 2

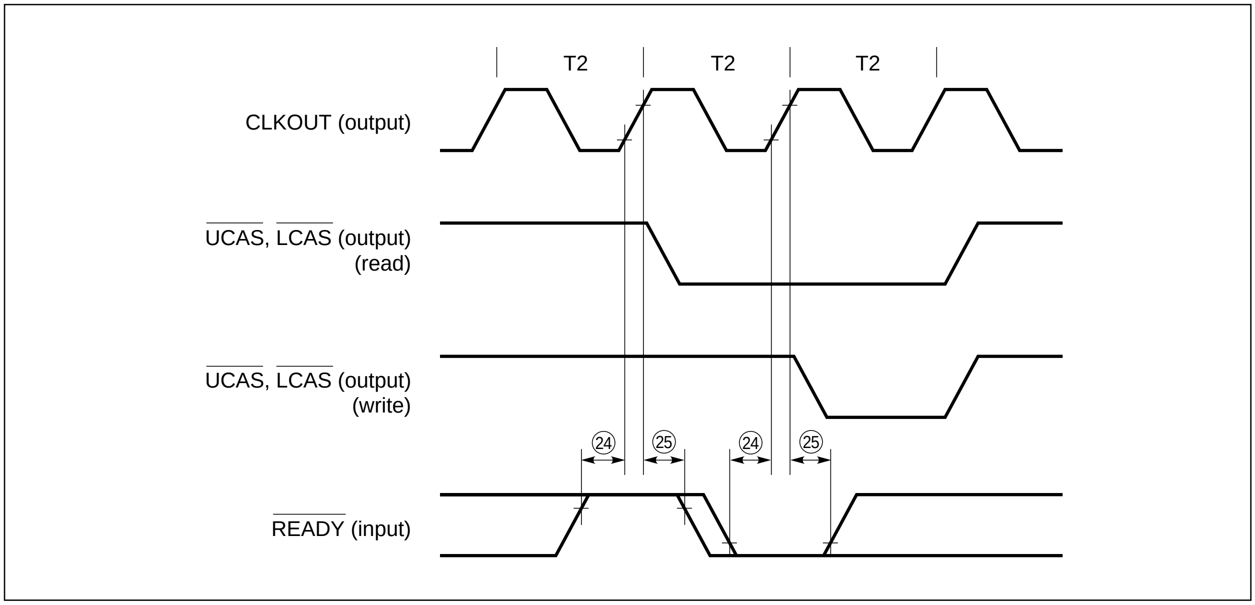
(b) Write timing (normal access: off-page) (2/2)


T1
T2
T2
T2
CLKOUT (output)
73
74
83
84
A0-A23, $\overline{\text{UBE}}$ (output)
COL.
ROW
COL.
56
64
65
66
77
$\overline{\text{RAS}}$ (output)
80
71
82
81
69
$\overline{\text{UCAS}}$, $\overline{\text{LCAS}}$ (output)
90
85
$\overline{\text{WE}}$ (output)
87
86
$\overline{\text{MRD}}$ (output)
88
89
D0-D15 (input/output) (ADC = 0)
88
89
D0-D15 (input/output) (ADC = 1)

Remark Broken lines indicate high impedance.

(c) $\overline{\text{READY}}$ input timing (normal access)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
$\overline{\text{READY}}$ setup time (relative to CLKOUT↑)	㉔	t_{SRYK}		6		ns
$\overline{\text{READY}}$ hold time (relative to CLKOUT↑)	㉕	t_{HKRY}		6		ns



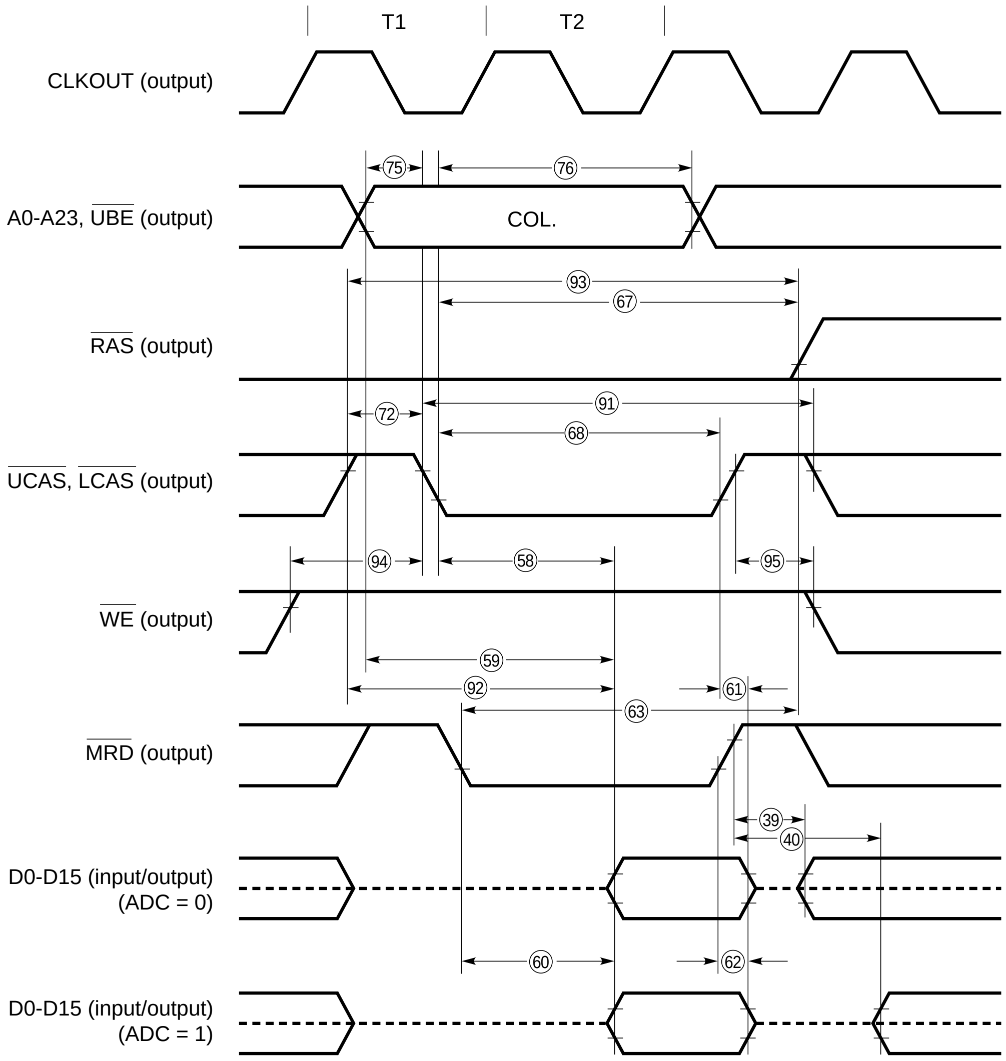
[MEMO]

(d) Read timing (high-speed page access: on-page) (1/2)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Delay from $\overline{\text{RD}}$↑ to write data output (ADC = 0)	39	t_{DRD0}		0.5T - 10		ns
Delay from $\overline{\text{RD}}$↑ to write data output (ADC = 1)	40	t_{DRD1}		1T - 10		ns
$\overline{\text{CAS}}$ access time	58	t_{CAC}			(w + 1)T - 20	ns
Access time from column address	59	t_{AA}			(w + 1)T - 3	ns
Output enable access time	60	t_{OEA}			1.5T - 20	ns
Output buffer turn-off delay (relative to $\overline{\text{CAS}}$)	61	t_{OFF}		0		ns
Output buffer turn-off delay (relative to $\overline{\text{MRD}}$)	62	t_{OEZ}		0		ns
$\overline{\text{RD}}$ setup time (relative to $\overline{\text{RAS}}$↑)	63	t_{OES}		1.5T		ns
$\overline{\text{RAS}}$ hold width (read)	67	t_{RSH}		(w + 1.5)T - 20		ns
$\overline{\text{CAS}}$ pulse width (read)	68	t_{CAS}		(w + 1)T - 15		ns
$\overline{\text{CAS}}$ precharge time	72	t_{CP}		0.5T - 10		ns
Column address setup time (read)	75	t_{ASC}		0.5T - 15		ns
Column address hold time (read)	76	t_{CAH}		(w + 1)T - 15		ns
Cycle time in high-speed page mode	91	t_{PC}		1.5T - 10		ns
Access time from $\overline{\text{CAS}}$ precharge	92	t_{ACP}			2T - 20	ns
$\overline{\text{RAS}}$ hold time relative to $\overline{\text{CAS}}$ precharge	93	t_{RHCP}		2T		ns
Read command setup time	94	t_{RCS}		0.5T		ns
Read command hold time	95	t_{RCH}		0.5T - 15		ns

Remark T : t_{CYK}
w : 0

(d) Read timing (high-speed page access: on-page) (2/2)


T1
T2
CLKOUT (output)
A0-A23, $\overline{\text{UBE}}$ (output)
COL.
$\overline{\text{RAS}}$ (output)
$\overline{\text{UCAS}}$, $\overline{\text{LCAS}}$ (output)
$\overline{\text{WE}}$ (output)
$\overline{\text{MRD}}$ (output)
D0-D15 (input/output) (ADC = 0)
D0-D15 (input/output) (ADC = 1)
75
76
93
67
91
72
68
94
58
95
59
92
61
63
39
40
60
62

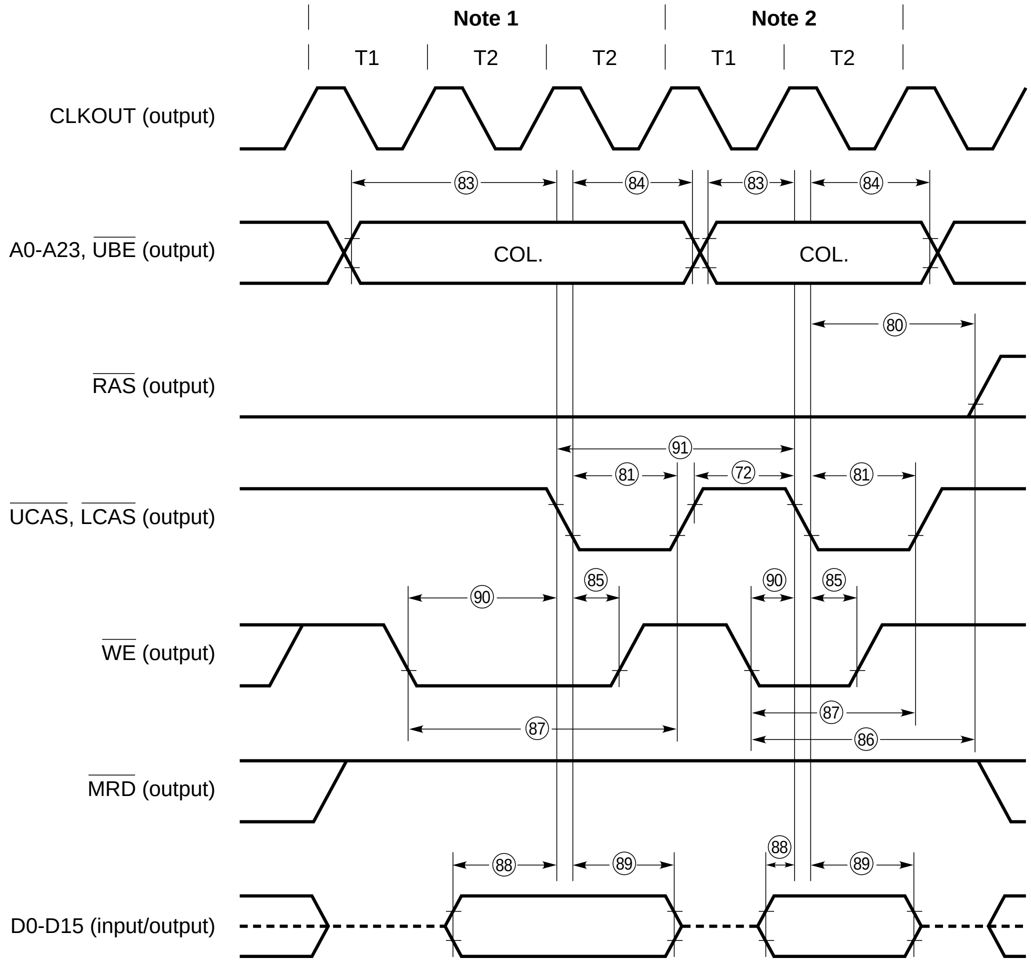
Remark Broken lines indicate high impedance.

(e) Write timing (high-speed page access: on-page) (1/2)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
$\overline{\text{CAS}}$ precharge time	72	t_{CP}		0.5T - 10		ns
$\overline{\text{RAS}}$ hold width (write)	80	t_{RSH}		1.5T - 20		ns
$\overline{\text{CAS}}$ pulse width (write)	81	t_{CAS}		1T - 15		ns
Column address setup time (write)	83	t_{ASC}		(w + 0.5)T - 15		ns
Column address hold time (write)	84	t_{CAH}		1T - 15		ns
Write command hold time	85	t_{WCH}		0.5T - 10		ns
Write command read time relative to $\overline{\text{RAS}}$	86	t_{RWL}		1.5T		ns
Write command read time relative to $\overline{\text{CAS}}$	87	t_{CWL}		1T		ns
Data setup time (relative to $\overline{\text{CAS}}$↓)	88	t_{DS}		0.5T - 15		ns
Data hold time (relative to $\overline{\text{CAS}}$↓)	89	t_{DH}		1T - 20	1T + 10	ns
Write command setup time	90	t_{WCS}		0.5T - 15		ns
Cycle time in high-speed page mode	91	t_{PC}		1.5T - 10		ns

Remark T : t_{CYK}
w : 0

(e) Write timing (high-speed page access: on-page) (2/2)



Notes 1. When ADC = 1 and other than DRAM access was performed in the previous cycle

2. Other than the above

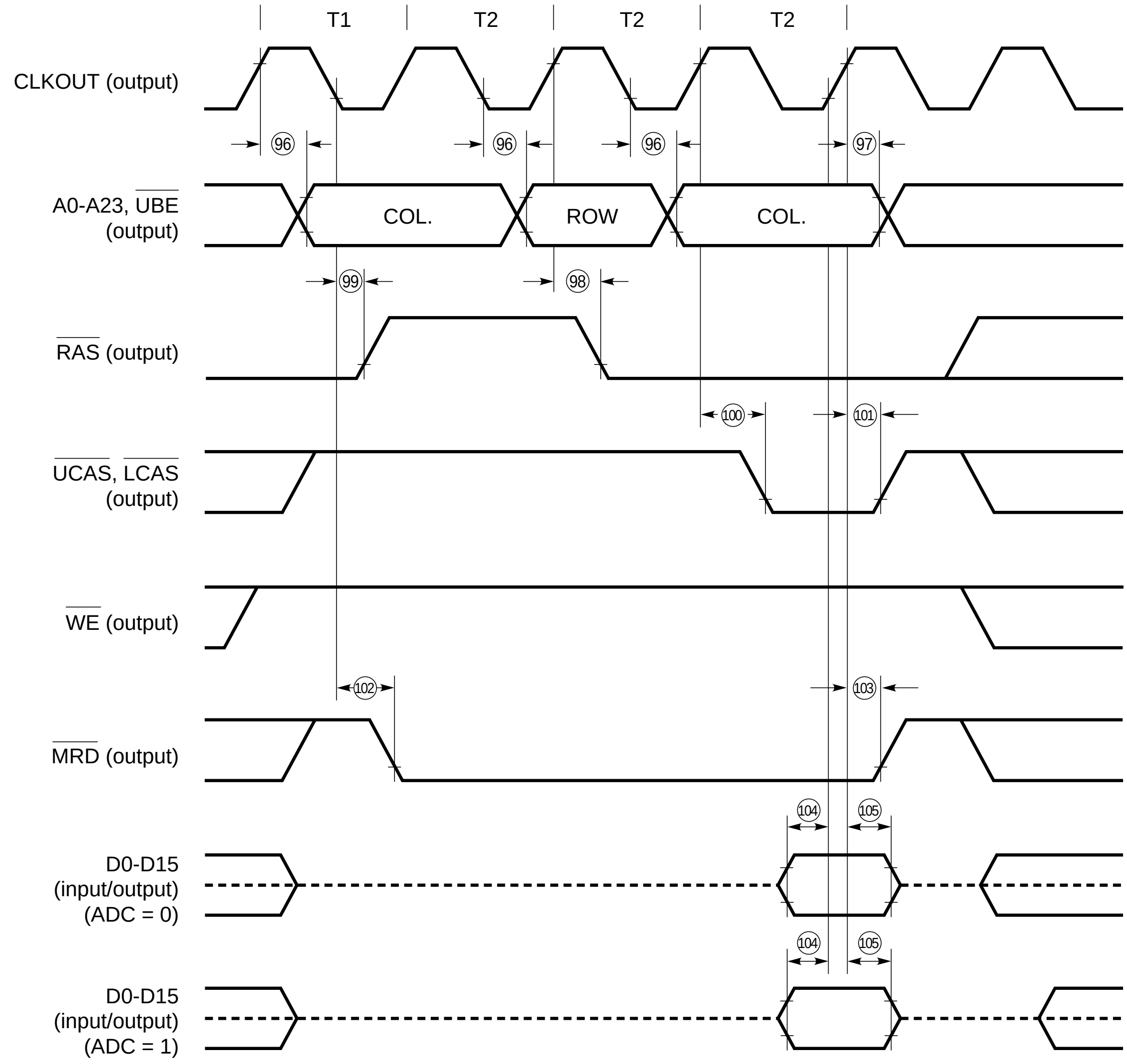
Remark Broken lines indicate high impedance.

(6) DRAM access timing (when a control circuit is configured using a gate array or other devices)

(a) Read timing (normal access: off-page) (1/2)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Address output delay (relative to CLKOUT)	96	t_{DKA}		2	15	ns
Address output hold time (relative to CLKOUT↑)	97	t_{HKA}		2	15	ns
$\overline{\text{RAS}}$ output delay (relative to CLKOUT↑)	98	t_{DKRAS}		1	12	ns
$\overline{\text{RAS}}$ output hold time (relative to CLKOUT↓)	99	t_{HKRAS}		1	12	ns
$\overline{\text{CAS}}$ output delay (relative to CLKOUT↑)	100	t_{DKCAS}		1	12	ns
$\overline{\text{CAS}}$ output hold time (relative to CLKOUT↑)	101	t_{HKCAS}		1	12	ns
$\overline{\text{MRD}}$ output delay (relative to CLKOUT↓)	102	t_{DKRD}		2	15	ns
$\overline{\text{MRD}}$ output hold time (relative to CLKOUT↑)	103	t_{HKRD}		2	15	ns
Data input setup time (relative to CLKOUT↑)	104	t_{SDK}		6		ns
Data input hold time (relative to CLKOUT↑)	105	t_{HKD}		6		ns

(a) Read timing (normal access: off-page) (2/2)

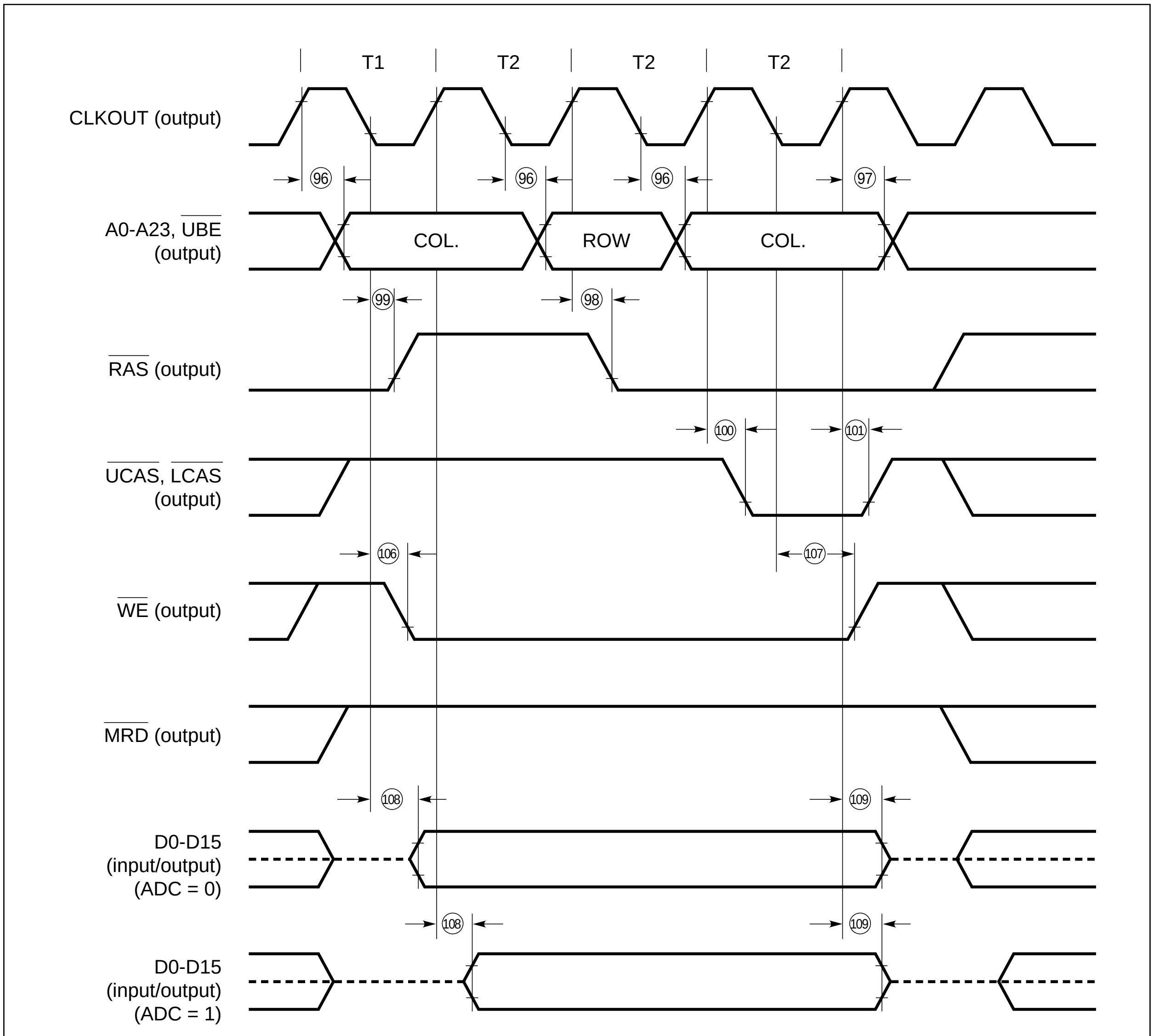


Remark Broken lines indicate high impedance.

(b) Write timing (normal access: off-page) (1/2)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Address output delay (relative to CLKOUT↑)	96	t_{DKA}		2	15	ns
Address output hold time (relative to CLKOUT↑)	97	t_{HKA}		2	15	ns
$\overline{RAS}$ output delay (relative to CLKOUT↑)	98	t_{DKRAS}		1	12	ns
$\overline{RAS}$ output hold time (relative to CLKOUT↓)	99	t_{HKRAS}		1	12	ns
$\overline{CAS}$ output delay (relative to CLKOUT↑)	100	t_{DKCAS}		1	12	ns
$\overline{CAS}$ output hold time (relative to CLKOUT↑)	101	t_{HKCAS}		1	12	ns
$\overline{WE}$ output delay (relative to CLKOUT↓)	106	t_{DKWE}		1	12	ns
$\overline{WE}$ output hold time (relative to CLKOUT↓)	107	t_{HKWE}		1	12	ns
Data active delay (from float, relative to CLKOUT)	108	t_{LZKDT}		2	15	ns
Data inactive hold time (to float, relative to CLKOUT↑)	109	t_{HZKDT}		2	15	ns

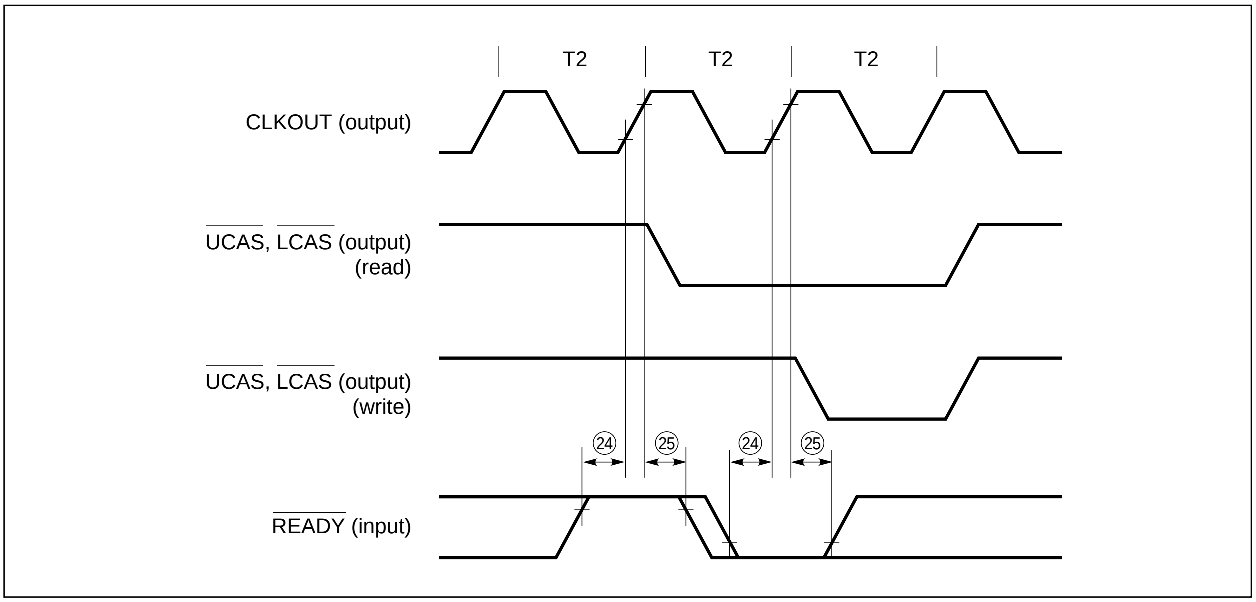
(b) Write timing (normal access: off-page) (2/2)



Remark Broken lines indicate high impedance.

(c) $\overline{\text{READY}}$ input timing (normal access)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
$\overline{\text{READY}}$ setup time (relative to CLKOUT↑)	㉔	t_{SRYK}		6		ns
$\overline{\text{READY}}$ hold time (relative to CLKOUT↑)	㉕	t_{HKRY}		6		ns

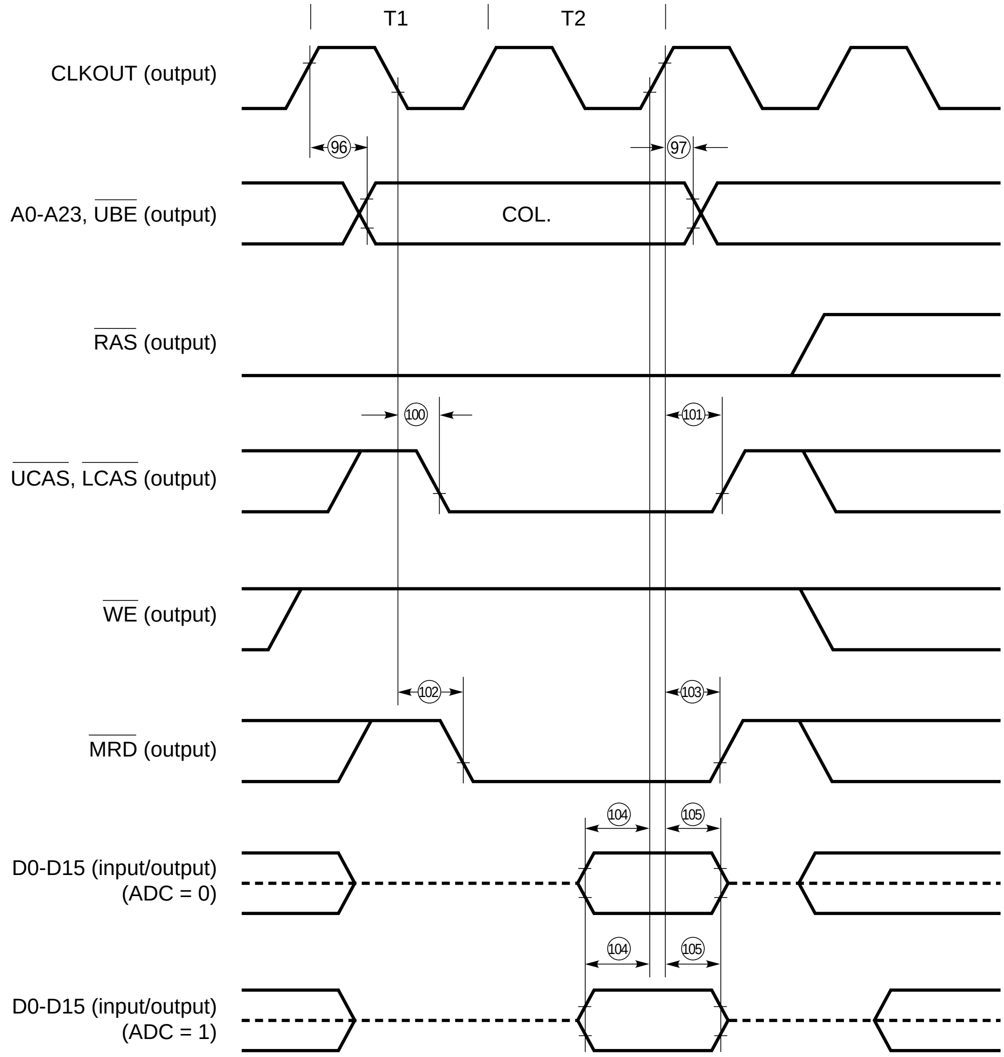


[MEMO]

(d) Read timing (high-speed page access: on-page) (1/2)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Address output delay (relative to CLKOUT)	96	t_{DKA}		2	15	ns
Address output hold time (relative to CLKOUT↑)	97	t_{HKA}		2	15	ns
$\overline{\text{CAS}}$ output delay (relative to CLKOUT↑)	100	t_{DKCAS}		1	12	ns
$\overline{\text{CAS}}$ output hold time (relative to CLKOUT↑)	101	t_{HKCAS}		1	12	ns
$\overline{\text{MRD}}$ output delay (relative to CLKOUT↓)	102	t_{DKRD}		2	15	ns
$\overline{\text{MRD}}$ output hold time (relative to CLKOUT↑)	103	t_{HKRD}		2	15	ns
Data input setup time (relative to CLKOUT↑)	104	t_{SDK}		6		ns
Data input hold time (relative to CLKOUT↑)	105	t_{HKD}		6		ns

(d) Read timing (high-speed page access: on-page) (2/2)

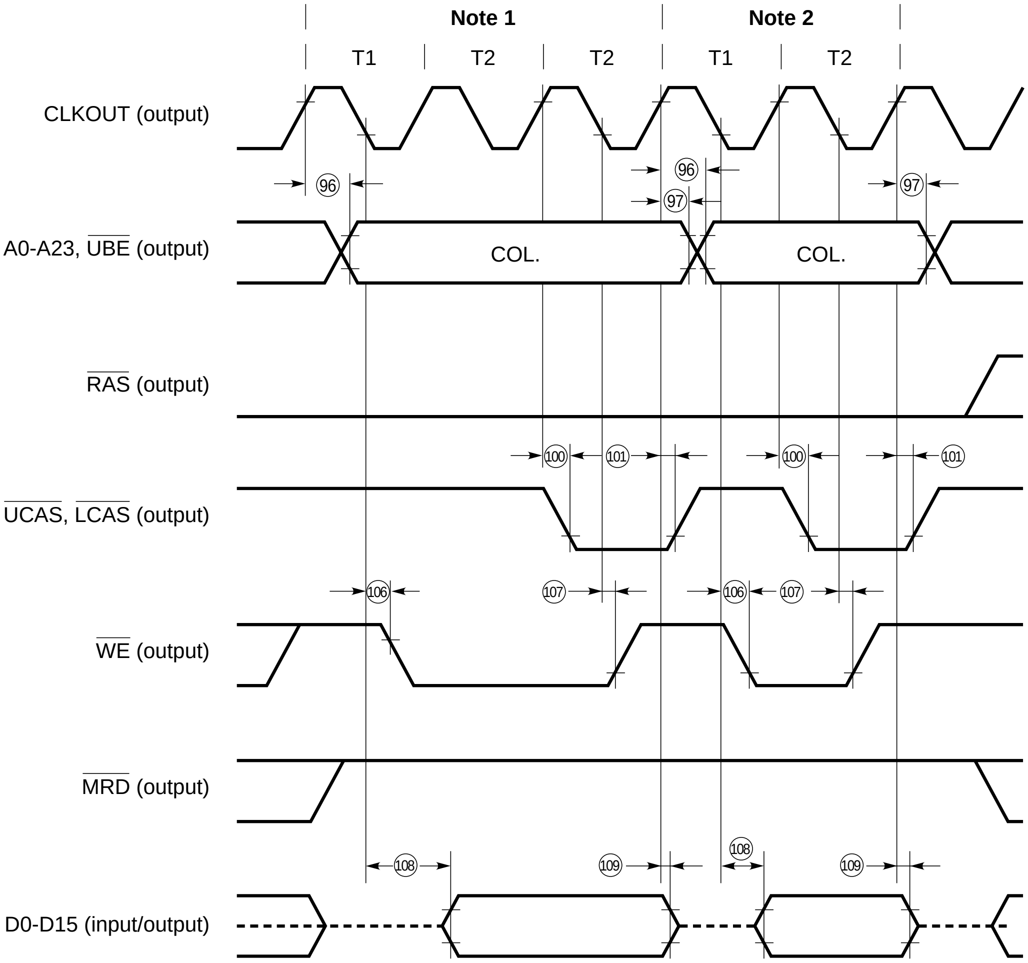


Remark Broken lines indicate high impedance.

(e) Write timing (high-speed page access: on-page) (1/2)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Address output delay (relative to CLKOUT↑)	96	t_{DKA}		2	15	ns
Address output hold time (relative to CLKOUT↑)	97	t_{HKA}		2	15	ns
$\overline{\text{CAS}}$ output delay (relative to CLKOUT↑)	100	t_{DKCAS}		1	12	ns
$\overline{\text{CAS}}$ output hold time (relative to CLKOUT↑)	101	t_{HKCAS}		1	12	ns
$\overline{\text{WE}}$ output delay (relative to CLKOUT↓)	106	t_{DKWE}		1	12	ns
$\overline{\text{WE}}$ output hold time (relative to CLKOUT↓)	107	t_{HKWE}		1	12	ns
Data active delay (from float, relative to CLKOUT)	108	t_{LZKDT}		2	15	ns
Data inactive hold time (to float, relative to CLKOUT↑)	109	t_{HZKDT}		2	15	ns

(e) Write timing (high-speed page access: on-page) (2/2)


Note 1
Note 2
T1
T2
T2
T1
T2
CLKOUT (output)
96
96
97
97
A0-A23, $\overline{\text{UBE}}$ (output)
COL.
COL.
$\overline{\text{RAS}}$ (output)
100
101
100
101
$\overline{\text{UCAS}}$, $\overline{\text{LCAS}}$ (output)
106
107
106
107
$\overline{\text{WE}}$ (output)
$\overline{\text{MRD}}$ (output)
108
109
108
109
D0-D15 (input/output)

Notes 1. When ADC = 1 and other than DRAM access was performed in the previous cycle

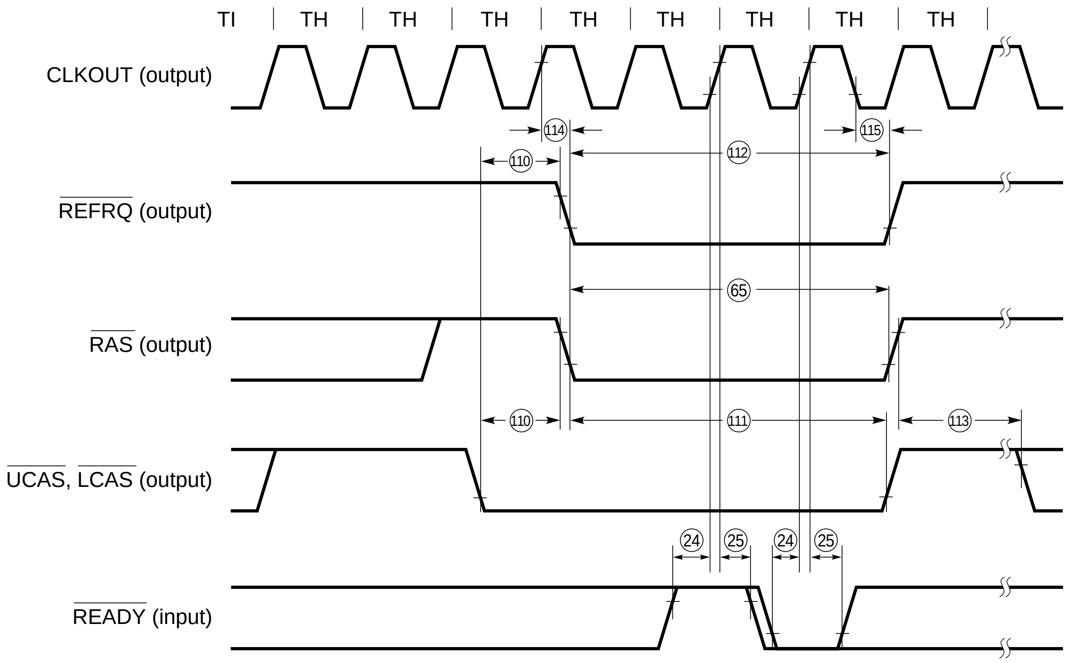
2. Other than the above

Remark Broken lines indicate high impedance.

(7) DRAM, CBR refresh timing

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
$\overline{\text{READY}}$ setup time (relative to CLKOUT↑)	24	t_{SRYK}		6		ns
$\overline{\text{READY}}$ hold time (relative to CLKOUT↑)	25	t_{HKRY}		6		ns
$\overline{\text{RAS}}$ pulse width	65	t_{RAS}		(w + 2.5)T - 20		ns
$\overline{\text{CAS}}$ setup time	110	t_{CSR}		1T - 20		ns
$\overline{\text{CAS}}$ hold time	111	t_{CHR}		(w + 2.5)T - 20		ns
Refresh pulse width	112	t_{REF}		(w + 2.5)T - 20		ns
$\overline{\text{RAS}}$ precharge to $\overline{\text{CAS}}$ hold time	113	t_{RPC}		4.5T - 20		ns
$\overline{\text{REFRQ}}$ active delay (relative to CLKOUT↑)	114	t_{DKREF}		1	12	ns
$\overline{\text{REFRQ}}$ inactive delay (relative to CLKOUT↓)	115	t_{HKREF}		1	12	ns

Remark T: t_{CYK}
w: Wait state count for CBR refresh

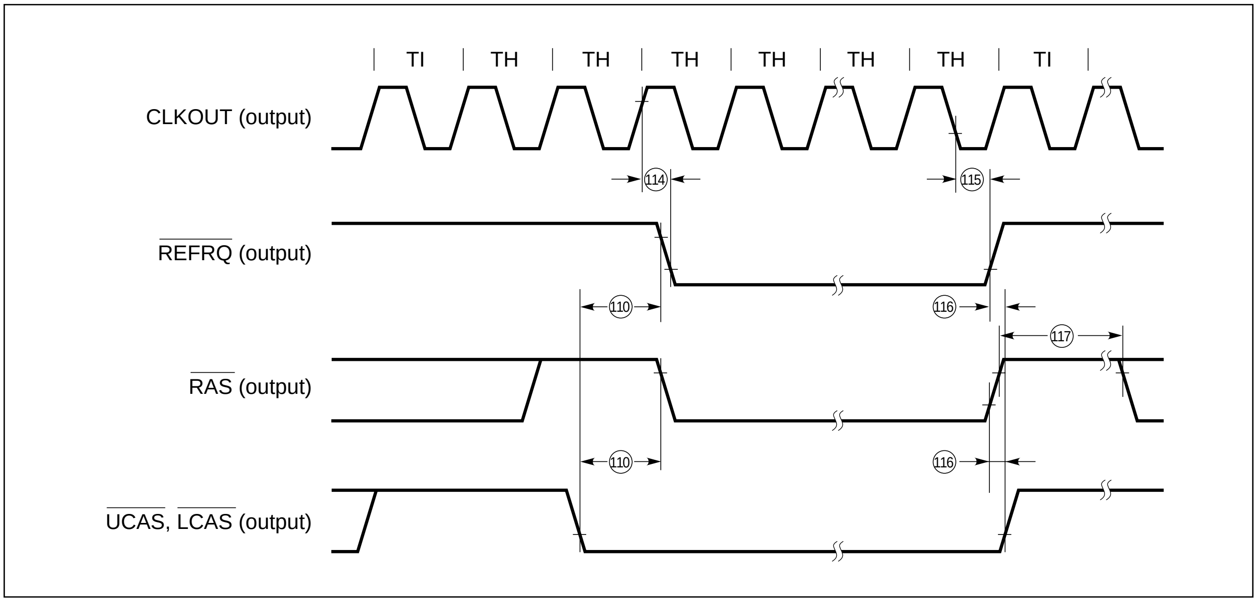


Remark In the above timing chart, w = 1 is assumed.

(8) DRAM, CBR self-refresh timing

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
$\overline{\text{CAS}}$ setup time	110	t_{CSR}		1T - 20		ns
$\overline{\text{REFRQ}}$ active delay (relative to CLKOUT↑)	114	t_{DKREF}		1	12	ns
$\overline{\text{REFRQ}}$ inactive delay (relative to CLKOUT↓)	115	t_{HKREF}		1	12	ns
$\overline{\text{CAS}}$ hold time	116	t_{CHS}		-10		ns
$\overline{\text{RAS}}$ precharge time	117	t_{RPS}		4.5T - 20		ns

Remark T: t_{CYK}

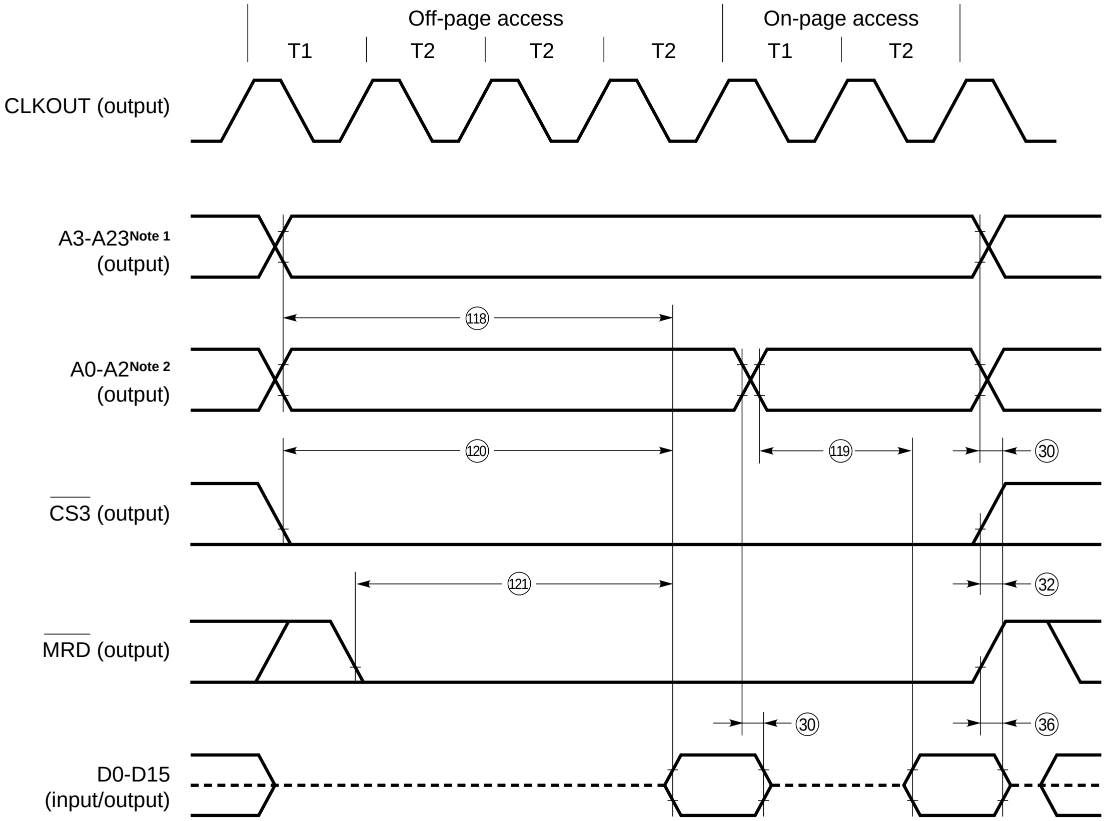

TI
TH
TH
TH
TH
TH
TH
TI
CLKOUT (output)
114
115
$\overline{\text{REFRQ}}$ (output)
110
116
117
$\overline{\text{RAS}}$ (output)
110
116
$\overline{\text{UCAS}}$, $\overline{\text{LCAS}}$ (output)

(9) Page-ROM access timing (1/2)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Hold time from address to data input	30	t_{ADH}		0		ns
Hold time from $\overline{CSn}$ to data input	32	t_{CDH}		0		ns
Hold time from $\overline{RD}$ to data input	36	t_{RDH}		0		ns
Off-page address access time	118	t_{OFPA}			$(n_{OFF} + 2)T - 25$	ns
On-page address access time	119	t_{ONPA}			$(n_{ON} + 2)T - 25$	ns
Off-page $\overline{CSn}$ access time	120	t_{OFCS}			$(n_{OFF} + 2)T - 25$	ns
Off-page $\overline{RD}$ access time	121	t_{OFRD}			$(n_{OFF} + 1.5)T - 25$	ns

Remark T : t_{CYK}
n_{OFF} : Wait state count for off-page access (n_{OFF} = 0-7)
n_{ON} : Wait state count for on-page access (n_{ON} = 0, 1)

(9) Page-ROM access timing (2/2)



Notes 1. The address pins to be used vary with the settings of bits MA5 to MA3 of the page-ROM configuration register (PRC).

MA5	MA4	MA3	Address
0	0	0	A3-A23
0	0	1	A4-A23
0	1	1	A5-A23
1	1	1	A6-A23

2. The address pins to be used vary with the settings of bits MA5 to MA3 of the page-ROM configuration register (PRC).

MA5	MA4	MA3	Address
0	0	0	A0-A2
0	0	1	A0-A3
0	1	1	A0-A4
1	1	1	A0-A5

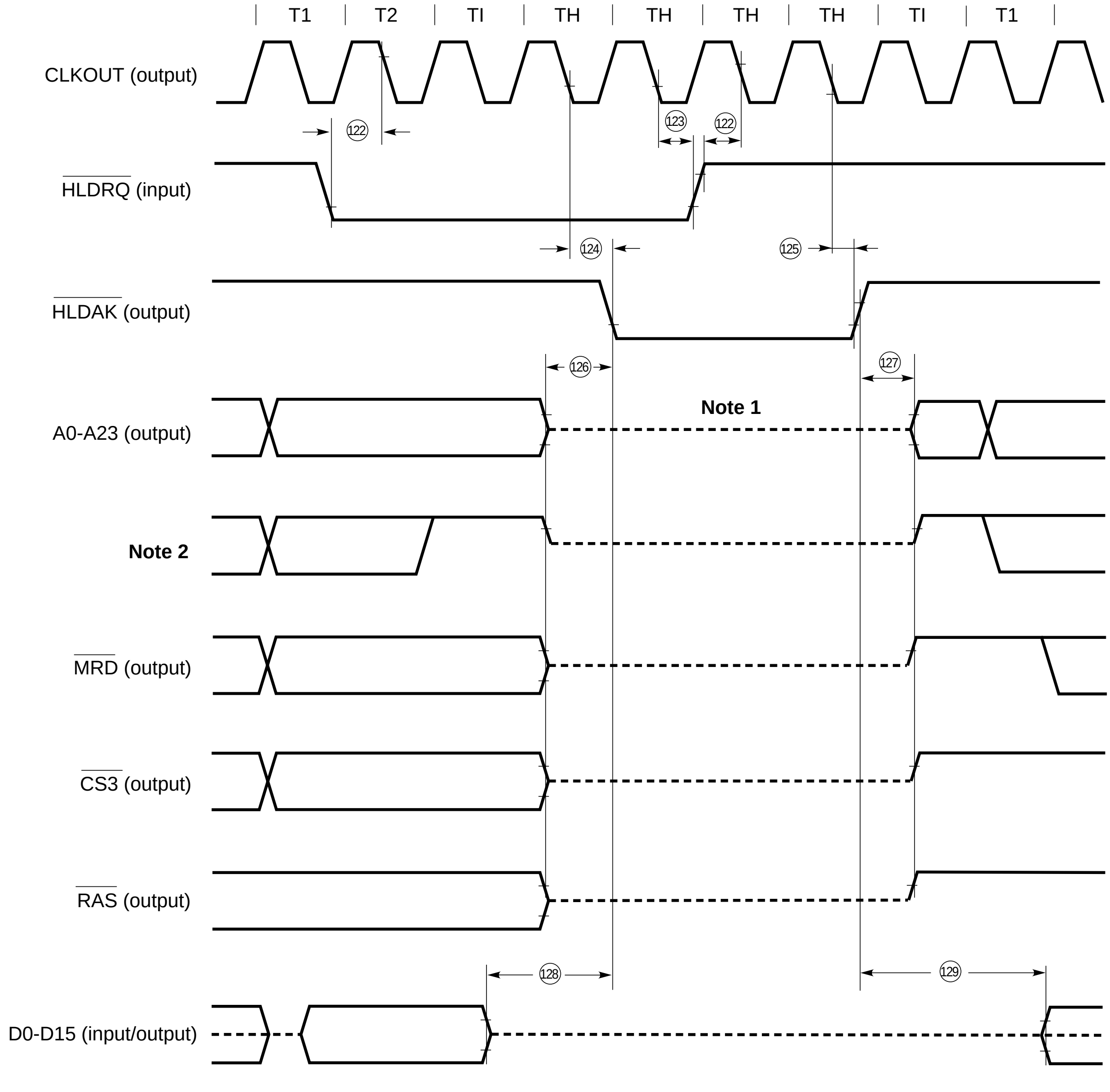
Remark Broken lines indicate high impedance.

(10) Bus hold timing (1/2)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
$\overline{\text{HLDRQ}}$ setup time (relative to CLKOUT↓)	$\textcircled{122}$	t_{SHQK}		6		ns
$\overline{\text{HLDRQ}}$ hold time (relative to CLKOUT↓)	$\textcircled{123}$	t_{HKHQ}		6		ns
$\overline{\text{HLDAK}}$ output delay (relative to CLKOUT↓)	$\textcircled{124}$	t_{DKHA}		2	15	ns
$\overline{\text{HLDAK}}$ output hold time (relative to CLKOUT↓)	$\textcircled{125}$	t_{HKHA}		2	15	ns
Delay from address float to $\overline{\text{HLDAK}}$↓	$\textcircled{126}$	t_{DAHA}		0.5T - 10		ns
Delay from $\overline{\text{HLDAK}}$↑ to address output	$\textcircled{127}$	t_{DHAA}		0.5T - 10		ns
Delay from data float to $\overline{\text{HLDAK}}$↓	$\textcircled{128}$	t_{DDHA}		1.5T - 15		ns
Delay from $\overline{\text{HLDAK}}$↑ to data output	$\textcircled{129}$	t_{DHAD}		2T - 15		ns

Remark T: t_{CYK}

(10) Bus hold timing (2/2)



Notes 1. The level existing immediately before the high-impedance state is held internally.

2. $\overline{\text{CS2}}$-$\overline{\text{CS0}}$ (output), $\overline{\text{UCAS}}$ (output), $\overline{\text{LCAS}}$ (output), $\overline{\text{LMWR}}$/$\overline{\text{WE}}$ (output), $\overline{\text{UMWR}}$ (output), $\overline{\text{IORD}}$ (output), $\overline{\text{IOWR}}$ (output)

Remark Broken lines indicate high impedance.

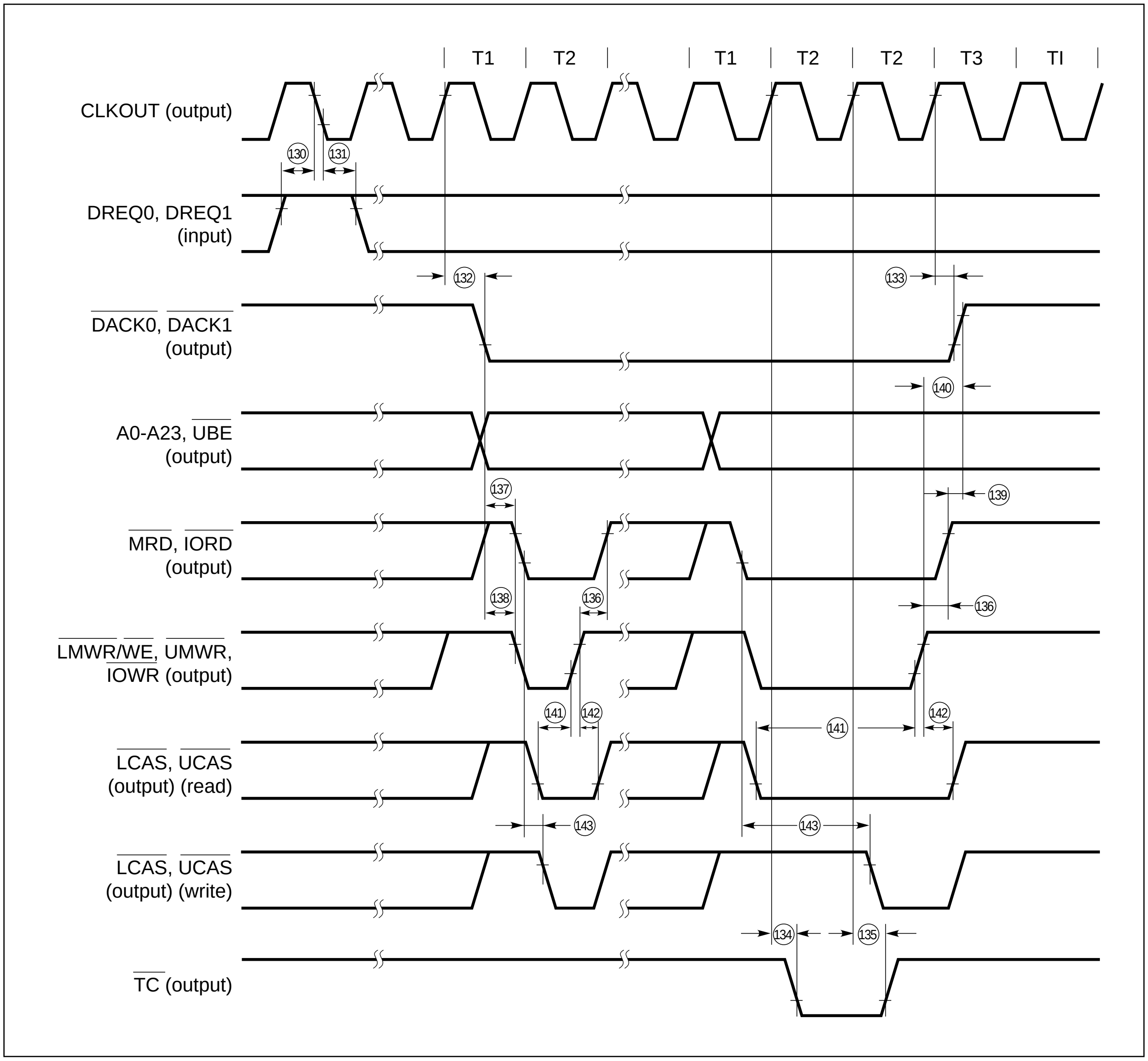
(11) DMAC timing (1/2)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
DREQn setup time (relative to CLKOUT↓)	130	t_{SDQK}		6		ns
DREQn hold time (relative to CLKOUT↓)	131	t_{HKDQ}		6		ns
$\overline{\text{DACKn}}$ output delay (relative to CLKOUT↑)	132	t_{DKDAK}		2	15	ns
$\overline{\text{DACKn}}$ output hold time (relative to CLKOUT↑)	133	t_{HKDAK}		2	15	ns
$\overline{\text{TC}}$ output delay (relative to CLKOUT↑)	134	t_{DKTC}		2	15	ns
$\overline{\text{TC}}$ output hold time (relative to CLKOUT↑)	135	t_{HKTC}		2	15	ns
Delay from $\overline{\text{WR}}$↑ to $\overline{\text{RD}}$↑	136	t_{DWRD}		0.5T - 10		ns
Delay from $\overline{\text{DACK}}$↓ to $\overline{\text{RD}}$↓	137	t_{DAKRD}		0.5T - 10		ns
Delay from $\overline{\text{DACK}}$↓ to $\overline{\text{WR}}$↓	138	t_{DAKWR}		0.5T - 10		ns
Delay from $\overline{\text{RD}}$↑ to $\overline{\text{DACK}}$↑	139	t_{RDDAK}		-4		ns
Delay from $\overline{\text{WR}}$↑ to $\overline{\text{DACK}}$↑	140	t_{WRDAK}		0.5T - 10		ns
Delay from $\overline{\text{CAS}}$↓ to $\overline{\text{IOWR}}$↑ (DRAM read)	141	t_{CASWR}		(n + 1)T - 10		ns
Delay from $\overline{\text{IOWR}}$↑ to $\overline{\text{CAS}}$↑ (DRAM read)	142	t_{WRCAS}		0.5T - 10		ns
Delay from $\overline{\text{IORD}}$↓ to $\overline{\text{CAS}}$↓ (DRAM write)	143	t_{RDCAS}		(n + 0.5)T - 10		ns

Remark T: t_{CYK}

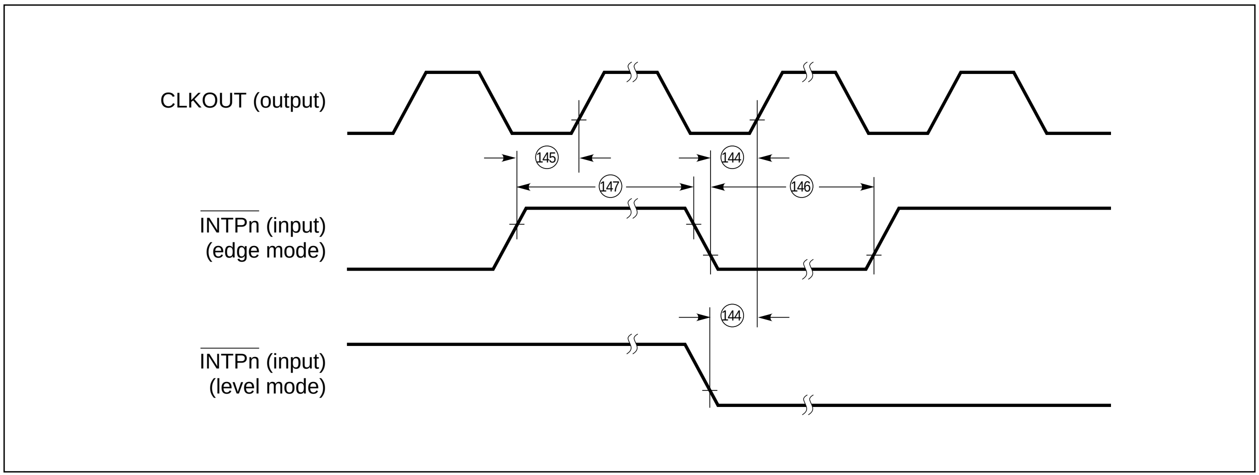
n: DMA wait state count

(11) DMAC timing (2/2)



(12) $\overline{\text{INTPn}}$ input setup time, hold time

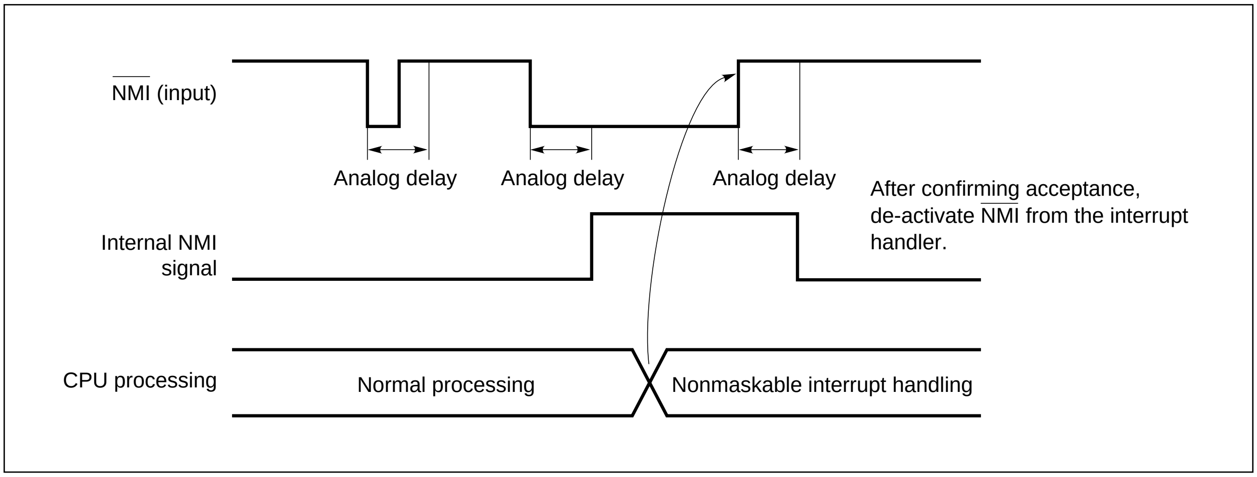
Parameter	Symbol		Conditions	MIN.	MAX.	Unit
$\overline{\text{INTPn}}$ input low setup time	144	t_{SILK}		9		ns
$\overline{\text{INTPn}}$ input high setup time	145	t_{SIHK}		9		ns
$\overline{\text{INTPn}}$ input low pulse width	146	t_{CYIL}		2		t_{CYK}
$\overline{\text{INTPn}}$ input high-level width	147	t_{CYIH}		2		t_{CYK}



(13) $\overline{\text{NMI}}$ input

The $\overline{\text{NMI}}$ pin incorporates a noise eliminator which is based on an analog delay (60 to 300 ns). The input setup time and input hold time are not, therefore, specified for $\overline{\text{NMI}}$.

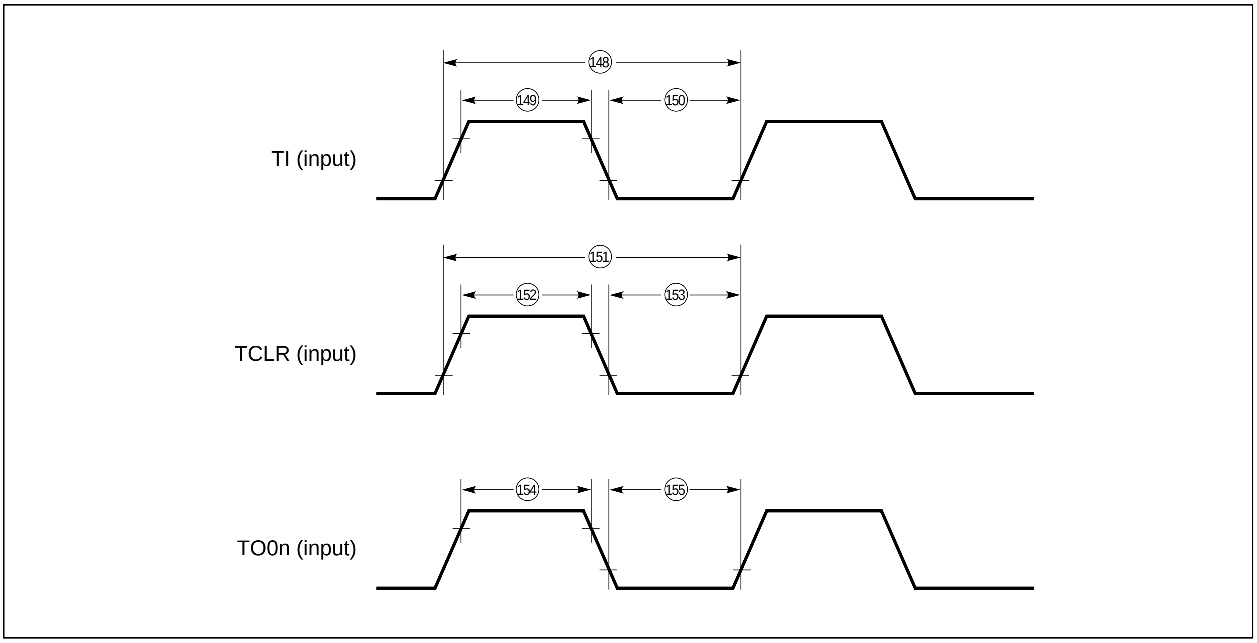
The $\overline{\text{NMI}}$ pin accepts a level input, such that the input level must be held until the acceptance of the input is confirmed after a branch to the handler.



(14) RPU block timing

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Timer clock cycle time	148	t_{TCYK}		4		t_{CYK}
Timer clock high-level width	149	t_{TKH}		2		t_{CYK}
Timer clock low-level width	150	t_{TKL}		2		t_{CYK}
Timer clear cycle time	151	t_{TCLRY}		4		t_{CYK}
Timer clear high-level width	152	t_{TCLRH}		2		t_{CYK}
Timer clear low-level width	153	t_{TCLRL}		2		t_{CYK}
Timer output high-level width	154	t_{WTOH}		2T - 7		ns
Timer output low-level width	155	t_{WTOL}		2T - 7		ns

Remark T: t_{CYK}



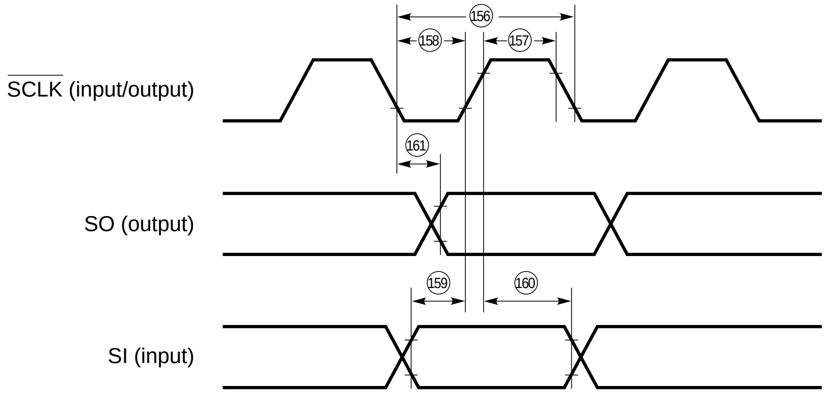
(15) CSI timing

(a) Master mode

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Serial clock cycle time	156	t_{CYSK}		4		t_{CYK}
Serial clock high-level width	157	t_{SKH}		30		ns
Serial clock low-level width	158	t_{SKL}		30		ns
SI setup time (relative to SCLK↑)	159	t_{SSISK}		20		ns
SI hold time (relative to SCLK↑)	160	t_{HSKSI}		20		ns
SO output delay (relative to SCLK↓)	161	t_{DSKSO}			30	ns

(b) Slave mode

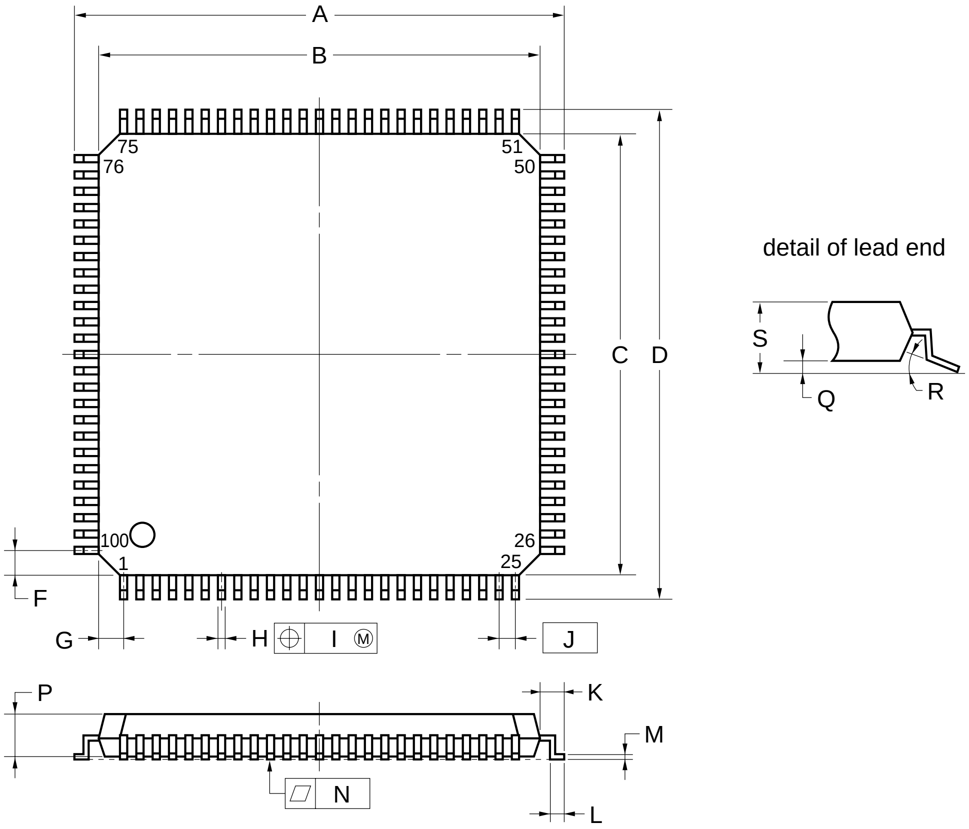
Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Serial clock cycle time	156	t_{CYSK}		4		t_{CYK}
Serial clock high-level width	157	t_{SKH}		30		ns
Serial clock low-level width	158	t_{SKL}		30		ns
SI setup time (relative to SCLK↑)	159	t_{SSISK}		20		ns
SI hold time (relative to SCLK↑)	160	t_{HSKSI}		20		ns
SO output delay (relative to SCLK↓)	161	t_{DSKSO}			30	ns



Remark Broken lines indicate high impedance.

★ 17. PACKAGE DRAWINGS

100 PIN PLASTIC LQFP (FINE PITCH) (14×14)



NOTE

Each lead centerline is located within 0.08 mm (0.003 inch) of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS	INCHES
A	16.00±0.20	0.630±0.008
B	14.00±0.20	$0.551^{+0.009}_{-0.008}$
C	14.00±0.20	$0.551^{+0.009}_{-0.008}$
D	16.00±0.20	0.630±0.008
F	1.00	0.039
G	1.00	0.039
H	$0.22^{+0.05}_{-0.04}$	0.009±0.002
I	0.08	0.003
J	0.50 (T.P.)	0.020 (T.P.)
K	1.00±0.20	$0.039^{+0.009}_{-0.008}$
L	0.50±0.20	$0.020^{+0.008}_{-0.009}$
M	$0.17^{+0.03}_{-0.07}$	$0.007^{+0.001}_{-0.003}$
N	0.08	0.003
P	1.40±0.05	0.055±0.002
Q	0.10±0.05	0.004±0.002
R	$3°^{+7°}_{-3°}$	$3°^{+7°}_{-3°}$
S	1.60 MAX.	0.063 MAX.

S100GC-50-8EU

18. RECOMMENDED SOLDERING CONDITIONS

The μPD70741 should be soldered and mounted under the conditions recommended in the table below.

For details of recommended soldering conditions, refer to the information document **Semiconductor Device Mounting Technology Manual** (C10535E).

For soldering methods and conditions other than those recommended below, contact an NEC sales representative.

★ **Table 18-1. Surface Mounting Type Soldering Conditions**

μPD70741GC-25-8EU: 100-pin plastic LQFP (fine pitch) (14 × 14 × 1.40 mm)

Soldering method	Soldering conditions	Recommended condition symbol
Infrared reflow	Package peak temperature: 235 °C, Duration: 30 sec. Max. (at 210 °C or above), Number of times: Twice Max., Time limit: 7 days[Note] (thereafter 10 hours prebaking required at 125 °C) <Caution> Non-heat-resistant trays, such as magazine and taping trays, cannot be baked before unpacking.	IR35-107-2
VPS	Package peak temperature: 215 °C, Duration: 40 sec. Max. (at 200 °C or above), Number of times: Twice Max., Time limit: 7 days[Note] (thereafter 10 hours prebaking required at 125 °C) <Caution> Non-heat-resistant trays, such as magazine and taping trays, cannot be baked before unpacking.	VP15-107-2
Partial heating	Pin temperature: 300 °C Max., Duration: 3 sec. Max. (per device side)	-

Note For the storage period after dry-pack decapsulation, storage conditions are Max. 25 °C, 65 % RH.

Caution Use of more than one soldering method should be avoided (except for partial heating).

NOTES FOR CMOS DEVICES

① PRECAUTION AGAINST ESD FOR SEMICONDUCTORS

Note: Strong electric field, when exposed to a MOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop generation of static electricity as much as possible, and quickly dissipate it once, when it has occurred. Environmental control must be adequate. When it is dry, humidifier should be used. It is recommended to avoid using insulators that easily build static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work bench and floor should be grounded. The operator should be grounded using wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions need to be taken for PW boards with semiconductor devices on it.

② HANDLING OF UNUSED INPUT PINS FOR CMOS

Note: No connection for CMOS device inputs can be cause of malfunction. If no connection is provided to the input pins, it is possible that an internal input level may be generated due to noise, etc., hence causing malfunction. CMOS device behave differently than Bipolar or NMOS devices. Input levels of CMOS devices must be fixed high or low by using a pull-up or pull-down circuitry. Each unused pin should be connected to V_{DD} or GND with a resistor, if it is considered to have a possibility of being an output pin. All handling related to the unused pins must be judged device by device and related specifications governing the devices.

③ STATUS BEFORE INITIALIZATION OF MOS DEVICES

Note: Power-on does not necessarily define initial status of MOS device. Production process of MOS does not define the initial operation status of the device. Immediately after the power source is turned ON, the devices with reset function have not yet been initialized. Hence, power-on does not guarantee out-pin levels, I/O settings or contents of registers. Device is not initialized until the reset signal is received. Reset operation must be executed immediately after power-on for devices having reset function.

Regional Information

Some information contained in this document may vary from country to country. Before using any NEC product in your application, please contact the NEC office in your country to obtain a list of authorized representatives and distributors. They will verify:

- Device availability
- Ordering information
- Product release schedule
- Availability of related technical literature
- Development environment specifications (for example, specifications for third-party tools and components, host computers, power plugs, AC supply voltages, and so forth)
- Network requirements

In addition, trademarks, registered trademarks, export restrictions, and other legal issues may also vary from country to country.

NEC Electronics Inc. (U.S.)
Santa Clara, California
Tel: 408-588-6000
800-366-9782
Fax: 408-588-6130
800-729-9288

NEC Electronics (Germany) GmbH
Duesseldorf, Germany
Tel: 0211-65 03 02
Fax: 0211-65 03 490

NEC Electronics (UK) Ltd.
Milton Keynes, UK
Tel: 01908-691-133
Fax: 01908-670-290

NEC Electronics Italiana s.r.1.
Milano, Italy
Tel: 02-66 75 41
Fax: 02-66 75 42 99

NEC Electronics (Germany) GmbH
Benelux Office
Eindhoven, The Netherlands
Tel: 040-2445845
Fax: 040-2444580

NEC Electronics (France) S.A.
Velizy-Villacoublay, France
Tel: 01-30-67 58 00
Fax: 01-30-67 58 99

NEC Electronics (France) S.A.
Spain Office
Madrid, Spain
Tel: 01-504-2787
Fax: 01-504-2860

NEC Electronics (Germany) GmbH
Scandinavia Office
Taeby, Sweden
Tel: 08-63 80 820
Fax: 08-63 80 388

NEC Electronics Hong Kong Ltd.
Hong Kong
Tel: 2886-9318
Fax: 2886-9022/9044

NEC Electronics Hong Kong Ltd.
Seoul Branch
Seoul, Korea
Tel: 02-528-0303
Fax: 02-528-4411

NEC Electronics Singapore Pte. Ltd.
United Square, Singapore 1130
Tel: 65-253-8311
Fax: 65-250-3583

NEC Electronics Taiwan Ltd.
Taipei, Taiwan
Tel: 02-719-2377
Fax: 02-719-5951

NEC do Brasil S.A.
Cumbica-Guarulhos-SP, Brasil
Tel: 011-6465-6810
Fax: 011-6465-6829

J98. 2

[MEMO]

The related documents indicated in this publication may include preliminary versions. However, preliminary versions are not marked as such.

V810, V821, and V810 Family are trademarks of NEC Corporation.

While NEC Corporation has been making continuous effort to enhance the reliability of its semiconductor devices, the possibility of defects cannot be eliminated entirely. To minimize risks of damage or injury to persons or property arising from a defect in an NEC semiconductor device, customers must incorporate sufficient safety measures in its design, such as redundancy, fire-containment, and anti-failure features.
NEC devices are classified into the following three quality grades:
"Standard", "Special", and "Specific". The Specific quality grade applies only to devices developed based on a customer designated "quality assurance program" for a specific application. The recommended applications of a device depend on its quality grade, as indicated below. Customers must check the quality grade of each device before using it in a particular application.

Standard: Computers, office equipment, communications equipment, test and measurement equipment, audio and visual equipment, home electronic appliances, machine tools, personal electronic equipment and industrial robots
Special: Transportation equipment (automobiles, trains, ships, etc.), traffic control systems, anti-disaster systems, anti-crime systems, safety equipment and medical equipment (not specifically designed for life support)
Specific: Aircrafts, aerospace equipment, submersible repeaters, nuclear reactor control systems, life support systems or medical equipment for life support, etc.

The quality grade of NEC devices is "Standard" unless otherwise specified in NEC's Data Sheets or Data Books.
If customers intend to use NEC devices for applications other than those specified for Standard quality grade, they should contact an NEC sales representative in advance.
Anti-radioactive design is not implemented in this product.