To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: http://www.renesas.com

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (http://www.renesas.com)

Send any inquiries to http://www.renesas.com/inquiry.

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: “Standard”, “High Quality”, and “Specific”. The recommended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as “Specific” without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as “Specific” or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is “Standard” unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

   “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.

   “High Quality”: 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”: Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

RENESAS

User's Manual

V830™

32-Bit Microprocessor

Hardware

μPD705100

Document No. U10064EJ3V0UM00 (3rd edition)
Date Published January 1998 J CP(K)

[MEMO]

SUMMARY OF CONTENTS

[MEMO]

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 devices 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.

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

No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Corporation. NEC Corporation assumes no responsibility for any errors which may appear in this document.

NEC Corporation does not assume any liability for infringement of patents, copyrights or other intellectual property rights of third parties by or arising from use of a device described herein or any other liability arising from use of such device. No license, either express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC Corporation or others.

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.

M7 96.5

Phase-out/Discontinued

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: 253-8311
Fax: 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

J97. 8

Major Changes

Page	Description
P.68	Description for the CM bit in **Section 7.4** has been modified.
	Chapter 8 has been included in the V830 Family™ User's Manual, Architecture.

The mark ⋆ shows major revised points.

PREFACE

Intended readers

This manual is intended for those users who wish to become familiar with the functions of the V830 (μPD705100), and those involved in the design of systems based on the V830.

Purpose

The purpose of this manual is to assist users in understanding the hardware functions of the V830, i.e., the topics listed in “Configuration” below.

Configuration

The V830 User's Manual consists of two manuals: the V830 User's Manual, Hardware (this manual) and V830 Family™ User's Manual, Architecture.

Hardware	Architecture
• Pin functions	• Register set
• Address space	• Data set
• 32-bit bus mode	• Address space
• 16-bit bus mode	• Instructions
• Interrupts	• Interrupts and exceptions
• Clock controller	• Internal memory
	• Reset

How to use this manual

Readers of this manual are assumed to have a general knowledge of electronics, logic circuits, and microcomputers.

To check functions whose names are known in detail,
-> See **Appendix B**.

To gain a particular understanding of the instruction functions,
-> Refer to the V830 Family User's Manual, Architecture, bounded separately.

To check the electrical characteristics of the V830,
-> Refer to the separate data sheet.

To gain an overall understanding of the functions provided by the V830,
-> Read this manual in its entirety.

Legend

Significance of a data representation	: Left high, right low
Representation of active low	: $\overline{xxx}$ (bar above a pin or signal name)
Memory map address	: Top upper, bottom lower
Note	: Explanation of Note that appears in text
Caution	: Point to which the user must pay particular attention
Remark	: Supplementary explanation of the contents of the text

Numeric representations : xxxx or xxxxB for a binary number
xxxx for a decimal number
xxxxH for a hexadecimal number

Prefixes indicating powers of two (address space, memory capacity):
K (kilo) : $2^{10} = 1024$
M (mega) : $2^{20} = 1024^2$
G (giga) : $2^{30} = 1024^3$

Related document The related document may be a preliminary edition; if so, however, this is not indicated in this manual.

- **Documents related to devices**

Document name	Document number
V830 Family User's Manual, Architecture	U12496E
V830 User's Manual, Hardware	This manual
μPD705100 (V830) Data Sheet	U11483E

- **Documents related to development tools**

Document name		Document number
IE-705100-MC-EM1 User's Manual (In-Circuit Emulator for V830)		To be created
CA830 User's Manual (C Compiler Package)	Operation (UNIX™-based)	U11013E
	Operation (Windows™-based)	U11068E
	Assembly Language	U11014E
	C	U11010E
	Project Manager	U11991E
ID830 User's Manual (C Source Debugger)	Operation (UNIX-based)	-
	Operation (Windows-based)	U12206E
	Installation (UNIX-based)	-
RX830 User's Manual (Real-Time OS)	Basic	U11730E
	Installation	U11731E
	Technical	U11713E

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 INTRODUCTION

The V830 microprocessor is the first V830 Family product to be offered by NEC for data processing applications. The V830 is overviewed below.

1.1 OVERVIEW

The V830 is a high-performance 32-bit RISC microprocessor. With an operating (internal) frequency of 100 MHz, the V830 can perform the data processing demanded by multimedia devices in only a few cycles. Besides a high interrupt responsibility and an optimized pipeline structure, a sum-of-products instruction, double-word shift instruction, and high-speed branch instruction using branch predication have been added to support multimedia functions.

Furthermore, by inheriting the V810 Family™ basic instruction set at the object level, V810 Family software can be used as is.

The V830 offers high-performance data processing for applications such as image processing, game machines, car navigation, high-performance TVs, color facsimile machines, etc.

1.2 FEATURES

- Number of instructions: 102
- Minimum number of instruction execution cycles: 1
- General-purpose registers: 32 bits x 32
- Instruction set: V810 basic instruction set
  Sum-of-products operation (32 bits x 32 bits + (upper/lower) 32 bits): 1-3 cycles
  Saturatable arithmetic operation (with a saturation detection function)
  Double-word shift (64-bit data shift): 1-2 cycles
  High-speed branch
  Block transfer instruction
- Memory space
  Memory space, I/O space: 4G-byte linear address
- External bus interface:
  Supports the (16-byte) burst bus cycle.
  Address/data separate bus
  Bus hold
  Bus lock
  Four chip select outputs
- Internal memory
  Instruction cache (direct mapping): 4K bytes
  Data cache (direct mapping/write-through): 4K bytes
  Instruction RAM: 4K bytes
  Data RAM: 4K bytes

- Interrupts
  - Nonmaskable interrupt: 1
  - Maskable interrupt: 16 levels
- Power control
  - Stop mode
  - Sleep mode
- Package
  - 144-pin plastic LQFP (fine pitch) (20 x 20 mm)
- CMOS structure

1.3 APPLICATIONS

- Systems requiring high-speed data processing
  (game machines, car navigation, high-performance TVs, color facsimile machines, etc.)

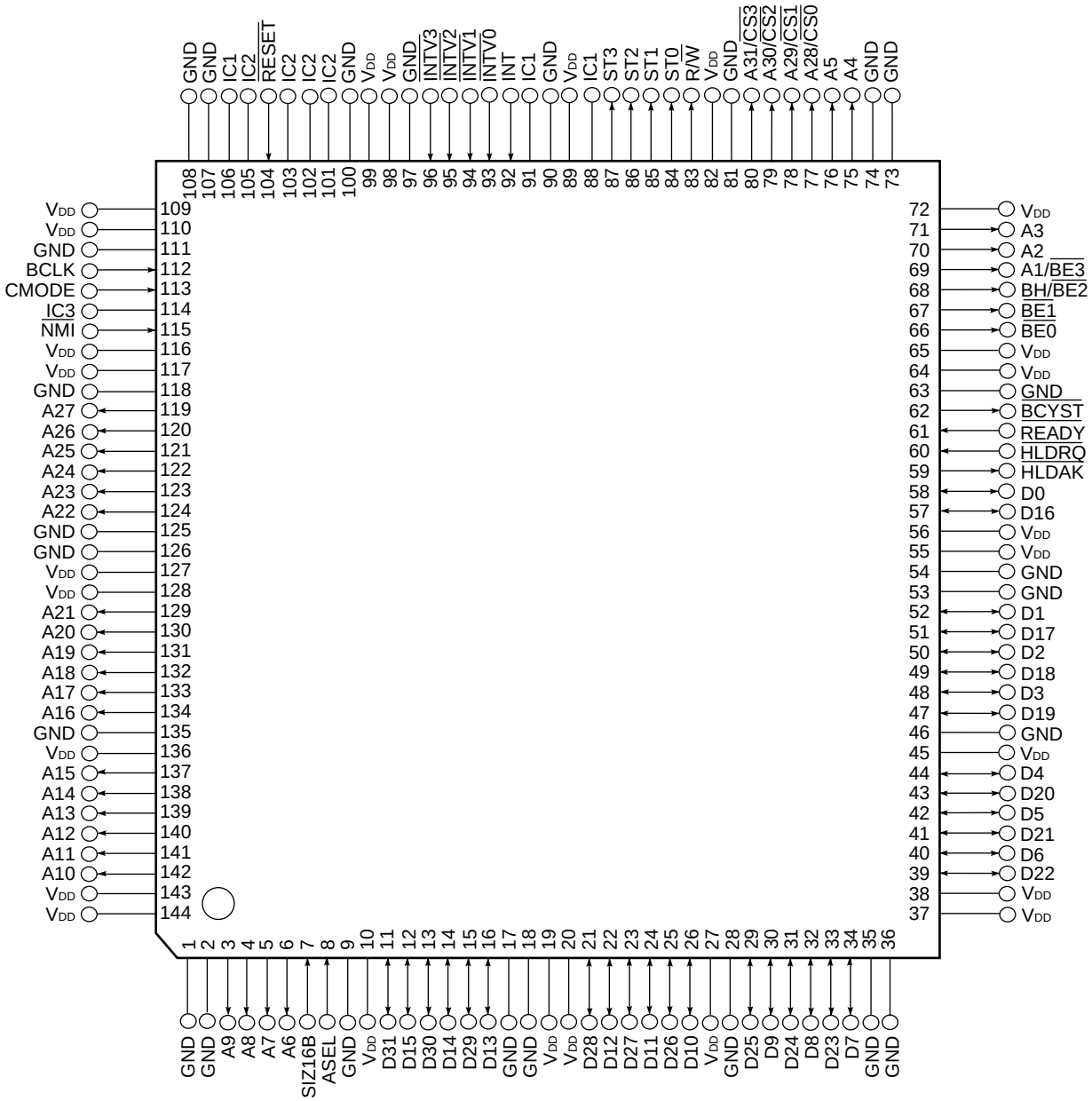
- Systems requiring lower power consumption
  (PDA, PHS, etc.)

1.4 ORDERING INFORMATION

Part number	Package
μPD705100GJ-100-8EU	144-pin plastic LQFP (fine pitch) (20 x 20 mm)

1.5 PIN CONFIGURATION

144-pin plastic LQFP (fine pitch) (20 x 20 mm)
μPD705100GJ-100-8EU

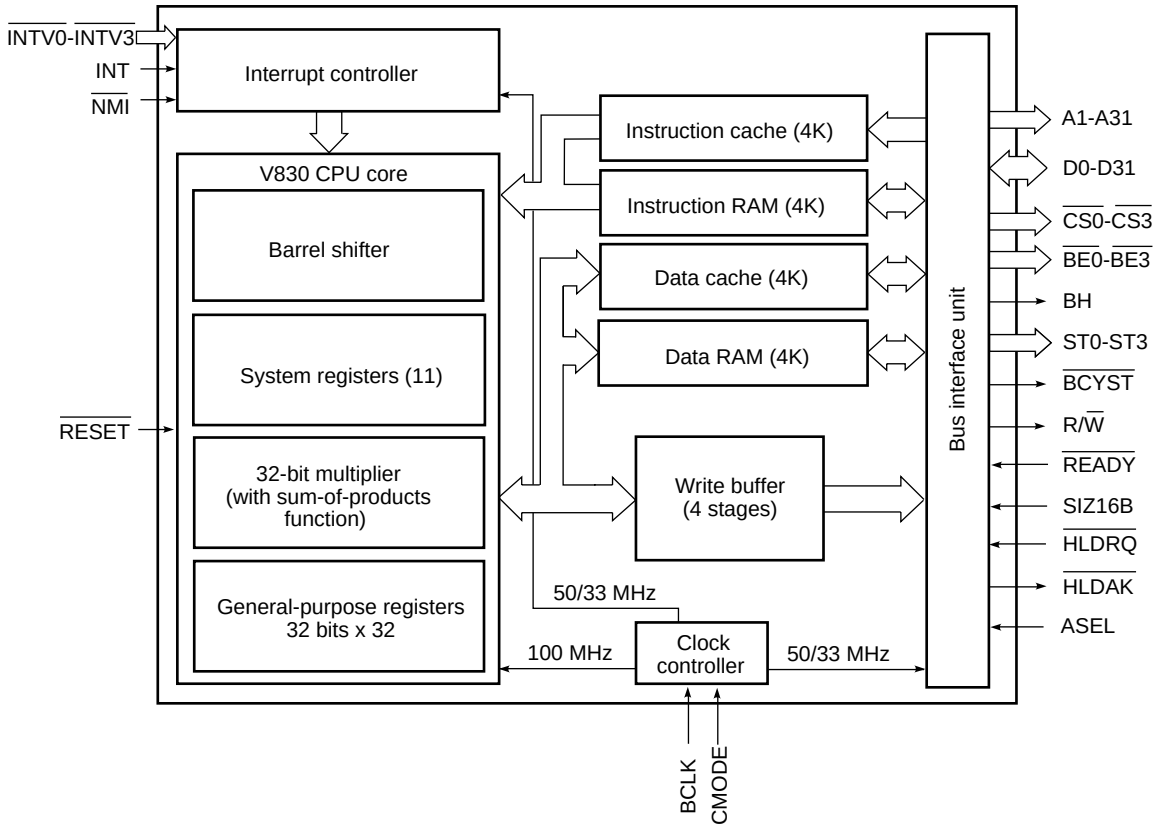


Caution Leave the IC1 pins open.
Connect each IC2 pin to GND via a dedicated resistor.
Connect each IC3 pin to V_{DD} via a dedicated resistor.

Pin names

A1-A31	: Address Bus
$\overline{CS0}$-$\overline{CS3}$	: Chip Select
D0-D31	: Data Bus
$\overline{BE0}$-$\overline{BE3}$	: Byte Enable
BH	: Byte or Halfword
ST0-ST3	: Status
$\overline{BCYST}$	: Bus Cycle Start
R/$\overline{W}$	: Read/Write
$\overline{READY}$	: Ready
$\overline{HLDRQ}$	: Hold Request
$\overline{HLDAK}$	: Hold Acknowledge
SIZ16B	: Bus Size 16 bit
$\overline{NMI}$	: Non-Maskable Interrupt Request
INT	: Interrupt Request
$\overline{INTV0}$-$\overline{INTV3}$	: Interrupt Level
BCLK	: Bus Clock
CMODE	: Clock Mode
ASEL	: Address Select
$\overline{RESET}$	: Reset
V_{DD}	: Power Supply
GND	: Ground
IC1-IC3	: Internally Connected

1.6 BLOCK DIAGRAM



1.7 INTERNAL UNITS

(1) CPU core

Executes the processing of the majority of instructions, including address calculation, arithmetic and logic operations, and data transfer within one cycle, by means of 5-stage pipeline control.

Dedicated hardware, such as an adder with a sum-of-products function (32 bits x 32 bits + (upper/lower) 32 bits) and barrel shifter (capable of 64-bit data shift) are built in to enable the high-speed processing of complicated instructions.

(2) Bus interface

Activates a required bus cycle according to the physical address acquired by the CPU. The bus interface unit supports both 32-bit bus mode, in which the external data bus has a 32-bit configuration, and 16-bit bus mode, in which it has a 16-bit configuration. It outputs appropriate control signals according to the mode set when a bus cycle is activated.

(3) Interrupt controller

Handles received hardware interrupt requests (nonmaskable and maskable interrupt requests) . Up to 16 maskable interrupt request sources can be handled using level input. The handler for maskable interrupts can be placed in the built-in instruction RAM.

(4) Clock controller

Creates external bus clock pulses for the interface between the CPU internal clock and the external hardware. It also supports the halting of clock pulses to the CPU (sleep mode) as well as PLL circuit halt (stop mode).

(5) Write buffer

Stores data write (up to four data items) when the CPU performs write to external hardware. When data is written into the write buffer, the CPU no longer has to wait for the end of the bus cycle and can continue processing.

(6) Internal memory

16K-byte memory. This memory consists of four 4K-byte blocks, an instruction cache, data cache, instruction RAM, and data RAM. The instruction RAM uses direct mapping, while the data cache uses direct mapping/write-through.

CHAPTER 2 PIN FUNCTIONS

2.1 PIN FUNCTIONS

Pin name	Input/output	Function	At hold	At reset
A2-A27	Tristate output	Address bus	Hi-Z	H
A28-A31/$\overline{CS0}$-$\overline{CS3}$**Note**		Address bus/chip select	Hi-Z/H	H
D0-D31	Tristate input/output	Bidirectional data bus	Hi-Z	Hi-Z
$\overline{BE0}$, $\overline{BE1}$	Tristate output	Indicates which data bus can be used for data access.	Hi-Z	H
$\overline{BE2}$/BH		Indicates access to D16-D23/byte or halfword access.	Hi-Z	H
$\overline{BE3}$/A1		Indicates most significant byte access/A1 address.	Hi-Z	H
ST0-ST3		Indicates the status of a bus.	Hi-Z	0101
$\overline{BCYST}$		Indicates the start of a bus cycle.	Hi-Z	H
R/$\overline{W}$		Indicates whether the bus cycle is a read or write cycle.	Hi-Z	H
$\overline{READY}$	Input	Terminates a bus cycle.	-	-
$\overline{HLDRQ}$		Requests bus mastership.	-	-
$\overline{HLDAK}$	Output	Response to $\overline{HLDRQ}$	L	H
SIZ16B	Input	Fixes the bus width to 16 bits.	-	-
$\overline{NMI}$		Nonmaskable interrupt request	-	-
INT		Maskable interrupt request	-	-
$\overline{INTV0}$-$\overline{INTV3}$		Indicates an interrupt level.	-	-
BCLK		Bus clock input	-	-
CMODE		Specifies the frequency ratio for the external bus and the internal circuit.	-	-
ASEL		Selects A28-A31/$\overline{CS0}$-$\overline{CS3}$.	-	-
$\overline{RESET}$		Resets the internal state.	-	-
V_{DD}	-	Supplies positive power.	-	-
GND		Ground potential	-	-

Note When used for a chip select signal, this is held at the high level.

2.2 EXPLANATION OF PIN FUNCTIONS

This section describes the pin functions of the V830. Each state of a bus cycle regards the time between the rise of one bus clock pulse to the rise of the next as one unit.

(1) A2-A27 (Address Bus): tristate output

Pins from which the CPU outputs an address signal when accessing main external storage or an input/output device. The output signal is synchronized with the rise of a bus clock pulse (BCLK) in Ta.

(2) A28-A31/$\overline{CS0}$-$\overline{CS3}$ (Address Bus/Chip Select): tristate output

Pins from which the CPU outputs an address or chip select signal when accessing main external storage or an input/output device. An address signal is output from A28-A31 if ASEL is at the high level at reset, while a chip select signal is output if it is at the low level. These signals are output from A28-A31/$\overline{CS0}$-$\overline{CS3}$ in synchronization with the rise of a clock pulse in the Ta and Tb2-Tb4 states (Ta, Tw2, and Tb2-Tb8 states in 16-bit mode). A chip select signal is output only during a bus cycle. The areas where the chip select output is valid are fixed statically according to the hardware.

Table 2-1. Chip Select Output vs. Address Space

Pin	Memory	I/O	Address space
$\overline{CS0}$	o	x	FE001000H-FFFFFFFFH (Cachable area) 7E000000H-7FFFFFFFH (Uncachable area)
$\overline{CS1}$	o	x	40000000H-4FFFFFFFH (Uncachable area) 00001000H-0FFFFFFFH (Cachable area)
$\overline{CS2}$	o	x	50000000H-5FFFFFFFH (Uncachable area) 10000000H-1FFFFFFFH (Cachable area)
$\overline{CS3}$	o	x	60000000H-6FFFFFFFH (Uncachable area) 20000000H-2FFFFFFFH (Cachable area)

(3) D0-D31 (Data Bus): tristate input/output

Pins which the CPU uses to input and output read and write data when accessing main external storage or an input/output device. During a write cycle, an output signal is synchronized with the rise of a bus clock pulse in the Ta, Ts, Tb1-Tb8, Tw1, and Tw2 states. For details, see **Section 4.2** (for 32-bit bus mode) and **Section 5.3** (for 16-bit bus mode).

(4) $\overline{BE0}$, $\overline{BE1}$ (Byte Enable): tristate output

Output the signal indicating which byte in the 32-bit data bus should be accessed when the CPU is accessing main external storage or an input/output device. The pins change in synchronization with the rise of a bus clock pulse in the Ta state.

$\overline{BE0}$: D0-D7

$\overline{BE1}$: D8-D15

Data size		Address		32-bit mode				16-bit mode			
		A1	A0	$\overline{BE3}$	$\overline{BE2}$	$\overline{BE1}$	$\overline{BE0}$	A1	BH	$\overline{BE1}$	$\overline{BE0}$
Byte access		0	0	1	1	1	0	0	1	1	0
		0	1	1	1	0	1	0	1	0	1
		1	0	1	0	1	1	1	1	1	0
		1	1	0	1	1	1	1	1	0	1
Halfword access		0	0	1	1	0	0	0	1	0	0
		1	0	0	0	1	1	1	1	0	0
Word access	First time	0	0	0	0	0	0	0	0	0	0
	Second							1	0	0	0
Burst transfer**Note**		0	0	0	0	0	0	0/1	0	0	0

Note Data is transferred four times in 32-bit bus mode and eight times in 16-bit bus mode. For details, see **Section 4.1** (for 32-bit bus mode) and **Section 5.2** (for 16-bit bus mode).

(5) $\overline{BE2}$/BH (Byte Enable/Byte or Halfword): tristate output

Indicates access to D16-D23 on the 32-bit data bus when the CPU is accessing main external storage or an input/output device. In 16-bit bus mode, a signal indicating byte or halfword access is output. The pin changes in synchronization with the rise of a bus clock pulse in the Ta state.

(6) $\overline{BE3}$/A1 (Byte Enable/Address): tristate output

Outputs a signal indicating access to the most significant byte on the 32-bit data bus when the CPU is accessing main external storage or an input/output device. The pin changes in synchronization with the rise of a bus clock pulse in the Ta state. In 16-bit bus mode, address A1 is output. The pin changes in synchronization with the rise of a bus clock pulse in the Ta, Tw2, Tb2-Tb8 states. For details, see **Section 4.2** (for 32-bit bus mode) and **Section 5.3** (for 16-bit bus mode).

(7) ST0-ST3 (Status): tristate output

Indicate the status of the current bus cycle and the CPU. The pins change in synchronization with a bus clock pulse. They also change in cycles other than a bus cycle.

ST3	ST2	ST1	ST0	R/$\overline{W}$	Bus cycle type	Data size
0	0	0	0	0	I/O write	1/2/4 bytes
0	0	0	0	1	I/O read	1/2/4 bytes
0	0	0	1	*	RFU	
0	0	1	0	*	Hald acknowledge**Notes 1 and 2**	
0	0	1	1	*	Stop acknowledge**Notes 1 and 2**	
0	1	*	*	*	RFU	
1	0	0	0	0	RFU	
1	0	0	0	1	Single instruction read	4 bytes
1	0	0	1	0	Single data write	1/2/4 bytes
1	0	0	1	1	Single data read	1/2/4 bytes
1	0	1	0	*	Idle state (with no external access request)	
1	0	1	1	0	Bus lock write	4 bytes
1	0	1	1	1	Bus lock read	4 bytes
1	1	0	0	0	RFU	
1	1	0	0	1	Instruction cache refill	16 bytes
1	1	0	1	0	RFU	
1	1	0	1	1	Data cache refill	16 bytes
1	1	1	0	*	RFU	
1	1	1	1	0	Block data write	16 bytes
1	1	1	1	1	Block data read	16 bytes

*: Don't care

Notes 1. ST0-ST3 do not change at the time of internal memory access or internal I/O access.

2. $\overline{\text{BCYST}}$ does not become active at hold acknowledge or stop acknowledge.

(8) $\overline{\text{BCYST}}$ (Bus Cycle Start): tristate output

Outputs a signal indicating the start of a bus cycle. The pin is at the low level in the Ta state.

(9) R/$\overline{W}$ (Read/Write): tristate output

Outputs a signal indicating whether the current bus cycle is a read or write cycle. The pin is valid only during a bus cycle. It outputs a high-level signal during a read cycle and a low-level signal during a write cycle. The pin changes in synchronization with the rise of a bus clock pulse in the Ta state.

(10) $\overline{\text{READY}}$ (Ready): input

Receives the signal for terminating a bus cycle. The signal is sampled at the rise of a bus clock pulse at the end of the Ts, Tb1-Tb8, Tw1-Tw2 states.

(11) $\overline{\text{HLDRQ}}$ (Hold Request): input

Receives a signal input to the CPU to request bus mastership. The signal is sampled at the rise of the last of the Ts and Tb4 states (Ts, Tb8, and Tw2 in 16-bit bus mode) or at the rise of the Ti state.

(12) $\overline{\text{HLDAK}}$ (Hold Acknowledge): output

Outputs a signal acknowledging input to $\overline{\text{HLDRQ}}$. The pin changes in synchronization with the rise of a bus clock pulse.

The pin becomes active after the CPU relinquishes bus mastership, and D0-D31, $\overline{\text{BE0}}$-$\overline{\text{BE3}}$, ST0-ST3, and R/$\overline{\text{W}}$ are set to high impedance from the next clock pulse until the pin changes to the high level. When $\overline{\text{HLDRQ}}$ becomes inactive, the CPU makes $\overline{\text{HLDAK}}$ inactive and regains bus mastership.

(13) SIZ16B (Bus Size 16 bits): input

Receives a signal for fixing the external data bus width to 16 bits. When SIZ16B becomes active, the CPU enters the mode in which signals corresponding to the 16-bit data bus system are output from $\overline{\text{BE0}}$, $\overline{\text{BE1}}$, BH, and A1 and D16-D31 are set to high impedance. The signal is sampled at the rise of the first bus clock pulse issued after $\overline{\text{RESET}}$ becomes high.

SIZ16B can be changed at reset only. If an attempt is made to change the pin at any other time, the operation of the CPU cannot be guaranteed.

(14) $\overline{\text{NMI}}$ (Non-maskable Interrupt Request): input

Receives a nonmaskable interrupt request signal issued to the CPU. The signal is sampled at the rise of a bus clock pulse, and the interrupt request is detected when $\overline{\text{NMI}}$ changes from high to low.

(15) INT (Interrupt Request): input

Receives a maskable interrupt request signal issued to the CPU. The signal at INT is sampled at the rise of a bus clock pulse. The interrupt request is detected when the following three conditions are satisfied:

(a) All of the NP, EP, and ID flags in PSW are set to 0.
(b) The interrupt level of $\overline{\text{INTV0}}$-$\overline{\text{INTV3}}$ is higher than the interrupt permit level for PSW.
(c) INT is active.

The V830 checks whether an interrupt request exists at the end of an instruction, or while it is not performing internal processing. If an interrupt request is detected, the V830 accepts it.

INT must be held at the active level, together with $\overline{\text{INTV0}}$-$\overline{\text{INTV3}}$, until the CPU starts interrupt handling and posts external notification, by software, that the interrupt request has been accepted.

(16) $\overline{\text{INTV0}}$-$\overline{\text{INTV3}}$ (Interrupt Level): input

Receive a maskable interrupt request signal, issued to the CPU. The signal at $\overline{\text{INTV0}}$-$\overline{\text{INTV3}}$ is sampled at the rise of a bus clock pulse.

$\overline{\text{INTV0}}$-$\overline{\text{INTV3}}$ must be held at the active level, together with INT, until the CPU starts interrupt handling and posts external notification, by software, that the interrupt request has been accepted. (The interrupt level, however, can be changed to a higher priority.)

(17) BCLK (Bus Clock): input

Receives bus clock pulses. A bus cycle is activated each time BCLK is input. The frequency ratio for the internal clock (CLK) and bus clock (BCLK) is specified with CMODE.

(18) CMODE (Clock Mode): input

Receives the signal specifying the frequency ratio for the external bus and the internal circuit. The signal is sampled at the first bus clock pulse issued after $\overline{\text{RESET}}$ goes high. The frequency of the internal clock is double that of the bus clock if CMODE is high at reset, or three times as high if CMODE is low.
CMODE must be connected to V_{DD} or GND via a resistor.

(19) ASEL (Address Select): input

Receives the signal for selecting the output of A28-A31/$\overline{\text{CS0}}$-$\overline{\text{CS3}}$. The signal at ASEL is sampled at the first bus clock pulse issued after $\overline{\text{RESET}}$ goes high. ASEL cannot be changed at any other timing.
A signal is output from A28-A31 if the input sampled at ASEL input is high, or from $\overline{\text{CS0}}$-$\overline{\text{CS3}}$ if the input is low.

(20) $\overline{\text{RESET}}$ (Reset): input

Receives the signal for initializing the CPU. The signal is sampled at the fall of a clock pulse. It must be held at the active level for at least 20 clock pulses.
When the input of $\overline{\text{RESET}}$ is accepted, the CPU initializes its pins and internal registers and executes instructions starting from address FFFFFFF0H.

(21) V_{DD} (Power Supply)

Pins used to supply positive power. All V_{DD} pins must be connected to positive power.

(22) GND (Ground)

Ground pins. All GND pins must be connected to ground.

(23) IC1 (Internally Connected 1)

All IC1 pins must be left open.

(24) IC2 (Internally Connected 2)

Each IC2 pin must be connected to ground via a dedicated resistor.

(25) IC3 (Internally Connected 3)

Each IC3 pin must be connected to positive power via a dedicated resistor.

CHAPTER 3 ADDRESS SPACE

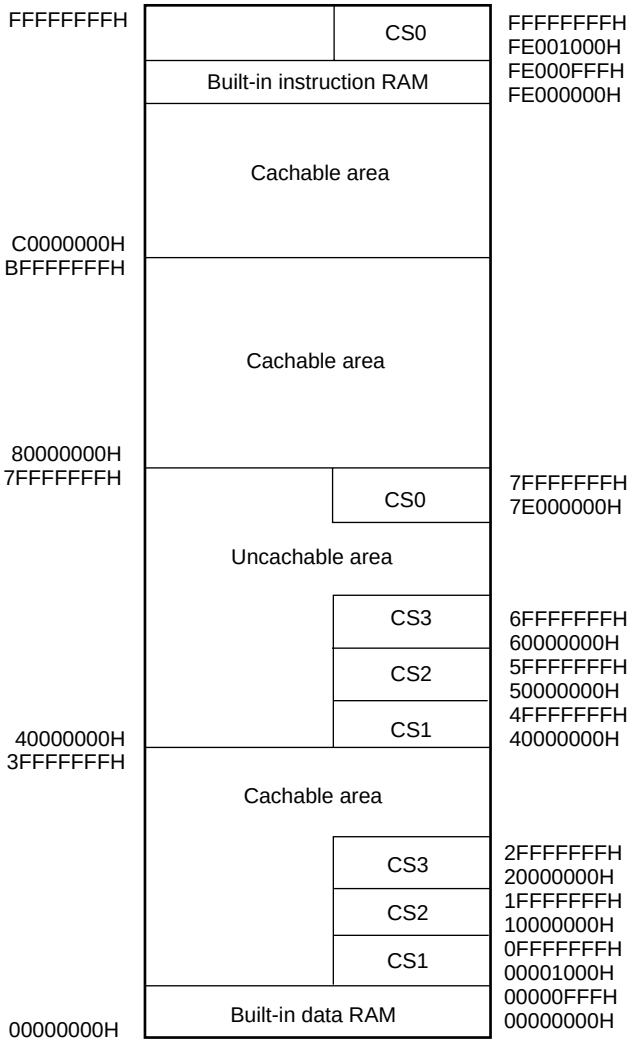
3.1 MEMORY SPACE

The V830 uses four chip select/address pins and 26 address bus pins to represent a 32-bit address. When the chip select function is used, a 256M-byte image space is created as three spaces and a 32M-byte image space is created as one space. When the chip select function is not used, a 4G-byte linear address space is created.

Area 40000000H-7FFFFFFFH in the memory space is reserved as an uncachable area. When this area is accessed, the cache function is not effective. For all other areas, the cache function is effective.

Within the memory space, built-in instruction RAM and built-in data RAM are mapped. By accessing these areas, an instruction can be fetched and data loaded/stored within one cycle (internal clock) without activating a bus cycle externally. Data in the built-in instruction RAM, however, cannot be accessed by using the load/store instructions. Nor can instructions be fetched from the built-in data RAM. These built-in RAMs are mapped to the cachable area; however, they are not cached.

Figure 3-1. Memory Map



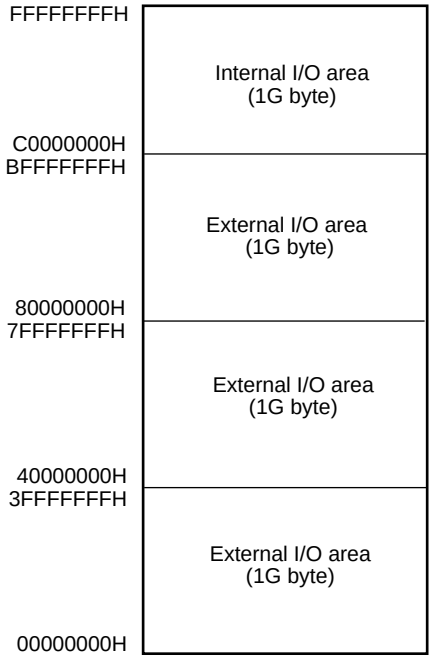
Chip select signal	Address space	Cachable
$\overline{CS0}$	7E000000H-7FFFFFFH	X
	FE001000H-FFFFFFFFH	○
$\overline{CS1}$	40000000H-4FFFFFFFH	X
	00001000H-0FFFFFFFH	○
$\overline{CS2}$	50000000H-5FFFFFFFH	X
	10000000H-1FFFFFFFH	○
$\overline{CS3}$	60000000H-6FFFFFFFH	X
	20000000H-2FFFFFFFH	○

3.2 I/O SPACE

The V830 represents the I/O space using 32 bits and supports a linear address space of up to 4G-bytes.

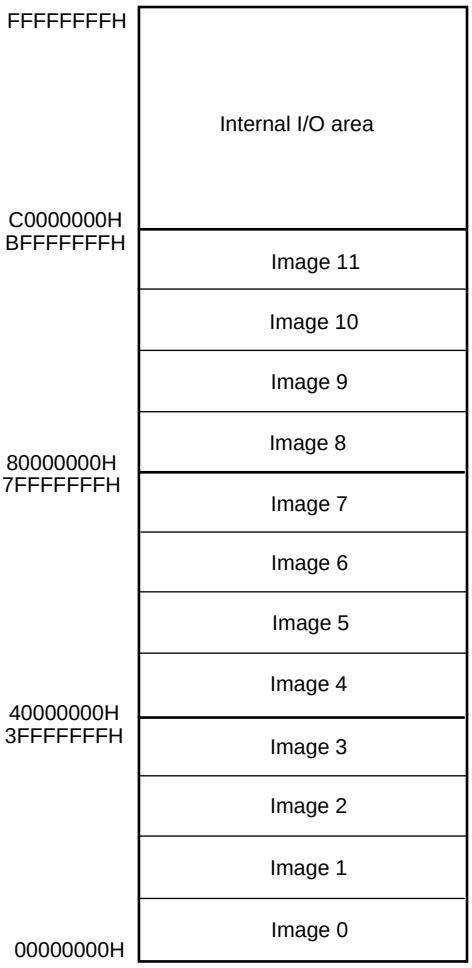
The 1G-byte area C0000000H-FFFFFFFFH is reserved as an internal I/O area. External I/O cannot be placed in this area. When accessing that part of the internal I/O area to which internal I/O is not allocated, normal operation cannot be quaranteed.

Figure 3-2. I/O Map



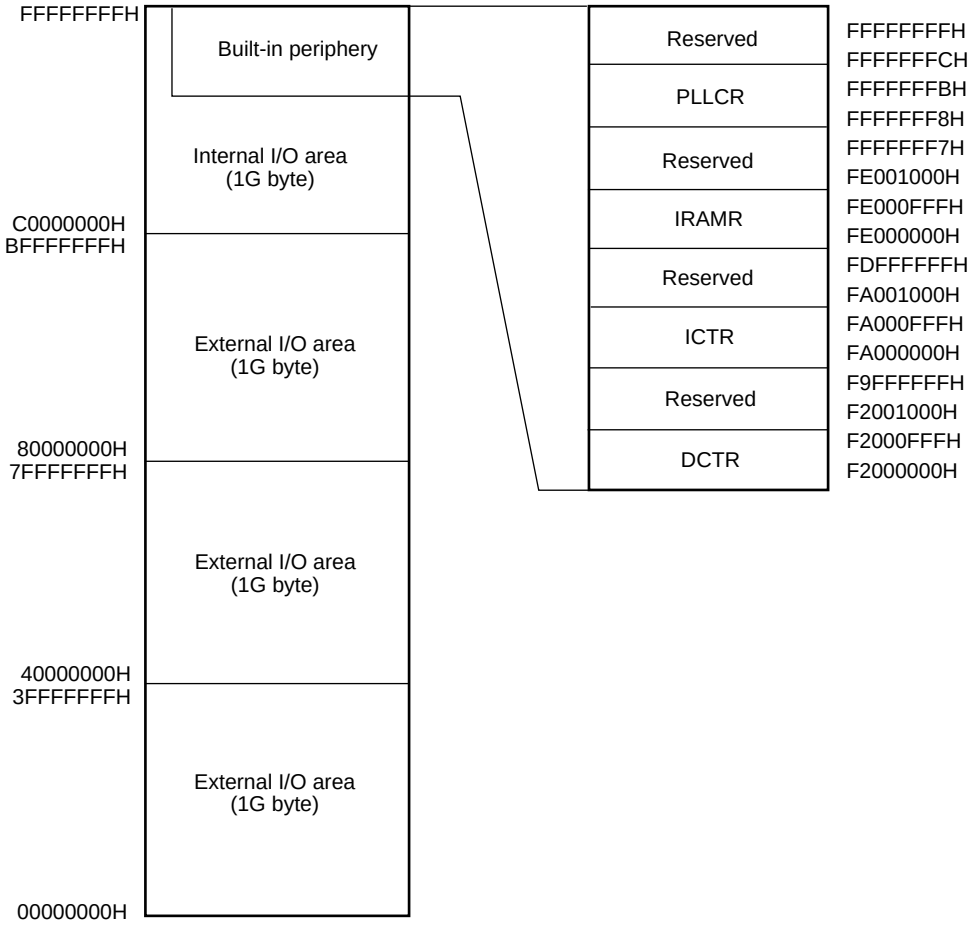
The cache function is not effective within the I/O space. When the chip select function is used, the area is used as the 256M-bytes image space represented by A2-A27.

Figure 3-3. Image Space Used When Chip Select Function is Used



The upper 1G-byte area (C0000000H-FFFFFFFFH) in the I/O space is reserved for internal I/O. To access internal I/O, the IN.W/OUT.W instructions (in words) must be used. When the internal I/O area is accessed, an external bus cycle is not activated.

Figure 3-4. Internal I/O Area



[MEMO]

CHAPTER 4 32-BIT BUS MODE

If the SIZ16B input, sampled at reset, is inactive, the external bus width becomes 32 bits (32-bit bus mode). In this mode, $\overline{BE2}$/BH acts as $\overline{BE2}$ and $\overline{BE3}$/A1 acts as $\overline{BE3}$.

4.1 RELATIONSHIP BETWEEN EXTERNAL ACCESS AND DATA BUS

This section explains data I/O and the byte enable signals ($\overline{BEn}$) at external accesses.

4.1.1 Relationship between External Accesses and Byte Enable Signals

In 32-bit bus mode, $\overline{BE0}$-$\overline{BE3}$ are output. External accesses are related to byte enable signals as indicated below.

Table 4-1. 32-Bit Bus Mode

Data length	Operand address		Byte enable				State
	Bit 1	Bit 0	$\overline{BE3}$	$\overline{BE2}$	$\overline{BE1}$	$\overline{BE0}$	
Byte	0	0	1	1	1	0	Ta,Ts
	0	1	1	1	0	1	Ta,Ts
	1	0	1	0	1	1	Ta,Ts
	1	1	0	1	1	1	Ta,Ts
Halfword	0	0	1	1	0	0	Ta,Ts
	1	0	0	0	1	1	Ta,Ts
Word	0	0	0	0	0	0	Ta,Ts
Burst transfer	0	0	0	0	0	0	Ta,Tb1
	0	0	0	0	0	0	Tb2
	0	0	0	0	0	0	Tb3
	0	0	0	0	0	0	Tb4

4.1.2 Operand Read

When an operand is being read, the relationship between the 32-bit data bus and internal registers is as shown below.

In Figure 4-1, Bm indicates that the m-th byte of the external data bus has been read. The relationship between Bm and the external data bus is as follows:

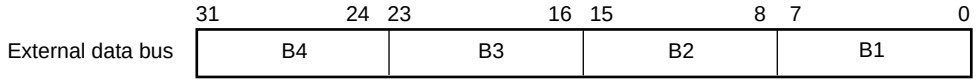


Figure 4-1. Read Cycle (32-Bit Bus Mode)

(a) Byte (8 bits)

Operand address		Internal registers				
Bit 1	Bit 0	31		16 15		0
0	0				B1	
0	1				B2	
1	0				B3	
1	1				B4	

(b) Halfword (16 bits)

Operand address		Internal registers				
Bit 1	Bit 0	31		16 15		0
0	0			B2	B1	
1	0			B4	B3	

(c) Word (32 bits)

Operand address		Internal registers				
Bit 1	Bit 0	31		16 15		0
0	0	B4		B3	B2	B1

4.1.3 Operand Write

When an operand is being written, the relationship between the 32-bit data bus and internal registers is as shown below.

In Figure 4-2, OPm indicates the byte position in the internal register. The relationship between OPm and the internal register is as follows:

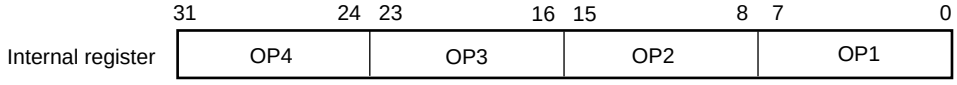


Figure 4-2. Write Cycle (32-Bit Bus Mode)

(a) Byte (8 bits)

Operand address		External data bus			
Bit 1	Bit 0	31	16	15	0
0	0				OP1
0	1			OP1	
1	0		OP1		
1	1	OP1			

(b) Halfword (16 bits)

Operand address		Internal registers			
Bit 1	Bit 0	31	16	15	0
0	0			OP2	OP1
1	0	OP2	OP1		

(c) Word (32 bits)

Operand address		External data bus			
Bit 1	Bit 0	31	16	15	0
0	0	OP4	OP3	OP2	OP1

4.2 BUS CYCLES

Bus cycles can be classified as being either single transfer or burst transfer bus cycles.

Single transfer bus cycles are used for the IN, OUT, and ST instructions, for the LD instruction when executed on an uncachable area, and for instruction fetches. In single transfer mode, data is read or written in single units. Single transfer mode can be activated in either the memory space or I/O space.

Burst transfer bus cycles are used for refilling the cache after a cache miss and for executing the BILD, BIST, BDLD, and BDST instructions. Burst transfer mode can be set only in the memory space, not in the I/O space. In burst transfer mode, four words (16 bytes) of data can be processed in each bus cycle, but the low-order four bits of the source and destination addresses must be 0.

4.2.1 Bus States

The V830 handles data accesses with a bus width of either 32 or 16 bits. A width is selected when the V830 is reset. The bus width cannot be changed during operation. In 32-bit bus mode, the bus assumes one of the states explained below:

In 32-bit bus mode, the bus cycles can assume one of eight states. The status signal (ST0-ST3) always changes in sync with the bus clock. Note, however, that it can also change at times other than changes in the bus clock.

(1) Ti state

The bus remains in the Ti state provided it does not receive a request to perform an access to a point external to the processor. The bus also enters this state after exiting from the hold state (Th state). In the Ti state, the $\overline{\text{BCYST}}$, $\overline{\text{CSn}}$, and other outputs are inactive. $\overline{\text{HLDRQ}}$ (input) and the interrupt request (input) are sampled in sync with the rising edge of the bus clock.

(2) Ta state

The bus enters the Ta state at the beginning of a bus cycle. In this state, $\overline{\text{BCYST}}$ (output) is active. A valid address is placed on the address bus at the rising edge of the bus clock. During a single write cycle, valid data is placed on the data bus at the rising edge of the bus clock. During a burst write cycle, data is not placed on the data bus.

Upon the termination of the Ta state, the bus enters either the Ts state in single transfer mode or the Tb1 state in burst transfer mode.

(3) Ts state

The bus enters Ts state upon the termination of a bus cycle or when the bus is forced to wait during a single transfer. $\overline{\text{HLDRQ}}$ (input) is sampled at the rising edge of the bus clock during the last Ts state. $\overline{\text{READY}}$ (input) is sampled at the rising edge of the bus clock upon the termination of the Ts state. If the sampled $\overline{\text{READY}}$ (input) is inactive, the bus re-enters the Ts state.

During a read, $\overline{\text{READY}}$ (input) is sampled. If it is active, the data on the data bus is read and the read cycle ends. The bus then enters the Ti state.

(4) Tb1-Tb4 states

The bus enters the Tbn state upon the termination of the n-th access to a point external to the processor, or when it is forced to wait. $\overline{\text{READY}}$ (input) is sampled at the first rising edge of the bus clock after termination of the state. If the sampled $\overline{\text{READY}}$ (input) is inactive, the bus re-enters the Tbn state.

If $\overline{\text{READY}}$ (input), sampled during a read, is active, the data on the data bus is read and the state terminates. If the bus is originally in the Tb4 state, however, it enters the Ti state.

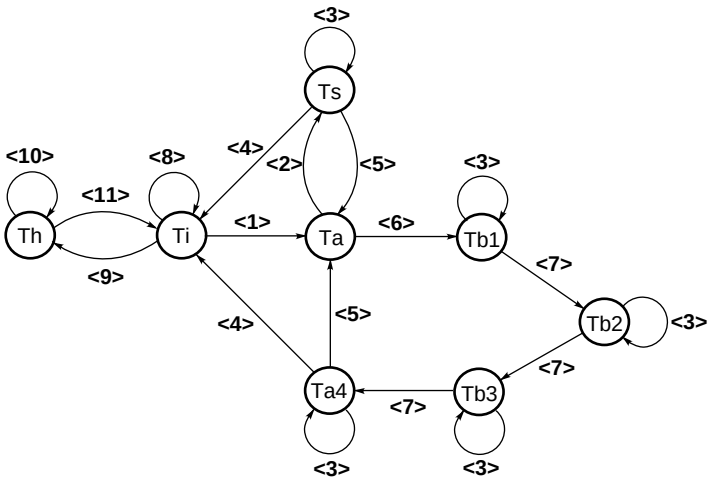
During a write in this state, write data is output in sync with the rising edge of the bus clock.

$\overline{\text{HLDRQ}}$ (input) is sampled at the rising edge of the bus clock in the last Tb4 state.

(5) Th state

The bus enters the Th state when it is held by $\overline{\text{HLDRQ}}$ (input). In this state, $\overline{\text{HLDRQ}}$ (input) is sampled at the rising edge of the bus clock. If it is inactive, the bus enters the Ti state.

Figure 4-3. Bus Cycle State Transitions (32-Bit Bus Mode)



<1> ($\overline{\text{HLDRQ}}$=1 or bus locked) and (cause of access)
<2> Single transfer
<3> $\overline{\text{READY}}$=1
<4> $\overline{\text{READY}}$=0 and ($\overline{\text{HLDRQ}}$=0 or [no cause of access] or [read bus cycle in effect] or [bus locked])
<5> $\overline{\text{READY}}$=0 and ($\overline{\text{HLDRQ}}$=1 and [cause of access] and [write bus cycle in effect] and [bus not locked])
<6> Burst transfer
<7> $\overline{\text{READY}}$=0
<8> $\overline{\text{HLDRQ}}$=1 and (no cause of access)
<9> $\overline{\text{HLDRQ}}$=0 and (bus not locked)
<10> $\overline{\text{HLDRQ}}$=0
<11> $\overline{\text{HLDRQ}}$=1

4.2.2 Memory Access and I/O Access

(1) Single read cycle

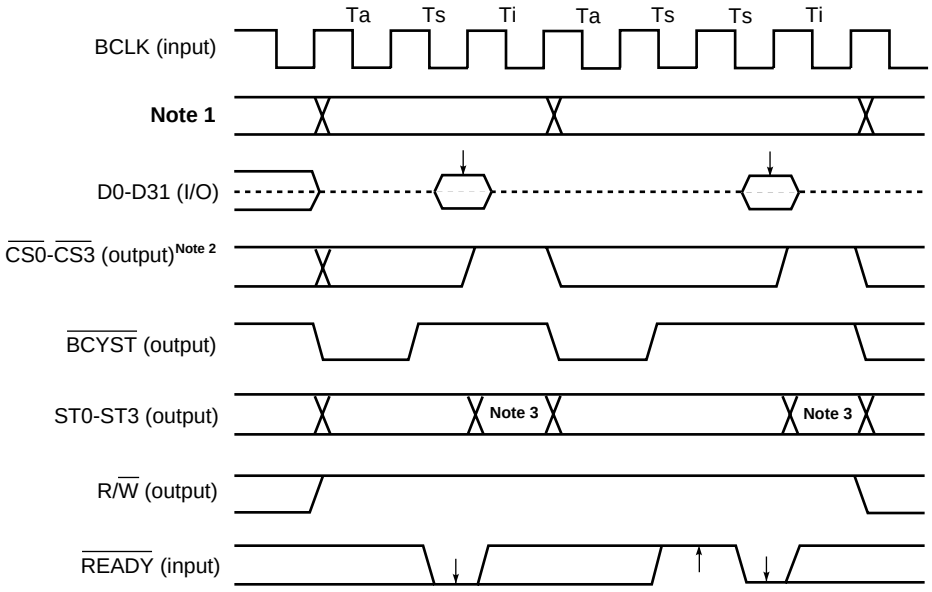
Single read cycles can be classified into memory read and I/O read cycles. Between these two types of read cycles, only the status signal (ST0-ST3) is different.

The rising edge of the bus clock in the Ta state triggers the output of the address and the activation of $\overline{\text{BCYST}}$ to signify the beginning of a bus cycle. For access to a space which involves chip select output, $\overline{\text{CSn}}$ is also output at the same timing.

The bus then enters the Ts state. In this state, the address is output as is and $\overline{\text{BCYST}}$ becomes inactive. $\overline{\text{READY}}$ is sampled at the first rising edge of the bus clock after the end of the Ts state. If $\overline{\text{READY}}$ is inactive, the bus re-enters the Ts state (wait). Thereafter, each time the Ts state ends, $\overline{\text{READY}}$ is sampled. The Ts state is continues for as long as $\overline{\text{READY}}$ is inactive.

Once $\overline{\text{READY}}$ becomes active, the read data will be sampled at the rising edge of the bus clock. Immediately after a single read cycle, an idle state (Ti) is always inserted.

Figure 4-4. Single Read Cycle (32-Bit Bus Mode)



Notes 1. A2-A27 (output) and $\overline{\text{BE0}}$-$\overline{\text{BE3}}$ (output)

2. If the chip select function is not used, A28-A31 are output to $\overline{\text{CS0}}$-$\overline{\text{CS3}}$.

3. Idle state

Remarks 1. A broken line indicates high impedance.

2. An arrow indicates a sampling timing.

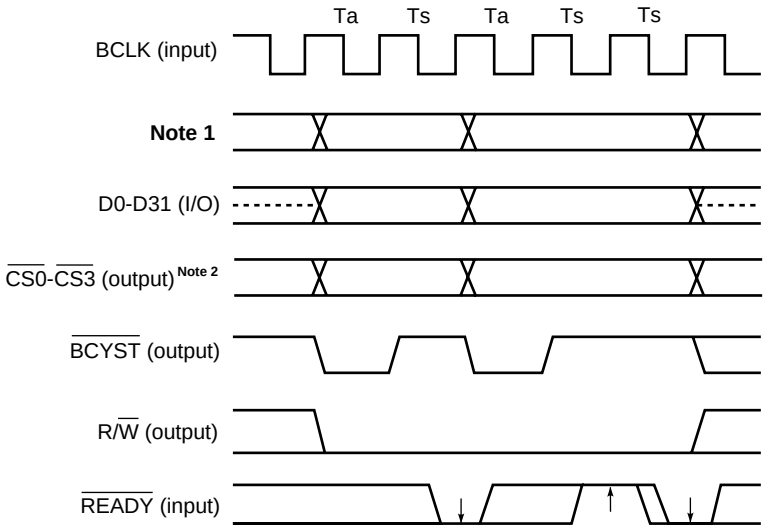
(2) Single write cycle

Single write cycles are classified into memory write and I/O write cycles. Between these two types of write cycles, only the status signal (ST0-ST3) is different.

The rising edge of the bus clock in the Ta state triggers the output of the address and write data and the activation of $\overline{\text{BCYST}}$ to signify the beginning of a bus cycle. For access to a space which involves chip select output, $\overline{\text{CSn}}$ is also output at the same timing.

The bus then enters the Ts state. In this state, the address and write data are output as is and $\overline{\text{BCYST}}$ becomes inactive. $\overline{\text{READY}}$ is sampled at the first rising edge of the bus clock after the end of the Ts state. If $\overline{\text{READY}}$ is inactive, the bus re-enters the Ts state (wait). Thereafter, each time the Ts state ends, $\overline{\text{READY}}$ is sampled. The Ts state continues as long as $\overline{\text{READY}}$ is inactive. The output write data is held until the bus next enters the Ta or Th state.

Figure 4-5. Single Write Cycle (32-Bit Bus Mode)



Notes 1. A2-A27 (output), $\overline{\text{BE0}}$-$\overline{\text{BE3}}$ (output), and ST0-ST3 (output)

2. If the chip select function is not used, A28-A31 are output to $\overline{\text{CS0}}$-$\overline{\text{CS3}}$.

Remarks 1. A broken line indicates high impedance.

2. An arrow indicates a sampling timing.

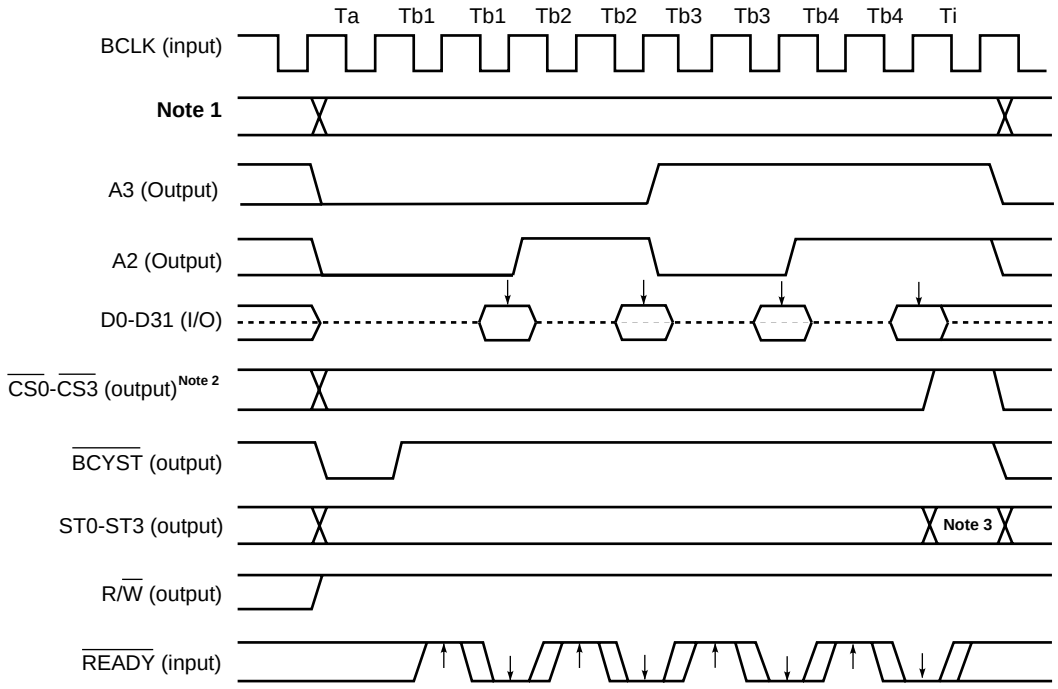
(3) Burst read cycle

Burst read cycles are used to transfer data from a memory space to built-in memory in four-word (16-byte) units. They cannot be used to access an I/O space.

The rising edge of the bus clock in the Ta state triggers the output of the address and the activation of $\overline{\text{BCYST}}$ to signify the beginning of a bus cycle. For access to a space which involves chip select output, $\overline{\text{CSn}}$ is also output at the same timing.

The bus then enters the Tb1 state. In this state, the address is output as is and $\overline{\text{BCYST}}$ becomes inactive. $\overline{\text{READY}}$ is sampled at the rising edge of the first bus clock after the end of the Tb1 state. If $\overline{\text{READY}}$ is inactive, the bus re-enters the Tb1 state (wait). Thereafter, each time the Tb1 state ends, $\overline{\text{READY}}$ is sampled. Once $\overline{\text{READY}}$ becomes active, the rising edge of the bus clock triggers the sampling of the read data, as well as transition to the Tb2 state. In the Tb2 state, the address is output at the rising edge of the bus clock and $\overline{\text{READY}}$ and the read data are sampled at the same timing as in the Tb1 state. The same timing also applies to the subsequent Tb3 and Tb4 states. An idle state (Ti) is always inserted immediately after a burst read cycle.

Figure 4-6. Burst Read Cycle (32-Bit Bus Mode)



Notes 1. A4-A27 (output) and $\overline{\text{BE0}}$-$\overline{\text{BE3}}$ (output)

2. If the chip select function is not used, A28-A31 are output to $\overline{\text{CS0}}$-$\overline{\text{CS3}}$.

3. Idle state

Remarks 1. A broken line indicates high impedance.

2. An arrow indicates a sampling timing.

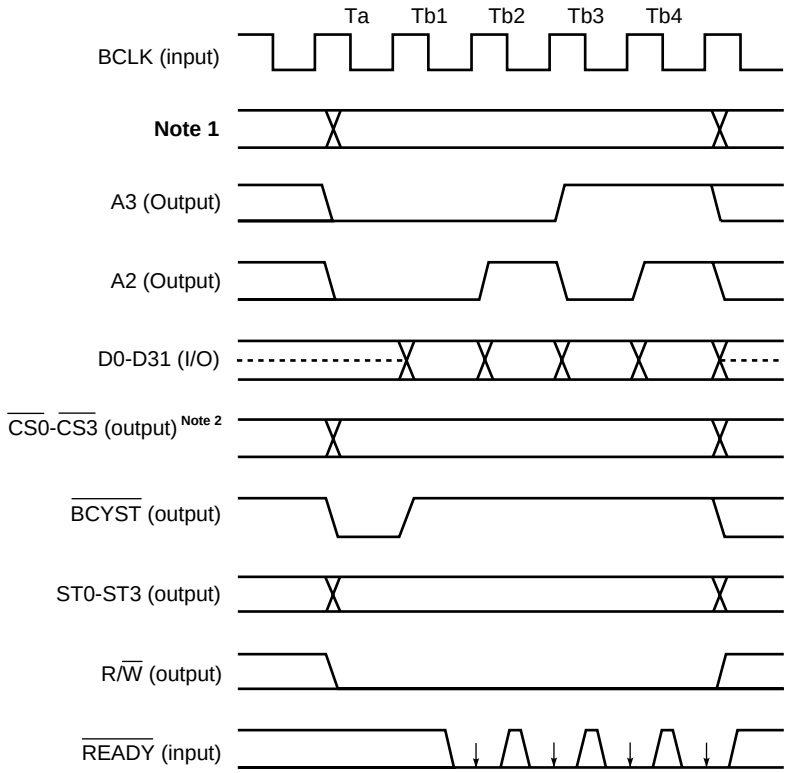
(4) Burst write cycle

Burst write cycles are used to transfer data from built-in memory to a memory space in four-word (16-byte) units. They cannot be used to access an I/O space.

The rising edge of the bus clock in the Ta state triggers the output of the address and the activation of $\overline{\text{BCYST}}$ to signify the beginning of a bus cycle. For access to a space which involves chip select output, $\overline{\text{CSn}}$ is also output at the same timing.

The bus then enters the Tb1 state. In this state, the address is output as is and $\overline{\text{BCYST}}$ becomes inactive. Also, the write data is output at the rising edge of the bus clock. $\overline{\text{READY}}$ is sampled at the rising edge of the first bus clock after the end of the Tb1 state. If $\overline{\text{READY}}$ is inactive, the bus enters the Tb1 state (wait) again. Thereafter, each time the Tb1 state ends, $\overline{\text{READY}}$ is sampled. When $\overline{\text{READY}}$ becomes active, the bus enters the Tb2 state. In the Tb2 state, the address and write data are output at the rising edge of the bus clock and $\overline{\text{READY}}$ is sampled at the same timing as in the Tb1 state. The same timing also applies to the subsequent Tb3 and Tb4 states.

Figure 4-7. Burst Write Cycle (32-Bit Bus Mode)



Notes 1. A4-A27 (output) and $\overline{\text{BE0}}$-$\overline{\text{BE3}}$ (output)

2. If the chip select function is not used, A28-A31 are output to $\overline{\text{CS0}}$-$\overline{\text{CS3}}$.

Remarks 1. A broken line indicates high impedance.

2. An arrow indicates a sampling timing.

4.3 CONTROL SIGNAL TIMING

This section explains bus lock timing and bus hold timing.

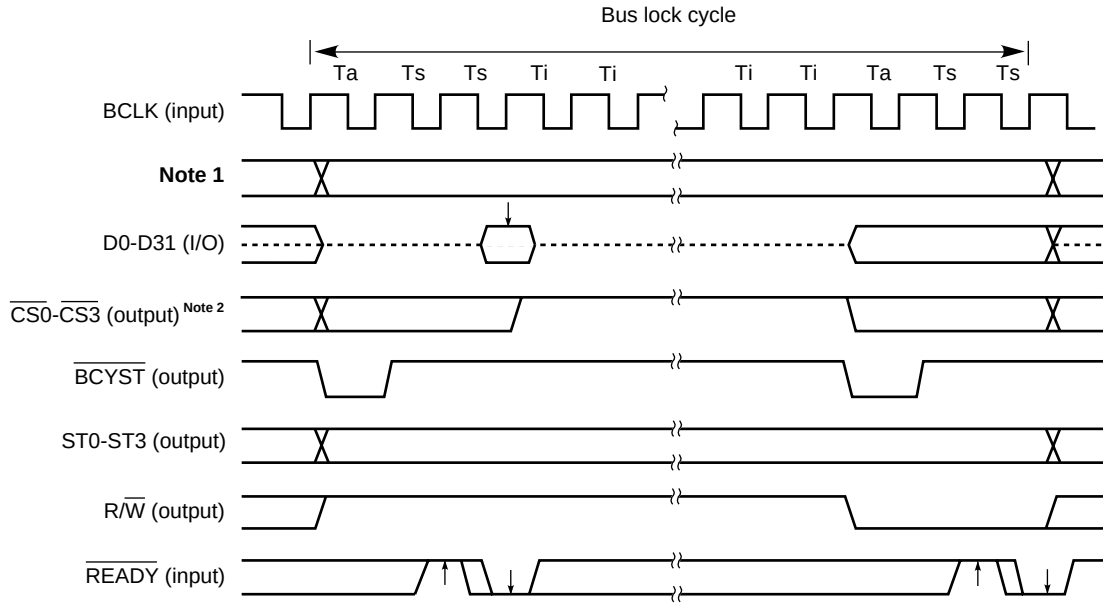
4.3.1 Bus Lock

The bus can be locked by encoding its status to prevent, its use by a non-V830 bus master. Bus locking is applied whenever a CAXI instruction is executed. During lock word access with the CAXI instruction, the status is as follows:

- The bus locked status is output to the status lines in sync with the beginning of a read cycle ($\overline{\text{BCYST}}$ active).
- The bus locked status changes in sync with the end of the last write cycle.

The bus lock timing chart (CAXI instruction execution) is given below.

Figure 4-8. Bus Lock Cycle (32-Bit Bus Mode)



Notes 1. A2-A31 (output) and $\overline{\text{BE0}}$-$\overline{\text{BE3}}$ (output)

2. If the chip select function is not used, A28-A31 are output to $\overline{\text{CS0}}$-$\overline{\text{CS3}}$.

Remarks 1. A broken line indicates high impedance.

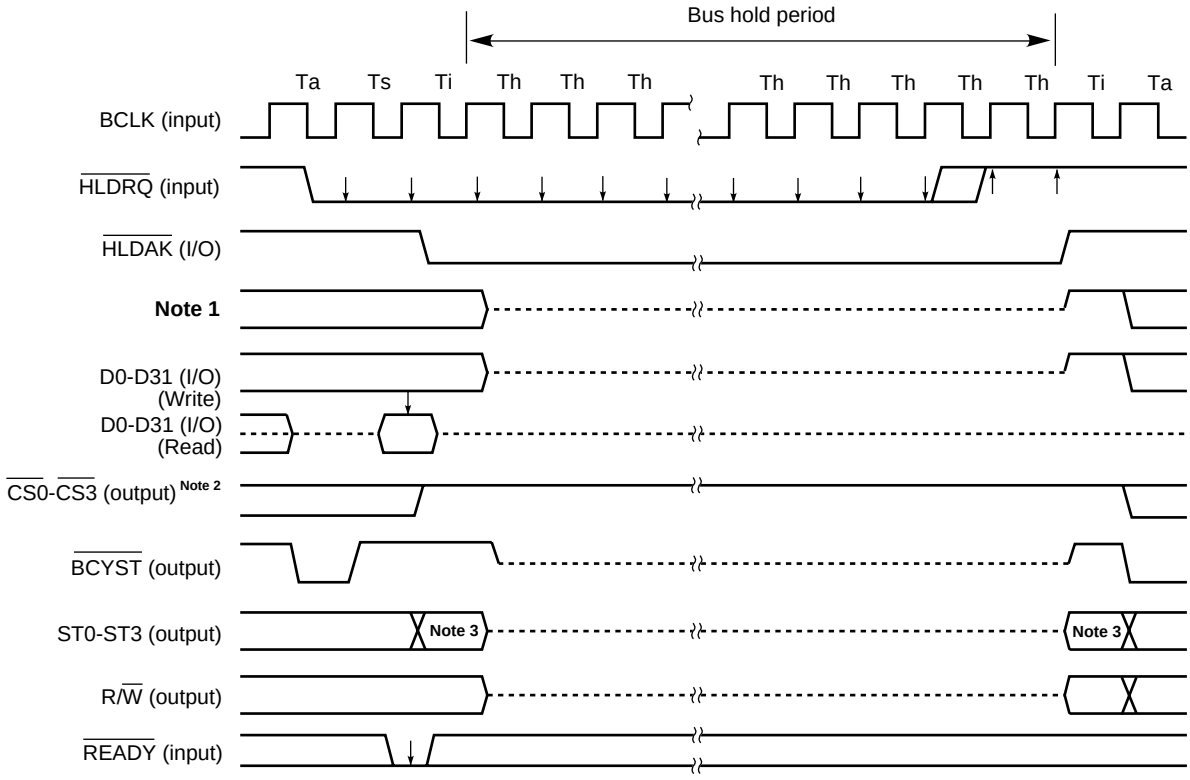
2. An arrow indicates a sampling timing.

4.3.2 Bus Hold

A bus master other than the V830 can use $\overline{\text{HLDRQ}}$ to request bus mastership from the V830. In 32-bit bus mode, $\overline{\text{HLDRQ}}$ is sampled at the rising edge of the bus clock in the Ts or Tb4 state and $\overline{\text{READY}}$ (input) is sampled when that state ends. If both signals are active, the bus enters the Ti state. In this state, $\overline{\text{HLDAK}}$ is activated and $\overline{\text{HLDRQ}}$ is again sampled at the rising edge of bus clock. If $\overline{\text{HLDRQ}}$ is active, the bus enters the Th state (bus hold state).

In the Th state, the bus is floating. Provided $\overline{\text{HLDRQ}}$ remains active, the bus will remain in the Th state. When $\overline{\text{HLDRQ}}$ becomes inactive, the bus exits from the Th state such that the V830 regains bus mastership and enters the Ti state. While the bus is locked, it does not accept bus hold requests until immediately before the last bus cycle. In the last bus cycle, the bus accepts a bus hold request.

Figure 4-9. Bus Hold



Notes 1. A2-A27 (output) and $\overline{\text{BE0}}$-$\overline{\text{BE3}}$ (output)

2. If the chip select function is not used, A28-A31 are output to $\overline{\text{CS0}}$-$\overline{\text{CS3}}$.

3. Idle state

Remarks 1. A broken line indicates high impedance.

2. An arrow indicates a sampling timing.

4.4 WRITE BUFFER

The V830 has a four-stage write buffer (capable of holding up to 16 bytes of data).

When a store instruction is executed, the write data is buffered and written to external memory. After the CPU has transferred data to the write buffer, the next instruction is processed. If, however, the write buffer is full, the CPU is halted until the write cycle for the data at the head of the write buffer ends, allowing data to be written to the write buffer. When accessing internal memory, however, instruction execution does not stop because the write buffer is not used.

To assure the instruction execution sequence, instruction execution is halted in the following cases, not resuming until the contents of the write buffer have all been written to external memory:

- Data cache miss/hit is encountered during load instruction execution
- An uncachable area is accessed during load/store instruction execution
- I/O access instruction
- Block transfer instruction
- HALT instruction
- STBY instruction
- CAXI instruction

A bus hold operation (triggered by $\overline{\text{HLDRQ}}$) is performed regardless of the state of the write buffer.

CHAPTER 5 16-BIT BUS MODE

If the SIZ16B input, sampled at reset, is active, the external bus width becomes 16 bits (16-bit bus mode). In this mode, the low-order 16 bits (D0-D15) of the data bus are valid, $\overline{BE2}$/BH acts as BH and $\overline{BE3}$/A1 acts as A1. The high-order 16 bits (D16-D31) of the data bus enter the high-impedance state.

5.1 16-BIT BUS SIZING

The V830 has a bus sizing function by which, to enable access from the data bus to 16 bits of memory or the I/O space, data can be transferred using only the low-order 16 bits of the 32-bit data bus.

When the SIZ16B input is activated upon a reset, the external data bus width becomes 16 bits (16-bit bus mode). In 16-bit bus mode, D16-D31 are all set to the high-impedance state and $\overline{BE0}$, $\overline{BE1}$, BH, and A1 are output in a way suited to a 16-bit bus system. Connection to D16-D31 is not necessary. The SIZ16B input can be changed only when the V830 is reset. It cannot be changed at any other time.

5.1.1 Byte/Halfword Access

Bus cycles in either of two bus states (Ta and Ts) are used for byte/halfword access.

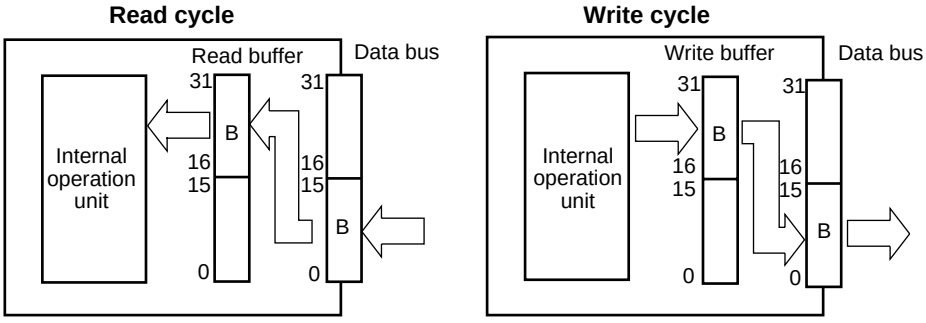
(1) Upper halfword

During read cycles, data is read from D0-D15.

During write cycles, D16-D31 data read from the write buffer is output to D0-D15.

Figure 5-1 illustrates the operation for upper halfword access. In this figure, B indicates the upper halfword (high-order 16 bits of the word).

Figure 5-1. Upper Halfword Access



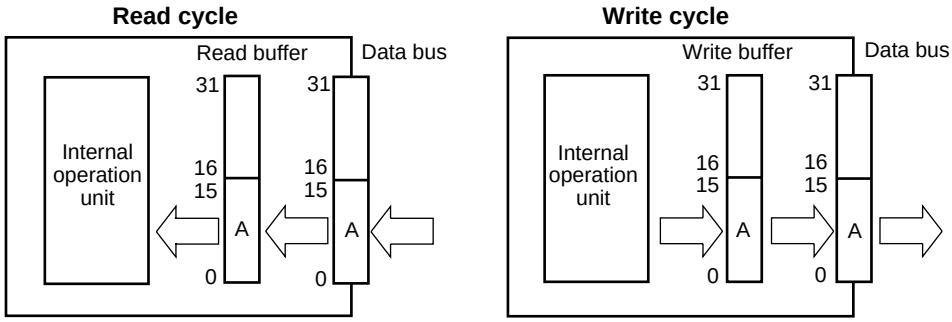
(2) Lower halfword

During read cycles, data is read from D0-D15.

During write cycles, D0-D15 data read from the write buffer is output to D0-D15.

Figure 5-2 shows the operation for lower halfword access. In this figure, A indicates the lower halfword (low-order 16 bits of the word).

Figure 5-2. Lower Halfword Access



5.1.2 Word Access

Bus cycles in any of three bus states (Ta, Tw1, and Tw2) are used for word access.

During a read cycle, the low-order 16 bits of data and high-order 16 bits of data are sampled from D0-D15 in the Tw1 and Tw2 state, respectively. During write cycles, the low-order 16 bits of data and high-order 16 bits of data are output to D0-D15 in the Ta/Tw1 state and Tw2 states, respectively.

Figure 5-3. Read Cycle

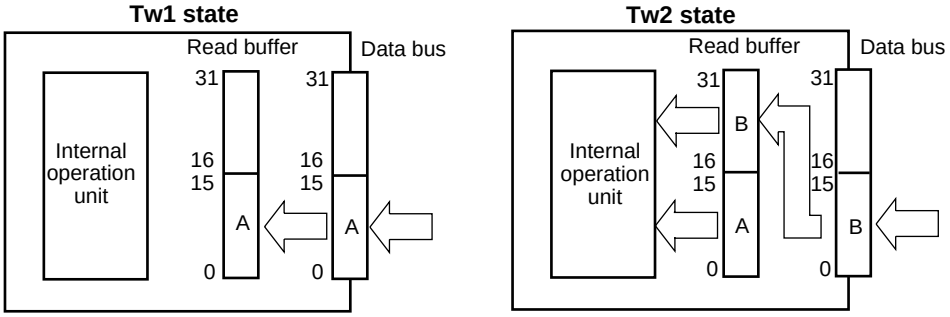
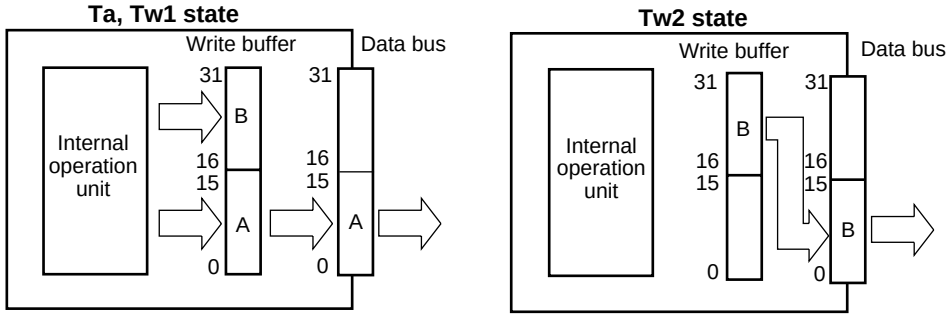


Figure 5-4. Write Cycle



5.2 RELATIONSHIP BETWEEN EXTERNAL ACCESS AND DATA BUS

This section explains data I/O and the byte enable signals ($\overline{BEn}$) for external access.

5.2.1 Relationship between External Access and Byte Enable Signals

In 16-bit bus mode, the $\overline{BE3}$/A1 output acts as A1 and $\overline{BE2}$/BH output acts as BH. External accesses are related to the byte enable signals as indicated below.

Table 5-1. 16-Bit Bus Mode

Data length	Operand address		Byte enable				State
	Bit 1	Bit 0	A1	BH	$\overline{BE1}$	$\overline{BE0}$	
Byte	0	0	0	1	1	0	Ta,Ts
	0	1	0	1	0	1	Ta,Ts
	1	0	1	1	1	0	Ta,Ts
	1	1	1	1	0	1	Ta,Ts
Halfword	0	0	0	1	0	0	Ta,Ts
	1	0	1	1	0	0	Ta,Ts
Word	0	0	0	0	0	0	Ta,Tw1
			1	0	0	0	Tw2
Burst transfer	0	0	0	0	0	0	Ta,Tb1
			1	0	0	0	Tb2
			0	0	0	0	Tb3
			1	0	0	0	Tb4
			0	0	0	0	Tb5
			1	0	0	0	Tb6
			0	0	0	0	Tb7
			1	0	0	0	Tb8

5.2.2 Operand Read

While an operand is being read, the relationship between the 16-bit data bus and internal registers is as explained below.

In Figure 5-5, n;Bm indicates that the m-th byte of the external data bus has been read at the n-th external access. The relationship between Bm and the external data bus is as follows:

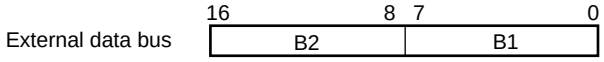


Figure 5-5. Read Cycle (16-Bit Bus Mode)

(a) Byte (8 bits)

Operand address		Internal registers			
Bit 1	Bit 0	31	16	15	0
0	0				1;B1
0	1				1;B2
1	0				1;B1
1	1				1;B2

(b) Halfword (16 bits)

Operand address		Internal registers			
Bit 1	Bit 0	31	16	15	0
0	0			1;B2	1;B1
1	0			1;B2	1;B1

(c) Word (32 bits)

Operand address		Internal registers				
Bit 1	Bit 0	31	16	15	0	
0	0			1;B2	1;B1	Tw1 state
		2;B4	2;B3			Tw2 state

5.2.3 Operand Write

While an operand is being written, the relationship between the 16-bit data bus and internal registers is as explained below.

In Figure 5-6, n;OPm indicates that the m-th byte of data from the internal register has been output to the data bus at the n-th external access. The relationship between OPm and the internal register is as follows:

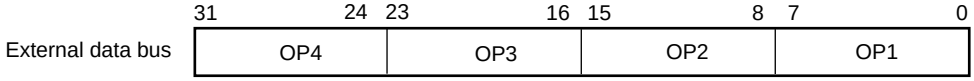


Figure 5-6. Write Cycle (16-Bit Bus Mode)

(a) Byte (8 bits)

Operand address Bit 1	Operand address Bit 0	External data bus 15–8	External data bus 7–0
0	0		1;OP1
0	1	1;OP1	
1	0		1;OP1
1	1	1;OP1	

(b) Halfword (16 bits)

Operand address Bit 1	Operand address Bit 0	External data bus 15–8	External data bus 7–0
0	0	1;OP2	1;OP1
1	0	1;OP2	1;OP1

(c) Word (32 bits)

Operand address Bit 1	Operand address Bit 0	External data bus 15–8	External data bus 7–0	
0	0	1;OP2	1;OP1	Tw1 state
		2;OP4	2;OP3	Tw2 state

5.3 BUS CYCLES

Bus cycles can be classified into either single transfer or burst transfer bus cycles.

Single transfer bus cycles are used for the IN, OUT, and ST instructions, for the LD instruction when executed on an uncachable area, and upon instruction fetches. In single transfer mode, data is read or written in single units. Single transfer mode can be activated in either the memory space or I/O space.

Burst transfer bus cycles are used for refilling the cache after a cache miss and for executing the BILD, BIST, BDLD, and BDST instructions. Burst transfer mode can be activated only in the memory space, but in the I/O space. In burst transfer mode, four words (16 bytes) of data can be processed in each bus cycle, but the low-order four bits of the source and destination addresses must be 0.

5.3.1 Bus States

The V830 handles data accesses with the bus width fixed to either 32 or 16 bits. A width is selected when the V830 is reset. The bus width cannot be changed during operation. In 16-bit bus mode, the bus assumes one of the states explained below:

In 16-bit bus mode, the bus cycles can assume one of 14 states. During the bus cycle for word access, two external accesses are performed. The state always changes in sync with the bus clock. Note, however, that it can also change at times either than changes in the bus clock.

(1) Ti state

The bus remains in the Ti state provided it does not receive a request to perform an access to a point external to the processor. It also enters this state after exiting from the hold state (Th state). In the Ti state, the $\overline{\text{BCYST}}$, $\overline{\text{CSn}}$, and other outputs are inactive. $\overline{\text{HLDRQ}}$ (input) and the interrupt request (input) are sampled in sync with the rising edge of the bus clock.

(2) Ta state

The bus enters the Ta state at the beginning of a bus cycle. In this state, $\overline{\text{BCYST}}$ (output) is active. A valid address is placed on the address bus at the rising edge of the bus clock. During a single write cycle, valid data is placed on the data bus at the rising edge of the bus clock. During a burst write cycle, data is not placed on the data bus.

Upon the termination of the Ta state, the bus enters the Ts state in single byte/halfword transfer mode, the Tw1 state in single word transfer mode, or the Tb1 state in burst transfer mode.

(3) Ts state

The bus enters the Ts state upon the termination of a bus cycle or when the bus is forced to wait during a single byte/halfword transfer. $\overline{\text{HLDRQ}}$ (input) is sampled at the rising edge of the bus clock during the last Ts state. $\overline{\text{READY}}$ (input) is sampled at the rising edge of the bus clock upon the termination of the Ts state. If the sampled $\overline{\text{READY}}$ (input) is inactive, the bus re-enters the Ts state starts.

If $\overline{\text{READY}}$ (input), sampled during a read, is active, the data on the data bus is read and the read cycle ends. The bus then enters the Ti state.

(4) Tw1 state

The bus enters the Tw1 state upon the termination of the first access in the bus cycle or when the bus is forced to wait during a single word transfer. $\overline{\text{READY}}$ (input) is sampled at the rising edge of the bus clock upon termination of the Tw1 state. If the sampled $\overline{\text{READY}}$ (input) is inactive, the bus re-enters the Tw1 state. If $\overline{\text{READY}}$ (input) sampled during a read is active, data is read from the data bus and the Tw1 state ends.

(5) Tw2 state

The bus enters the Tw2 state upon the termination of the second access in the bus cycle or when the bus is forced to wait during a single word transfer. $\overline{\text{HLDRQ}}$ (input) is sampled at the rising edge of the bus clock in this state. $\overline{\text{READY}}$ (input) is sampled at the rising edge of the bus clock after termination of the Tw2 state. If the sampled $\overline{\text{READY}}$ (input) is inactive, the bus re-enters the Tw1 state and the sampled $\overline{\text{HLDRQ}}$ (input) is ignored.

If $\overline{\text{READY}}$ (input), sampled during a read, is active, data is read from the data bus and the Tw2 state terminals. During a write, write data is output to the data bus in sync with the rising edge of the bus clock.

(6) Tb1-Tb8 states

The bus enters the Tbn state upon the termination of the n-th access to a point external to the processor, or when it is forced to wait. $\overline{\text{READY}}$ (input) is sampled at the first rising edge of the bus clock after termination of the state. If the sampled $\overline{\text{READY}}$ (input) is inactive, the bus re-enters the Tbn state.

If $\overline{\text{READY}}$ (input), sampled during a read, is active, the data on the data bus is read and the state terminals. If the bus is originally in the Tb8 state, however, it enters the Ti state.

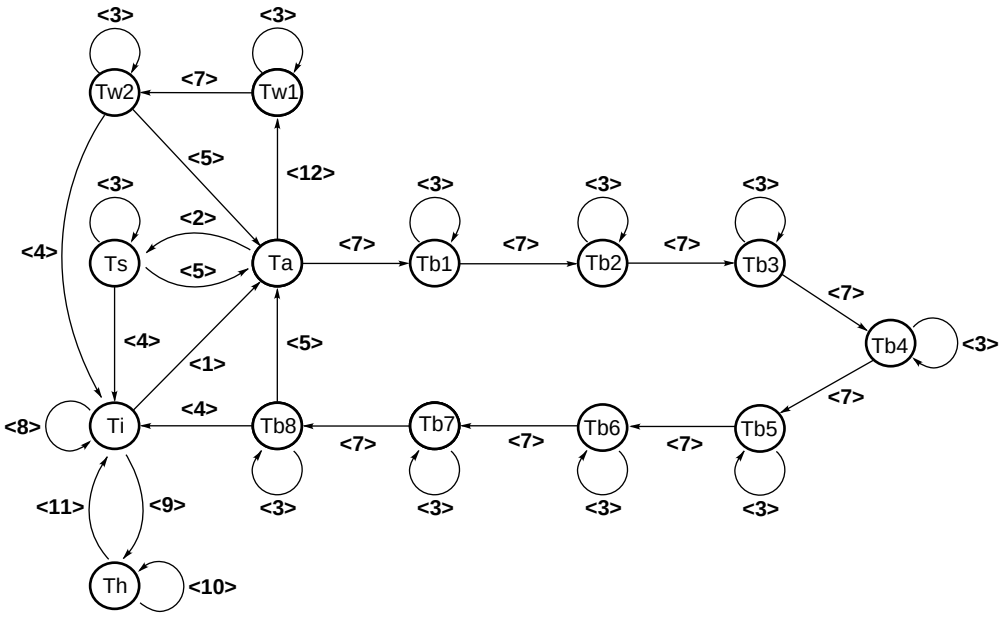
During a write in this state, write data is output in sync with the rising edge of the bus clock.

$\overline{\text{HLDRQ}}$ (input) is sampled at the rising edge of the bus clock in the last Tb8 state.

(7) Th state

The bus enters the Th state when it is held by $\overline{\text{HLDRQ}}$ (input). In this state, $\overline{\text{HLDRQ}}$ (input) is sampled at the rising edge of the bus clock. If it is inactive, the bus enters the Ti state.

Figure 5-7. Bus Cycle State Transitions (16-Bit Bus Mode)



<1> ($\overline{\text{HLDRQ}}$=1 or bus locked) and (cause of access)
<2> Single byte/halfword transfer
<3> $\overline{\text{READY}}$=1
<4> $\overline{\text{READY}}$=0 and ($\overline{\text{HLDRQ}}$=0 or [no cause of access] or [read bus cycle in effect] or [bus locked])
<5> $\overline{\text{READY}}$=0 and ($\overline{\text{HLDRQ}}$=1 and [cause of access] and [write bus cycle in effect] and [bus not locked])
<6> Burst transfer
<7> $\overline{\text{READY}}$=0
<8> $\overline{\text{HLDRQ}}$=1 and (no cause of access)
<9> $\overline{\text{HLDRQ}}$=0 and (bus not locked)
<10> $\overline{\text{HLDRQ}}$=0
<11> $\overline{\text{HLDRQ}}$=1
<12> Single word transfer

5.3.2 Memory Access and I/O Access

(1) Single read cycle

Single read cycles can be classified into memory read and I/O read cycles. Between these two types of read cycles, only the status signal (ST0-ST3) is different. Also, the timing is differs slightly between byte/halfword access and word access.

For byte/halfword access, the rising edge of the bus clock in the Ta state triggers the output of the address and the activation of $\overline{\text{BCYST}}$ to signify the beginning of a bus cycle. For access to a space which involves chip select output, $\overline{\text{CSn}}$ is also output at the same timing.

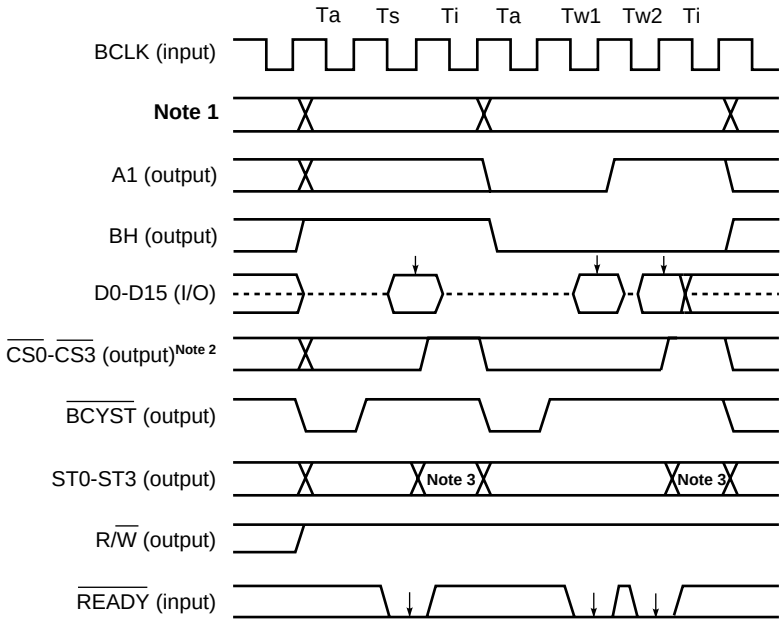
The bus then enters to the Ts state. In this state, the address is output as is and $\overline{\text{BCYST}}$ becomes inactive. $\overline{\text{READY}}$ is sampled at the first rising edge of the bus clock after the end of the Ts state. If $\overline{\text{READY}}$ is inactive, the bus re-enters the Ts state (wait). Thereafter, each time the Ts state ends, $\overline{\text{READY}}$ is sampled. The Ts state continues for as long as $\overline{\text{READY}}$ is inactive.

Once $\overline{\text{READY}}$ becomes active, the read data will be sampled at the rising edge of the bus clock.

For word access, the address is output when the bus clock rises in the Tw2 state. Otherwise, Tw1 and Tw2 work for timing the same way as Ts for byte/halfword access.

Immediately after a single read cycle, an idle state (Ti) is always inserted.

Figure 5-8. Single Read Cycle (16-Bit Bus Mode)



Notes 1. A2-A27 (output) and $\overline{\text{BE0}}$-$\overline{\text{BE3}}$ (output)

2. If the chip select function is not used, A28-A31 are output to $\overline{\text{CS0}}$-$\overline{\text{CS3}}$.

3. Idle state

Remarks 1. A broken line indicates high impedance.

2. An arrow indicates a sampling timing.

(2) Single write cycle

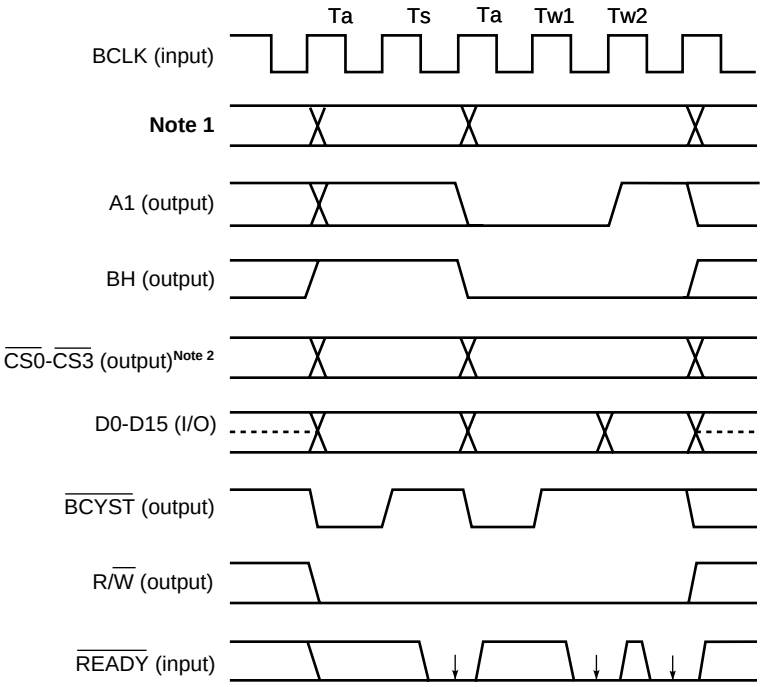
Single write cycles are classified into memory write and I/O write cycles. Between these two types of write cycles, only the status signal (ST0-ST3) is different. Also, the timing differs slightly between byte/halfword access and word access.

For byte/halfword access, the rising edge of the bus clock in the Ta state triggers the output of the address and write data and the activation of $\overline{\text{BCYST}}$ to signify the beginning of a bus cycle. For access to a space which involves chip select output, $\overline{\text{CSn}}$ is also output at the same timing.

The bus then enters the Ts state. In this state, the address and write data are output as is and $\overline{\text{BCYST}}$ becomes inactive. $\overline{\text{READY}}$ is sampled at the first rising edge of the bus clock after the end of the Ts state. If $\overline{\text{READY}}$ is inactive, the bus re-enters the Ts state (wait). Thereafter, each time the Ts state ends, $\overline{\text{READY}}$ is sampled. The Ts state continues as long as $\overline{\text{READY}}$ is inactive.

For word access, the address and data are output at the rising edge of the bus clock in the Tw2 state. Otherwise, Tw1 and Tw2 work for timing in the same way as Ts for byte/halfword access.

Figure 5-9. Single Write Cycle (16-Bit Bus Mode)



Notes 1. A2-A27 (output), $\overline{\text{BE0}}$-$\overline{\text{BE3}}$ (output), and ST0-ST3 (output)

2. If the chip select function is not used, A28-A31 are output to $\overline{\text{CS0}}$-$\overline{\text{CS3}}$.

Remarks 1. A broken line indicates high impedance.

2. An arrow indicates a sampling timing.

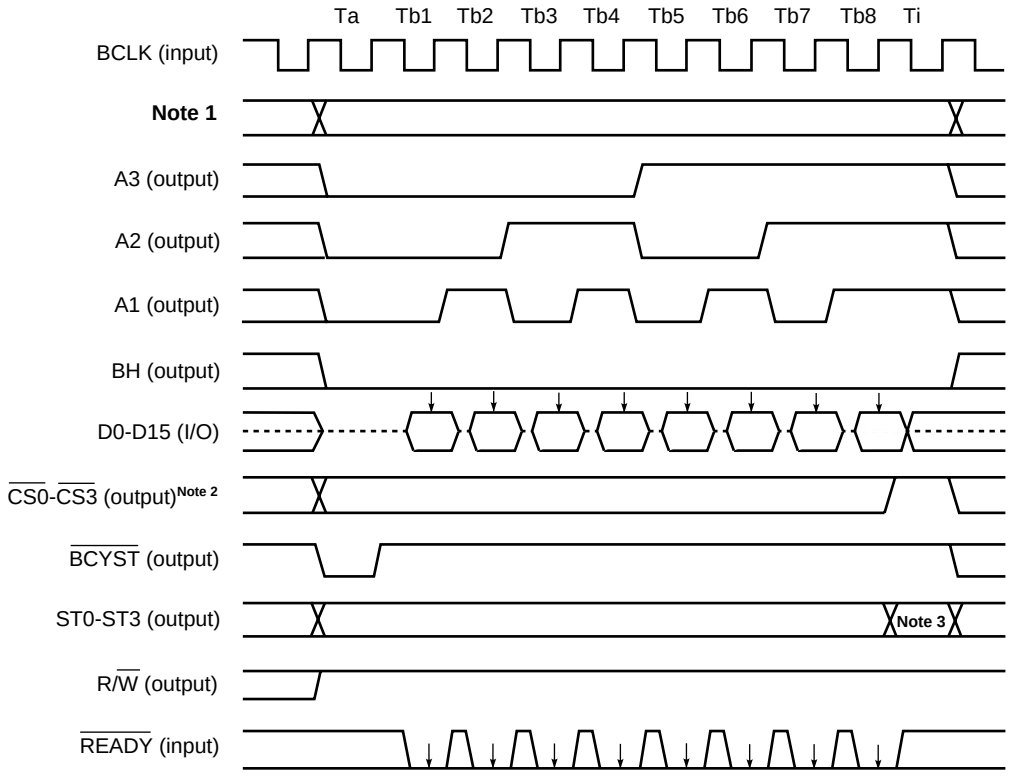
(3) Burst read cycle

Burst read cycles are used to transfer data from a memory space to built-in memory in four-word (16-byte) units. They cannot be used to access an I/O space.

The rising edge of the bus clock in the Ta state triggers the output of the address and the activation of $\overline{\text{BCYST}}$ to signify the beginning of a bus cycle. For access to a space which involves chip select output, $\overline{\text{CSn}}$ is also output at the same timing.

The bus then enters to the Tb1 state. In this state, the address is output as is and $\overline{\text{BCYST}}$ becomes inactive. $\overline{\text{READY}}$ is sampled at the rising edge of the first bus clock after the end of the Tb1 state. If $\overline{\text{READY}}$ is inactive, the bus re-enters the Tb1 state (wait). Thereafter, each time the Tb1 state ends, $\overline{\text{READY}}$ is sampled. Once $\overline{\text{READY}}$ becomes active, the rising edge of the bus clock triggers the sampling of the read data, as well as transition to the Tb2 state. In the Tb2 state, the address is output at the rising edge of the bus clock and $\overline{\text{READY}}$ and the read data are sampled at the same timing as in the Tb1 state. The same timing also applies to the subsequent Tb3 to Tb8 states. An idle state (Ti) is always inserted immediately after a burst read cycle.

Figure 5-10. Burst Read Cycle (16-Bit Bus Mode)



Notes 1. A4-A27 (output) and $\overline{\text{BE0}}$-$\overline{\text{BE3}}$ (output)

2. If the chip select function is not used, A28-A31 are output to $\overline{\text{CS0}}$-$\overline{\text{CS3}}$.

3. Idle state

Remarks 1. A broken line indicates high impedance.

2. An arrow indicates a sampling timing.

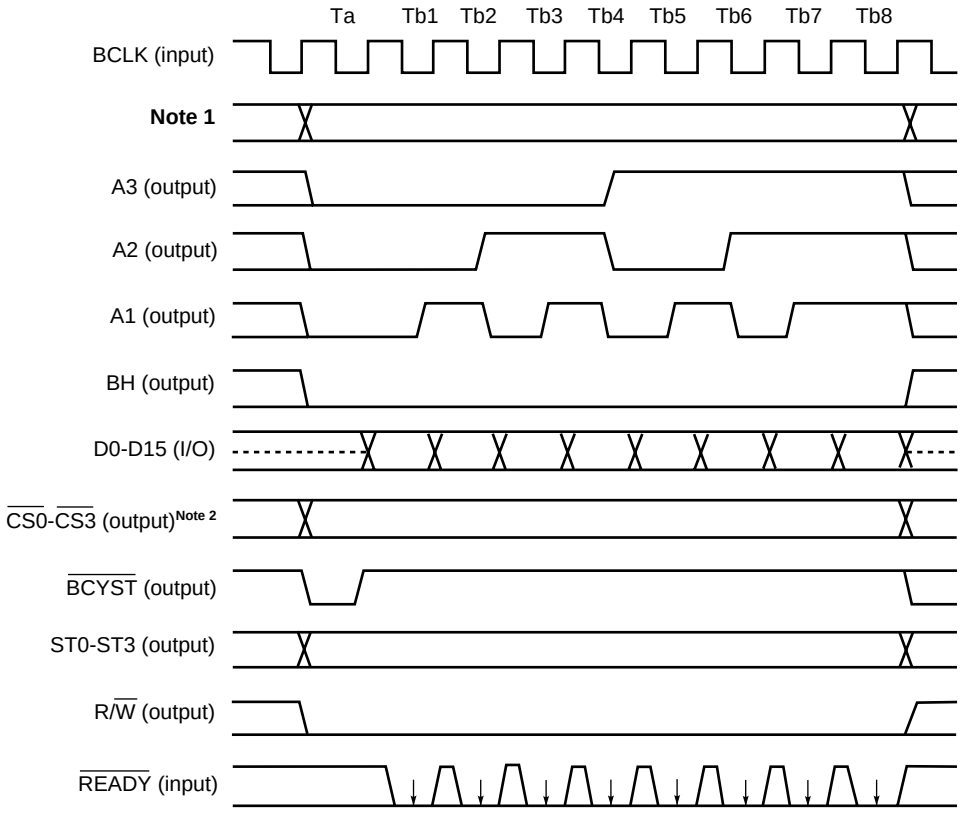
(4) Burst write cycle

Burst write cycles are used to transfer data from built-in memory to a memory space in four-word (16-byte) units. They cannot be used to access an I/O space.

The rising edge of the bus clock in the Ta state triggers the output of the address and the activation of $\overline{\text{BCYST}}$ to signify the beginning of a bus cycle. For access to a space which involves chip select output, $\overline{\text{CSn}}$ is also output at the same timing.

The bus then enters to the Tb1 state. In this state, the address is output as is and $\overline{\text{BCYST}}$ becomes inactive. Also, the write data is output at the rising edge of the bus clock. $\overline{\text{READY}}$ is sampled at the rising edge of the first bus clock after the end of the Tb1 state. If $\overline{\text{READY}}$ is inactive, the bus enters the Tb1 state (wait) again. Thereafter, each time the Tb1 state ends, $\overline{\text{READY}}$ is sampled. When $\overline{\text{READY}}$ becomes active, the bus enters the Tb2 state. In the Tb2 state, the address and write data are output on the rising edge of the bus clock and $\overline{\text{READY}}$ is sampled at the same timing as in the Tb1 state. The same timing also applies to the subsequent Tb3 to Tb8 states.

Figure 5-11. Burst Write Cycle (16-Bit Bus Mode)



Notes 1. A4-A27 (output) and $\overline{\text{BE0}}$-$\overline{\text{BE3}}$ (output)

2. If the chip select function is not used, A28-A31 are output to $\overline{\text{CS0}}$-$\overline{\text{CS3}}$.

Remarks 1. A broken line indicates high impedance.

2. An arrow indicates a sampling timing.

5.4 CONTROL SIGNAL TIMING

This section explains bus lock timing and bus hold timing.

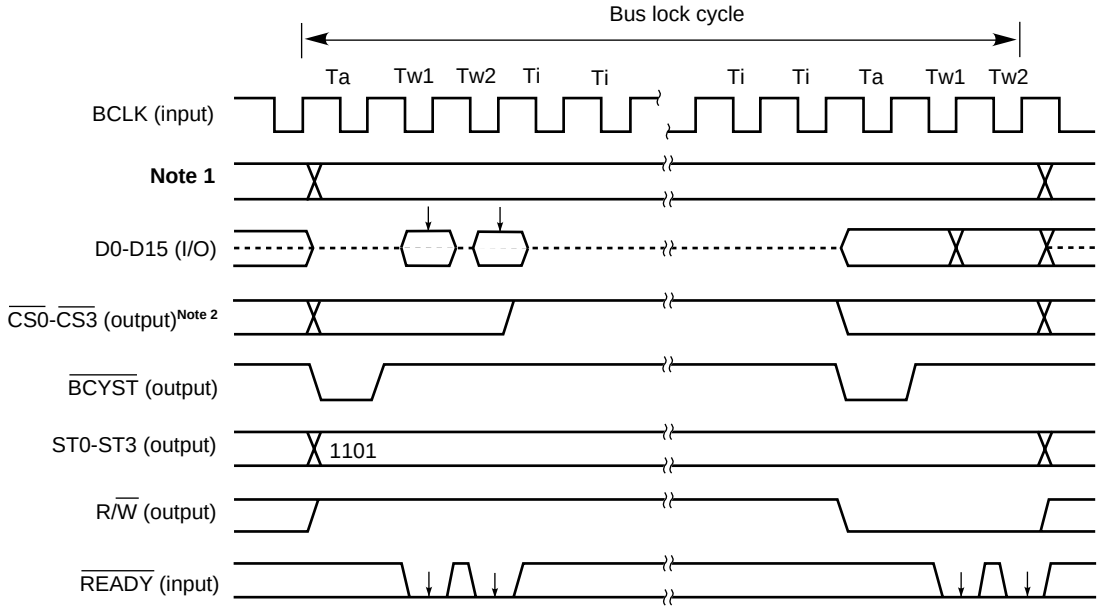
5.4.1 Bus Lock

The bus can be locked by encoding its status to prevent its use by a non-V830 bus master. Bus locking is applied whenever a CAXI instruction is executed. During lock word access with the CAXI instruction, the status is as follows:

- The bus locked status is output to the status lines in sync with the beginning of a read cycle ($\overline{\text{BCYST}}$ active).
- The bus locked status changes in sync with the end of the last write cycle.

The bus lock timing chart (CAXI instruction execution) is given below. In 16-bit bus mode, the read/write cycle consists of Ta, Tw1, and Tw2 states.

Figure 5-12. Bus Lock Cycle (16-Bit Bus Mode)



Notes 1. A2-A31 (output) and $\overline{\text{BE0}}$-$\overline{\text{BE3}}$ (output)

2. If the chip select function is not used, A28-A31 are output to $\overline{\text{CS0}}$-$\overline{\text{CS3}}$.

Remarks 1. A broken line indicates high impedance.

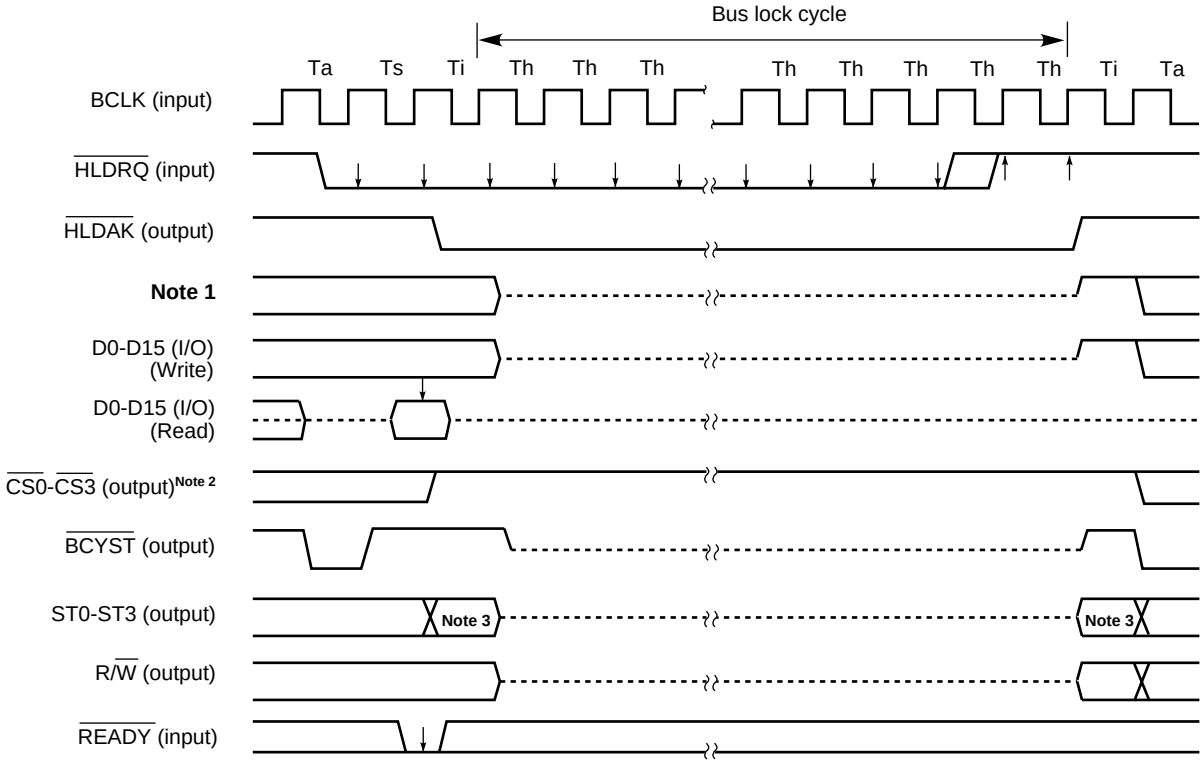
2. An arrow indicates a sampling timing.

5.4.2 Bus Hold

A bus master other than the V830 can use $\overline{\text{HLDRQ}}$ to request bus mastership from the V830. In 16-bit bus mode, $\overline{\text{HLDRQ}}$ is sampled at the rising edge of the bus clock in the Ts, Tw2, or Tb8 state and $\overline{\text{READY}}$ (input) is sampled when that state ends. If both signals are active, the bus enters the Ti state. In this state, $\overline{\text{HLDAK}}$ is activated and $\overline{\text{HLDRQ}}$ is again sampled at the rising edge of the bus clock. If $\overline{\text{HLDRQ}}$ is active, the bus enters the Th state (bus hold state).

In the Th state, the bus is floating. Provided $\overline{\text{HLDRQ}}$ remains active, the bus will remain in the Th state. When $\overline{\text{HLDRQ}}$ becomes inactive, the bus exits from the Th state such that the V830 regains bus mastership and enters the Ti state. While the bus is locked, it does not accept bus hold requests until immediately before the last bus cycle. In the last bus cycle, the bus accepts a bus hold request.

Figure 5-13. Bus Hold



Notes 1. A2-A27 (output) and $\overline{\text{BE0}}$-$\overline{\text{BE3}}$ (output)

2. If the chip select function is not used, A28-A31 are output to $\overline{\text{CS0}}$-$\overline{\text{CS3}}$.

3. Idle state

Remarks 1. A broken line indicates high impedance.

2. An arrow indicates a sampling timing.

5.5 WRITE BUFFER

The V830 has a four-stage write buffer (capable of holding up to 16 bytes of data).

When a store instruction is executed, the write data is buffered and written to external memory. After the CPU has transferred data to the write buffer, the next instruction is processed. If, however, the write buffer is full, the CPU is halted until the write cycle for the data at the head of the write buffer ends, allowing data to be written to the write buffer. When accessing internal memory, however, instruction execution does not stop because the write buffer is not used.

To assure the instruction execution sequence, instruction execution is halted in the following cases, not resuming until the contents of the write buffer have all been written to external memory:

- Data cache miss/hit is encountered during load instruction execution
- An uncachable area is accessed during load/store instruction execution
- I/O access instruction
- Block transfer instruction
- HALT instruction
- STBY instruction
- CAXI instruction

A bus hold operation (triggered by $\overline{\text{HLDRQ}}$) is performed regardless of the state of the write buffer.

[MEMO]

CHAPTER 6 INTERRUPTS

V830 interrupts include maskable interrupts, nonmaskable interrupts, and reset operations.

6.1 MASKABLE INTERRUPTS

Maskable interrupt requests are themselves denoted by INT, and their interrupt levels by $\overline{\text{INTV0}}$ to $\overline{\text{INTV3}}$. The following lists pin states and the corresponding interrupt levels.

Table 6-1. Interrupt Levels

Interrupt level	$\overline{\text{INTV3}}$	$\overline{\text{INTV2}}$	$\overline{\text{INTV1}}$	$\overline{\text{INTV0}}$
15	0	0	0	0
14	0	0	0	1
13	0	0	1	0
12	0	0	1	1
11	0	1	0	0
10	0	1	0	1
9	0	1	1	0
8	0	1	1	1
7	1	0	0	0
6	1	0	0	1
5	1	0	1	0
4	1	0	1	1
3	1	1	0	0
2	1	1	0	1
1	1	1	1	0
0	1	1	1	1

INT and $\overline{\text{INTV0}}$ to $\overline{\text{INTV3}}$ are level inputs. The V830 samples an INT at the rising edge of a bus clock pulse. INT and $\overline{\text{INTV0}}$ to $\overline{\text{INTV3}}$ should be held at the active level until the V830 accepts the interrupt request and posts to a peripheral, by software, notification of the acceptance of the interrupt request. Although a change to a higher interrupt level is possible, the timing at which an interrupt request is detected cannot then be posted to peripheral.

Hence, an interrupt request made before such a change may be accepted. If the interrupt request input (INT, $\overline{\text{INTV0}}$-$\overline{\text{INTV3}}$) fails to satisfy the setup time requirement for the bus clock pulse, the interrupt request will be detected at the rising edge of the next bus clock pulse.

Upon accepting an interrupt request, the V830 jumps to a fixed address to start interrupt handling. The target address of the jump is set to FE0000n0H (built-in RAM) or FFFFFEn0H (external memory), where n is the interrupt level, either of which may be specified with the IHA bit of the system register, HCCW.

Caution Interrupt level 15 is reserved for use by development tools (in-circuit emulator, ROM emulator, etc). If the user uses interrupt level 15, those development tools may fail to operate.

6.2 NONMASKABLE INTERRUPTS

The V830 samples an $\overline{\text{NMI}}$ at the rising edge of a bus clock pulse. When the $\overline{\text{NMI}}$ changes from the high to low level, an interrupt request is detected. Once a nonmaskable interrupt request has been detected, the $\overline{\text{NMI}}$ can subsequently be deactivated at any time because the NMI is detected at the falling edge. An interrupt request thus detected is retained in the CPU until the CPU starts interrupt handling.

Upon accepting a nonmaskable interrupt, the V830 jumps to the fixed address (FFFFFFD0H). If another nonmaskable interrupt is issued during nonmaskable interrupt handling (the NP bit of PSW is set to 1), it is retained in the processor. If, however another nonmaskable interrupt request is issued during clearing of the latch circuit by internal processing after the start of nonmaskable interrupt handling, it is not retained in the processor.

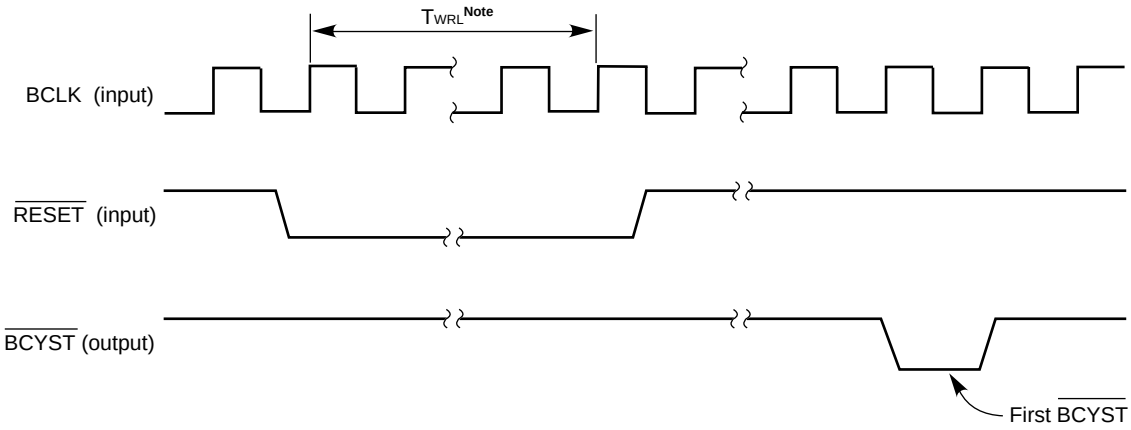
6.3 RESET

The V830 can be reset by inputting a low-level signal of 20 or more clock pulses to $\overline{\text{RESET}}$. After the V830 has been reset, the CPU starts program execution from address FFFFFFF0H.

If $\overline{\text{RESET}}$ is driven high, the CPU starts instruction fetching from the reset address.

Immediately after power-on or in the stop-mode state, the active pulse width of the $\overline{\text{RESET}}$ should be determined by adding the PLL oscillation settling time to the active level of 20 clock pulses.

Figure 6-1. Reset



Note The minimum value of T_{WRL} (the period during which the $\overline{RESET}$ remains low) depends on the following states:

- At power-on or upon return from a stop state (with a STBY instruction)
  : 10 ms (oscillation settling time of external oscillator + PLL oscillation settling time of V830)
- When normal (other than the above)
  : 20 clock pulses

The following lists the pin states existing immediately after a reset operation.

Table 6-2. Pin States Immediately After Reset

Pin	Pin state
D0-D31	High impedance
A31/$\overline{CS3}$	High level
A30/$\overline{CS2}$	
A29/$\overline{CS1}$	
A28/$\overline{CS0}$	
A3-A27	
$\overline{BE3}$/A1	
$\overline{BE2}$/BH	
$\overline{BE1}$,$\overline{BE0}$	
ST0-ST3	
R/$\overline{W}$	
$\overline{BCYST}$	
$\overline{HLDAK}$	

The internal registers are initialized to the following values.

Table 6-3. Initial Register Values

Register	Initial value
PC	FFFFFFF0H
r0	00000000H
r1-r31	Not defined
EIPC	
EIPSW	
FEPC	
FEPSW	
DPC	
DPSW	
ECR	0000FFF0H
PSW	00008000H
PIR	00008300H
TKCW	000000E0H
HCCW	00000000H
PLLCR	000000#0H
CMCR	00000000H
ICTR	IV0=0,IV1=0,ICTAG=Not defined
DCTR	DV=0,DCTAG=Not defined
IRAMR	Not defined

CHAPTER 7 CLOCK CONTROLLER

7.1 OPERATION MODES

The V830 supports two clock stop functions, namely, sleep mode and stop mode. Transition from one mode to another is made by executing special instructions HALT or STBY. The following lists the features of these modes:

Table 7-1. Operation Modes

	Sleep mode	Stop mode
V830 internal state	Internal clock stop PLL operation continuous Bus hold acceptable Built-in RAM/cache data hold	Internal clock stop PLL operation stop Bus hold unacceptable Built-in RAM/cache data hold
Entry to mode	HALT instruction	STBY instruction
Escape from mode	Maskable interrupt/NMI/reset	NMI/reset

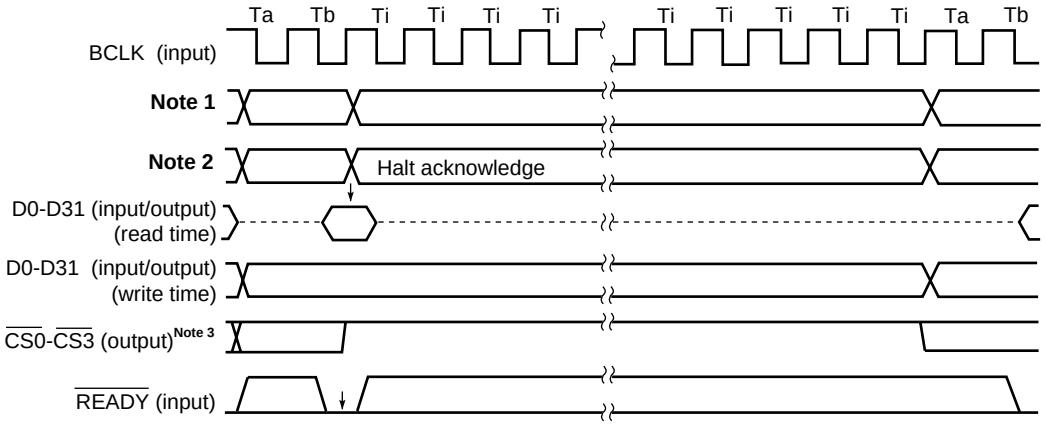
7.2 SLEEP MODE

The V830 enters sleep mode upon the execution of a HALT instruction. On the other hand, escape from sleep mode can be realized by a maskable interrupt, NMI, or reset operation.

In sleep mode, bus hold requests can be accepted. During bus hold, the status becomes high impedance and no halt acknowledge status is output. At the end of bus hold, a halt acknowledge status is output in sync with the rising edge of a bus clock pulse.

The timing in sleep mode is as shown below.

Figure 7-1. Halt Acknowledge Cycle



Notes 1. A2-A27 (output), BE0-BE3 (output)

2. ST0-ST3, R/W

3. If no chip select function is used, A28-A31 are output to CS0-CS3.

Remarks 1. A broken line indicates high impedance.

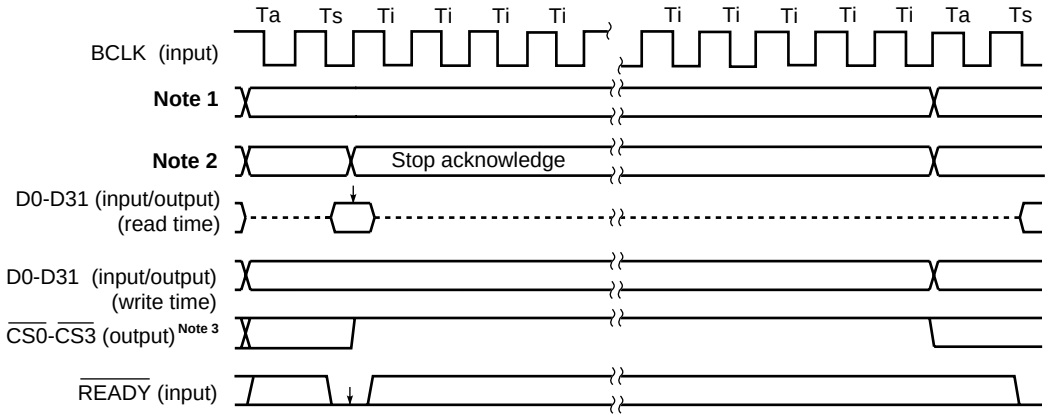
2. An arrow indicates a sampling timing.

7.3 STOP MODE

The V830 enters stop mode when an STBY instruction is executed. On the other hand, escape from stop mode can be realized using an NMI or a reset operation. The power consumption in stop mode is less than that in the sleep mode because the PLL circuit stops.

Also, no bus hold requests are accepted in the stop mode.

Figure 7-2. Stop Acknowledge Cycle



Notes 1. A2-A27 (output), BE0-BE3 (output)

2. ST0-ST3, R/W

3. If no chip select function is used, A28-A31 are output to CS0-CS3.

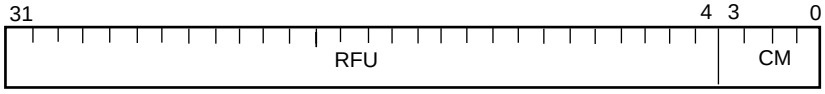
Remarks 1. A broken line indicates high impedance.

2. An arrow indicates a sampling timing.

7.4 PLL CONTROL REGISTER

The PLL control register indicates the state of the PLL circuit. This read-only register is mapped in the I/O space. To access this register, use the IN.W instruction. After being reset, this register will contain 00000001H or 00000002H.

PLLCR(FFFFFFF8H)



Bit location	Bit name	Meaning
31-4	RFU	Reserved field (fixed to 0.)
3-0	CM	Clock Mode Indicates the state of the CMODE signal sampled at reset.

CM	Internal clock: bus clock
0000	1:1
0001	2:1 (where CMODE = 1)
0010	3:1 (where CMODE = 0)
0011	4:1
01xx	RFU
1xxx	RFU

*

APPENDIX A INSTRUCTIONS (LISTED ALPHABETICALLY)

The instructions are listed below in alphabetic order of their mnemonics.

Explanation of list format

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
ADD	reg1, reg2	I	*	*	*	*	

Instruction mnemonic

Instruction format

Indicates how each flag changes.
- : Does not change.
* : Changes.
0 : Becomes 0.
1 : Becomes 1.

Abbreviations of operands

Abbreviation	Meaning
reg1	General-purpose register (used as a source register)
reg2	General-purpose register (used mainly as a destination register, but in some instructions, used as a source register)
reg3	General-purpose register (used mainly as a destination register, but in some instructions, used as a source register)
immX	X bits of immediate data
dispX	X-bit displacement
regID	System register number
vector adr	Trap handler address corresponding to trap vector

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
ABC	disp9	III	-	-	-	-	High-speed conditional branch (if Carry) relative to PC.
ABE	disp9	III	-	-	-	-	High-speed conditional branch (if Equal) relative to PC.
ABGE	disp9	III	-	-	-	-	High-speed conditional branch (if Greater than or Equal) relative to PC.
ABGT	disp9	III	-	-	-	-	High-speed conditional branch (if Greater than) relative to PC.
ABH	disp9	III	-	-	-	-	High-speed conditional branch (if Higher) relative to PC.
ABL	disp9	III	-	-	-	-	High-speed conditional branch (if Lower) relative to PC.
ABLE	disp9	III	-	-	-	-	High-speed conditional branch (if Less than or Equal) relative to PC.
ABLT	disp9	III	-	-	-	-	High-speed conditional branch (if Less than) relative to PC.
ABN	disp9	III	-	-	-	-	High-speed conditional branch (if Negative) relative to PC.
ABNC	disp9	III	-	-	-	-	High-speed conditional branch (if Not Carry) relative to PC.
ABNE	disp9	III	-	-	-	-	High-speed conditional branch (if Not Equal) relative to PC.
ABNH	disp9	III	-	-	-	-	High-speed conditional branch (if Not Higher) relative to PC.
ABNL	disp9	III	-	-	-	-	High-speed conditional branch (if Not Lower) relative to PC.
ABNV	disp9	III	-	-	-	-	High-speed conditional branch (if Not Overflow) relative to PC.
ABNZ	disp9	III	-	-	-	-	High-speed conditional branch (if Not Zero) relative to PC.
ABP	disp9	III	-	-	-	-	High-speed conditional branch (if Positive) relative to PC.
ABR	disp9	III	-	-	-	-	High-speed unconditional branch (Always) relative to PC.
ABV	disp9	III	-	-	-	-	High-speed conditional branch (if Overflow) relative to PC.
ABZ	disp9	III	-	-	-	-	High-speed conditional branch (if Zero) relative to PC.
ADD	reg1, reg2	I	*	*	*	*	Addition. reg1 is added to reg2 and the sum is written into reg2.
	imm5, reg2	II	*	*	*	*	Addition. imm5, sign-extended to a word, is added to reg2 and the sum is written into reg2.

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
ADDI	imm16, reg1, reg2	V	*	*	*	*	Addition. imm16, sign-extended to a word, is added to reg1, and the sum is written into reg2.
AND	reg1, reg2	I	-	0	*	*	AND. reg2 and reg1 are ANDed and the result is written into reg2.
ANDI	imm16, reg1, reg2	V	-	0	0	*	AND. reg1 is ANDed with imm16, zero-extended to a word, and result is written into reg2.
BC	disp9	III	-	-	-	-	Conditional branch (if Carry) relative to PC.
BDLD	[reg1], [reg2]	VII	-	-	-	-	Block transfer. 4 words of data are transferred from external memory to built-in data RAM.
BDST	[reg2], [reg1]	VII	-	-	-	-	Block transfer. 4 words of data are transferred from built-in data RAM to external memory.
BE	disp9	III	-	-	-	-	Conditional branch (if Equal) relative to PC.
BGE	disp9	III	-	-	-	-	Conditional branch (if Greater than or Equal) relative to PC.
BGT	disp9	III	-	-	-	-	Conditional branch (if Greater than) relative to PC.
BH	disp9	III	-	-	-	-	Conditional branch (if Higher) relative to PC.
BILD	[reg1], [reg2]	VII	-	-	-	-	Block transfer. 4 words of data are transferred from external memory to built-in instruction RAM.
BIST	[reg2], [reg1]	VII	-	-	-	-	Block transfer. 4 words of data are transferred from built-in instruction RAM to external memory.
BL	disp9	III	-	-	-	-	Conditional branch (if Lower) relative to PC.
BLE	disp9	III	-	-	-	-	Conditional branch (if Less than or Equal) relative to PC.
BLT	disp9	III	-	-	-	-	Conditional branch (if Less than) relative to PC.
BN	disp9	III	-	-	-	-	Conditional branch (if Negative) relative to PC.
BNC	disp9	III	-	-	-	-	Conditional branch (if Not Carry) relative to PC.
BNE	disp9	III	-	-	-	-	Conditional branch (if Not Equal) relative to PC.
BNH	disp9	III	-	-	-	-	Conditional branch (if Not Higher) relative to PC.
BNL	disp9	III	-	-	-	-	Conditional branch (if Not Lower) relative to PC.
BNV	disp9	III	-	-	-	-	Conditional branch (if Not Overflow) relative to PC.
BNZ	disp9	III	-	-	-	-	Conditional branch (if Not Zero) relative to PC.
BP	disp9	III	-	-	-	-	Conditional branch (if Positive) relative to PC.
BR	disp9	III	-	-	-	-	Unconditional branch (Always) relative to PC.
BRKRET		IX	-	-	-	-	Return from fatal exception handling
BV	disp9	III	-	-	-	-	Conditional branch (if Overflow) relative to PC.
BZ	disp9	III	-	-	-	-	Conditional branch (if Zero) relative to PC.

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
CAXI	disp16 [reg1], reg2	VI	*	*	*	*	Inter-processor synchronization in multi-processor system.
CMP	reg1, reg2	I	*	*	*	*	Comparison. reg2 is compared with reg1 sign-extended to a word and the condition flag is set according to the result. The comparison involves subtracting reg1 from reg2.
	imm5, rag2	II	*	*	*	*	Comparison. reg2 is compared with imm5 sign-extended to a word and the condition flag is set according to the result. The comparison involves subtracting imm5, sign-extended to a word, from reg2.
DI		II	-	-	-	-	Disable interrupt. Maskable interrupts are disabled. DI instruction cannot disable nonmaskable interrupts.
DIV	reg1, reg2	I	-	*	*	*	Division of signed operands. reg2 is divided by reg1 (signed operands). The quotient is stored in reg2 and the remainder in r30. The division is performed so that the sign of the remainder will match that of the dividend.
DIVU	reg1, reg2	I	-	0	*	*	Division of unsigned operands. reg2 is divided by reg1 (unsigned operands). The quotient is stored in reg2 and the remainder in r30. The division is performed so that the sign of the remainder will match that of the dividend.
EI		II	-	-	-	-	Enable interrupt. Maskable interrupts are enabled. The EI instruction cannot enable nonmaskable interrupts.
HALT		IX	-	-	-	-	Processor halt. The processor is placed in sleep mode.
IN.B	disp16 [reg1], reg2	VI	-	-	-	-	Port input. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit port address. A byte of data is read from the resulting resulting port address, zero-extended to a word, then stored in reg2.
IN.H	disp16[reg1], reg2	VI	-	-	-	-	Port input. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit port address. A halfword of data is read from the produced port address, zero-extended to a word, and stored in reg2. Bit 0 of the unsigned 32-bit port address is masked to 0.
IN.W	disp16[reg1], reg2	VI	-	-	-	-	Port input. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit port address. A word of data is read from the resulting port address, then written into reg2. Bits 0 and 1 of the unsigned 32-bit port address are masked to 0.

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
JAL	disp26	IV	-	-	-	-	Jump and link. The sum of the current PC and 4 is written into r31. disp26, sign-extended to a word, is added to the PC and the sum is set to the PC for control transfer. Bit 0 of disp26 is masked.
JMP	[reg1]	I	-	-	-	-	Indirect unconditional branch via register. Control is passed to the address designated by reg1. Bit 0 of the address is masked to 0.
JR	disp26	IV	-	-	-	-	Unconditional branch. disp26, sign-extended to a word, is added to the current PC and control is passed to the address specified by that sum. Bit 0 of disp26 is masked to 0.
LD.B	disp16[reg1], reg2	VI	-	-	-	-	Byte load. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit address. A byte of data is read from the produced address, sign-extended to a word, then written into reg2.
LD.H	disp16[reg1], reg2	VI	-	-	-	-	Halfword load. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit address. A halfword of data is read from the produced address, sign-extended to a word, then written into reg2. Bit 0 of the unsigned 32-bit address is masked to 0.
LD.W	disp16[reg1], reg2	VI	-	-	-	-	Word load. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit address. A word of data is read from the produced address, then written into reg2. Bits 0 and 1 of the unsigned 32-bit address are masked to 0.
LDSR	reg2, regID	II	*	*	*	*	Load into system register. The contents of reg2 are set in the system register identified by the system register number (regID).
MAC3	reg1, reg2, reg3	VIII	-	-	-	-	Saturatable operation on signed 32-bit operands. reg1 and reg2 are multiplied together as signed integers and the product is added to reg3. [If no overflow has occurred:] The result is stored in reg3. [If an overflow has occurred:] The SAT bit is set. If the result is positive, the positive maximum is written into reg3; if the result is negative, the negative maximum is written into reg3.

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
MACI	imm16, reg1, reg2	V	-	-	-	-	Saturatable operation on signed 32-bit operands. reg1 and imm16, sign-extended to 32 bits, are multiplied together as signed integers and the product is added to reg2 as a signed integer. [If no overflow has occurred:] The result is written into reg2. [If an overflow has occurred:] The SAT bit is set. If the result is positive, the positive maximum is written into reg2; if the result is negative, the negative maximum is written into reg2.
MACT3	reg1, reg2, reg3	VIII	-	-	-	-	Saturatable operation on signed 32-bit operands. reg1 and reg2 are multiplied together as signed integers and the high-order 32 bits of the product are added to reg3 as signed integers. [If no overflow has occurred:] The result is written into reg3. [If an overflow has occurred:] The SAT bit is set. If the result is positive, the positive maximum is written into reg3; if the result is negative, the negative maximum is written into reg3.
MAX3	reg1, reg2, reg3	VIII	-	-	-	-	Maximum. reg2 and reg1 are compared as signed integers. The larger value is written into reg3.
MIN3	reg1, reg2, reg3	VIII	-	-	-	-	Minimum. reg2 and reg1 are compared as signed integers. The smaller value is written into reg3.
MOV	reg1, reg2	I	-	-	-	-	Data transfer. reg1 is copied to reg2 for data transfer.
	imm5, reg2	II	-	-	-	-	Data transfer. imm5, sign-extended to a word, is copied into reg2 for data transfer.
MOVEA	imm16, reg1, reg2	V	-	-	-	-	Addition. The high-order 16 bits (imm16), sign-extended to a word, are added to reg1 and the sum is written into reg2.
MOVHI	imm16, reg1, reg2	V	-	-	-	-	Addition. A word consisting of the high-order 16 bits (imm16) and low-order 16 bits (0) is added to reg1 and the sum is written into reg2.
MUL	reg1, reg2	I	-	*	*	*	Multiplication of signed operands. reg2 and reg1 are multiplied together as signed values. The high-order 32 bits of the product (double word) are written into r30 and low-order 32 bits are written into reg2.
MUL3	reg1, reg2, reg3	VIII	-	-	-	-	Multiplication of signed 32-bit operands. reg2 and reg1 are multiplied together as signed integers. The high-order 32 bits of the product are written into reg3.

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
MULI	imm16, reg1, reg2	V	-	-	-	-	Saturatable multiplication of signed 32-bit operands. reg1 and imm16, sign-extended to 32 bits, are multiplied together as signed integers. [If no overflow has occurred:] The result is written into reg2. [If an overflow has occurred:] The SAT bit is set. If the result is positive, the positive maximum is written into reg2; if the result is negative, the negative maximum is written into reg2.
MULT3	reg1, reg2, reg3	VIII	-	-	-	-	Saturatable multiplication of signed 32-bit operands. reg1 and reg2 are multiplied together as signed integers. The high-order 32 bits of the product are written into reg3.
MULU	reg1, reg2	I	-	*	*	*	Multiplication of unsigned operands. reg1 and reg2 are multiplied together as unsigned values. The high-order 32 bits of the product (double word) are written into r30 and the low-order 32 bits are written into reg2.
NOP		III	-	-	-	-	No operation.
NOT	reg1, reg2	I	-	0	*	*	NOT. The NOT (ones complement) of reg1 is taken and written into reg2.
OR	reg1, reg2	I	-	0	*	*	OR. The OR of reg2 and reg1 is taken and written into reg2.
ORI	imm16, reg1, reg2	V	-	0	*	*	OR. The OR of reg1 and imm16, zero-extended to a word, is taken and written into reg2.
OUT.B	reg2, disp16[reg1]	VI	-	-	-	-	Port output. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit port address. The low-order one byte of the data in reg2 is output to the resulting port address.
OUT.H	reg2, disp16[reg1]	VI	-	-	-	-	Port output. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit port address. The low-order two bytes of the data in reg2 are output to the resulting port address. Bit 0 of the unsigned 32-bit port address is masked to 0.
OUT.W	reg2, disp16[reg1]	VI	-	-	-	-	Port output. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit port address. The word of data in reg2 is output to the produced port address. Bits 0 and 1 of the unsigned 32-bit port address are masked to 0.
RETI		IX	*	*	*	*	Return from trap/interrupt handling routine. The return PC and PSW are read from the system registers so that program execution will return from the trap or interrupt handling routine.

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
SAR	reg1 ,reg2	I	*	0	*	*	Arithmetic right shift. reg2 is arithmetically shifted to the right by the displacement specified by the low-order five bits of reg1 (MSB value is copied to the MSB in sequence). The result is written into reg2.
	imm5, reg2	II	*	0	*	*	Arithmetic right shift. reg2 is arithmetically shifted to the right by the displacement specified by imm5, zero-extended to a word. The result is written into reg2.
SATADD3	reg1, reg2, reg3	VIII	*	*	*	*	Saturatable addition. reg1 and reg2 are added together as signed integers. [If no overflow has occurred:] The result is written into reg3. [If an overflow has occurred:] The SAT bit is set. If the result is positive, the positive maximum is written into reg3; if the result is negative, the negative maximum is written into reg3.
SATSUB3	reg1, reg2, reg3	VIII	*	*	*	*	Saturatable subtraction. reg1 is subtracted from reg2 as signed integers. [If no overflow has occurred:] The result is written into reg3. [If an overflow has occurred:] The SAT bit is set. If the result is positive, the positive maximum is written into reg3; if the result is negative, the negative maximum is written into reg3.
SETF	imm5, reg2	II	-	-	-	-	Set flag condition. reg2 is set to 1 if the condition specified by the low-order four bits of imm5 matches the condition flag; otherwise it is set to 0.
SHL	reg1, reg2	I	*	0	*	*	Logical left shift. reg2 is logically shifted to the left (0 is put on the LSB) by the displacement specified by the low-order five bits of reg1. The result is written into reg2.
	imm5, reg2	II	*	0	*	*	Logical left shift. reg2 is logically shifted to the left by the displacement specified by imm5, zero-extended to a word. The result is written into reg2.
SHLD3	reg1, reg2, reg3	VIII	-	-	-	-	Left shift of concatenation. The 64 bits consisting of reg3 (high order) and reg2 (low order) are logically shifted to the left by the displacement specified by the low-order five bits of reg1. The high-order 32 bits of the result are written into reg3.

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
SHR	reg1, reg2	I	*	0	*	*	Logical right shift. reg2 is logically shifted to the right by the displacement specified by the low-order five bits of reg1 (0 is put on the MSB). The result is written into reg2.
	imm5, reg2	II	*	0	*	*	Logical right shift. reg2 is logically shifted to the right by the displacement specified by imm5, zero-extended to a word. The result is written into reg2.
SHRD3	reg1, reg2, reg3	VIII	-	-	-	-	Right shift of concatenation. The 64 bits consisting of reg3 (high order) and reg2 (low order) are logically shifted to the right by the displacement specified by the low-order five bits of reg1. The low-order 32 bits of the result are written into reg3.
ST.B	reg2, disp16[reg1]	VI	-	-	-	-	Byte store. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit address. The low-order one byte of data in reg2 is stored at the resulting address.
ST.H	reg2, disp16[reg1]	VI	-	-	-	-	Halfword store. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit address. The low-order two bytes of the data in reg2 are stored at the resulting address. Bit 0 of the unsigned 32-bit address is masked to 0.
ST.W	reg2, disp16[reg1]	VI	-	-	-	-	Word store. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit address. The word of data in reg2 is stored at the resulting address. Bits 0 and 1 of the unsigned 32-bit address are masked to 0.
STBY		IX	-	-	-	-	Processor stop. The processor is placed in stop mode.
STSR	regID,reg2	II	-	-	-	-	System register store. The contents of the system register identified by the system register number (regID) are set in reg2.
SUB	reg1,reg2	I	*	*	*	*	Subtraction. reg1 is subtracted from reg2. The difference is written into reg2.
TRAP	vector	II	-	-	-	-	Software trap. The return PC and PSW are saved in the system registers: PSW.EP = 1 -> Save in FEPC, FEPSW PSW.EP = 0 -> Save in EIPC, EIPSW The exception code is set in the ECR: PSW.EP = 1 -> Set in FECC PSW.EP = 0 -> Set in EICC PSW flags are set: PSW.EP = 1 -> Set NP and ID PSW.EP = 0 -> Set EP and ID Program execution jumps to the trap handler address corresponding to the trap vector (0-31) specified by vector and begins exception handling.

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
XOR	reg1,reg2	I	-	0	*	*	Exclusive OR. The exclusive OR of reg2 and reg1 is taken and written into reg2.
XORI	imm16, reg1,reg2	V	-	0	*	*	Exclusive OR. The exclusive OR of reg1 and imm16, zero-extended to a word, is taken and written into reg2.

APPENDIX B INDEX

Phase-out/Discontinued

NEC

Facsimile Message

From:

Name

Company

Tel. FAX

Address

Although NEC has taken all possible steps to ensure that the documentation supplied to our customers is complete, bug free and up-to-date, we readily accept that errors may occur. Despite all the care and precautions we've taken, you may encounter problems in the documentation. Please complete this form whenever you'd like to report errors or suggest improvements to us.

Thank you for your kind support.

North America
NEC Electronics Inc.
Corporate Communications Dept.
Fax: 1-800-729-9288
1-408-588-6130

Europe
NEC Electronics (Europe) GmbH
Technical Documentation Dept.
Fax: +49-211-6503-274

South America
NEC do Brasil S.A.
Fax: +55-11-6465-6829

Hong Kong, Philippines, Oceania
NEC Electronics Hong Kong Ltd.
Fax: +852-2886-9022/9044

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

Taiwan
NEC Electronics Taiwan Ltd.
Fax: 02-719-5951

Asian Nations except Philippines
NEC Electronics Singapore Pte. Ltd.
Fax: +65-250-3583

Japan
NEC Corporation
Semiconductor Solution Engineering Division
Technical Information Support Dept.
Fax: 044-548-7900

I would like to report the following error/make the following suggestion:

Document title: _____

Document number: _____ Page number: _____

If possible, please fax the referenced page or drawing.

Document Rating	Excellent	Good	Acceptable	Poor
Clarity	❑	❑	❑	❑
Technical Accuracy	❑	❑	❑	❑
Organization	❑	❑	❑	❑