HD63185
Document Image Compression and Expansion Processor Enhanced (DICEP-E)

Product Overview

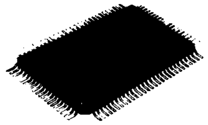
HITACHI®

DICEP-E is an LSI that performs compression (encoding) and expansion (decoding) of 2-value data used to represent a document image. It is positioned at the high end of the DICEP (HD63085) and DICEP-A (HD63183), and has been designed for application in such electronic office devices as facsimiles apparatus (G3, G4), intelligent copiers, image workstations, terminals, and filing systems. DICEP-E adopts the MH (Modified Huffman), MR (Modified Relative Element Address Designate) and M^2R (Modified MR) coding scheme, which are the G3 and G4 standards recommended by CCITT (International Telegraph and Telephone Consultative Committee).

Features

- Compatible with the CCITT recommendations for G3 and G4 facsimile apparatus
- Bit unit page processing using simple MPU instructions
  The number of picture elements in each line and the line length can be specified in bit units. Encoding (compression) and decoding (expansion) of 2-value data for multiple scan lines can also be performed.
- On-chip line memory for high-speed encoding and decoding
  An on-chip line memory (6,144 bits/line) for reference and code lines provides high-speed encoding and decoding.
  —Transfer of data to the on-chip line memory is performed by accessing 1 word (8-bit/16-bit) in 5 machine cycles (312.5ns minimum) for 1-line continuous access.
  —On-chip line memory also allows reference line access during 2-dimensional encoding to be performed by accessing 1 word in 3 cycles, and parallel processing is also possible.
  —An on-chip changed picture element detector and decoded picture element generator perform changed picture element detection and image data generation for 1 word of data per cycle.
- Specification of split processing memory area for encoding/decoding
  Byte-by-byte split processing of code data and image data can be performed in any memory area, and processing can be restarted from any address. This allows split processing in memory block units, rather than line units.
- Flexible system configuration
  —Connectable with either 68 Series or 80 Series MPU.
  —Two bus interfaces: system bus interface and image bus interface. Both the system bus and image bus sizes are selectable between 8-bit and 16-bit.
  —The port for code data and image data write to and read from DICEP-E can be selected as desired for connection to the system bus interface.
- Line-by-line alternate encode/decode
  Line-by-line alternate encoding and decoding, or time sharing simultaneous processing of inter-bus transfers and decoding can be performed.
- High-speed DMA transfer
  —2-channel simultaneous DMA transfer can be performed for transfer between the I/O device on the image bus and image memory, and between two image memories. Maximum transfer speed is 6.4M bytes/second.
  —Including a DMA controller on the system bus of the DICEP-E allows 4-channel (write to/read from FIFO and line memory) independent DMA transfer.
- Code-code conversion
  Code data can be converted to different code data without using external memory.
- A selection of coded data configurations
  The configuration of code generated during the encode procedure can be specified as MSB first or LSB first.
- Conversion of memory data configuration on the image bus
  During DMA transfer between memories, memory data on the image bus can be converted to four different configurations in 8-bit or 16-bit unis.
- Read-modify-write
  During decoding, logical operations can be performed using the decoded data in the on-chip line memory and the image data stored in external memory. Up to 16 logical operations, including AND, OR, and EOR are available, and operations can be performed line by line.
- DRAM control signal
  Memory ROW and COLUMN switching signals ($\overline{RAS}$, $\overline{CAS}$, MUX) are output for image memory DRAM on the image bus.

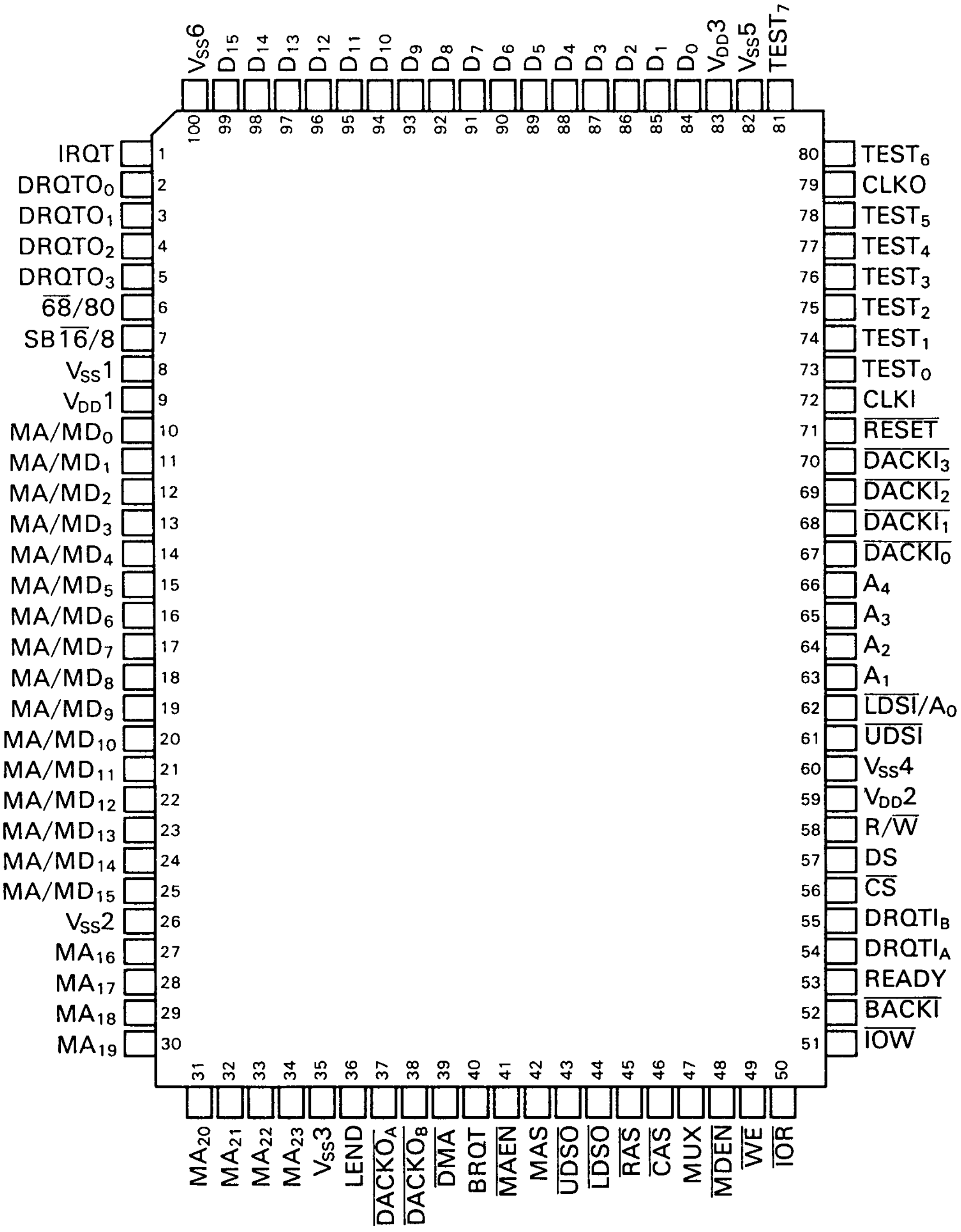
- A host of programmable parameters
  On-chip programmable registers provide flexibility for application in a wide variety of systems.
  —Parameters can be changed before issuance of each instruction.
  —Scan line length is programmable in 1-bit units, ranging from the image memory word size (8-bit/16-bit) up to 6,144 bits.
  —Encoding, decoding, and data transfer can be selected individually or in any combination.
  —MH, MR or M^2R coding scheme can be selected.
  —The minimum number of code bits per line is programmable within the range of $0-2^{16}$-1.
  —The number EOLs (end of line) is programmable within the range of $0-2^{16}$-1.
  —The number RTCs (return to control) is programmable within the range of $0-2^{16}$-1.
  —The number lines that must be processed is programmable within the range of $1-2^{16}$-1.
  —The number lines that must be transferred during DMA transfer is programmable within the range of $1-2^{16}$-1.
  —The number lines that are 2-dimension encoded following 1-dimension encoding during MR encoding is programmable within the range of $0-2^{16}$-1.
  —The area for document image data area which is read and the area for the code data which is written during encoding can be specified within the range of $1-2^{16}$-1 bytes.
  —The area for document image data area which is generated and the area for the code data which is read during decoding can be specified within the range of $1-2^{16}$-1 bytes.
- Data transfer between buses
  Data can be transferred between the system bus and image bus via the on-chip line memory, without encoding/decoding. Burst DMA transfer is performed in line units, also, access from the system bus is performed using write to and read from the control register line memory, allowing access from the MPU.
- Desired part of a document
  Specific portions of the original can be specified in bit units for cut and encoding/decoding. Also, data can be cut out during DMA transfer on the image bus in units that are the same size as the bus width of the I/O device.
- Octet mode
  Each line or the total number of code bits per page can be set in 8-bit or 16-bit units, depending on the bus size.
- High-speed operation
  —Minimum instruction cycle time of 62.5ns
  —Maximum input clock frequency of 16HMz
- Output of 1-line processing end signal
  In the multi-line processing mode, a line processing end signal is output from the LEND pin after completion of processing for 1 line. During encoding, this signal is output following generation of 1 line's code (when it is set in the register immediately before E-FIFO), and during decoding it is output following writing from internal line memory to external memory.
- 100-pin plastic QFP (FP-100A)



(FP-100A)

Pin Arrangement

● FP-100A



(Top view)

Functional Summary

Item	Function
Coding scheme	MH, MR, M^2R
Maximum scan-line length	6,144 picture elements/line
Number of processing lines	64k lines (2^{16}-1)
Image memory address space	8M words (16M bytes)
Picture element processing units	Bit units (number of processing picture elements, line length)
Maximum DMA transfer rate	6.4M bytes/second
Internal line memory maximum access speed	6.4M bytes/second
Maximum split processing memory area (image data, code data)	64k bytes (byte units)
Generated code configuration	LSB/MSB first
Read-modify-write	Operation processing–16 types (AND, OR, EOR, etc.)
Coding speed	See figure 1

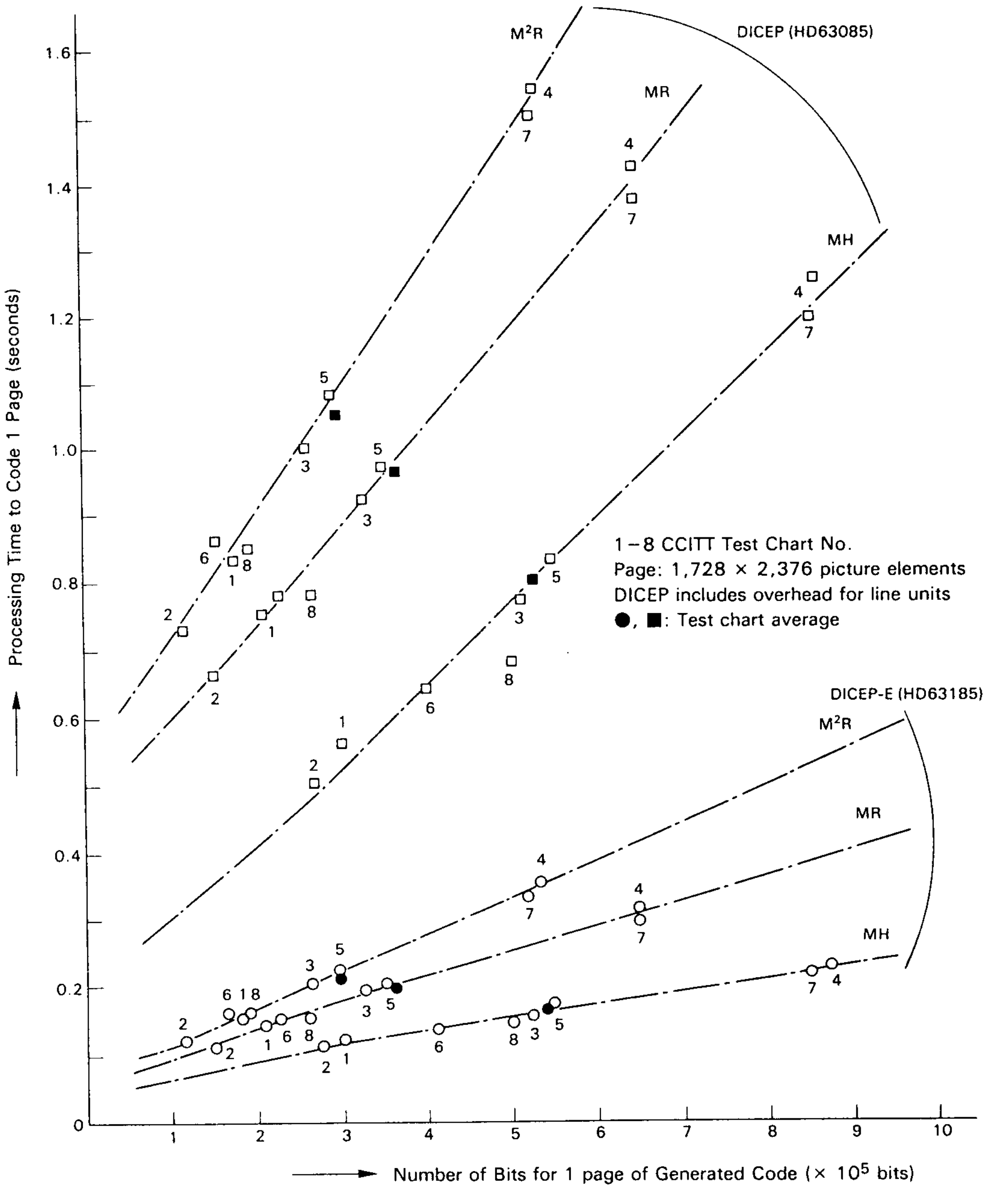


Figure 1 Average Processing Time for MH, MR and M²R Coding

Functional Comparison of DICEP Family

	DICEP (HD63085)	**DICEP-A (HD63183)**	**DICEP-E (HD63185)**
Coding scheme	MH, MR, M^{2}R run length	MH, MR, M^{2}R run length	MH, MR, M^{2}R
Image memory address space	64k bytes	4M bytes (FP) 64k bytes (P)	16M bytes
Bus width (system/image)	(S) 8 bits (D) 8/16 bits	(S) 8 bits (D) 8/16 bits	(S) 8/16 bits (D) 8/16 bits
Page mode function	×	○	○
Picture element processing unit	Byte	Byte	Bit
Coding time (M^{2}R) (standard chart average, including overhead)	1.06 seconds/A4 page	0.94 seconds/A4 page	0.22 seconds/A4 page
Bus interface	1-channel FIFO	2-channel FIFO	Line internal memory (6kb × 2) 2-channel FIFO

Block Diagrams

System Bus Interface

The system bus interface allows interfacing with such 16-bit MPUs such as the H16 Series and 68000, and with 8-bit MPUs such as the 6800 Series and 64180 Series. This block includes control registers that can directly access the MPU, and can also accommodate a 4-channel DMA for data transfer to FIFO and line memory during encoding/decoding.

Microprogram Controller

The microprogram controller stores the encoded/decoded program in microprogram ROM and controls all of the other blocks.

This block consists of a 48-bit × 512-word microprogram ROM, and sequence controller and a pipeline registers.

Execution Unit

The execution unit generates image memory addresses, and calculates run lengths, the position of the changed picture elements and the relative distance between changed picture elements on the code line and those on the reference line.

This block consists of 38 16-bit registers, and arithmetic unit (AU), and an adder. Registers can contain any value sent from the MPU, via the system bus interface control registers.

Coder and Decoder

The coder generates code table ROM addresses in accordance with the input run lengths and relative distances, and references the code table ROM to generate an appropriate code. The decoder, on the other hand, generates a code table ROM address based on the input code, and references the code table ROM to generate references or relative distances.

This block consists of 8-bit × 1024-word code table ROM, a code table ROM interface, 16-bit × 2-word or 8-bit × 4-word (optional) encode FIFO (E-FIFO) and decode FIFO (D-FIFO). The code table ROM stores data for reference during encoding and decoding, while the code table ROM interface generates code table ROM addresses. E-FIFO is used for temporary storage of code during encoding, while D-FIFO is used for temporary storage of code during decoding.

Changed Picture Element Detector

The changed picture element detector detects picture elements whose color (black/white) is different from the immediately preceding picture element on the same scan line. This circuit can detect changed picture elements for 1 word (16 bits) of image memory per instruction cycle.

This block consists of two changed picture element detectors — a reference line detector and an encoded line detector.

Decoded Picture Element Generator

The decoded picture element generator can generate 1 word (16 bits) per instruction cycle of restored picture element data, based on changed picture element position data.

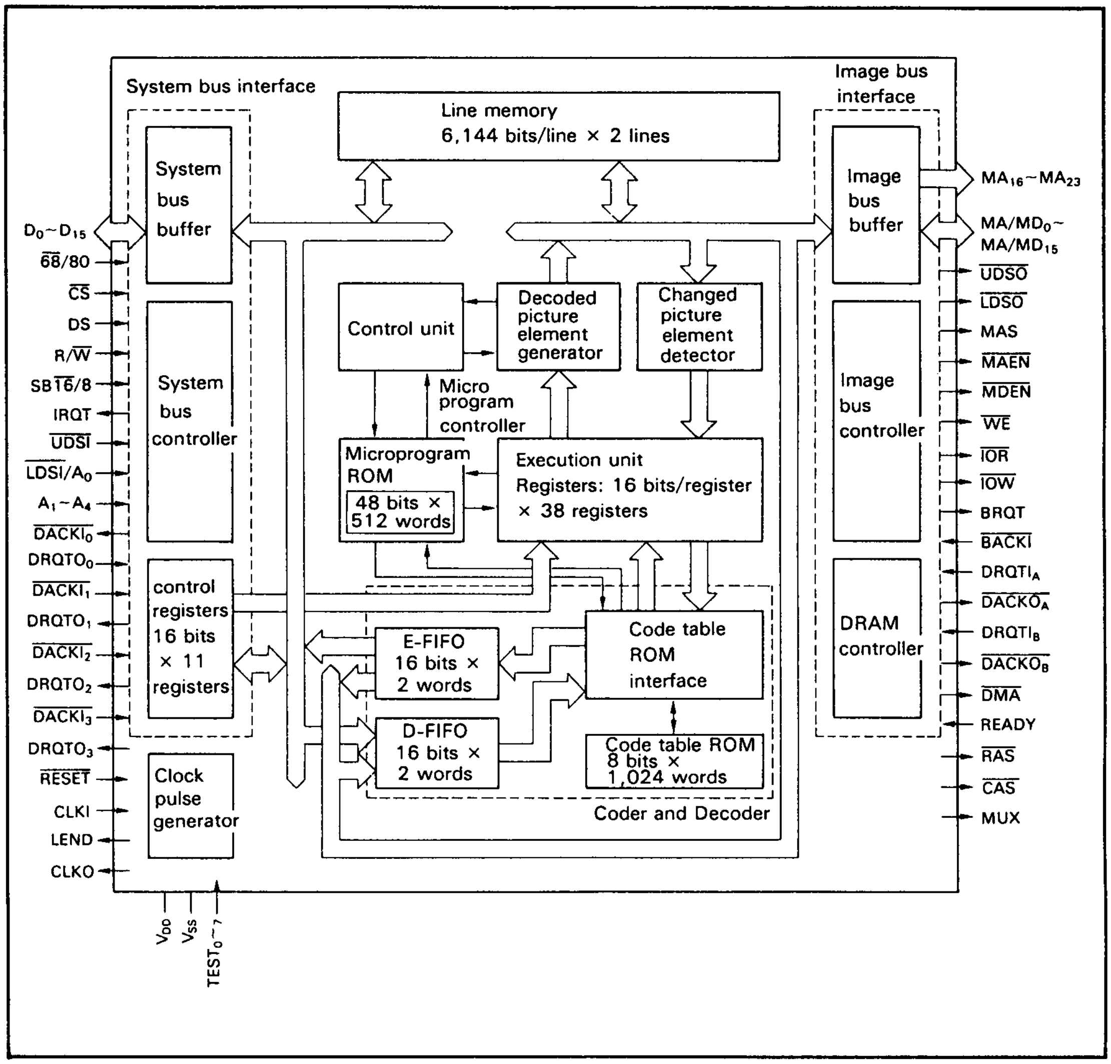
Line Memory

Line memory can be used for accumulation of 2-lines of image data during encoding/decoding, or as buffer memory during data transfer between the system bus and image bus.

This memory is consists of static RAM, in the capacity of 6,144 bits per line. One line of line memory is used as the encode/decode reference line, while another line is used to store data during encoding and data transfer between buses.

Document Image Bus Interface

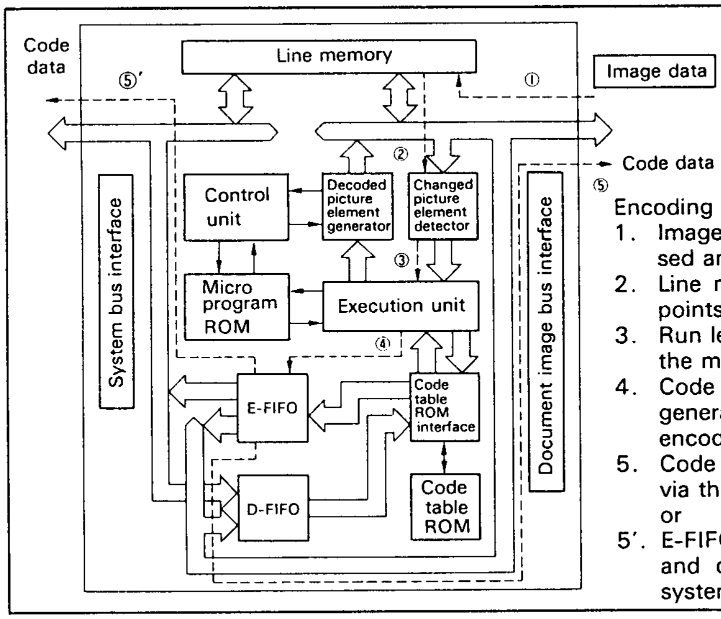
The document image bus interface provides an interface between DICEP-E and the image bus, with a multiplexed 16 bits for addresses and data, and 8 bits for the upper 8 bits of the address. In addition to a bus arbitration function, this block outputs signals for controls of such image bus functions as 2-channel DMA control.


System bus interface
Line memory
6,144 bits/line × 2 lines
Image bus interface
System bus buffer
Image bus buffer
$D_0 \sim D_{15}$
$\overline{68}/80$
$\overline{CS}$
DS
$R/\overline{W}$
$SB\overline{16}/8$
IRQT
$\overline{UDSI}$
$\overline{LDSI}/A_0$
$A_1 \sim A_4$
$\overline{DACKI_0}$
$DRQTO_0$
$\overline{DACKI_1}$
$DRQTO_1$
$\overline{DACKI_2}$
$DRQTO_2$
$\overline{DACKI_3}$
$DRQTO_3$
$\overline{RESET}$
CLKI
LEND
CLKO
System bus controller
control registers 16 bits × 11 registers
Clock pulse generator
Control unit
Decoded picture element generator
Changed picture element detector
Micro program controller
Microprogram ROM
48 bits × 512 words
Execution unit
Registers: 16 bits/register × 38 registers
E-FIFO 16 bits × 2 words
D-FIFO 16 bits × 2 words
Code table ROM interface
Code table ROM 8 bits × 1,024 words
Coder and Decoder
Image bus controller
DRAM controller
$MA_{16} \sim MA_{23}$
$MA/MD_0 \sim MA/MD_{15}$
$\overline{UDSO}$
$\overline{LDSO}$
MAS
$\overline{MAEN}$
$\overline{MDEN}$
$\overline{WE}$
$\overline{IOR}$
$\overline{IOW}$
BRQT
$\overline{BACKI}$
$DRQTI_A$
$\overline{DACKO_A}$
$DRQTI_B$
$\overline{DACKO_B}$
$\overline{DMA}$
READY
$\overline{RAS}$
$\overline{CAS}$
MUX
V_{DD}
V_{SS}
$TEST_{0 \sim 7}$

Data Processing Flow

Encoding Flow

See figure 2 and 3 for encoding flow.



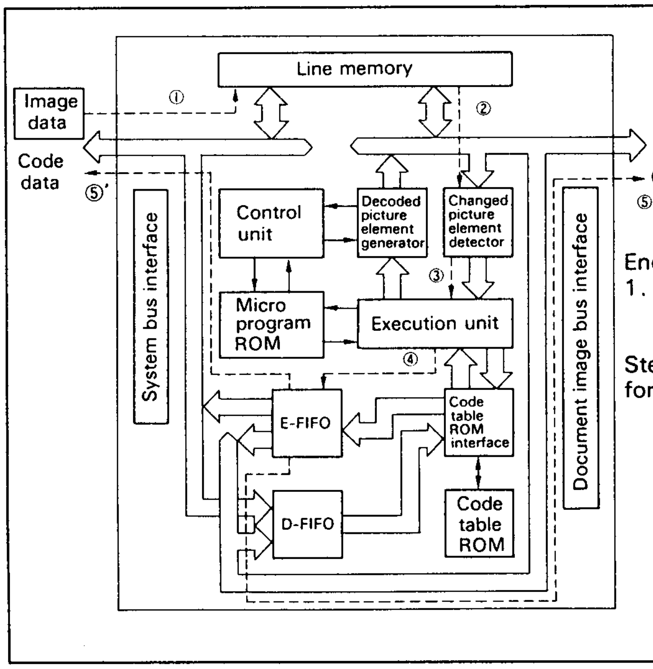
Encoding Flow (1)

1. Image data on the image bus are accessed and stored in line memory.
2. Line memory is accessed and changed points (changed pixels) are detected.
3. Run length calculation is performed and the mode is determined.
4. Code table ROM is accessed, code is generated, and code data are set in encode FIFO (E-FIFO).
5. Code data are written to the image bus via the image bus buffer.

or

5'. E-FIFO is accessed by external DMAC and code data are transferred to the system bus (DRQTO₀, $\overline{\text{DACKI}_0}$).

Figure 2 DICEP-E Encoding Flow (1)



Encoding Flow (2)

1. External DMAC transfers image data on the system bus to line memory (DRQTO₂, $\overline{\text{DACKI}_2}$)

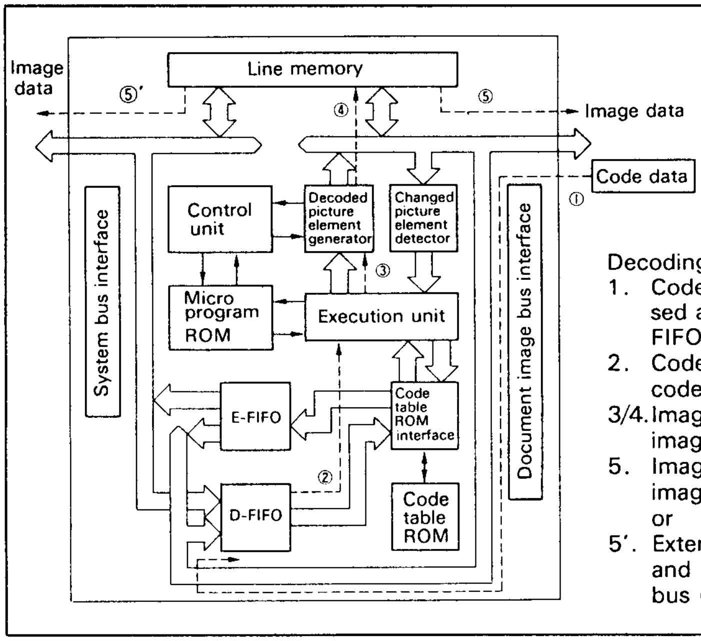
Steps 2 through 5 (5') are the same as those for Encoding Flow (1).

Figure 3 DICEP-E Encoding Flow (2)

9

Decoding Flow

See figure 4 and 5 for decoding flow.



Decoding Flow (1)

1. Code data on the image bus are accessed and code data are set in the decode FIFO (D-FIFO).
2. Code table ROM is accessed and the code is determined.

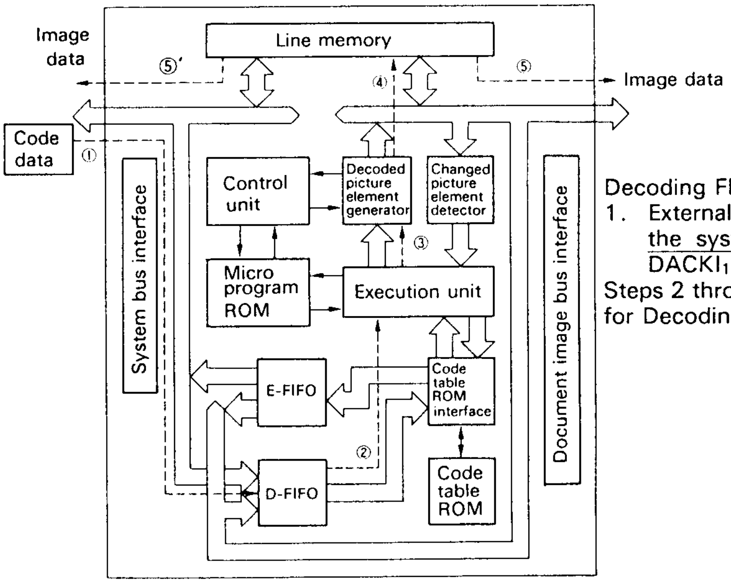
3/4. Image data are set in line memory via the image data restore circuit.

5. Image data are written to memory on the image bus via the image bus buffer.

or

5'. External DMAC accesses line memory and transfers image data to the system bus ($\overline{\mathrm{DRQTO}_3}$, $\overline{\mathrm{DACKI}_3}$).

Figure 4 DICEP-E Decoding Flow (1)



Decoding Flow (2)

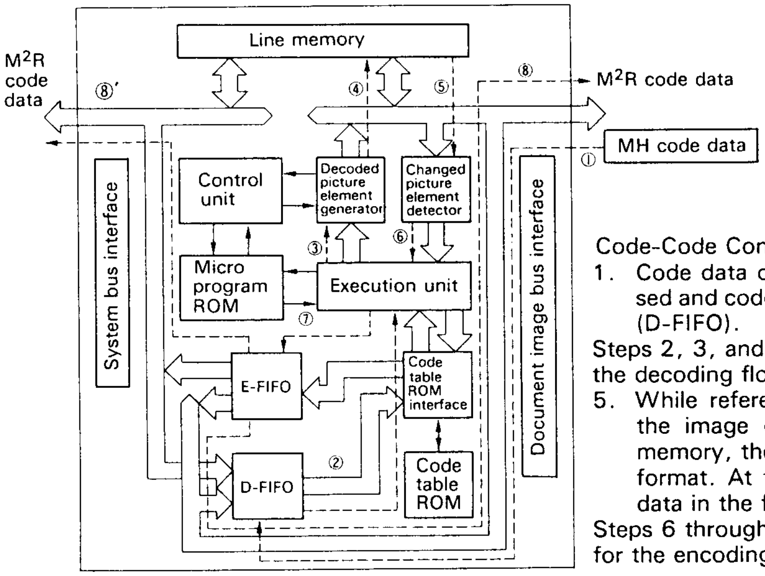
1. External DMAC transfers code data on the system bus to D-FIFO ($\overline{\mathrm{DRQTO}_1}$, $\overline{\mathrm{DACKI}_1}$).

Steps 2 through 5 (5') are the same as those for Decoding Flow (1).

Figure 5 DICEP-E Decoding Flow (2)

Code-Code Conversion Flow

The example shown here is for conversion of MH code data into M^{2}R code data. Other combinations are also possible.



Code-Code Conversion Flow (1)

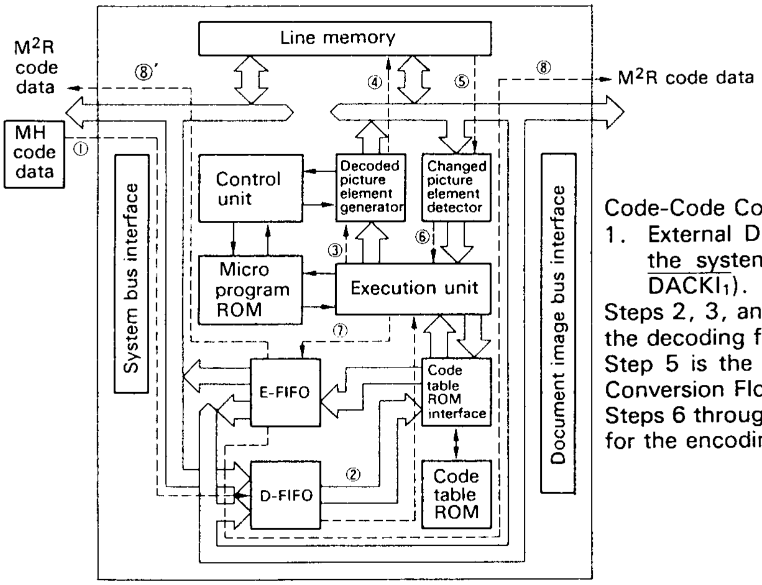
1. Code data on the image bus are accessed and code data are set in decode FIFO (D-FIFO).

Steps 2, 3, and 4 are the same as those for the decoding flow.

5. While referencing the reference line of the image data being restored in line memory, the data are converted to M^{2}R format. At this time, the reference line data in the first line is reset.

Steps 6 through 8 (8') are the same as that for the encoding flow.

Figure 6 DICEP-E Code/Code Conversion Flow (1)



Code-Code Conversion Flow (2)

1. External DMAC transfers code data on the system bus to D-FIFO ($DRQTO_1$, $\overline{DACKI_1}$).

Steps 2, 3, and 4 are the same as those for the decoding flow.

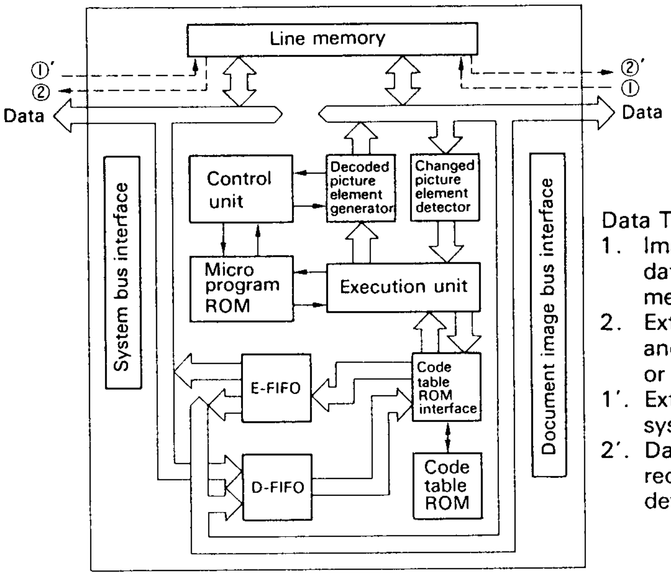
Step 5 is the same as that for Code-Code Conversion Flow (1).

Steps 6 through 8 (8') are the same as those for the encoding flow.

Figure 7 DICEP-E Code/Code Conversion Flow (2)

Data Transfer Between Buses Flow

The following shows the flow for transfer between the system bus and document image bus without data encoding/decoding.



Data Transfer Between Buses-Flow

1. Image bus memory data or I/O device data are accessed and stored in line memory.
2. External DMAC accesses line memory and transfers data to the system bus.

or

1'. External DMAC transfers data on the system bus to line memory.

2'. Data stored in line memory are transferred to image bus memory or an I/O device.

Figure 8 DICEP-E Data Transfer Between Buses Flow

System Configuration

DICEP-E has been designed to allow the widest possible application. It features both an image bus and system bus, and data encoding/decoding can be performed using either one of these busses or both of them. Figure 9 shows a sample of a system that applies DICEP-E.

MPU

- Can be used with 8-bit or 16-bit system data bus.
- Can be interfaced with 68 Series or 80 Series.

Main Memory

Both image data and code data can be stored in main memory via the system bus.

DMAC

The following 4-channel independent DMA transfer operations are possible:

- Read of code data from encode FIFO (E-FIFO). (DRQTO$_0$, $\overline{\text{DACKI}_0}$)
- Write of code data to decode FIFO (D-FIFO). (DRQTO$_1$, $\overline{\text{DACKI}_1}$)
- Image data read to or write from line memory during encoding or inter-bus data transfer. (DRQTO$_2$, $\overline{\text{DACKI}_2}$)
- Image data read to or write from line memory during decoding. (DRQTO$_3$, $\overline{\text{DACKI}_3}$)

Image Memory

Both image data and code data can be stored in image memory via the image bus. Such functions as read-modify-write and data bit configuration change are available for transfer of data between the on-chip line memory and image memory, and between two image memories. DRAM control signals $\overline{\text{RAS}}$, $\overline{\text{CAS}}$, and MUX are output to support DRAM operations.

I/O Devices

A 2-channel DMA function allows independent control of such I/O devices as scanners, printers, etc. (DRQTI$_A$, $\overline{\text{DACKO}_A}$, DRQTI$_B$, $\overline{\text{DACKO}_B}$).

13

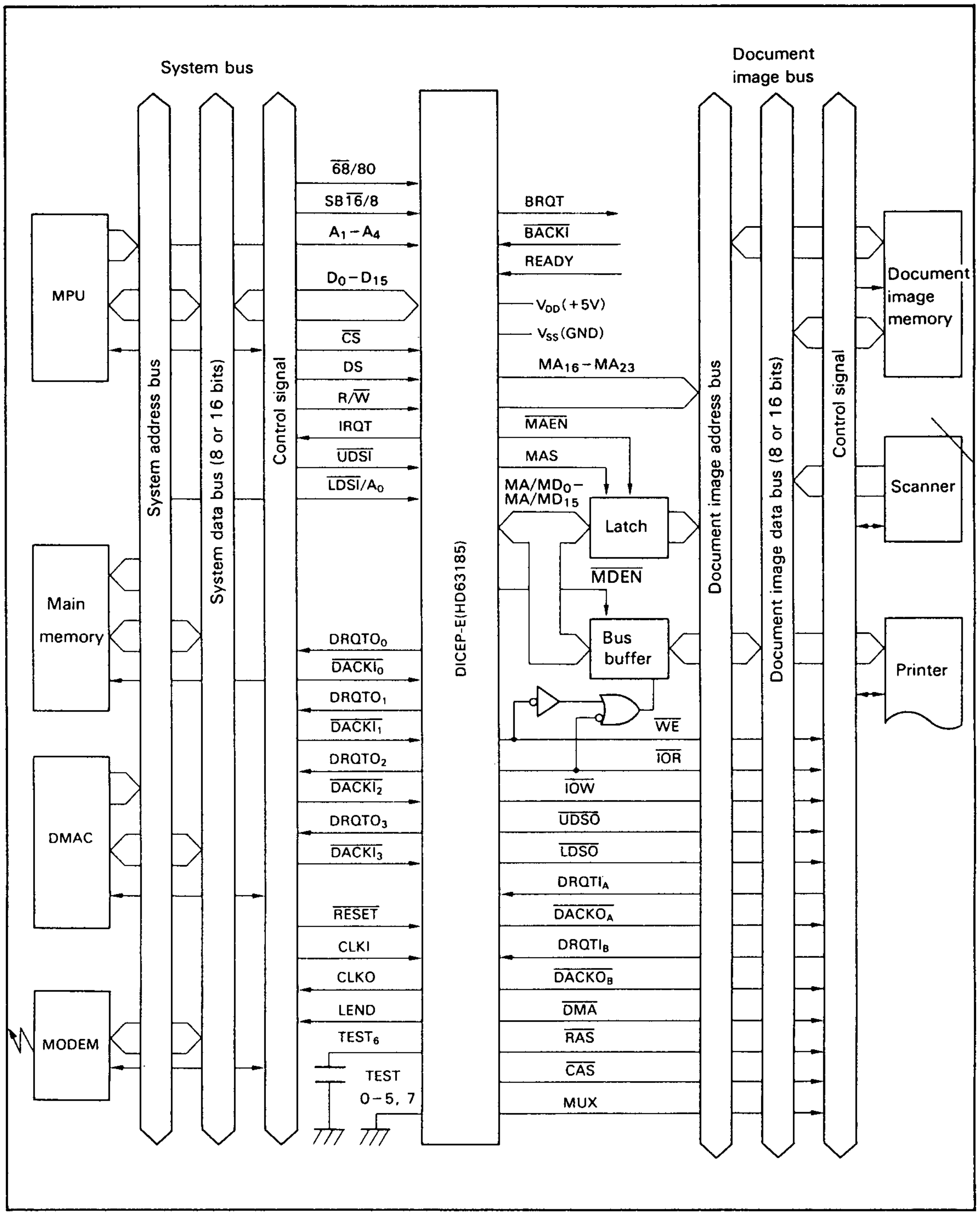


Figure 9 DICEP-E System Configuration Example

Internal Registers

DICEP-E registers can be classified into two groups: control registers that are directly accessible by the MPU, and parameter registers that can be accessed via the parameter buffer registers (PBR). The following details the access methods used for each type of register.

Registers directly accessible from the MPU

The control registers (CR), data buffer register (DBR), and FIFO can be accessed directly from the MPU, in accordance with A1—A4, when $\overline{CS}$ is L-level.

Table 1 shows the directly accessible registers.

Registers accessible via the parameter buffer registers (PBR)

The parameter registers can be accessed via the parameter buffer registers (PBR). The MPU specifies a parameter register using the address register (AR), and a parameter written to the parameter buffer registers is written to the corresponding parameter register. The MPU writes parameters to parameter registers in the following sequence:

1. The PBR ready flag of the status register (SR) is checked to see if it is 1.
2. The desired parameter register is specified using the address register (AR).
3. The parameter is written to the parameter buffer registers (PBR).
4. When the parameter is written to PBR, DICEP-E judges whether address and parameter preparation are complete, and then writes the parameter to the specified register.

Table 2 shows the parameter registers.

Table 1 Control Registers

Address				R/W	Symbol	Register Name	Register Bits															
A4	A3	A2	A1				15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	W	SCR	System control register	×	×	×	×	DFIFO ENB	EFIFO ENB	LMMEM ENB	CODE	RMW3	RMW2	RMW1	RMW0	DMABE	DMAAE	IOW/B	MW/B
				R	SSR	System status register	PBRWR RDY	PBRRD RDY	×	×	DFIFO ENB	EFIFO ENB	LMMEM ENB	CODE	RMW3	RMW2	RMW1	RMW0	DMABE	DMAAE	IOW/B	MW/B
0	0	0	1	R	IRR	Interrupt request register	×	×	×	DFEND FLG	EFEND FLG	MEMED FLG	DINI FLG	EINI FLG	ERR FLG	RTC FLG	×	INI FLG	DMAB FLG	DMAA FLG	DCE FLG	ECE FLG
0	0	1	0	W	ENCCR	Encoding control register	×	×	×	×	×	EM/IO	ENCS	ENCD	EC CNV	OCTET	"0"	PE	"0"	EC2	EC1	EC0
				R	ENCSR	Encoding status register	ECRWR RDY	LM RDYE	EFIFO RDY	×	×	EM/IO	ENCS	ENCD	EC CNV	OCTET	"0"	PE	"0"	EC2	EC1	EC0
0	0	1	1	W	DECCR	Decoding control register	×	×	×	×	×	"0"	DECS	DECD	DC CNV	RMWE	"0"	EOLE	"0"	DC2	DC1	DC0
				R	DECSR	Decoding status register	DCRWR RDY	LM RDYD	DFIFO RDY	×	×	"0"	DECS	DECD	DC CNV	RMWE	"0"	EOLE	"0"	DC2	DC1	DC0
0	1	0	0	W	DMACR	DMAA control register	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	DMAA ST
				R	DMASR	DMAA status register	DMAWR RDY	×	×	×	×	×	×	×	×	×	×	×	×	×	×	DMAA ST
0	1	0	1	W	DMBCR	DMAB control register	×	×	×	×	×	×	×	×	×	×	BC1	BC0	MTOM1	MTOM0	MTOME	DMAB ST
				R	DMBSR	DMAB status register	DMBWR RDY	×	×	×	×	×	×	×	×	×	BC1	BC0	MTOM1	MTOM0	MTOME	DMAB ST
0	1	1	0	R	EFIFO	Encode FIFO																
				W	DFIFO	Decode FIFO																
0	1	1	1	R/W	LM	Line memory																
1	0	0	0	R/W	AR	Address register	×	×	×	×	×	×	×	×	×	×	AR5	AR4	AR3	AR2	AR1	AR0
1	0	0	1	R/W	PBR	Parameter buffer register																

Table 2 Parameter Registers

Address (AR)						Operation Mode			Register Name	
AR5	AR4	AR3	AR2	AR1	AR0	Encode	Decode	Transfer	Symbol	Name
0	0	0	0	0	0	○			ERSARH	Encode reference line start address (H)
0	0	0	0	0	1	○			ERSARL	Encode reference line start address (L)
0	0	0	0	1	0	○	○	○	EESARH	Encode code line start address (H)
0	0	0	0	1	1	○	○	○	EESARL	Encode code line start address (L)
0	0	0	1	0	0	○		○	ESBAR	Encode start bit address register
0	0	0	1	0	1	○		○	EPNR	Number of encode bits register
0	0	0	1	1	0	○		○	EHWR	Encode scan-line length register
0	0	0	1	1	1	○		○	ELNR	Number of encode lines register
0	0	1	0	0	0	○			EEOLR	Encode EOL register
0	0	1	0	0	1	○			EMCLR	Minimum number of encode bits register
0	0	1	0	1	0	○			EKPR	Encode K parameter register
0	0	1	0	1	1	○			EKCNTR	Encode K counter register
0	0	1	1	0	0	○			ECSARH	Encode code area start address (H)
0	0	1	1	0	1	○			ECSARL	Encode code area start address (L)
0	0	1	1	1	0	○			ECBLKR	Encode code area size register
0	0	1	1	1	1		○		DRSARH	Decode reference line start address (H)
0	1	0	0	0	0		○		DRSARL	Decode reference line start address (L)
0	1	0	0	0	1		○		DDSARH	Decode code line start address (H)
0	1	0	0	1	0		○		DDSARL	Decode code line start address (L)
0	1	0	0	1	1		○		DSBAR	Decode start bit address register
0	1	0	1	0	0		○		DPNR	Number of decode bits register
0	1	0	1	0	1		○		DHWR	Decode scan-line length register
0	1	0	1	1	0		○		DLNR	Number of decode lines register
0	1	0	1	1	1		○		DRTCR	Decode RTC register
0	1	1	0	0	0		○		DCSARH	Decode code area start address (H)
0	1	1	0	0	1		○		DCSARL	Decode code area start address (L)
0	1	1	0	1	0		○		DCBLKR	Decode code area size register
0	1	1	0	1	1	○	○	○	MBLKR	Image memory block register
0	1	1	1	0	0	(○)	(○)	(○)	DMASAH	DMAA channel start address register (H)
0	1	1	1	0	1	(○)	(○)	(○)	DMASAL	DMAA channel start address register (L)
0	1	1	1	1	0	(○)	(○)	(○)	DMABSR	DMAA channel block size register
0	1	1	1	1	1	(○)	(○)	(○)	DMAHWR	DMAA channel scan-line length register

(to be continued)

Table 2 Parameter Registers (continued)

Address (AR)						Operation Mode			Register Name	
AR5	AR4	AR3	AR2	AR1	AR0	Encode	Decode	Transfer	Symbol	Name
1	0	0	0	0	0	(○)	(○)	(○)	DMALNR	Number of DMAA channel processing lines register
1	0	0	0	0	1	(○)	(○)	(○)	DMBSAH	DMAB channel start address register (H)
1	0	0	0	1	0	(○)	(○)	(○)	DMBSAL	DMAB channel start address register (L)
1	0	0	0	1	1	(○)	(○)	(○)	DMBBSR	DMAB channel block size register
1	0	0	1	0	0	(○)	(○)	(○)	DMBHWR	DMAB channel scan-line length register
1	0	0	1	0	1	(○)	(○)	(○)	DMBLNR	Number of DMAB channel processing lines register

When using this document, keep the following in mind:

1. This document may, wholly or partially, be subject to change without notice.
2. All rights are reserved: No one is permitted to reproduce or duplicate, in any form, the whole or part of this document without Hitachi's permission.
3. Hitachi will not be held responsible for any damage to the user that may result from accidents or any other reasons during operation of the user's unit according to this document.
4. Circuitry and other examples described herein are meant merely to indicate the characteristics and performance of Hitachi's semiconductor products. Hitachi assumes no responsibility for any intellectual property claims or other problems that may result from applications based on the examples described herein.
5. No license is granted by implication or otherwise under any patents or other rights of any third party or Hitachi, Ltd.
6. MEDICAL APPLICATIONS: Hitachi's products are not authorized for use in MEDICAL APPLICATIONS without the written consent of the appropriate officer of Hitachi's sales company. Such use includes, but is not limited to, use in life support systems. Buyers of Hitachi's products are requested to notify the relevant Hitachi sales offices when planning to use the products in MEDICAL APPLICATIONS.

HITACHI®
Our Standards Set Standards

Hitachi America, Ltd.
Semiconductor & I.C. Division
Hitachi Plaza
2000 Sierra Point Parkway, Brisbane, CA 94005-1819 1-415-589-8300

Printed in U.S.A.

790/GI/RB
Order Number: M23T001