Am7971A

Compression Expansion Processor
(CEP with image bit-boundary processing)
FINAL

DISTINCTIVE CHARACTERISTICS

- Image preserving compression and expansion of two-tone image using run-length (one-dimensional) coding and relative element address (two-dimensional) coding.
- Compatible with internationally accepted CCITT Group III and IV (Recommendations T.4 and T.6) image compression standards.
- Image bit-boundary operations.
- High performance of 1 to 12 MHz pixel rates with 3, 5, and 8MHz clock.
- CPU bus and optional local Document Store Bus with on-chip DMA. The CEP can address up to 16Mbytes on each bus.
- Handles four memory buffers: source and destination buffers for both the compressor and expander.
- Full duplex mode for simultaneous compressor and expander operations with each processor independently programmable.
- On-chip error detection to catch data corruptions and support for easy error recovery.
- 46 user programmable registers allow for very easy and highly flexible system implementation. Includes:
  - Programmable page width (up to 16K pels), frame width and top, left and right margins.
  - Optional Express mode during compression and Granularity mode during expansion for vertical resolution conversion.
  - Programmable K parameter.
  - Optional Wraparound mode.
  - Transparent mode.

GENERAL DESCRIPTION

The Am7971A Compression Expansion Processor (CEP) with Image Bit-Boundary Processing capacity is a high performance peripheral which compresses and expands two-tone bit mapped images or documents in accordance with internationally accepted CCITT standards. These fully image preserving algorithms reduce storage requirements and data transmission time for systems handling bit-mapped data.

The Am7971A is a functionally enhanced version of the Am7971 offering improved negative compression and error recovery performance. The Am7971A can replace the Am7971 in existing systems without board/system/timing alterations.

The Am7971A performs one-dimensional Modified Hoffman (MH) run-length coding as well as two-dimensional Modified READ (MR/MMR) relative coding as specified in CCITT Recommendations T.4 and T.6 for Group III and Group IV compatible equipments. The typical compression ratio for the eight CCITT test documents is 5:1 to 50:1.

The compressor and expander operate not only in full duplex mode but each processor can be independently programmed for one-dimensional encoding/decoding, two-dimensional encoding/decoding, or transparent data transfer.

Equipped with an on-chip error detection mechanism, the Am7971A detects data corruptions by checking for illegal codes, negative run-lengths and incorrect line lengths. Furthermore, its architecture allows for error recovery with minimal CPU intervention.

With 46 user programmable registers, standard Am8088-like microprocessor bus interface, dual bus architecture and on-chip DMA the Am7971A offers tremendous system flexibility and ease of implementation. After initialization the Am7971A will operate with minimal CPU overhead. Its status is available through polled registers and exception conditions may be signalled using an external interrupt.

Document page width is programmable up to 16K picture elements (pels). Programmable frame width enable windowing features and programmable top, left and right margins allow image boundaries to be left blank.

Optional express mode allows one line to be skipped after every 'nth' line to accelerate compression (n = 1 to 255). On the expansion side, the granularity option allows the processor to duplicate every mth line (m = 1 to 7).

In two-dimensional mode, the programmable K-parameter (k = 1 to 255 and infinity) defines the number of lines to be encoded in 2-D coding sequence before a 1-D line is inserted. For error free environments (Group 4) K = infinity allows for maximum compression.

The CEP can address up to 16 Mbytes of memory on each bus and two buffers (source and destination) on both the compressor and expander. Starting address, buffer length and current address for image and coded data are stored in internal registers independently for both the compressor and expander.

Publication # 08681 Rev. B Amendment /0
Issue Date: July 1988

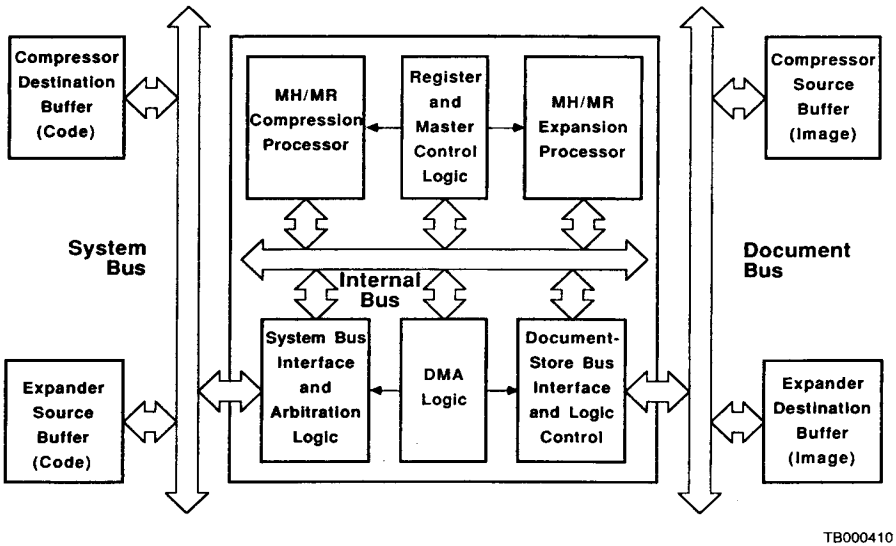


Figure 1. Am7971A Block Diagram

RELATED AMD PRODUCTS	
Part No.	**Description**
Am7971A EVAL	Am7971A Evaluation Board

CONNECTION DIAGRAMS
Top View

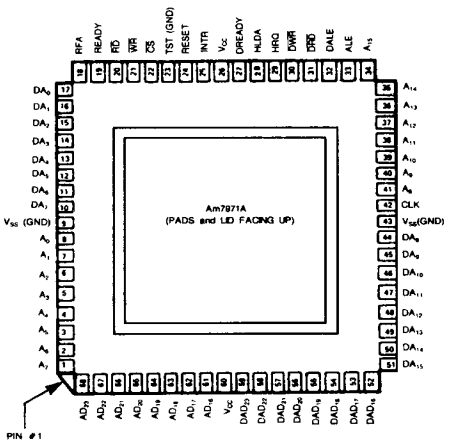


Figure 2. Am7971A Pinout for Leadless Chip Carrier (LCC)

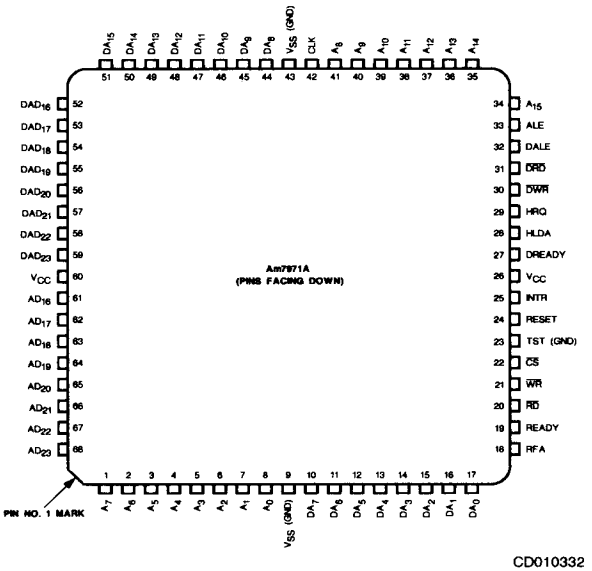


Figure 3. Am7971A Pinout for Plastic Leaded Chip Carrier (PLCC)

5

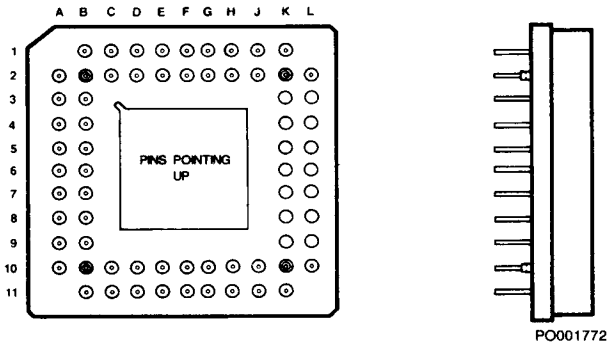


Figure 4. Am7971A (CEP) Pinout for a Pin Grid Array (PGA) Package

PIN DESIGNATIONS							
(SORTED BY PIN NAME)				(SORTED BY PIN NUMBER)			
PIN NAME	PIN NO.	PIN NAME	PIN NO.	PIN NO.	PIN NAME	PIN NO.	PIN NAME
A_0	E-2	DA_7	F-2	A-2	AD_{23}	G-1	DA_6
A_1	E-1	DA_8	F-10	A-3	AD_{21}	G-2	DA_5
A_2	D-2	DA_9	E-11	A-4	AD_{19}	G-10	CLK
A_3	D-1	DA_{10}	E-10	A-5	AD_{17}	G-11	A_8
A_4	C-2	DA_{11}	D-11	A-6	V_{CC}	H-1	DA_4
A_5	C-1	DA_{12}	D-10	A-7	DAD_{22}	H-2	DA_3
A_6	B-2	DA_{13}	C-11	A-8	DAD_{20}	H-10	A_9
A_7	B-1	DA_{14}	C-10	A-9	DAD_{18}	H-11	A_{10}
A_8	G-11	DA_{15}	B-11	A-10	DAD_{16}	J-1	DA_2
A_9	H-10	DAD_{16}	A-10	B-1	A_7	J-2	DA_1
A_{10}	H-11	DAD_{17}	B-10	B-2	A_6	J-10	A_{11}
A_{11}	J-10	DAD_{18}	A-9	B-3	AD_{22}	J-11	A_{12}
A_{12}	J-11	DAD_{19}	B-9	B-4	AD_{20}	K-1	DA_0
A_{13}	K-10	DAD_{20}	A-8	B-5	AD_{18}	K-2	READY
A_{14}	K-11	DAD_{21}	B-8	B-6	AD_{16}	K-3	$\overline{WR}$
A_{15}	L-10	DAD_{22}	A-7	B-7	DAD_{23}	K-4	TST (GND)
AD_{16}	B-6	DAD_{23}	B-7	B-8	DAD_{21}	K-5	INTR
AD_{17}	A-5	DALE	L-9	B-9	DAD_{19}	K-6	DREADY
AD_{18}	B-5	$\overline{DRD}$	K-8	B-10	DAD_{17}	K-7	HRQ
AD_{19}	A-4	DREADY	K-6	B-11	DA_{15}	K-8	$\overline{DRD}$
AD_{20}	B-4	$\overline{DWR}$	L-8	C-1	A_5	K-9	ALE
AD_{21}	A-3	HLDA	L-7	C-2	A_4	K-10	A_{13}
AD_{22}	B-3	HRQ	K-7	C-10	DA_{14}	K-11	A_{14}
AD_{23}	A-2	INTR	K-5	C-11	DA_{13}	L-2	RFA
ALE	K-9	$\overline{RD}$	L-3	D-1	A_3	L-3	$\overline{RD}$
CLK	G-10	READY	K-2	D-2	A_2	L-4	$\overline{CS}$
$\overline{CS}$	L-4	RESET	L-5	D-10	DA_{12}	L-5	RESET
DA_0	K-1	RFA	L-2	D-11	DA_{11}	L-6	V_{CC}
DA_1	J-2	TST (GND)	K-4	E-1	A_1	L-7	HLDA
DA_2	J-1	V_{CC}	A-6	E-2	A_0	L-8	$\overline{DWR}$
DA_3	H-2	V_{CC}	L-6	E-10	DA_{10}	L-9	DALE
DA_4	H-1	V_{SS} (GND)	F-1	E-11	DA_9	L-10	A_{15}
DA_5	G-2	V_{SS} (GND)	F-11	F-1	V_{SS}		
DA_6	G-1	$\overline{WR}$	K-3	F-2	DA_7		
				F-10	DA_8		
				F-11	V_{SS}		

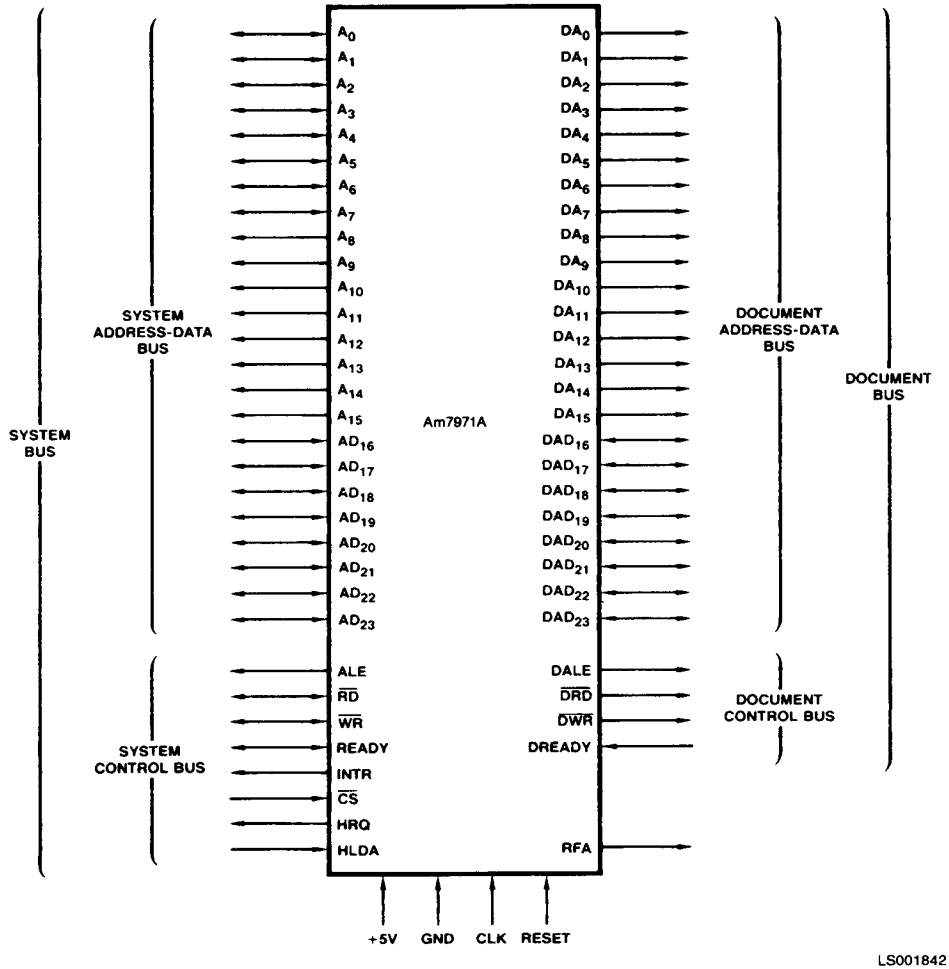
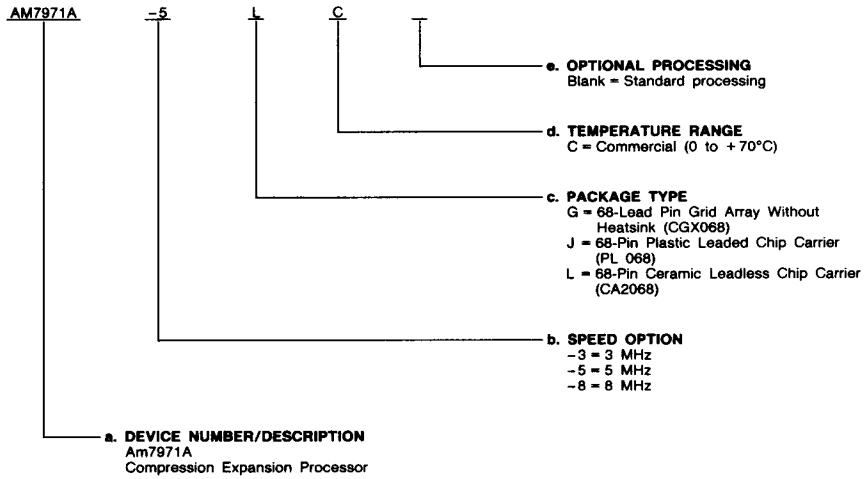


Figure 5. Am7971A Logic Symbol

ORDERING INFORMATION

Standard Products

AMD standard products are available in several packages and operating ranges. The order number (Valid Combination) is formed by a combination of: **a. Device Number**
b. Speed Option (if applicable)
c. Package Type
d. Temperature Range
e. Optional Processing



Valid Combinations	
AM7971A-3	JC
AM7971A-5	GC, JC, LC
AM7971A-8	GC, LC

Valid Combinations

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations, to check on newly released combinations, and to obtain additional data on AMD's standard military grade products.

PIN DESCRIPTION

CLK Clock (Input)
The Clock signal controls most of the CEP's internal operations and determines the rates of its data transfers. The Clock input accepts a TTL voltage level. The input signals $\overline{CS}$, HLDA, $\overline{RD}$, and $\overline{WR}$, can make transitions independent of the CEP clock (asynchronous operation).

RESET (Input)
RESET is an asynchronous, active-High input which initializes the CEP to an idle state. This input must be driven High for at least four clock cycles. The hardware reset initializes the internal state machine, the DMA, and the interrupt controller. Then it starts a software reset on the expander and compressor.

V_{CC}
Power supply input. Connect to +5 V.

V_{SS}
Ground (GND) input. Connect to 0 V.

SYSTEM BUS CONTROL SIGNALS

$A_0 - A_7$ Lower Address Outputs/Internal Register Address Inputs (Bidirectional)

$A_8 - A_{15}$ Lower Address (Three-state Outputs)
When the CEP is not in control of the system bus (HRQ and HLDA Low = Slave Mode), and the $\overline{CS}$ input is Low, $A_1 - A_7$ are used as input address lines to access the CEP's internal registers. During this time, the address lines A_0, $A_8 - A_{15}$ are ignored by the CEP. The input addresses on $A_1 - A_7$ are latched by the rising edge of $\overline{RD}$ or $\overline{WR}$. In the Bus Master mode (HRQ and HLDA High) $A_0 - A_{15}$ are three-state non multiplexed address outputs used for the system side memory transactions. The presence of valid address on $A_0 - A_{15}$ is defined by the falling edge of ALE. These lines are enabled 2 clock cycles after HREQ and HLDA = High. After the High-to-Low transition of HRQ, the $A_0 - A_{15}$ lines will float.

$AD_{16} - AD_{23}$ Address-Data (Input/Output, Bus Three-state)
The Address-Data Bus is a time-multiplexed (in Master Mode only), bidirectional, active-High, 3-state bus used for all system bus I/O and memory transactions. The presence of a valid address during Bus Master operations is defined by the falling edge of ALE and valid data is defined by the $\overline{WR}$ and $\overline{RD}$ signals; otherwise these lines are floating. While the CEP $\overline{RD}$ output is Low, $AD_{16} - AD_{23}$ must contain valid input data from the system while the READY input is High. When the CEP $\overline{WR}$ output is asserted Low, $AD_{16} - AD_{23}$ has valid CEP output data. When the CEP is acting as a Bus Slave (HRQ and HLDA Low) and the $\overline{CS}$ input is driven Low, $AD_{16} - AD_{23}$ are used strictly as data lines $D_0 - D_7$. They behave as input data lines when $\overline{WR}$ is asserted Low and as output data lines when $\overline{RD}$ is asserted Low. At all other times they are floating.

ALE Address Latch Enable (Output)
This active-High signal is provided by the CEP to latch the address signals $AD_{16} - AD_{23}$ into an address latch. This pin is never floated. ALE is asserted during address time when the CEP is Bus Master; otherwise it is Low. Address is defined as valid prior to the High-to-Low (trailing) transition of ALE.

$\overline{CS}$ Chip Select (Input, Active Low)
$\overline{CS}$ is an asynchronous, active-Low input. A CPU or other external device uses $\overline{CS}$ to activate the CEP for reading from or writing to its internal registers. Once asserted, this input can remain Low until all register accesses have been completed (block transfer). Once $\overline{CS}$ goes High, it may not go Low again for at least 100 ns. $\overline{CS}$ is ignored when the CEP is in control of the system bus.

HLDA Hold Acknowledge (Input)
HLDA is an asynchronous, active-High input indicating that the CPU has relinquished the bus and that no higher priority device has assumed bus control. HLDA is internally synchronized by the CEP. The HLDA input to the CEP can be set Low prior to HRQ (Preemption). This forces the CEP to release the bus within a maximum time of 5 clock periods (assuming READY is High and no wait states).

HRQ Hold Request (Output)
Hold Request is an active-High signal used by the CEP to obtain control of the bus from the system CPU.If the HLDA input is High after the HRQ output goes High, HRQ will remain High until the CEP has completed one memory transaction. The HLDA input may go Low prior to HRQ going Low. The HRQ signal remains Low for a minimum of 2 clocks to allow the bus master to arbitrate for the bus. If HLDA is not asserted, HRQ can be forced Low only by a hardware reset.

INTR Interrupt Request (Output)
Interrupt Request is an active-High output used to interrupt the CPU. It is driven High whenever an exception or terminating condition exists in either the Compressor (if the Compressor Interrupt Enable bit is set) or Expander (if the Expander Interrupt Enable bit is set). The INTR line is reset to Low when the CPU reads the CEP Master Status Register or when the CEP is hardware reset.

$\overline{RD}$ Read (Input/Output, Active LOW, Three-state)
$\overline{RD}$ is a bidirectional, active-Low, 3-state signal. A Low indicates that the $AD_{16} - AD_{23}$ bus is being used for a Read Data Transfer. When the CEP is not in control of the system bus and the external system is transferring information from the CEP, $\overline{RD}$ is an asynchronous timing input used by the CEP to move data between registers and the $AD_{16} - AD_{23}$ bus. $\overline{RD}$ is an output when the CEP is Bus Master (HRQ and HLDA are both High). The CEP asserts $\overline{RD}$ when data from system memory is required.

READY (Input/Output, Three-state)
READY is a synchronous, active-High, 3-state, bidirectional signal. READY is used as an input signal when the CEP is Bus Master. In Master Mode, the CEP samples the READY line with the rising edge of T2 before $\overline{RD}$ or $\overline{WR}$ are asserted by the CEP. If READY is Low during this time, wait cycles are inserted until READY is returned High. This input must be synchronized to the CEP clock. READY is used as an output signal when the CEP is Bus Slave. After $\overline{CS}$ has been asserted by the CPU, READY is kept Low by the CEP until it is ready to respond.

$\overline{WR}$ Write (Input/Output, Active Low, Three-state)
$\overline{WR}$ is a bidirectional, active-Low, three-state signal. A Low indicates that the $AD_{16} - AD_{23}$ bus is being used for a Write Data Transfer. When the CEP is not in control of the system bus and the external system is transferring information to the CEP, $\overline{WR}$ is an asynchronous timing input used by the CEP to move data from the $AD_{16} - AD_{23}$ bus into its internal registers. $\overline{WR}$ is an output when the CEP is Bus Master (HRQ and HLDA are both High). The CEP asserts $\overline{WR}$ when data is to be written into Main Memory.

DOCUMENT BUS CONTROL SIGNALS

$DA_0 - DA_{15}$ Document Store Lower Address Bus (Output, Three-state)

The Document Store Lower Address Bus is a non-multiplexed, active-High, Three-state bus used in addressing all local document memory transactions. When the CEP is in control of the Document Store Bus, the presence of a valid address on $DA_0 - DA_{15}$ is defined by the falling edge of DALE. When the CEP does not use the document bus for a memory transaction, these lines are floating.

$DAD_{16} - DAD_{23}$ Document Store Upper Address-Data Bus (Input/Output, Three-state)

The Document Store Upper Address-Data Bus is a time-multiplexed, bidirectional, active-High, Three-state bus used for all local document memory transactions. The presence of a valid address during a document bus memory transaction is defined by the falling edge of DALE and the valid data is defined by the $\overline{DWR}$ and $\overline{DRD}$ signals; otherwise these lines are floating.

DALE Document Store ALE (Output, Three-state)

This active-High output signal is provided by the CEP to indicate valid address signals $DAD_{16} - DAD_{23}$. When the CEP does not need the document interface for a data transfer, this pin is floating.

$\overline{DRD}$ Document Store Read (Output, Active Low, Three-state)

$\overline{DRD}$ is an active-Low, Three-state output signal. A Low on this signal indicates that the $DAD_{16} - DAD_{23}$ bus is being used for a Read Data Transfer. When the CEP does not need this interface for a data transfer, this pin is floating.

DREADY Ready (Input)

DREADY is a synchronous, active-High input. DREADY is used as an input signal when the CEP is Bus Master. The CEP samples the DREADY line with the rising edge of T2 before $\overline{DRD}$ or $\overline{DWR}$ is asserted by the CEP. If DREADY is Low during this time, wait cycles are inserted until DREADY is returned High. This input must be synchronized to the CEP clock.

$\overline{DWR}$ Document Store Write (Output, Active Low, Three-state)

$\overline{DWR}$ is an active-Low, Three-state output signal. A Low on this pin indicates that the $DAD_{16} - DAD_{23}$ bus is being used for a Document Bus write data transfer. When the CEP does not need this interface for a data transfer, this pin is floating.

RFA Reference Line Access (Output, Active-High)

This line indicates a reference line access in conjunction with an active "$\overline{RD}$ or $\overline{DRD}$" signal. If the CEP performs a reference line access either through the system or the document interface, this output is asserted High. It stays Low during all other data transactions.

TST Test (Input)

This input is used for factory testing. It must be tied to Ground.

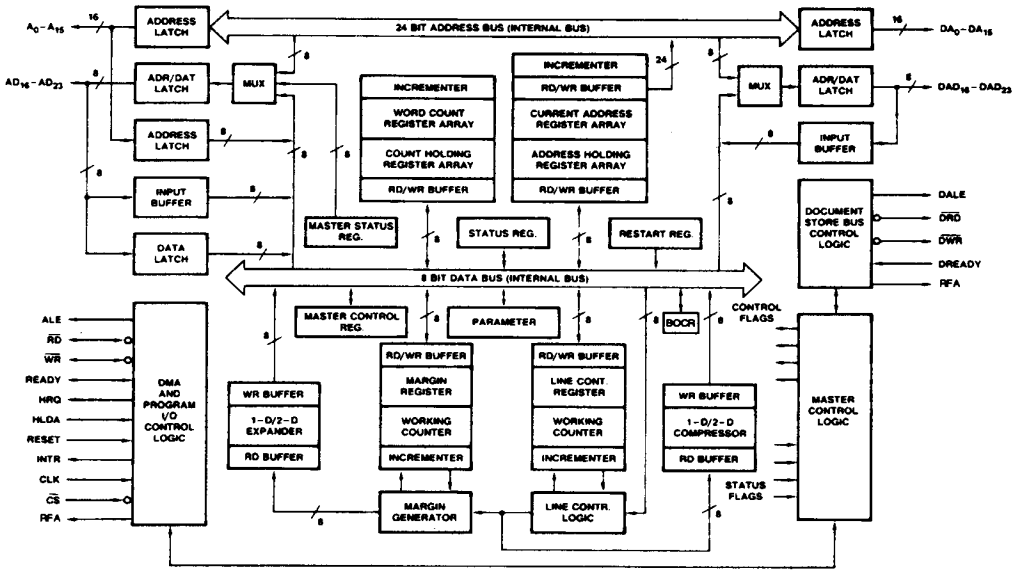


Figure 6. Am7971A Block Diagram

FUNCTIONAL DESCRIPTION

I. OVERVIEW

Figure 6 shows the internal structure of the CEP. The Am7971A contains two separate buses, the System bus and the Document Store bus. One DMA Controller on the CEP chip serves both buses. DMA data transfers cannot take place on both buses at the same time, however, slave transfers can occur on the system bus while a DMA transfer is taking place on the Document Store bus. Data transfers between the Am7971A and Main Memory take place on the System bus. Data transfers between the Am7971A and the Document Store Memory take place on the Document Store Bus.

The Am7971A processes two types of data: uncompressed or image data and coded or compressed data. Image data is stored in that portion of memory called the Image Buffer. Compressed data is stored in a portion of memory called Code Buffer. In an Am7971A system, the Code and Image Buffers are external to the CEP and each can be located in either the Main Memory or the Document Store in any combination.

Consideration should be given to the assignment of the buffers to memory. All control information exchanges between the Am7971A and the host processor take place on the System bus. Because of the high data rate of image data, it is recommended that the Image Buffer be placed in the Document Store. This way, it can be accessed with maximum speed without slowing down the host system. For maximum performance, the Image buffer should be large enough to store one uncompressed document. The Code Buffer can be placed in the Main Memory so that the CPU can access it rapidly during transmission or reception of data. Since the compressed code is considerably smaller than the image data, it does not seriously slow down the system bus and thus does not impact the CPU.

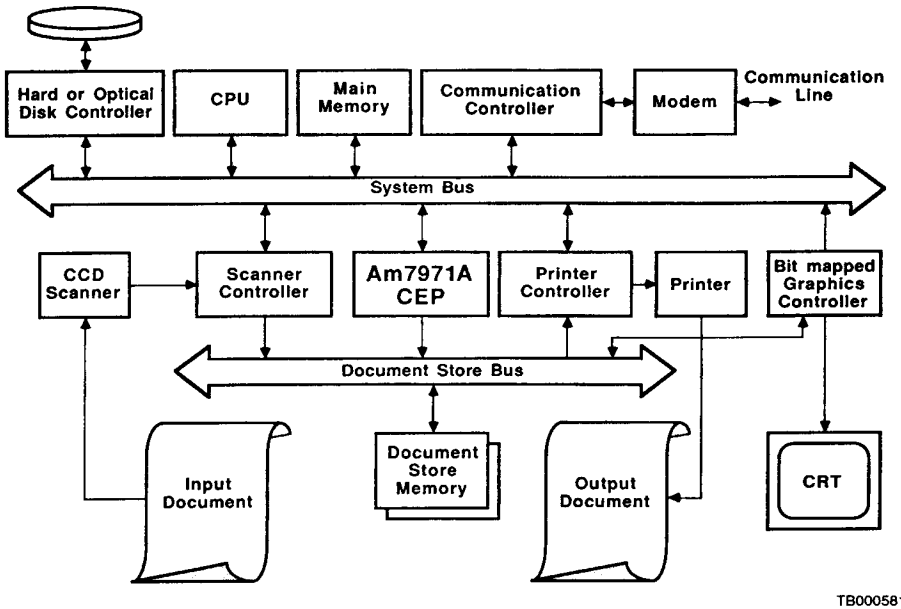


Figure 7. CEP Typical System Configuration

Figure 7 shows a typical configuration with the code buffers in the Main Memory and the image buffer in the dedicated Document Memory. All image handling devices like scanners or printers are located on this bus. It can also be utilized by a graphics processor. The system bus only carries coded data to a transmission line or a mass storage device. Thus, the system CPU is released from any significant task for image handling.

CEP operations consist of three phases: initialization, operation, and termination. In the first phase, the registers (for the compressor or expander) are initialized to specify and control the desired operation. In the second phase, the processing operation itself is started and performed. The final phase involves terminating the selected processor and performing any actions that are appropriate to that termination.

The Am7971A contains registers to specify the starting address and assigned length of both the Image Buffer and the Code Buffer. The Compressor takes image data from its Image Buffer and loads the resulting compressed data into its Code Buffer. The Am7971A Expander takes compressed data from its Code Buffer for processing and loads the resulting image data into its Image Buffer. In an Am7971A system, both the Compressor and the Expander are completely independent and can operate simultaneously.

In principle, the compression and expansion algorithm implemented in the CEP works with any image resolution (PELs per

inch). However, if the resolution is too low, the compression ratio becomes insignificant.

For certain images (such as half tone or low resolution), the compressed data representing a line may be longer than the original line of the image. This is called negative compression. The Am7971A checks for this condition after compressing a line and alerts the host processor via an interrupt and a status bit.

Each compressed line may be delimited by an End of Line (EOL) code according to the CCITT recommendation for Group 3 facsimile apparatus. However, this automatic EOL insertion can be suppressed by appropriate bit settings of the Am7971A.

The CCITT recommendation T.4 for Group 3 equipment requires each coded line to have a certain minimum length. Fill bits are added by the CEP to a short line when necessary to meet this requirement. The Am7971A contains a Time Fill Register to specify the minimum line lengths (including zero).

Data is vulnerable to modification by transmission errors. When erroneous data is expanded, the resulting image is very different from the original. The Am7971A checks the expanded line for the number of picture elements required by the specified paper width. If there is a discrepancy, the CPU is alerted via an interrupt.

II. INTERFACE DESCRIPTION

The two interfaces of the CEP both consist of a 24 bit address bus and a 8 bit data bus. The 8 bit data bus is multiplexed with the upper 8 address lines. ALE/DALE is used to latch the upper part of the address; $\overline{RD}/DRD$ and $\overline{WR}/\overline{DWR}$ are used to indicate the read or write access of the CEP. Any number of wait states can be inserted in a CEP memory access by keeping READY/DREADY Low. A memory access without wait-states takes 3 clock cycles.

The system interface is designed to perform an iAPX 8086 like bus arbitration for the system bus using the signals HRQ and HLDA. HRQ is asserted when the CEP wants to perform a Master DMA access on the system bus. This request is granted by an active HLDA.

The above sequence is called Master Mode because the CEP performs an independent Master DMA cycle on the system bus (see figure 8). No external DMA device is needed to supply the expander or compressor with data. The Am7971A usually performs one memory access cycle for each bus arbitration cycle. If the begin or end of an image line is not on a byte boundary (see bit boundary image processing) the Am7971A expander performs a Read Modify Write operation in order to mask expanded image bits into a frame picture which results in a two byte burst transfer. It only requests the bus once for this operation. The Read and Write accesses occur back to back which results in a two byte burst transfer.

- **CEP accesses memory for image data and coded data**

TB000570

Figure 8. CEP Master Mode.

- **CPU accesses/initializes registers**
- **CPU polls status**
- **Block Transfer mode (optional)**

TB000560

Figure 9. CEP Slave Mode.

The Am7971A has 46 registers for address pointers, parameters and status information. These registers provide a maximum of flexibility in memory management, format control and operational control over the CEP opertions. They are initialized before a compression or expansion sequence is started. During an expansion/compression operation, three status registers supply information about the current operation of the expander or compressor.

While the CEP is busy compressing or expanding a picture, all internal registers can be accessed whenever the CEP is not using the system bus. This is called Slave Mode (see figure 9).

Since the system interface is used for Master DMA accesses as well as for Slave accesses to the registers, several control signals are bidirectional ($\overline{RD}$, $\overline{WR}$, READY, $A_1 - A_7$). These input/output signals are 3-stated by the CEP when it is not in Master Mode. The CEP recognizes a register access request when the $\overline{CS}$ signal is asserted. All registers are directly addressed through the address lines A_1-A_7 (Only even addresses are used). The data transfer to and from the registers is channeled through $AD_{16} - AD_{23}$. $\overline{CS}$ can be kept Low for consecutive slave accesses (Block I/O Transaction).

With one exception, the registers are not directly connected to the system interface. On a Slave access, an internal microprogram takes care of the data transfers to and from the registers. After data is available from or successfully written into the registers, the CEP responds by asserting READY High. The response time of the CEP to these register access requests varies from register to register. READY might be suppressed between 4 to 20 clock cycles if the CEP is idle and up to 50 clock cycles if the CEP is busy. If the host system aborts the slave access by driving $\overline{RD}$ or $\overline{WR}$ High before the CEP responded to the access by asserting READY High, the slave access is disregarded and no register contents are changed (CPU Program Read/Write Access Abort).

The Master Status register (MSR) supplies the host system with an overview about the current status of the compressor and the expander. Only this register is accessible without delay (4 clock cycles) providing fast access to the status information in polling mode.

III. MEMORY BUFFER MANAGEMENT

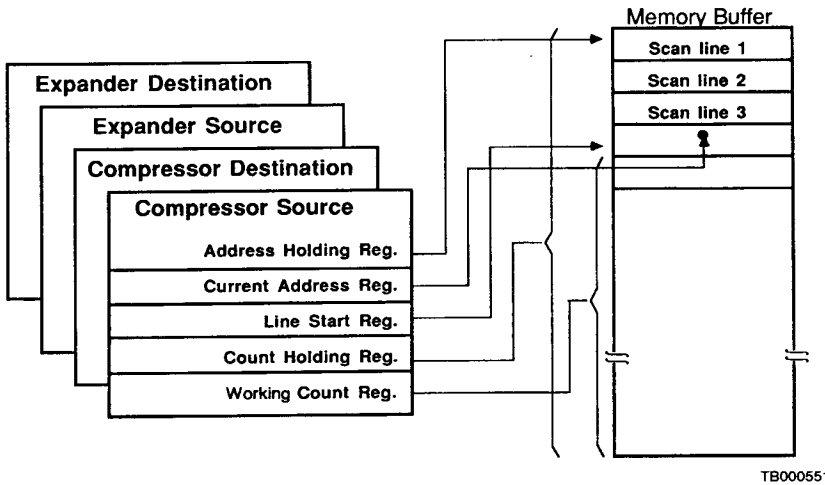


Figure 10. CEP Memory Buffer Management.

5

Each section of the CEP, the compressor and the expander, needs one set of source and destination buffers. Since full duplex operation is possible, the expander can expand an image from the expander source buffer into the expander destination buffer, while the compressor is compressing an image from the compressor source buffer into the compressor destination buffer. For each of these four possible buffers, there is a separate set of address pointers and byte counters as shown in the Figure 10.

A Buffer is defined by specifying which memory it is in (Main Memory or Document Store), the starting address and the capacity of the Buffer. The source and destination control bits in the expander/compressor master control register determine whether the source or the destination buffer is located on the system bus or the document bus.

The CEP always uses the Current Address Register (CAR) content to access memory for its current operations. It always points to the byte which will be accessed next. The CEP automatically increments this register after each memory access. When the CEP starts processing a new image line it always copies the content of the current address register into the Line Start Address Register (LSR). Thus, when an exception occurs, it is always possible to recover the beginning of the last processed image line and its corresponding code.

For each access to a buffer the CEP increments the corresponding Working Count Register (WCR). Before the CEP starts processing a buffer this register contains the total number (in 2's complement form) of bytes available in a buffer. Hence it always shows the number of bytes remaining unprocessed in a buffer. If the content of WCR reaches zero the CEP terminates its operation and flags a buffer overflow condition. The software has to decide if the whole page is finished or if the operation is to be resumed to complete a page (see exception processing).

For its operation, the CEP only needs and modifies the current registers (current address, working count) and the Line Start address register. In addition to these registers an Address Hold Register (AHR) and a Count Hold Register (CHR) are provided for each buffer to maintain a copy of the initial values. The AHR stores the starting address of a buffer and the CHR stores the 2's complement of the buffer size as a byte count. These Registers are never changed by the CEP but they are used to update the CAR or WCR when the same buffer is used over and over again. This feature reduces the software overhead for updating the address pointers and counter values to an absolute minimum (Refer to the section 'Buffer Overflow' for more details).

IV. PAPER FORMAT CONTROL

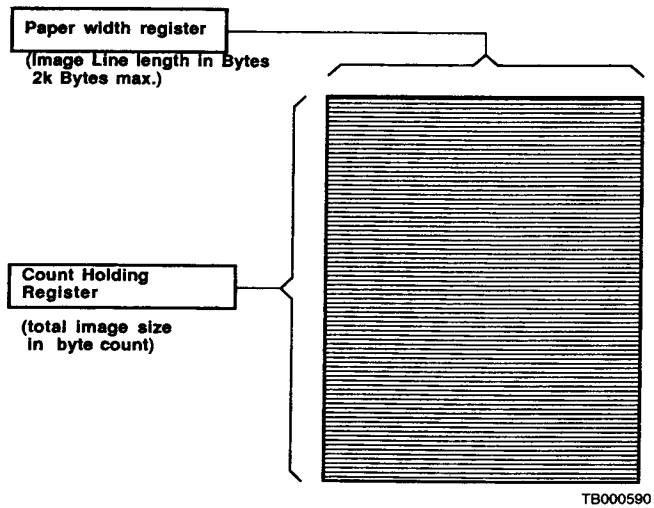


Figure 11. CEP Paper Size Control

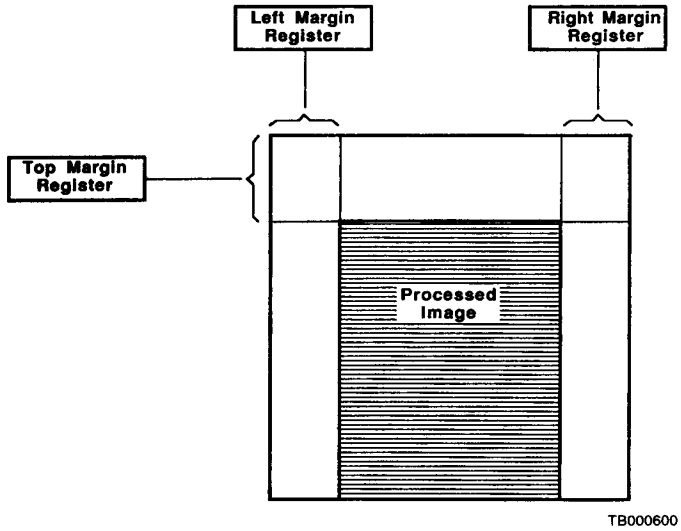


Figure 12. CEP Margin Control

Figure 11 shows, that the width of a document to be processed is defined by the Page Width Register. The length of a document during compression is determined by the user software. The CEP stops compressing whenever the source image buffer is empty. The Count Holding Register contains the number of bytes of image in the image buffer. If the source image buffer specified is smaller than the document, more processing control is required.

The simplest case is when the source image buffer length is the same length as the document (See Figure 12). In this case, the CEP is programmed to append an "end of document" code to the coded image at the end of compression. During expansion, processing continues until the end of document code (RTC for Group 3 or EOP for Group 4) is detected assuming that the destination image buffer is also large enough to hold the expanded document.

If the source image buffer for the compressor is smaller than the document, the document is processed in segments (Refer to the discussion of page fraction processing for details).

During expansion of a document larger than the image buffer, the CEP stops whenever there is a destination image buffer overflow. The CPU must then empty the image buffer and resume the expansion operation. For more details, refer to the page fraction processing discussion.

CCITT recommendation T.4 covers compression and expansion of scan lines up to 2560 bits. The Am7971A accommodates much wider pages by the use of multiple make-up codes. The CEP allows specification of scan line lengths up to 16K bits in both, 1-D and 2D mode operation.

Figure 12 shows how a white margin around the image can be specified for compressor operation. This feature is useful to suppress tolerances in scanner adjustments. Specifying a margin provides a clean surrounding for the scanned image.

The three margin control registers specify white right, left or top margins. If any of these registers are non zero, the compressor reads the image data within these margins but disregards the content and encodes it as white image. If the image following these margins begins with white pixels, the compressor combines them together with the margin into one white runlength code.

Since, by definition, the Top Margin white space is to occur only once per document, the compressor logic decrements the Top Margin Register by one after processing each scan line until it reaches "0", at which time normal compression proceeds. Because of this, the Top Margin Register must be initialized each time a new page is started.

Margins can only be defined in byte boundaries. If the margin continues into white data, a single white code is generated for the margin and the following white pels in the image line.

When the margin specifications are not consistent with the page width, the CEP will terminate operation after setting the Illegal Command bit in the appropriate Status register.

The expander does not have a margin control feature.

V. WINDOW PROCESSING

The CEP provides the possibility to define an image window of any size at any location in memory. The idea is to define a larger image as a frame, and a window as a page within this frame. Figure 13 shows how the frame width register specifies the scan line width of the frame picture in memory.

This ability provides window processing without any additional processing of the image. A partial image area of any size at any location can be compressed from a larger image and expanded into a different location of a target image.

The window is specified like a small page. If the frame width and the page width are the same, the full frame is processed and thus the window processing feature is disabled.

If the frame width is larger than the page width, window processing is performed. The position of the window is specified by the address hold register. The width of the window is specified by the page width register. The length of the window is specified by the number of bytes processed (CHR and WCR).

Whenever the CEP reaches the end of a scan line, it adds the frame width to the image buffer LSR which still contains the start address of the last line. Thus it then points to the first byte of the next line. After this, the same address is automatically copied into the image buffer's current address register (CAR). Then it continues operation from this new position. This scheme is used for the expander destination buffer and for the compressor source buffer (see also the description of the Line Start Register under buffer overflow processing and Figure 19).

Any memory location outside this defined window is untouched. If the expansion results in a longer runlength than specified in the page width register, which might happen due to a data error, the CEP will skip outputting any bits beyond the specified line length.

Margin control is effective during window processing. Therefore, it is possible to have a window with white margins.

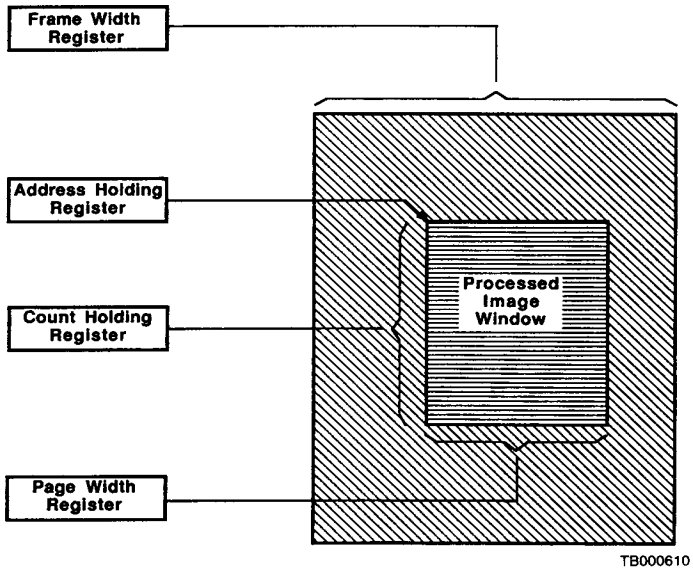


Figure 13. CEP Window Processing

VI. BIT OFFSET CONTROL

The Am7971A allows specification of an image scan line on any possible bit boundary. This is useful in preserving the information at the edges of documents which are scanned on a bit boundary, especially important for window processing. The Am7971A can compress from a window at any bit position in a picture and expand it into any position in a destination image.

To specify the left and right bit offset, the Am7971A provides a register for the compressor and the expander, called the expander or compressor bit offset control register (EBOCR/CBOCR). Figure 14 shows how the contents of these registers correlate to the definition of a bit offset at the right or left edge of a page (or window). Any Bit Offset between 0-7 bits can be selected. The Bit Offset specifies the number of bits in the outermost byte of a line that are not used. During expansion, the Am7971A performs a read-modify write operation at the line extremes to patch the bit offset into an existing image.

Note: If the bit offset value is "0", the Am7971A behaves exactly as the Am7970A with line termination parameter = "0".

VII. CODE FORMAT CONTROL

The CEP generates a code format as recommended by CCITT for Group III and Group IV apparatus. This standard coding procedure was originally intended for facsimile transmission but is also widely accepted as standard for graphics workstations and electronic archiving purposes. Since not all image processing environments necessarily need to follow the CCITT recommendation precisely, the CEP offers a variety of coding options which can be used to gain a higher compression ratio. The usefulness of these options depends on the specific application and must be evaluated from case to case.

Since the CEP maintains great flexibility in the code format to be used, much care must be taken in the correct initialization of the registers which affect the format of the compressed image. Only a specific combination of parameters will produce a Group III format, while a different combination will produce Group IV code. The internal microcontroller checks all parameters before starting compression or expansion, and determines if it is a Group III or Group IV data format.

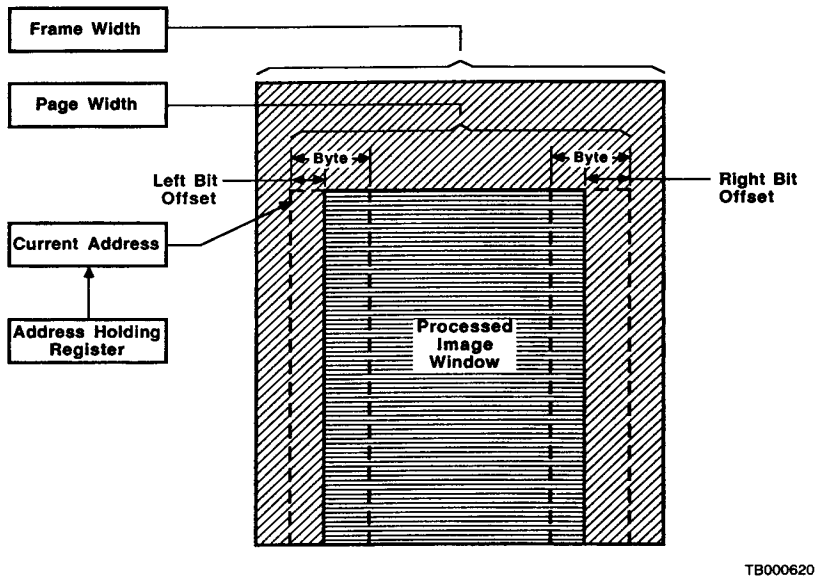


Figure 14. CEP Bit-Boundary Image Processing With Window

5

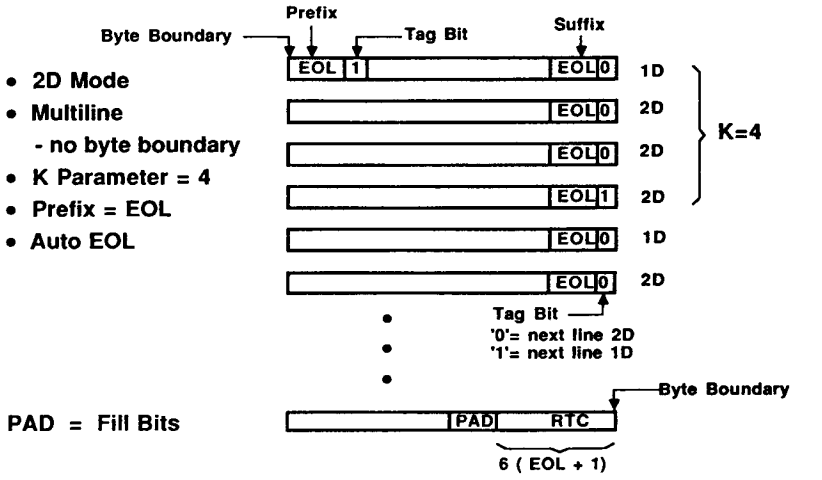


Figure 15. CCITT Group III Format Control

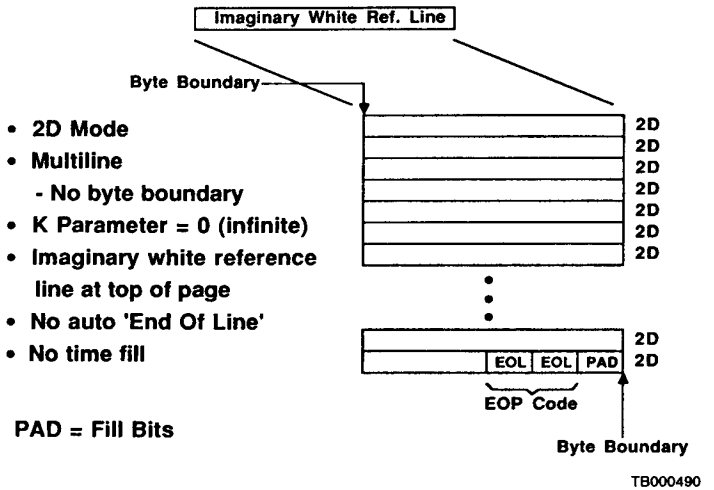


Figure 16. CCITT Group IV Format Control

Figures 15 and 16 are examples of Group III and Group IV code formats and show the differences of the selected parameters.

Auto EOL Control:

The EOL code terminating a coded image line plays an important role for resynchronization of the expansion process after the occurrence of corrupted data from transmission errors in Group III coded data. The compressor terminates each coded line with an EOL code if "Auto EOL" mode is selected in the Compressor Parameter Register (CPR). The expander recognizes EOL as the end of a coded line if "Auto EOL" is selected in the EOL control bit of the expander parameter register (EPR).

Byte Boundary Control:

The CEP is able to adjust the end of a coded line to either bit or byte boundary. After an exception condition, such as negative compression or data error, it is easier to resume from a byte-boundary adjusted line. However bit-boundary lines result in higher compression and throughput. The CCITT Group III standard leaves this option up to the user.

If the "byte boundary" option in the compressor parameter register (CPR) is selected, the CEP compressor adjusts each coded line to a byte boundary. It does that by adding fill bits (= "0") between the compressed image line and the EOL code. If a byte boundary adjusted compressed image is to be expanded, the expander knows from the expander restart control register (ERCR) whether to disregard these fill bits or not. If the expander input is byte boundary adjusted but the expander set to no byte boundary operation or vice versa, a data error will occur and the operation is terminated. However, if Auto EOL is programmed, the expander ignores the byte boundary control bit (fill bits are automatically disregarded if there are any).

Coding Mode Control:

The "mode control" bits in the Master Control Registers (EMCR/CMCR) determine the coding scheme used to compress or expand an image. The choices are 1-D, 2-D and transparent mode.

One Dimensional Mode (1-D):

In one dimensional mode, the CEP processes an image by the Modified Huffman coding scheme only. Image lines are scanned for groups of pixels with the same color (black or white) from left to right. The number of pixels between two color changes is the color runlength. This runlength is passed through a coding table. Two different code tables are used for black and white runlengths. The lengths of the codes are optimized according to the statistical probability of occurrence of a runlength.

For each new line a new code is started. Each line is assumed to start with a white pixel. If it actually starts with a black pixel, a white zero length code is inserted. The codes for each line are independent from previous or following lines. Thus the Huffman coding scheme is a one dimensional (1-D) or horizontal coding method (Refer to the CEP technical manual for detailed information about the coding methods and the CCITT specification).

When the CEP is in 1-D mode and auto EOL operation is selected, it will not append a Tag bit to any EOL since all lines are coded with the same method. A Tag Bit is used to distinguish the coding mode of the next line.

Two Dimensional Mode (2-D):

In two dimensional mode, the CEP processes data by the modified READ (Relative Element Address) coding scheme. This method detects similarities between two adjacent lines. Only the differences between lines are coded into very short code words of varying length. Since normal text images have a lot of equal or similar adjacent lines, this coding scheme is very efficient for such documents. To generate the code for a scan line, the compressor or expander must access the previous scan line for reference. Thus the READ coding scheme is called a two dimensional or vertical method.

The two dimensional scheme only deals with relative positions of color changes between two lines. If the first line of a document is to be coded in 2-D mode, it is necessary to assume an imaginary white reference line on top of a page. Starting from this assumption, it is possible to recover the image by relative information only. The vertical coding scheme is only efficient for small differences between lines. The READ scheme falls back to the Huffman scheme when the differences between two lines are too big (Refer to the CEP technical manual for detailed information about the coding methods and the CCITT specification).

The CCITT specification uses the 2-D mode in two different ways. In Group IV environments, READ code is used exactly as shown above (see also figure 16). All lines are coded in the 2-D scheme only. In Group III environment, the 2-D coding scheme is used in conjunction with the 1-D scheme. After N lines are encoded in 2- D, one line is encoded in 1-D. The K-parameter determines the number of 2-D lines for each 1-D coded line (K = N + 1). In this mode each coded line must be terminated by an EOL code including a Tag Bit which indicates the coding scheme of the next line (Also see figure 15).

The Tag Bit:

The compressor appends a Tag Bit to each EOL code when 2-D coding mode is selected. The expander determines from the Tag Bit the coding scheme of the following line. In 1-D mode the Tag Bit is unnecessary and therefore omitted.

EOL code for 1-D Mode: 0000 0000 0001
2-D Mode: 0000 0000 0001T

T = Tag Bit
= 0 if next line is 2-D
= 1 if next line is 1-D

In Group III (2-D) mode, when the SA bit is set (e.g. at the beginning of a page), the expander does not simply assume a 1-D coded line but expands the line according to the Tag Bit appended to the prefix EOL. A Tag Bit is also appended to each EOL code of the RTC suffix in Group III coding mode. For this purpose the Tag bits are always set "1". If the EOL codes are omitted (auto EOL mode off), the expander decodes data according to the K-parameter.

Transparent Mode:

In addition to the one dimensional and two dimensional scheme, the CEP provides an optional transparent mode. This mode disables the coding/encoding algorithm but keeps all format control parameters in effect. If data is not to be changed when being transferred by the compressor or expander in transparent mode, all format control parameters must be disabled (no EOL, no byte boundary, no SA, granularity = 0, express mode = 0, no time fill, no margins). Alternately, the expander in transparent mode can be used to strip off the prefix and suffixes by leaving Auto EOL mode, SA bit and RTC or EOP format on.

The transparent mode provides a plain DMA channel for data moves between the system interface and the document interface and for general DMA actions on either one of the two interfaces. While the CEP is busy expanding or compressing in half duplex mode, the unused counterpart could also be used for DMA transactions of the next picture to be processed.

K-Parameter:

If the image was compressed in 2-D only, it is impossible to recover a useful image after a data error. The CCITT. therefore, specifies for Group III type data a factor called the K-Parameter. Using this K-Parameter 1-D coded lines are interleaved with 2-D coded lines. There are K-1 lines coded in 2-D mode for each 1-D coded line. Unlike the 2-D mode, the expander in 1-D mode does not need to access an already expanded image line as reference for the next line. Therefore, a data error can be stopped from propagating through the whole remaining image by these 1D coded lines.

Since a maximum of K lines can be corrupted by a data error, the K-parameter represents the maximum tolerable distortion of an image. Thus, the K-parameter depends on the vertical resolution of an image and the probability of data errors during transmission.

The K-Parameter Register (CKPR) is always needed for 2-D mode compression. The expander disregards its K-Parameter register (EKPR) in CCITT Group III mode because the Tag Bit appended to the EOL suffixes indicates the coding scheme of the next line to be expanded. The expander K-Parameter is only used when suffix EOL codes are omitted (Non CCITT compatible mode, auto EOL mode off).

Examples of parameters for different formats:

The mode control parameter (in the MCR), the EOL field in the EPR/CPR and the K-parameter provides among others the following methods of decoding/encoding the image:

- 1D mode and Auto EOL
  All lines are coded in 1D mode only and terminated by a EOL code without a Tag bit.
- 2D mode with K = 1 and Auto EOL
  All lines are coded in 1D mode only and terminated by an EOL code with a Tag bit (= "1").
- 2D mode with 256 > K > 1 and Auto EOL.
  Lines are coded in 2D mode with interleaving 1D coded lines. Every line is terminated by an EOL code with a Tag bit indicating the coding scheme for the next line.
- 2D mode with K = 0 (infinite)
  All lines are coded in 2D mode. Each line is terminated with EOL code plus Tag bit (= "0") if Auto EOL is selected. Otherwise no EOL is appended.

Source Attribution:

The CCITT Group III recommendation defines a prefix indicating the beginning of a compressed image. For this prefix a single EOL code including a Tag bit is used. Group III apparatus starts transmitting signals as soon as the connection is established. The first EOL among these signals indicates the beginning of a compressed image page.

The system program must set the Source Attribute bit, SA, in the Parameter Registers (CPR/EPR) if the CEP is to recognize this prefix. If the SA bit is set, the CEP does not start expansion of the input data until it has received an EOL code. This feature is also very useful in searching for the next EOL in the code after a data error has occurred (see exception processing).

The Source Attribution bit must also be set before Group IV data is processed. Since all lines are coded in 2D mode, there is no reference line available to expand the first line of a page. For Group IV processing, the CEP must assume an imaginary all white line as reference for the first image line. The CEP does this if the following parameter setup is specified: No auto-EOL and K = 0 and 2-D mode and SA = 1. If the SA bit is not set, the CEP will not assume an imaginary white line (Refer to the technical manual for details on this non CCITT format). The SA bit is automatically reset by the CEP after processing of the first line. It usually does not need to be modified by the system until a new page is started or expander data error recovery is being attempted.

Format Control:

The two coding schemes also differ in the suffix used to indicate the end of a compressed image. Group III coding uses a sequence of six EOL's to indicate the end of the coded page. If 2-D mode is selected, each EOL is appended by a Tag bit set to '1'. The 1-D scheme uses EOL codes without Tag Bit. The compressor generates this type of suffix on a source buffer overflow if suffix RTC code format is selected in the parameter register (CPR). For non-Group IV processing, the expander needs three EOLs to recognize the end of a compressed image.

Group IV coding uses an EOP code to indicate the end of a compressed image. The EOP consists of two plain EOL codes without any tag bits. The compressor generates these two EOLs on a source buffer overflow if Suffix EOP Code is selected in the Compressor Parameter Register. When Group IV processing is selected the expander needs a sequence of two EOL codes to recognize the end of a compressed image.

If the source buffer is smaller than the entire document, a source buffer overflow may not necessarily indicate the end of a page. If it is not an end of page, RTC or EOP suffix should not be appended. Therefore, the first part of the image must be compressed in byte-boundaries or non byte-boundaries mode without selecting a suffix. When the last portion of the document is being processed, the data format control is switched to Suffix RTC or EOP code mode. Alternatively, if the size of the image buffer is large enough to permit processing of an entire image without stopping, the suffix options can be selected right at the beginning.

Both suffix RTC and EOP code formats imply non Byte-boundary formatted code. If the image has to be compressed in byte boundary mode, all scan lines except the last one have to be compressed with the byte boundary mode selected. Then for the last line the compression must be resumed with either Suffix RTC or EOP code mode and single line operation mode selected.

Changing modes between lines must be done very carefully because it might corrupt overhanging code from the previous line which is still in the output pipeline (Refer to the technical manual for more details).

In single line operation each compressed line is suffixed with the code selected in the Parameter Register.

When the expander recognizes either an RTC or EOP code, it terminates the operation and sets the EOP Code Detected flag in the Master Status Register (MSR).

Fill Bits:

If a coded line and the RTC code does not end on a byte boundary, the compressor adds fill bits between the compressed data and the RTC code to end the line on a byte boundary. Alternatively, if the code is terminated by an EOP code, the CEP adds fill bits after the EOP code to end the code on a byte boundary. Fill bits are always "0".

The compressor also inserts enough fill bits between a coded line and the EOL code to end a code line (including the EOL) on a byte boundary.

Time Fill:

The CCITT recommendation for Group III data requires each coded line to have certain minimum length. This is necessary for compatibility with older and slower equipment which cannot print fast enough.

The Time Fill Register (TFLR) specifies the time fill parameter which defines the minimum number of bytes in a coded scan line. If the generated code line is shorter than this, enough fill bytes (='0') are added between the code and the EOL to meet the minimum requirement.

The Time Fill Parameter can be calculated by the following formula:

TFLR = MSP * MTT / 8

MSP : Modem Speed
MTT : Minimum Transmission Time

When the Auto-EOL feature is supressed, the Am7971A ignores the time fill requirement; no time fill is inserted.

Bit Ordering in code and image bytes in memory:

In memory, the bits representing the pixels are stored as bytes. The first pixel at the top left corner of the image must be stored as the least significant bit of the first byte in the memory buffer. This is also the first bit to be transmitted. The compressed image follows the same rule.

This is contrary to normal computer convention where the LSB is the rightmost bit in the byte. Hand-coded data from the coding table (following traditional computer rules) result in the wrong bit order in each byte, and, therefore, requires the bit order in every byte to be reversed (bit 0 → bit 7, bit 1 → bit 6 etc).

Theoretically the bit ordering of image data does not matter, but the wrong bit order will almost always result in additional, unnecessary color changes, and therefore upset the compression ratio.

Note also, that scanners are not standardized. Some put the leftmost bit into the LSB (as required by the CEP), some do the opposite and, therefore, require reversal of bit ordering either by software or by hardware.

VIII. NON-CCITT OPTIONS

Wraparound Mode:

The wraparound mode groups a specified number of scan lines together into one effective line during encoding. This mode cannot be used simultaneously with 2-D compression or Express Mode. In 1-D compression, it gains approximately 15% code efficiency because fewer EOL codes are used and longer runlength can be encoded using multiple make-up codes. This coding scheme is not compatible with the CCITT recommendations.

The number of lines grouped together is specified by the content of the wraparound register plus 1. If the value is "0", then the effective line is identical to a scan line.

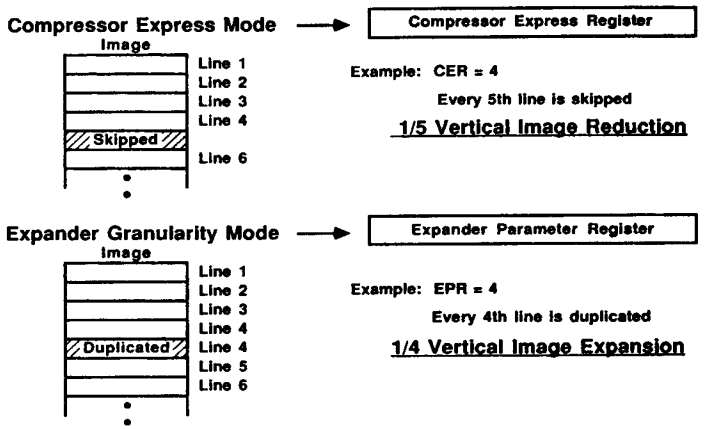


Figure 17. CEP Express and Granularity Modes

Vertical Image Resolution Conversion:

Figure 17 shows an example of the CEP's capability to reduce or expand the image vertically during compression or expansion. This feature is not compatible with the CCITT recommendation.

Express Mode:

The compressor express register specifies how many scan lines to compress before skipping one line. For example Figure 17 shows how every fifth line will be skipped resulting in a vertical image reduction of 20%. A '0' value in this register disables this feature.

If the Express Mode is defined with 2-D Compression, each line that is compressed is also the reference line for the next line that is compressed. Skipped lines are not used as reference lines.

Granularity Control:

The G-Parameter bits in the Expander Parameter Register contain the granularity factor (G-Parameter). It specifies how many lines to expand before duplicating the last line expanded. For instance, Figure 17 shows an example with G = 4. Here, every fourth scan line is duplicated resulting in a 25% vertical expansion.

IX. PROCESS CONTROL

Compression/Expansion Sequence:

Conceptually, the CEP operates as a coprocessor. The expansion and compression process is performed with minimum host CPU's assistance. Since the CEP is equipped with full Master DMA capability once a compression or expansion process is started, a whole page could be processed without any CPU intervention.

A compression or expansion sequence has the following steps:

- The host system provides the data in a memory location accessible by the CEP.
- The host system issues a software reset to the CEP
- The host system initializes the CEP's registers.
- The host system starts the CEP.
- The CEP accesses the data in the specified memory buffers and processes an image until a full page is completed or an exception condition occurs.
- The CEP alerts the system about the termination of its operation by asserting an interrupt.
- The system examines the CEP status registers to determine the cause of the interrupt. According to this information, it updates the memory buffers and initializes the CEP. The CEP resumes its operation or starts a new sequence for the next page.

The CEP executes three different operations, the Software Reset, Single Line Operation and Multi Line Operation. They are selected by the Operation Control Bits in the Master Control Register (CMCR, EMCR) and started by setting the 'GO' bit. The 'GO' bit is automatically reset after completion of the selected operation.

- The Software Reset clears the internal working registers, process control flags, the status and interrupt registers, sets "busy" to zero, flushes the input queue, and sets up the check for configurational errors and flags them. It is generally used to bring the CEP into an idle condition from where it can start on a new page. The Software Reset does not alter any user programmable registers.

- The Single Line Operation terminates the expansion or compression procedure after each processed effective image line. An effective image line can be longer than a single scan line if wraparound mode is selected (see options). The single line operation is useful for debugging, in systems with single line buffers. It is also used in some special situations (processing the last line in byte boundary mode, processing of negative compression or data errors).

- The Multi Line Operation terminates the process when an exception condition occurs. All pointers are automatically updated internally for each new image line. In this mode, the number of lines to be compressed or expanded is determined by the specified buffer sizes. Alternately the expansion can proceed until the end of a page is detected from a terminating code (RTC or EOP).

An entire image may be compressed or expanded in one operation if the code buffer and the image buffer are both large enough to contain the entire image. In this case the system program must specify a software reset operation before starting the compression or expansion of each new document.

The code buffer (compressor destination, expander source) can be smaller than an entire document. The CEP stops at the end of the buffer indicating a buffer overflow or underflow (This document makes no distinction between overflow and underflow of any buffer). Processing can be continued without issuing a reset after the buffers are updated. Similarily, the CEP is able to operate on image buffers that are smaller than the entire document. (Note: image buffer size should be an internal multiple of page width-PWR.)

Full duplex operation:

The Am7971A has two different operating configurations. In the full-duplex mode, the Expander and the Compressor are operated simultaneously. In the half-duplex mode, either the Expander or the Compressor are operating. Setting a '1' in the GO bit of the CMCR starts compression. A "1" in the GO bit of the EMCR starts expansion. For full-duplex operation, load a "1" into the GO bit of each register.

The expander and compressor sections work independently during full duplex operation. Either one can terminate its operation separately after an exception condition and assert an interrupt. The cause of the interrupt can be observed from the status registers and the terminated section can resume operation while the other section is still busy.

Initializing the CEP:

The Am7971A has the following initialization requirements:

- Source Buffer definition
- Destination Buffer definition
- Attributes
- Control Parameters
- Restart Condition
- Operating Mode
- Processing Mode
- Data Format
- Paper Format
- Minimum transmission time
- Options

These requirements are met by writing appropriate information into the 46 registers in the CEP. The system program should specify certain initial conditions before starting the operation of the Am7971A. After the CEP is started by setting the GO bit in the Master Control register, the CEP checks the parameters for consistency. If it finds an illogical condition, it terminates.

X. STATUS CONTROL

The Status Registers:

The CEP has three status registers: the Master Status Register (MSR), the Compressor Status Register (CSR), and the Expander Status Register (ESR). The MSR provides information about the general status of both the compressor and the expander. ESR and CSR inform specifically about the expander or compressor.

Most important are the EBY and CBY Bits in the MSR. The CBY Bit is set High when the compressor is busy and Low when the process is terminated. The EBY Bit provides the same information for the expander side. The CBY in the CSR and the EBY in the ESR are directly hardwired to those in the MSR (The meaning of all other status bits is explained under exception processing).

Interrupt Handling:

When interrupt mode is selected, the interrupt (INTR) signal is asserted upon termination of a process. This is when the busy bit returns to "0". The interrupt mode is selected by setting the interrupt enable bits (CIE,EIE) either in the expander or compressor master control registers (EMCR,CMCR). The interrupt can be enabled separately for the expander and the compressor.

The INTR line will remain High until the MSR has been accessed by the system. The system program must test the MSR register to distinguish Compressor interrupts from Expander interrupts by reading the busy bits. The system program then isolates the cause of the interrupt by reading the appropriate status register (CSR or ESR). Reading the Master Status Register clears the interrupt. The system program may then execute its interrupt service routine to respond to the interrupt.

Polling the Status:

If interrupts have not been enabled, the system program should periodically poll EBY and CBY by reading the MSR register. Polling the MSR is much faster than polling ESR or CSR because this register is directly accessible by the system without significant delay. All other registers are accessible only through the internal microprogram which is interrupted by a register access attempt of the system. Since this procedure is rather slow it is not recommended to access any other registers beside the MSR while the CEP is busy. System and CEP performance would be degraded.

Interrupt mode is better than polling operation because it frees the system from monitoring the busy bits. It is by far the most efficient way to control the CEP.

XI. EXCEPTION PROCESSING

The CEP will terminate its operation when there is a deviation from the normal compression or expansion sequence. These deviations are called exceptions and always require system interaction. An exception from normal processing occurs at the end of a page (EOP detected), on a buffer overflow, if negative compression is detected, when a data error or an illegal extension code is detected during expansion, if an illegal command was issued, or if the system aborts the procedure. Handling these exceptions properly is key to a successful expansion or compression of an image page.

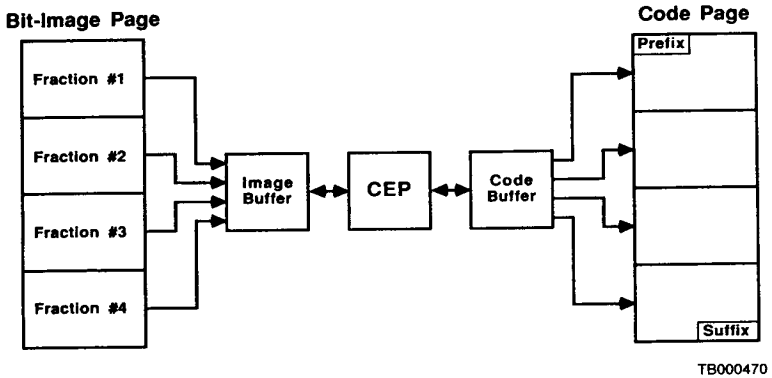


Figure 18. Fractional Page Processing (Memory Buffer Smaller Document Size)

5

Buffer Overflow (Operating on fractional code or image buffer):

The CEP is designed to work with any amount of memory available in a system. Thus the size of the source and destination buffers for expansion and compression may vary from 1 Byte to 16MByte. The sizes are defined in the Count Hold Registers. If a buffer is not large enough to accommodate the code of an entire document or the entire image data of a document, the CEP will run into an overflow condition.

When the CEP finishes processing the last byte of one of the buffers before the expansion or compression process is completed, it terminates the operation, asserts an interrupt and indicates this event by resetting the Busy Bit (Master Status Register) and setting the buffer overflow bits in the Expander or Compressor Status Registers.

After a buffer overflow, the CEP provides all the information needed to proceed with the compression or expansion from any boundary condition without producing code or image inconsistencies. There is only one restriction: since the CEP is designed for operation on image scan line image buffer must always be specified as an integral multiple of a scan line as defined in the page width registers (EPWR/CPWR) or an integral multiple of wrapped line length (if EWR/CWR is non-zero).

The buffer overflow status:

The host system exercises the CEP's status registers to determine which buffer is exhausted. There can be an overflow of the source or destination buffer or a simultaneous overflow on both sides. During full duplex operation four overflows can happen at the same time (expander/compressor source/destination). The overflow bits in the ESR and CSR indicate the source of an overflow.

Updating memory buffers after overflow:

The system must now decide if a destination buffer has to be saved or a source buffer has to be loaded with new data. If the two dimensional coding scheme is used, care must be taken that the last complete image line in the compressor source buffer is preserved, since it is needed as reference line for the following line except for the following conditions: when the two dimensional code is intermixed with one dimensionally coded lines (CCITT Group III) the reference line may be overwritten if the user specifies the buffer size such that the CEP compresses the first scan line at the beginning of a new buffer in the one dimensional coding scheme.

The system's response to an exception condition is crucial for the overall performance of the CEP. Therefore, the procedure for recovery from a buffer overflow condition needs to be highly optimized. The address hold registers and the restart

control registers were implemented to minimize the number of register accesses necessary to resume from a buffer overflow condition.

Optimizing the resume operation:

The address hold registers (EDAHR, ESAHR, CDAHR, CSAHR) keep a backup of the initial start positions of all memory buffers. The Count Hold Registers (EDCHR, ESCHR, CDCHR, CSCHR) keep a backup of the initial sizes of all buffers. The Restart Control Register (CRCR, ERCR) specifies whether or not an Address Hold Register is used to update the Current Address Register or the Count Hold Register is used to update the Working Count Registers.

Each bit in the Restart Control Registers specifies for a specific register whether its content is untouched (continue) or if it is loaded from the corresponding Hold Register (Restart) after the 'GO' bit is set for resumption of CEP operation. Accordingly, two different resume procedures are possible. The system can disregard the Hold Registers, initialize and update all Current and Working Registers and always keep the Restart Control Bits on Continue Mode. Under these circumstances, the Hold Registers are not used by the CEP at all. Alternately, the system can initialize the Hold Registers once at the beginning. Then the Restart Control Bits are used to specify which working registers are to be automatically loaded from the hold registers on start or resumption.

A typical system interaction:

The second choice is much more efficient because no address pointers have to be transferred. A typical system interaction on a buffer overflow exception would only consist of four CEP register accesses :

- Read Master status register. The MSR informs about the reason for terminating.
- Read the status of the CEP section in use (compressor or expander). The status register provides all the information needed to evaluate a situation and to decide what action to take and determine how the CEP can be initialized for a resume operation.
- Write the correct resume pattern (restart or continue) into the Restart Control Register. The use of the Address Hold Registers in the CEP by a Restart condition avoids any additional update of address pointers.
- Set the "GO" bit in the MCR.

If, for example, a source buffer overflow occurred, all Restart Control Bits are set to continue except the Source Address Control bit and the Source Count Control bit. Then, after the CEP is started, it will continue processing the destination buffer from the location where it stopped. In contrast, the CEP starts processing the source buffer from the beginning as defined in the Hold Register for a number of bytes as defined in the Count Hold Register.

Note, that a software reset is fatal for a proper resume operation since it clears the whole internal status and flushes the internal pipelines. If a software reset occurs after a buffer overflow termination, the CEP cannot encode or decode a consistent code string after resumption.

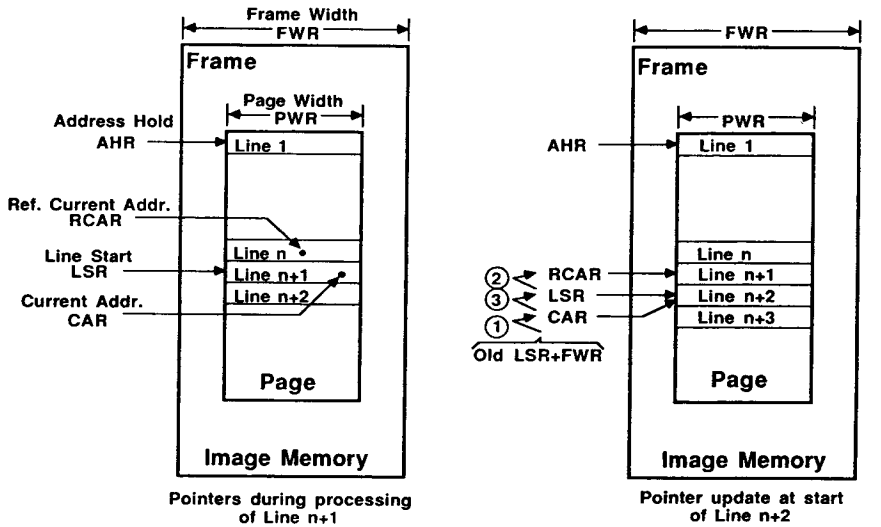


Figure 19. Address Pointer Management (During Multi-Line Operation)

The Line Start Register:

The Line Start Address Registers (EDLSR, ESLSR, CDLSR, CSLSR) have an important role for the management of the memory buffers and for proper recovery from data errors or negative compression. During multi-line operation the LSR is used to calculate the begin of consecutive image lines.

Figure 19 shows three important pointers used for processing an image line (expanding or compression). While the RCAR

and the CAR move through the reference and the current line, the LSR points to the first byte of the line currently being processed. After a line is completed, a new CAR is calculated by adding the frame width to the LSR. Then the previous line start address is loaded into the reference current address pointer. Finally the new CAR is loaded into the LSR and processing of the next line can begin. The Address Hold Register should normally be pointing to the beginning of the memory buffer.

Beside the Line Start Register for the image buffer (EDLSR, CSLSR) the CEP also provides LSRs for the code buffers (ESLSR, CDLSR). These registers always point to the EOL code (suffix) of the previous line or to the byte preceding it. They are updated after an EOL code is generated by the compressor or detected by the expander.

The Line Start Registers are very useful when an exception occurs because the information about the start position of the image line and its corresponding code is still available to the system. The system can also modify the LSRs to adjust the CEP's memory management to changed memory definitions. This is necessary for example during recovery from a data error (see data error) or for resumption after certain types of buffer overflow as described below.

Specifying the Line Start Registers:

At the beginning of a page the CEP is usually started in Restart Mode. Thus all working registers are loaded from the hold registers. If this is the case (Line Start Address Control Bit = '0'), the LSR is loaded from the address hold register. From there on 'Continue' Mode is selected, so that the LSR is only modified by the CEP for each new line it starts to process.

The Reference Line Register:

As shown in figure 19 in 2-D Compression Mode the content of the LSR is loaded into the Reference Current Address Register (ERCAR,CRCAR) before the CEP starts processing the next line. They serve as internal address registers pointing to the next address in the image buffer which the CEP will access for reference. Like the current address registers, they are always incremented by one. Altering the content of the Reference Line Registers will most likely corrupt the CEP's operation. However, there is no problem in reading them for debugging purposes. Also, it is necessary to write into these registers under certain conditions during expander error recovery from a transmission error.

If the position of the reference line must be changed, it can be specified through the LSR if 'Continue' operation is chosen and a new line is started. This is important in 2-D mode when the source buffer for the compressor is exhausted. When the system updates this buffer with new image data, the reference line is lost. The compressor starts a new line at the beginning of the new source buffer but expects the reference line still to exist at the end of this buffer. It thus generates incorrect code. To avoid corruption in this situation, the last line of image which was just processed, must be copied to a memory location outside this memory buffer, before the image lines are loaded into the source buffer. Before the compressor resumes, the position of the copied line is loaded into the CSLSR and the Compressor Source Line Start Control Bit set to continue. The CSCAR must be loaded with the start address of the source buffer (eg. from the Hold Register). The back-up line is then used correctly as reference line for the first line of the updated buffer.

For Group III processing this procedure can be avoided by specifying the compressor source buffer size as an integral multiple of the line length multiplied with the K-parameter. The source buffer will then always begin with a line which is coded in 1-D mode and no reference line access is necessary. On the expander side the the image buffer is not overwritten by the system, so it is still available at its old position after resumption. Therefore, no special expander destination overflow handling of the EDLSR is necessary in 2-D mode.

The Line Incomplete Bit:

The Line Incomplete Bit (LPI) in the Status Register is set when exception situation occurs while the CEP is in the middle of processing a scan line. The LPI of the Compressor is also set if there is partial code remaining in the CEP which has not yet been written out to the memory. This happens during non byte boundary processing because part of the code string has to be concatenated to the next line. LPI is always set on a data error during expansion (see section on data error).

Data Error:

The Group III CCITT-specification assumes the possibility of corrupted data because of transmission errors. If an image is coded in 2-D mode only, the remaining image after a data error would be lost. For this reason 2-D coded lines are interlaced with 1-D coded lines which are used by the expander to resynchronize the decoding procedure after a data error.

The code must be carefully screened for inconsistencies to keep distortion in the expanded image as small as possible. All logically possible methods to check the consistency of the coded input are implemented in the CEP, and it will cause a termination of the process with the Data Error Recognized Bit (DER) set in the Expander Status Register (ESR) if an inconsistent code is detected.

However, because of ambiguities of the coding scheme, not all data corruptions can be detected. There are situations in the 2D coding scheme where the code matches perfectly with the reference line and results in a correct image line length but still does not produce the correct image. An unpredictable number of lines are expanded after such a data error, until the code expander realizes a data corruption. A similar problem arises if the Tag-bit in the EOL code is flipped. The expander encodes a 2D line as a 1D line or vice versa and might not be able to detect the EOL code for an unpredictable number of lines. This is a problem with the coding scheme, not a lack of intelligence in the CEP.

Illegal Extension Codes:

A special case of data error is an illegal extension code which also terminates the process and sets the Extension Code Detected bit (ECD) in the MSR. In case of CCITT-compatible code this is always a data error and should be handled in the same manner. In non-CCITT compatible environments, this feature can be used to implement a variation from the specified CCITT coding table by using other extension codes than specified. This would have to be done by additional software. The expander detects these extension codes by flagging an illegal extension code. The software can than take over and resume processing the following code string.

The Error status:

The DER bit is set under the following circumstances:

- An illegal code is detected. The CEP stops immediately after detecting the illegal code, but the Current Address Register might point a couple of addresses ahead (The CEP prefetches up to 3 bytes in advance).
- A code expands into a negative runlength. This is possible in 2D code since the reference points in the reference line might be corrupted. The expander terminates as described above.
- After an EOL code is detected, the expanded image line does not match the specified page width (EPWR). The expander terminates on the end of this line.

The ECD bit is set under the following circumstances:

– An extension code is detected for which the least significant three bits are not all "1"s. The corresponding extension code can be read from the Extension Bits field in the MSR in reversed order.

A corrupted code might be expanded into a runlength larger than the specified page width before it is actually recognized as a data error. However, under all circumstances, the CEP clips an expanded image line to the specified line length (EPWR).

The LPI bit is always set in conjunction with the DER or ECD bit set.

Recovering from data errors:

After a data error is detected, an error recovery procedure iterates the CEP through the next lines until a resynchronization on a 1-D coded line is possible. The CEP proceeds from there on without further assistance from the system. Though the CEP cannot recover from a data error situation without support from the host system, it provides many features which reduce the systems burden to a minimum.

After beginning the expansion of a new image line, the CEP always stores the line start of the image and the line start of the code in the line start registers (ESLSR, EDLSR). After a data error, this information is very important. With this information the system can determine how much of the image was expanded correctly and the position of the last correct line. The corrupted image line can be overwritten either by a line of zeros (white) or, better, by duplicating the last correct line. The CEP's transparent mode operation can be used to perform this task.

Then the next EOL beyond the data error must be found as a point from which the CEP can resume its operation. This can be done by performing a software reset for the expander, then setting the source attribute bit (SA) in the expander parameter register and specifying a starting address between the data error and the next EOL and then starting the CEP. Under this condition, the CEP thinks it is starting at the beginning of a coded page and starts searching for an EOL because CCITT defines a prefix EOL code for Group III data. After it recognizes the next EOL, it automatically proceeds to expand the code following the EOL.

After resuming from a data error in a 2D coded line, the expander will most likely, but not necessarily, decode another 2D line and get another data error because the reference line is not correct. The above described procedure must be repeated until a 1D coded line is reached. From there on, the CEP can proceed without further corruptions in the expanded image.

Since the expander prefetches up to three bytes, the next EOL might be part of the prefetched code. It might also be corrupted itself. Therefore, great care must be taken to calculate the correct address to resume from (The CEP Technical Manual gives detailed information on how to determine the correct address).

Negative Compression:

For some images (such as half-tones or low resolution raster-pictures), the compressed data representing a line may be longer than the original line of the image. The Am7971A checks for this condition after compressing a line providing bit 0 of the Compressor Parameter Register (CPR) has been cleared to "0" by the user. The Am7971 automatically checks for negative compression regardless of the status of this bit. When it occurs, the host processor is alerted by an interrupt. The CEP then terminates the compression and sets the negative compression bit in its status register. If bit 0 in the CPR is set to "1" by the user, the Am7971A does not check for negative compression and continues to process the document.

The host system can react in one of the following three possible ways:

1. Set all restart control bits in the restart control register (CRCR) to continue and set compressors to "GO". This action tolerates the negative compression and just continues operation.
2. Read the source and destination line start addresses (CSLSR, CDLSR) from the compressor registers and re-encode the same line in uncompressed mode by software. Then continue compression by reinitializing the destination current address register and following the same steps as above.
3. Tolerate negative compression as described under 1. If the whole coded page is negatively compressed, transmit or store the image uncompressed.

Illegal Command Error:

The CEP immediately terminates its operation and indicates an illegal command condition under the following circumstances:

Compressor:

- 2D operation is selected and wraparound register is non-zero.
- Wraparound and express mode are selected together
- Left and right margins are overlapping or larger than paper width
- Reserved Modes are selected in the CMCR
- Compressor Page Width has been selected as '0'

Expander:

- 2D expansion mode and wraparound mode have both been specified
- A non-zero granularity parameter and a non-zero wraparound have both been specified
- Reserved Modes are selected in the EMCR
- The expander page width has been selected as "0"

Only conditions which are vital for a consistent operation of the CEP's internal logic are checked. Other parameters (like frame width, etc) are not checked for consistency. The check is only performed on a 'GO' after a reset was performed. For reasons of efficiency, the check is suppressed on all subsequent resume operations.

Abort Condition:

Any attempt to write into the EMCR or CMCR while the CEP is busy will cause the current expander or compressor operation to be aborted. The expander/compressor busy and new operation attempted bit is set and an interrupt asserted. The CEP cannot resume from such a condition. However, an abort is useful to stop the CEP in a system emergency situation. The CEP ignores any attempt to write into expander registers other than EMCR while the expander is busy and compressor registers other than EMCR while the compressor is busy.

XII. CEP REGISTERS

TABLE 1. COMPRESSOR REGISTER ADDRESS ASSIGNMENTS

Abbr.	Name	Size (Bits)	Number of Bytes	Port Address(es)
MSR*	Master Status Register	8	1	FE
CMCR	Compressor Master Control Register	8	1	76
CPR	Compressor Parameter Register	8	1	74
CSR	Compressor Status Register	8	1	78
CER	Compressor Express Register	8	1	68
CRCR	Compressor Restart Control Register	8	1	48
CKPR	Compressor K Parameter Register	8	1	66
TFLR	Time Fill Register	8	1	44
CBOCR	Compressor Bit Offset Control Register	8	1	7A
CWR	Compressor Wraparound Register	16	2	50 (LSB)/52 (MSB)
LMGR	Left Margin Register	16	2	40 (LSB)/42 (MSB)
RMGR	Right Margin Register	16	2	60 (LSB)/62 (MSB)
TMGR	Top Margin Register	16	2	30 (LSB)/32 (MSB)
CFWR	Compressor Frame Width Register	16	2	54 (LSB)/56 (MSB)
CPWR	Compressor Page Width Register	16	2	70 (LSB)/72 (MSB)
CSAHR	Compressor Source Address Holding Register	24	3	3A (LSB)/3C/3E (MSB)
CSCAR	Compressor Source Current Address Register	24	3	0A (LSB)/0C/0E (MSB)
CDAHR	Compressor Destination Address Holding Register	24	3	4A (LSB)/4C/4E (MSB)
CDCAR	Compressor Destination Current Address Register	24	3	2A (LSB)/2C/2E (MSB)
CSCHR	Compressor Source Count Holding Register	24	3	14 (LSB)/16/18 (MSB)
CSWCR	Compressor Source Working Count Register	24	3	04 (LSB)/06/08 (MSB)
CDCHR	Compressor Destination Count Holding Register	24	3	34 (LSB)/36/38 (MSB)
CDWCR	Compressor Destination Working Count Register	24	3	24 (LSB)/26/28 (MSB)
CSLSR	Compressor Source Line Start Address Register	24	3	5A (LSB)/5C/5E (MSB)
CDLSR	Compressor Destination Line Start Address Register	24	3	6A (LSB)/6C/6E (MSB)
CRCAR	Compressor Reference Current Address Register	24	3	1A (LSB)/1C/1E (MSB)

*This register is common to both the compressor and the expander.

TABLE 2. EXPANDER REGISTER ADDRESS ASSIGNMENTS

Abbr.	Name	Size (Bits)	Number of Bytes	Port Address(es)
MSR*	Master Status Register	8	1	FE
EBOCR	Expander Bit Offset Control Register	8	1	FA
EMCR	Expander Master Control Register	8	1	F6
EPR	Expander Parameter Register	8	1	F4
ESR	Expander Status Register	8	1	F8
ERCR	Expander Restart Control Register	8	1	C8
EKPR	Expander K Parameter Register	8	1	E6
EWR	Expander Wraparound Register	16	2	D0 (LSB)/D2 (MSB)
EPWR	Expander Page Width Register	16	2	F0 (LSB)/F2 (MSB)
EFWR	Expander Frame Width Register	16	2	D4 (LSB)/D6 (MSB)
ESAHR	Expander Source Address Holding Register	24	3	BA (LSB)/BC/BE (MSB)
ESCAR	Expander Source Current Address Register	24	3	8A (LSB)/8C/8E (MSB)
EDAHR	Expander Destination Address Holding Register	24	3	CA (LSB)/CC/CE (MSB)
EDCAR	Expander Destination Current Address Register	24	3	AA (LSB)/AC/AE (MSB)
ESCHR	Expander Source Count Holding Register	24	3	94 (LSB)/96/98 (MSB)
ESWCR	Expander Source Working Count Register	24	3	84 (LSB)/66/88 (MSB)
EDCHR	Expander Destination Count Holding Register	24	3	B4 (LSB)/B6/B8 (MSB)
EDWCR	Expander Destination Working Count Register	24	3	A4 (LSB)/A6/A8 (MSB)
ESLSR	Expander Source Line Start Address Register	24	3	DA (LSB)/DC/DE (MSB)
EDLSR	Expander Destination Line Start Address Register	24	3	EA (LSB)/EC/EE (MSB)
ERCAR	Expander Reference Current Address Register	24	3	9A (LSB)/9C/9E(MSB)

Note: All register addresses are even, the bytes in a register are, therefore, not addressed with contiguous addresses.

*This register is common to both the compressor and expander.

TABLE 3. CEP REGISTERS BY ADDRESS (LEAST SIGNIFICANT DIGIT)

Most Significant Digit	0	2	4	6	8	A	C	E
0			CSWCR (L)	CSWCR (M)	CSWCR (H)	CSCAR (L)	CSCAR (M)	CSCAR (H)
1			CSCHR (L)	CSCHR (M)	CSCHR (H)	CRCAR (L)	CRCHR (M)	CRCAR (H)
2			CDWCR (L)	CDWCR (M)	CDWCR (H)	CDCAR (L)	CDCAR (M)	CSCAR (H)
3	TMGR (L)	TMGR (H)	CDCHR (L)	CDCHR (M)	CDCHR (H)	CSAHR (L)	CSAHR (M)	CSAHR (H)
4	LMGR (L)	LMGR (H)	TFLR		CRCR	CDAHR (L)	CDAHR (M)	CDAHR (H)
5	CWR (L)	CWR (H)	CFWR (L)	CFWR (H)		CSLSR (L)	CSLSR (M)	CSLSR (H)
6	RMGR (L)	RMGR (H)		CKPR	CER	CDLSR (L)	CDLSR (M)	CDLSR (H)
7	CPWR (L)	CPWR (H)	CPR	CMCR	CSR	CBOCR		
8			ESWCR (L)	ESWCR (M)	ESWCR (H)	ESCAR (L)	ESCAR (M)	ESCAR (H)
9			ESCHR (L)	ESCHR (M)	ESCHR (H)	ERCAR (L)	ERCAR (M)	ERCAR (H)
A			EDWCR (L)	EDWCR (M)	EDWCR (H)	EDCAR (L)	EDCAR (M)	EDCAR (H)
B			EDCHR (L)	EDCHR (M)	EDCHR (H)	ESAHR (L)	ESAHR (M)	ESAHR (H)
C					ERCR	EDAHR (L)	EDAHR (M)	EDAHR (H)
D	EWR (L)	EWR (H)	EFWR (L)	EFWR (H)		ESLSR (L)	ESLSR (M)	ESLSR (H)
E				EKPR		EDLSR (L)	EDLSR (M)	EDLSR (H)
F	EPWR (L)	EPWR (H)	EPR	EMCR	ESR	EBOCR		MSR

(L): Low Byte
(M): Middle Byte
(H): High Byte

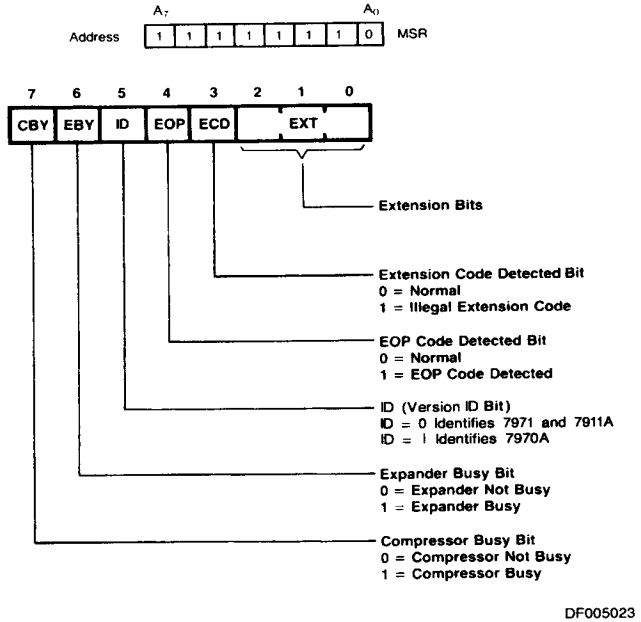


Figure 20. Master Status Register (MSR)

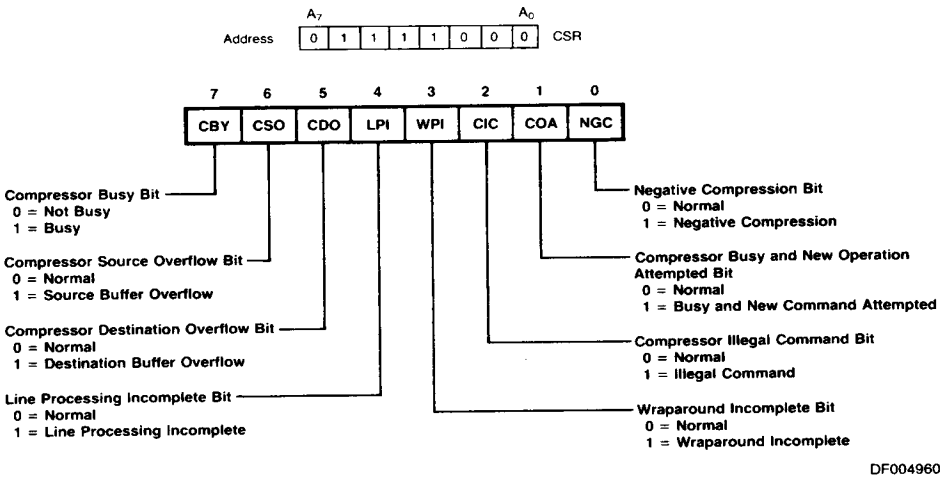


Figure 21. Compressor Status Register (CSR)

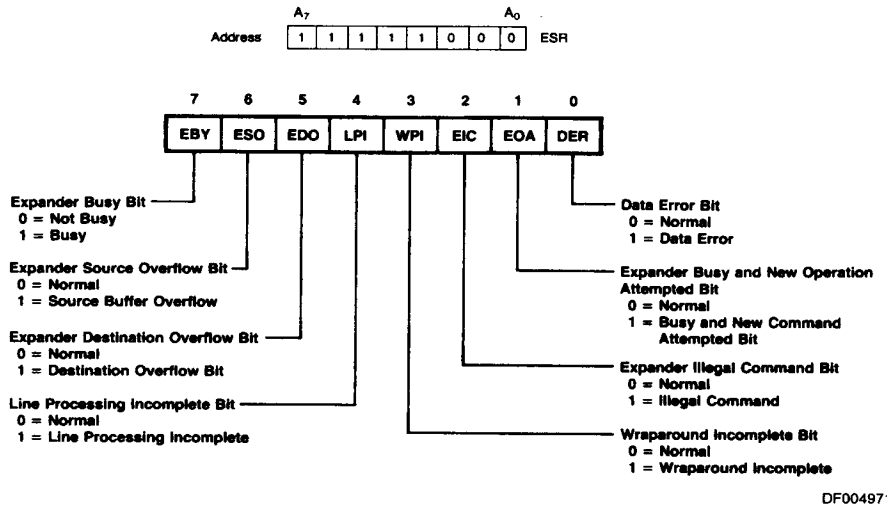


Figure 22. Expander Status Register (ESR)

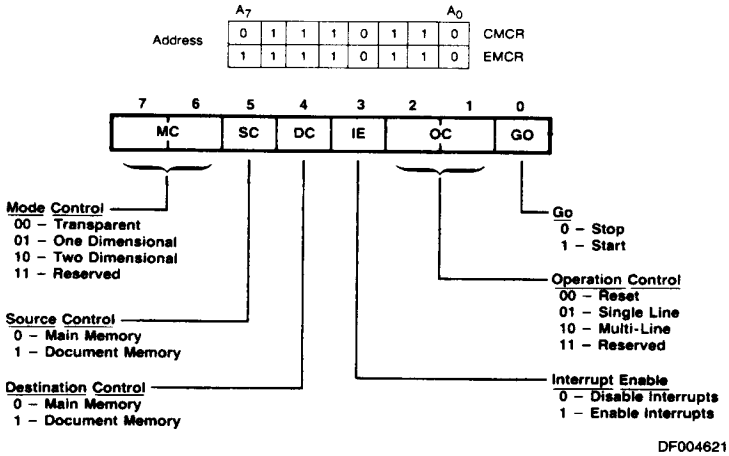
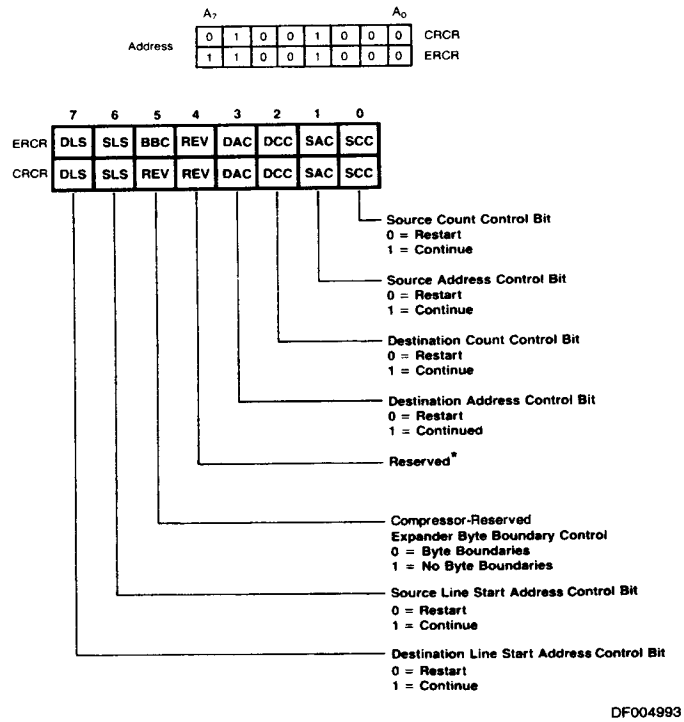


Figure 23. Master Control Register (CMCR, EMCR)



* All Reserved bits should be set to zero.

Figure 24. Restart Control Register (CRCR, ERCR)

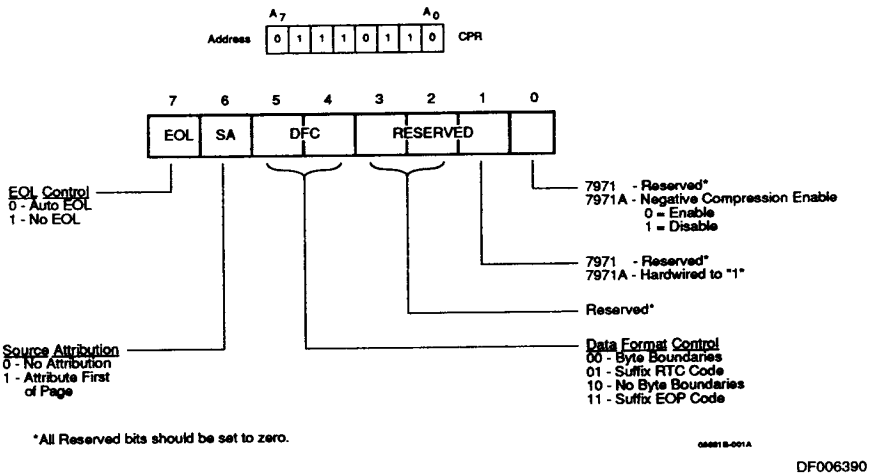
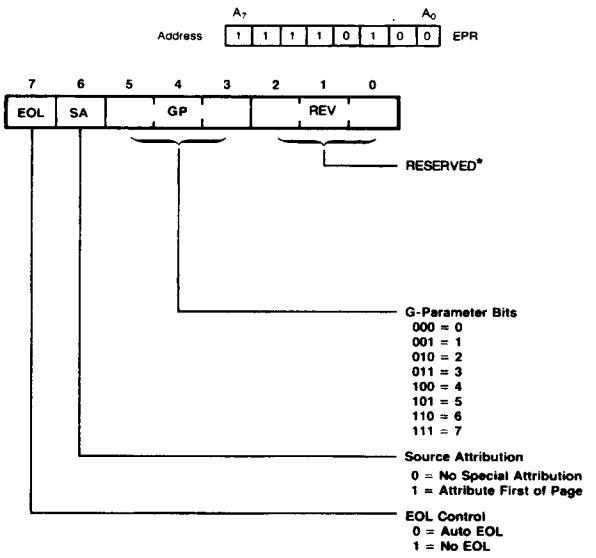


Figure 25. Compressor Parameter Register (CPR)



* All Reserved bits should be set to zero.

Figure 26. Expander Parameter Register (EPR)

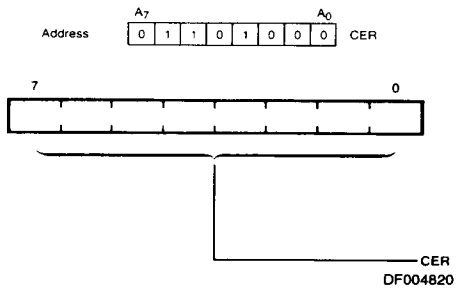


Figure 27. Compressor Express Register (CER)

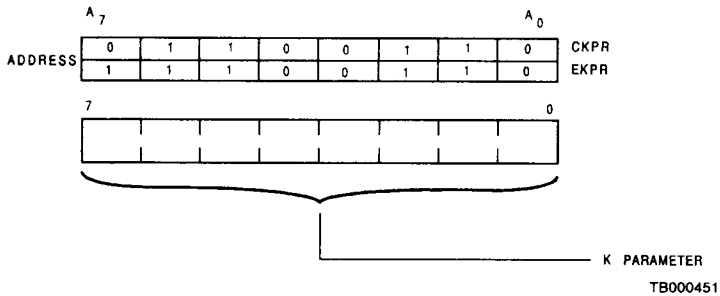


Figure 28. K Parameter Registers (CKPR, EKPR)

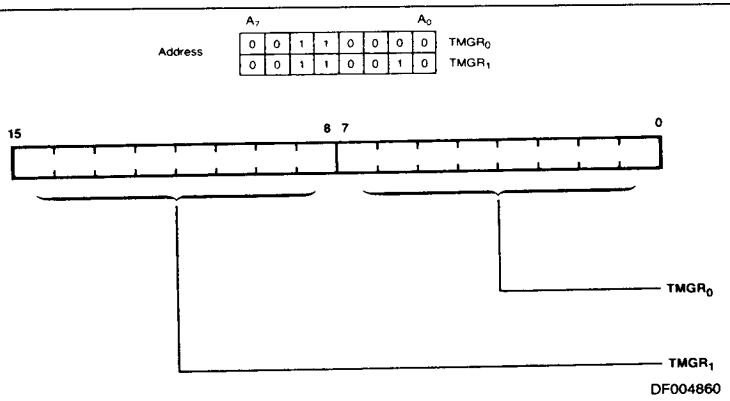


Figure 29. Top Margin Register (TMGR)

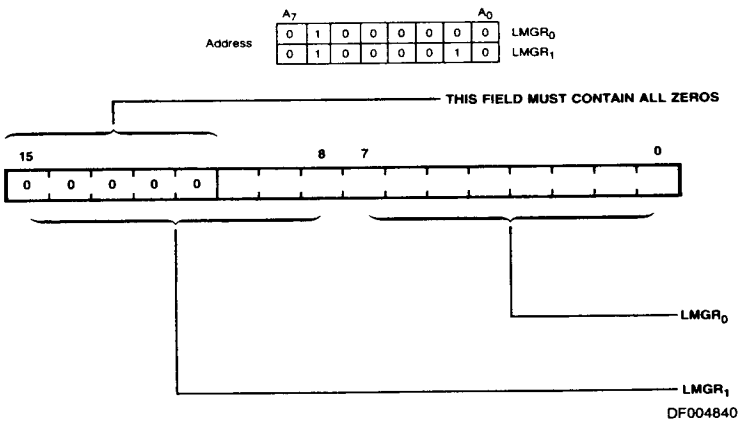


Figure 30. Left Margin Register (LMGR)

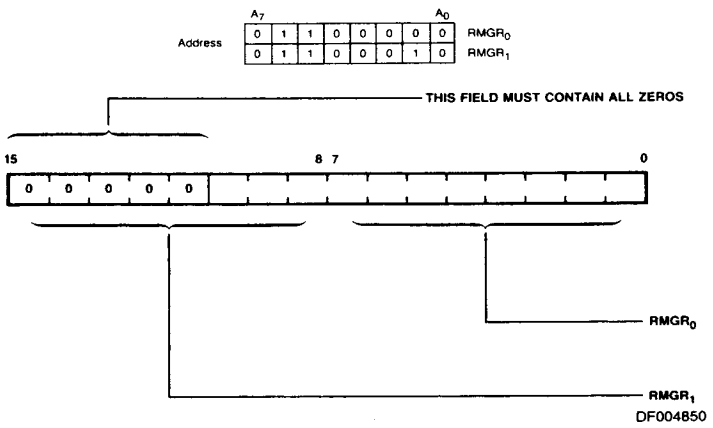


Figure 31. Right Margin Register (RMGR)

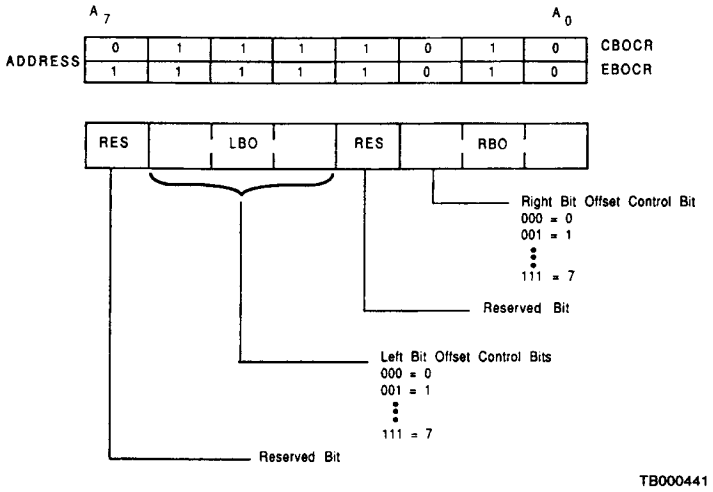


Figure 32. Bit Offset Control Register (CBOCR, EBOCR)

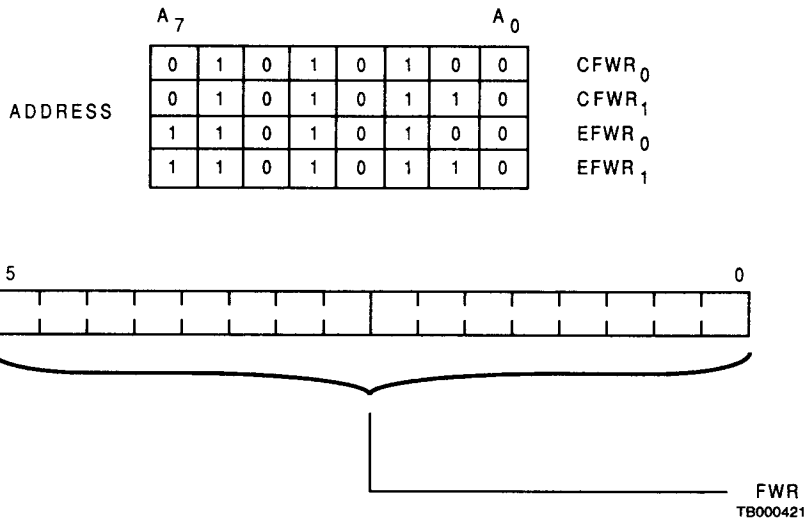


Figure 33. Frame Width Registers (CFWR, EFWR)

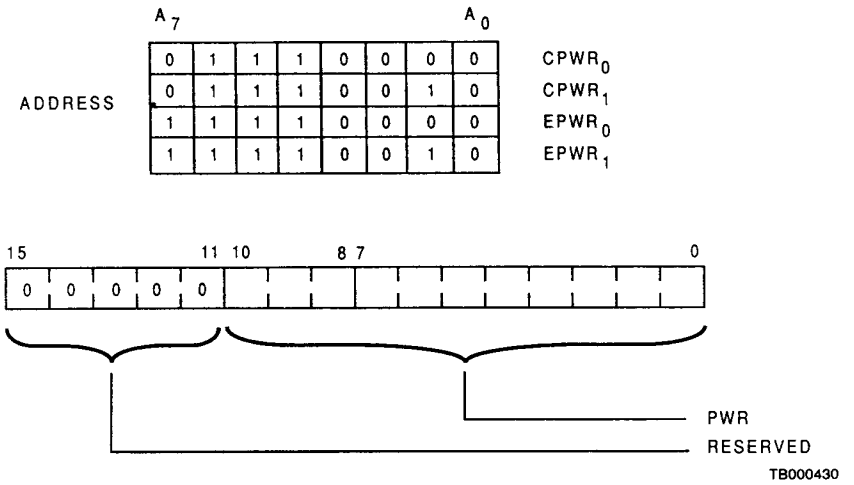


Figure 34. Page Width Registers (CPWR, EPWR)

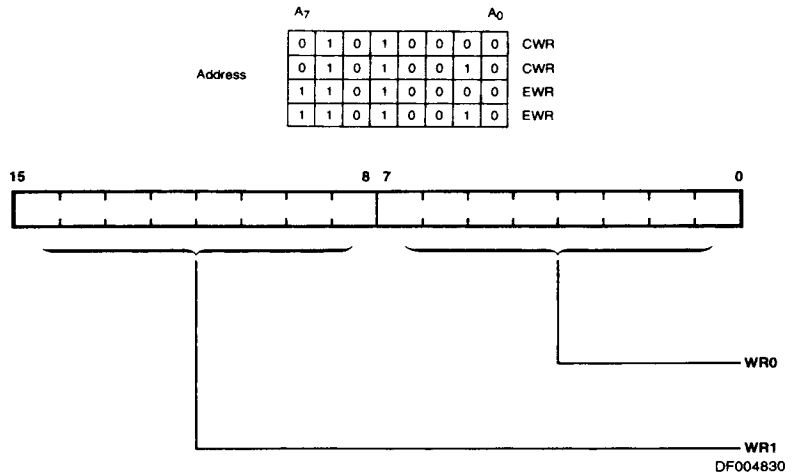


Figure 35. Wraparound Registers (CWR, EWR)

5

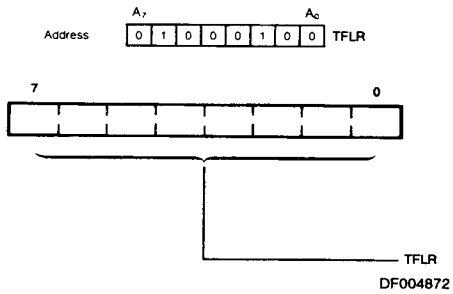


Figure 36. Time Fill Register (TFLR)

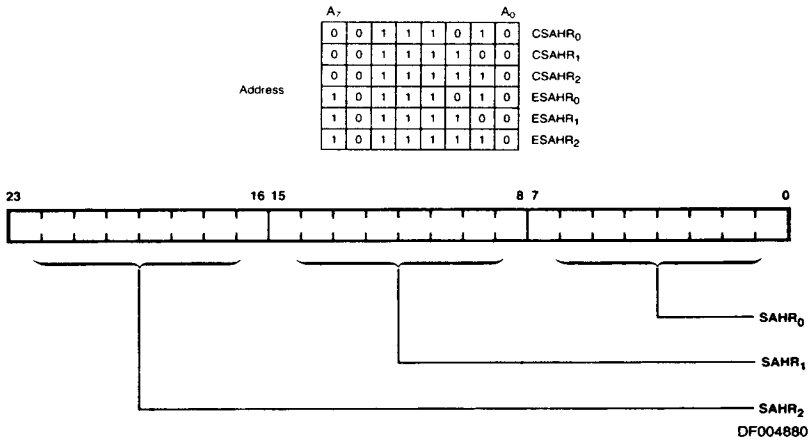


Figure 37. Source Address Holding Registers (CSAHR, ESAHR)

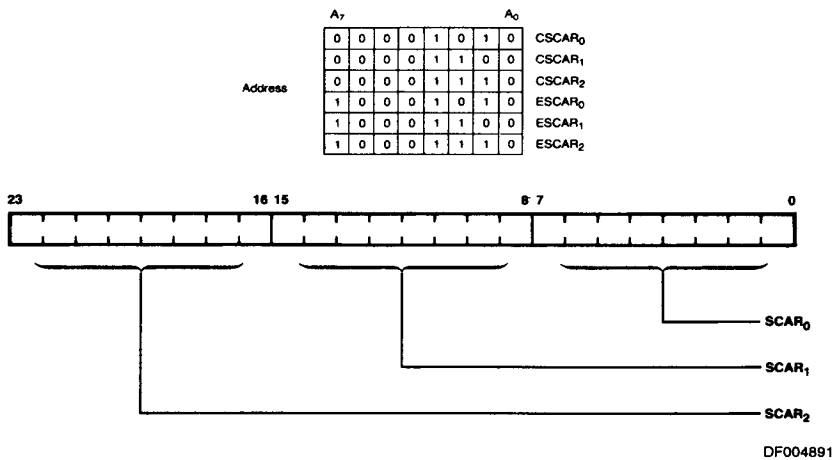


Figure 38. Source Current Address Registers (CSCAR, ESCAR)

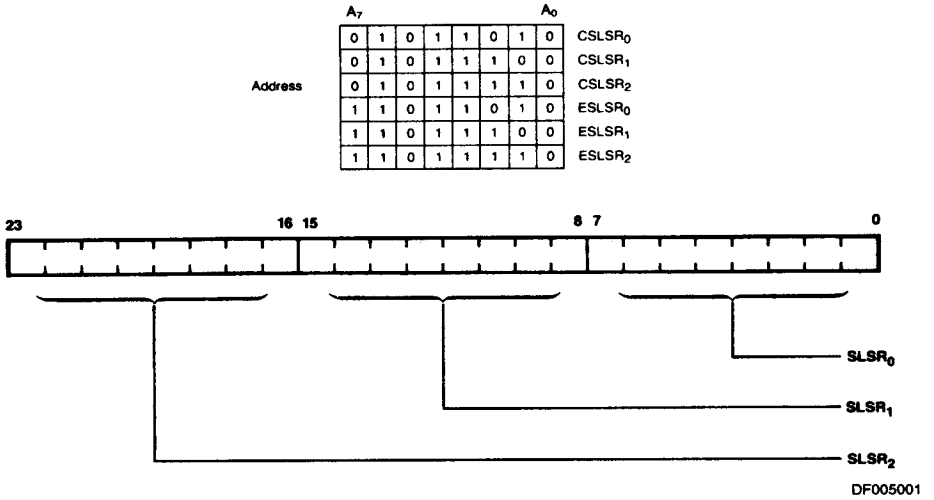


Figure 39. Source Line Start Address Registers (CSLSR, ESLSR)

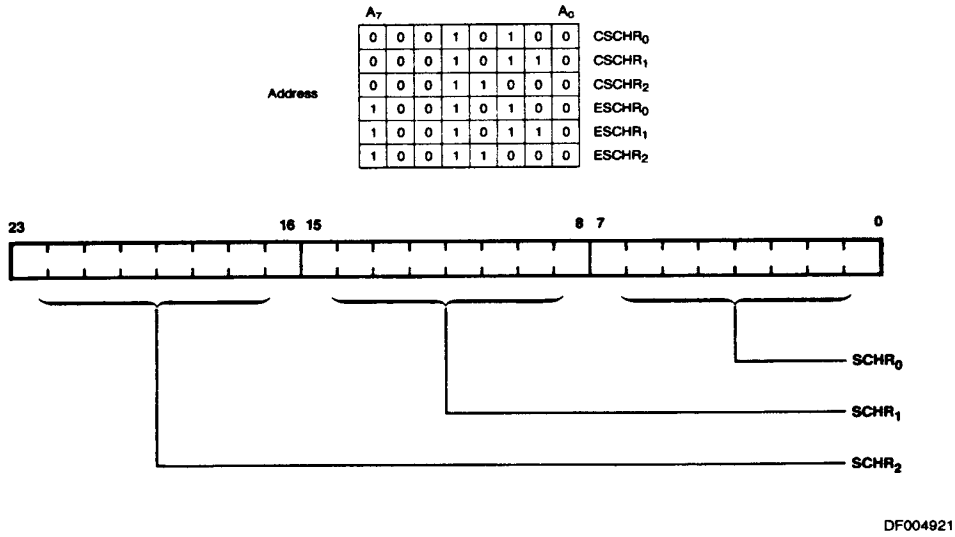


Figure 40. Source Count Holding Registers (CSCHR, ESCHR)

5

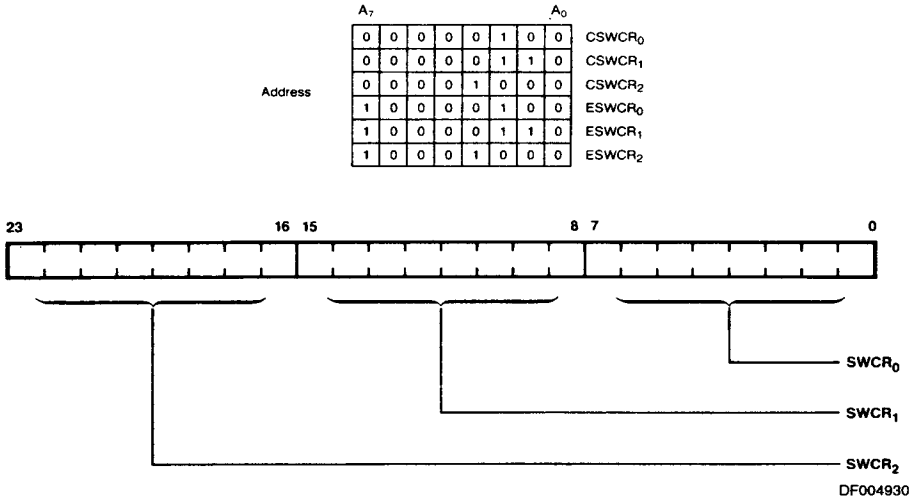


Figure 41. Source Working Count Registers (CSWCR, ESWCR)

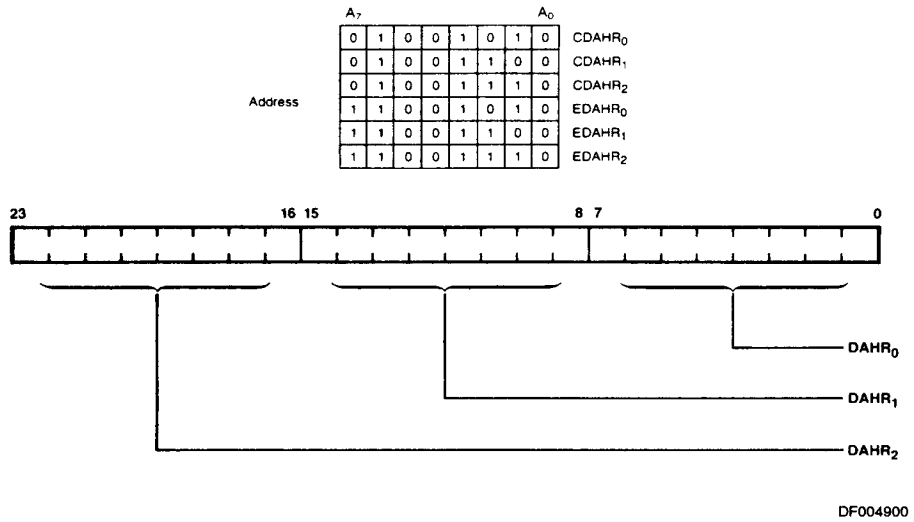


Figure 42. Destination Address Holding Registers (CDAHR, EDAHR)

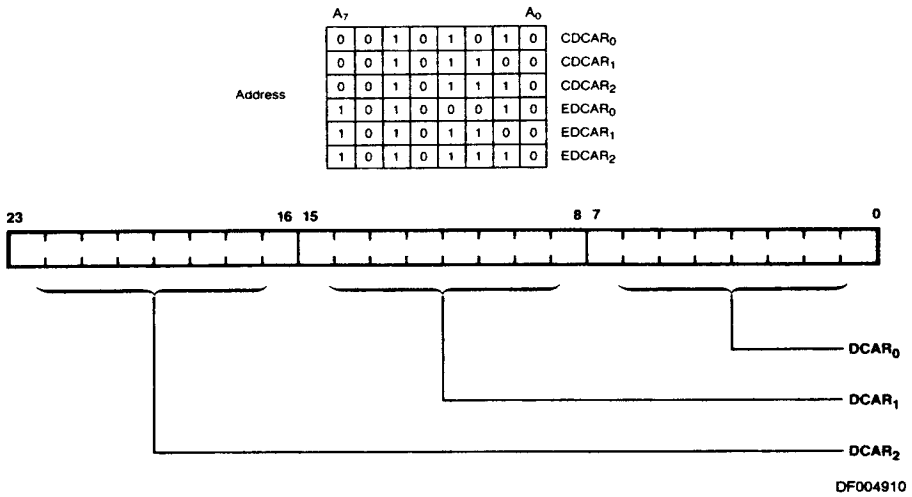


Figure 43. Destination Current Address Registers (CDCAR, EDCAR)

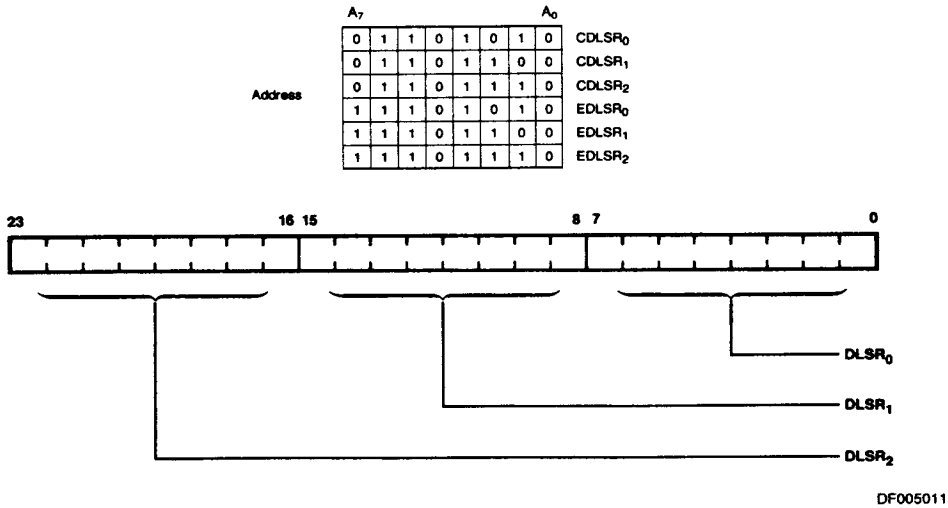


Figure 44. Destination Line Start Address Registers (CDLSR, EDLSR)

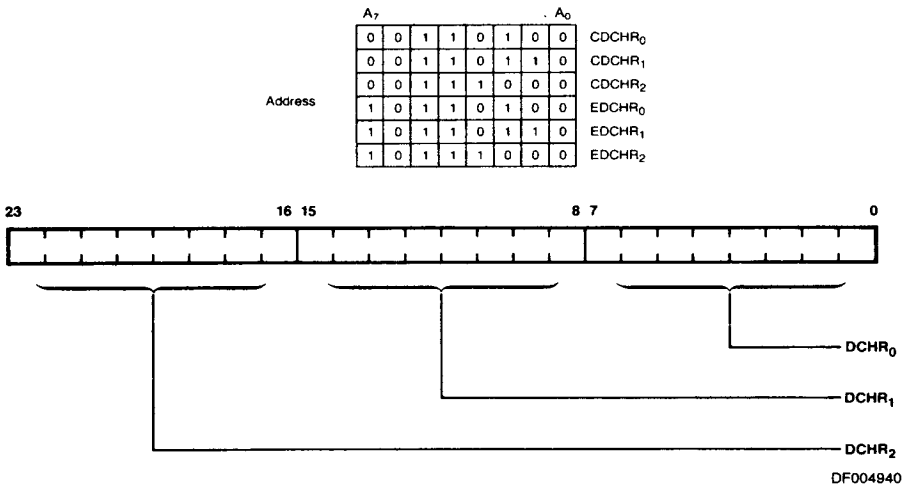


Figure 45. Destination Count Holding Registers (CDCHR, EDCHR)

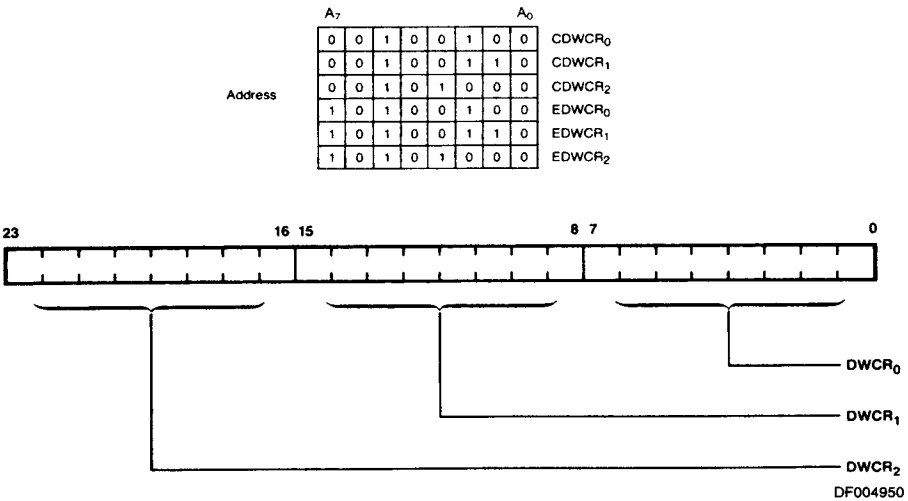


Figure 46. Destination Working Count Registers (CDWCR, EDWCR)

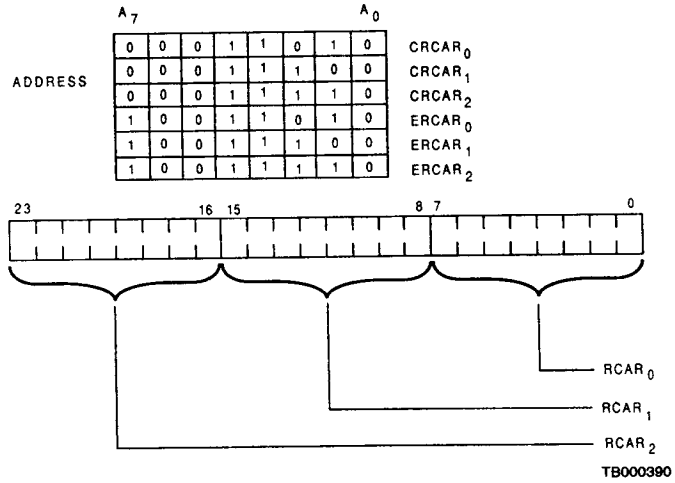
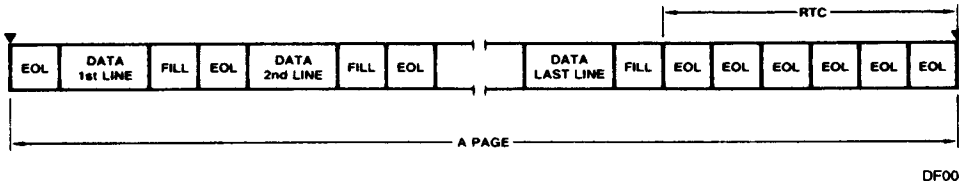


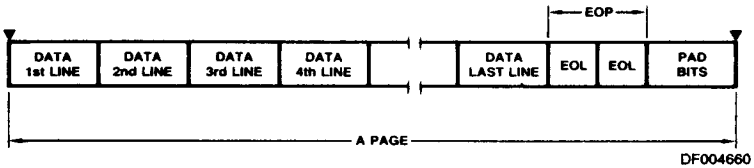
Figure 47. Reference Current Address Registers (CRCAR, ERCAR)

XIII. Am7971A CEP DATA FORMATS



▼ Byte Boundaries Mark

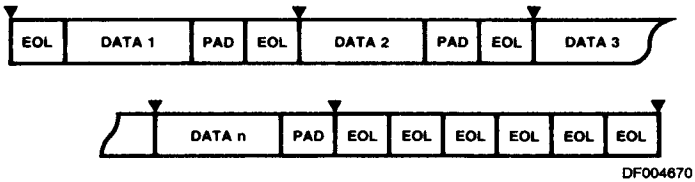
Figure 48. G-3 Compressed Data Format



▼ Byte Boundaries Mark

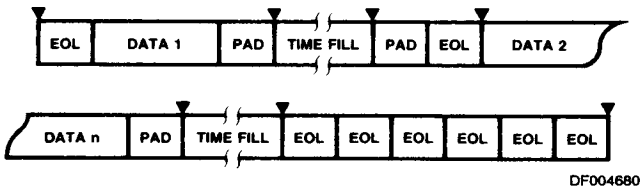
Figure 49. G-4 Compressed Data Format

Compressed Data Format of the 1-D Mode (Figures 50 through 55)



▼ Byte Boundaries Mark
PAD: Consecutive any numbers of 0's (if any)

Figure 50. Byte Boundary Conditioned with Auto EOL and No Time Fill



▼ Byte Boundaries Mark
PAD: 1 to 7 of 0's (if any)

Figure 51. Byte Boundary Conditioned with Auto EOL and Time Fill

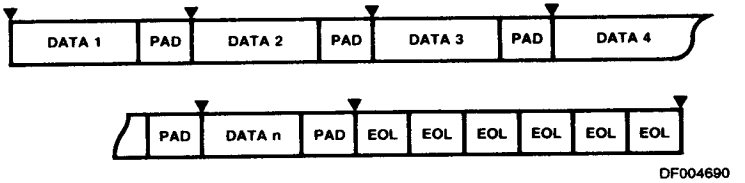
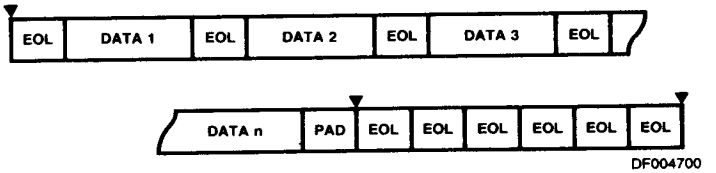


Figure 52. Byte Boundary Conditioned without EOL and Time Fill



▼ Byte Boundaries Mark
PAD: 1 to 7 of 0's (if any)

Figure 53. No Byte Boundary Conditioned with Auto EOL and No Time Fill

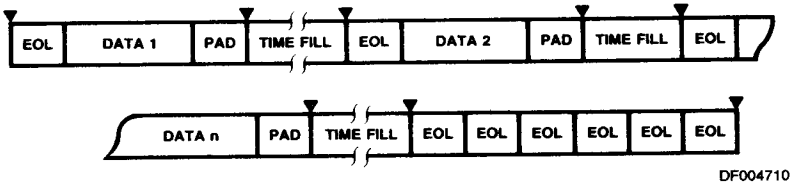
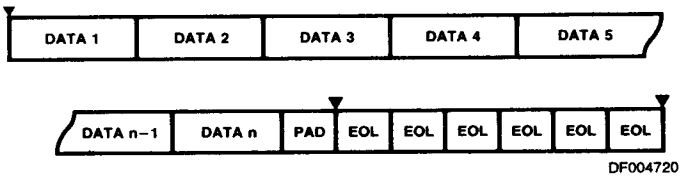


Figure 54. Byte Boundary Conditioned with Auto EOL and Time Fill



▼ Byte Boundaries Mark
PAD: 1 to 7 of 0's (if any)

Figure 55. No Byte Boundary Conditioned without EOL and Time Fill

Compressed Data Format of the 2-D Mode (Figures 56 through 64)

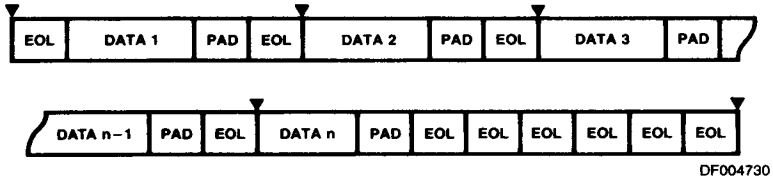


Figure 56. Byte Boundary Conditioned with Auto EOL and No Time Fill

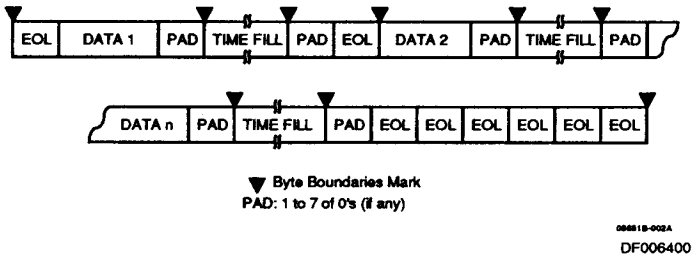


Figure 57. Byte Boundary Conditioned with Auto EOL and Time Fill

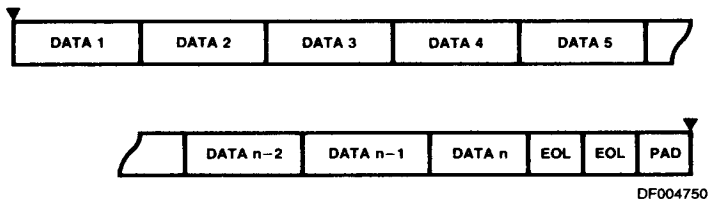
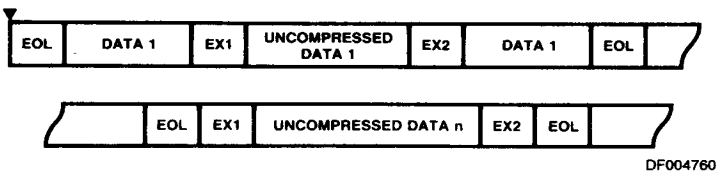


Figure 58. No Byte Boundary Conditioned without EOL and Time Fill (G-4)



▼ Byte Boundaries Mark
PAD: Consecutive any numbers of 0's (if any)
EX1: Extension code (entry code)
EX2: Extension code (exit code)

Figure 59. Uncompressed Data Format

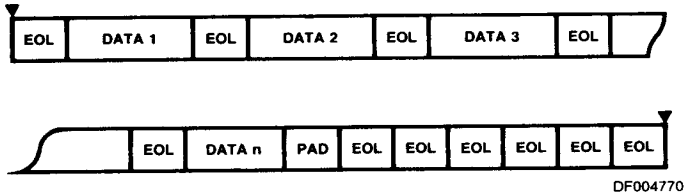
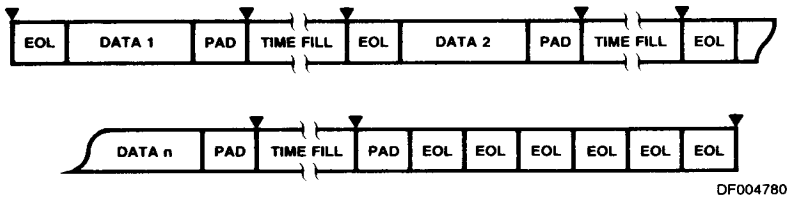


Figure 60. No Byte Boundary Conditioned with Auto EOL and No Time Fill



▼ Byte Boundaries Mark
PAD: 1 to 7 0's (if any)

Figure 61. No Byte Boundary Conditioned with Auto EOL and Time Fill

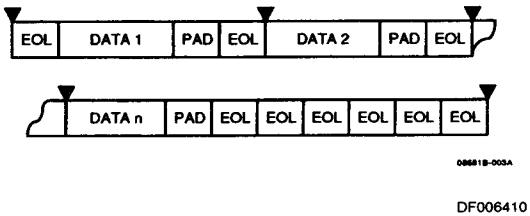


Figure 62. Byte Boundary Conditioned with Auto EOL and No Time Fill

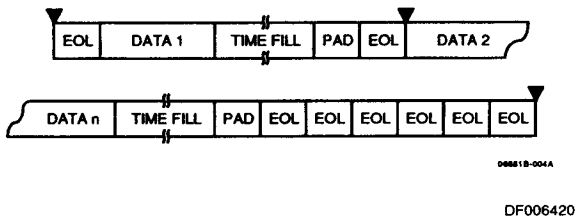


Figure 63. Byte Boundary Conditioned with Auto EOL and Time Fill

5

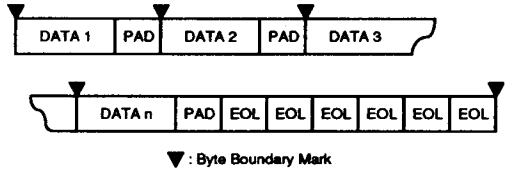


Figure 64. Byte Boundary Conditioned without EOL and Time Fill

ABSOLUTE MAXIMUM RATINGS

Storage Temperature ..... −65 to +150°C
Temperature Ambient Under Bias — T_C ..... 0 to +70°C
Supply Voltage to Ground
Potential Continuous ..... −0.5 to +7.0 V

Stresses above those listed under ABSOLUTE MAXIMUM RATINGS may cause permanent device failure. Functionality at or above these limits is not implied. Exposure to absolute maximum ratings for extended periods may affect device reliability.

OPERATING RANGES

Commercial (C) Devices
Ambient Temperature (T_A) ..... 0 to +70°C
Supply Voltage (V_{CC}) ..... 5.0 V ±5%

Operating ranges define those limits between which the functionality of the device is guaranteed.

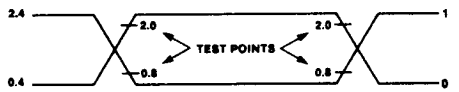
DC CHARACTERISTICS over operating range

Parameter Symbol	Parameter Description	Test Conditions	Min.	Max.	Unit
V_{IL}	Input LOW Voltage		−0.5	0.8	V
V_{IH}	Input HIGH Voltage		2.0	V_{CC} + 0.5	V
V_{OL}	Output LOW Voltage	I_{OL} = 3.2 mA		0.4	V
V_{OH}	Output HIGH Voltage	I_{OH} = 400 µA	2.4		V
I_{CC}	Power Supply Current (Note 1)	T_A = 0°C		600	mA
I_{LL}	Input Leakage Current	0 V ≤ V_{IN} ≤ V_{CC}		±10	µA
I_{LO}	Output Leakage Current	0.45 V ≤ V_{OUT} ≤ V_{CC}		±10	µA
V_{CL}	Clock Input LOW Voltage		−0.5	+0.8	V

Capacity of all inputs and outputs: 10 pF (Fc = 1 MHz). This parameter is not tested in every device.

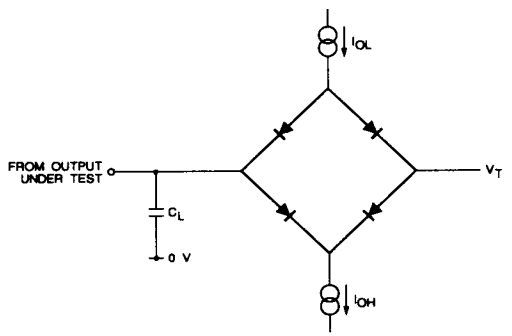
Notes: 1. I_{CC} Typical (T_A = 25°C) = 500 mA
I_{CC} Typical (T_A = 70°C) = 450 mA

SWITCHING TEST WAVEFORM



WF009542

SWITCHING TEST CIRCUIT



TC003862

$$V_T = \frac{V_{OL} + V_{OH}}{2}$$

SWITCHING CHARACTERISTICS over operating range
Timing Requirements

# Parameter	Description	Test Conditions	3 MHz Min.	3 MHz Max.	5 MHz Min.	5 MHz Max.	8 MHz Min.	8 MHz Max.	Unit
1 TCLCL	CLK Cycle Period	From 0.8 V to 0.8 V	330	1000	200	1000	125	1000	ns
2 TCHCL	CLK HIGH Time	From 2.0 V to 2.0 V	130		85		55		ns
3 TCLCH	CLK LOW Time	From 0.8 V to 0.8 V	130		85		55		ns
4 TWHRL	READY Hold Time after WRITE (Note 5)	From 0.8 V to 2.0 V		65		65		60	ns
5 TRDHRL	READY Hold Time after READ (Note 5)	From 0.8 V to 2.0 V		65		65		60	ns
6 TRHVH	Power Supply HIGH to RESET LOW Time		4TCLCL		4TCLCL		4TCLCL		ns
7 TRHRL	RESET HIGH TIME		4TCLCL		4TCLCL		4TCLCL		ns
8 TRLSL	RESET LOW to First $\overline{CS}$		2TCLCL		2TCLCL		2TCLCL		ns
9 THAHCH	HLDA RE Set-up Time (Note 1)		50		30		25		ns
10 THALCH	HLDA FE Set-up Time (Note 1)		50		30		25		ns
11 TAVCV	Address Valid to Control Active		20		20		20		ns
12 TSLRDL	$\overline{CS}$ LOW to $\overline{RD}$ LOW		(Note 6)		(Note 6)		(Note 6)		ns
13 TRDHSH	$\overline{RD}$ HIGH to Address Change		20		20		20		ns
14 TSLWRL	$\overline{CS}$ LOW to $\overline{WR}$ LOW		(Note 6)		(Note 6)		(Note 6)		ns
15 TDVRYH	Data Valid to READY RE (Note 2)		30		30		25		ns
15a TDVWH	Data Valid to $\overline{WR}$ HIGH		30		30		25		ns
16 TWRHDV	DATA Hold Time		20		20		20		ns
17 TWRHSH	$\overline{WR}$ HIGH to Address Change		20		20		20		ns
18 TRYLCH	READY FE Set-up Time		30		20		20		ns
19 TCHDXY	READY Hold Time		30		20		20		ns
20 TRYHCH	READY Active Set-Up Time		30		20		20		ns
21 TDVCH	DATA IN Set-Up Time		45		35		30		ns
22 TRDHDX	READ HIGH to Data not Valid		0		0		0		ns

Note: Switching characteristics are targetted numbers and are subject to change without notice.
See notes following table on page 47.

SWITCHING CHARACTERISTICS over operating range

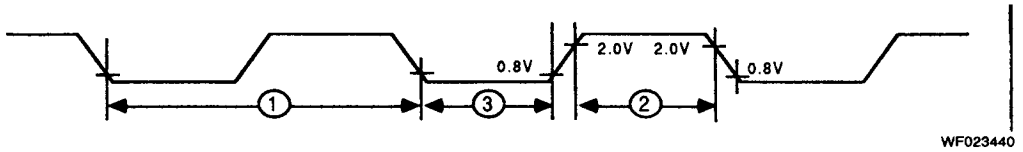
Timing Responses

# Parameter	Description	Test Conditions	3 MHz Min.	3 MHz Max.	5 MHz Min.	5 MHz Max.	8 MHz Min.	8 MHz Max.	Unit
24 TCLHRH	HRQ Active Delay	CL = 20 – 100 PF for all Am7971A Outputs (In addition to Am7971A selfload)		120		80		50	ns
25 TCLHRL	HRQ Inactive Delay			120		85		50	ns
26 TSLRYL	READY Active Delay			80		80		50	ns
27 TCLAV	Address Valid Delay			110		80		55	ns
28 TCLLH	ALE/DALE Active Delay			80		65		50	ns
29 TLHLL	ALE/DALE Width		TCLCH – 40		TCLCH – 20		TCLCH – 20		ns
30 TAVAL	Address Valid to ALE LOW		65		55		25		ns
31 TCHLL	ALE/DALE Inactive Delay			80		65		50	ns
32 TLLAX	Address Hold Time to ALE/DALE Inactive		70		50		40		ns
33 TCLAZ	Address Float Delay From Clock FE			70		50		40	ns
34 TCLAX	Address Hold Time		0		0		0		ns
35 TAZRL	Address Float to $\overline{RD}$/$\overline{DRD}$ Active		(Note 3)		(Note 3)		(Note 3)		ns
36 TCHRL	$\overline{RD}$/$\overline{DRD}$ Active Delay from Clock RE			60		50		40	ns
37 TRLRH	$\overline{RD}$/$\overline{DRD}$ Width		TCLCL – 50		TCLCL – 40		TCLCL – 30		ns
38 TCHRH	$\overline{RD}$/$\overline{DRD}$ Inactive Delay from Clock RE			60		50		40	ns
39 TRHAV	$\overline{RD}$/$\overline{DRD}$ Inactive to Next Address Active		TCHCL		TCHCL		TCHCL		ns
40 TCLDX	DATA Hold Time		0		0		0		ns
41									ns
42 TCLDV	DATA Valid Delay From Clock FE			90		70		70	ns
43 TCHWL	$\overline{WR}$/$\overline{DWR}$ Active Delay from Clock RE			60		50		40	ns
44 TWLWH	$\overline{WR}$/$\overline{DWR}$ Width		TCLCL – 50		TCLCL – 40		TCLCL – 30		ns
45 TCHWH	$\overline{WR}$/$\overline{DWR}$ Inactive Delay from Clock RE			60		50		40	ns
46 TWHDX	DATA Hold Time After $\overline{WR}$/$\overline{DWR}$		90		60		50		ns
47 TCADT3 a b	Control Active Delay from Float Control Inactive Delay to Float			65		65		55	ns
48 TCLRH	Clock to RFA HIGH	From 0.8 V to 2.0 V		75		65		55	ns
49 TCLRL	Clock to RFA LOW	From 2.0 V to 0.8 V		75		65		55	ns
50 TRYW	READY Width (Note 4)		2TCLCL – 75		2TCLCL – 75		2TCLCL – 55		ns
51 TCSH	$\overline{CS}$ Hold Time		0		0		0		ns
52 TRDHDV	DATA FLOAT Time			50		50		50	ns
53 THRLHR	HREQ LOW To HRQ HIGH		2TCLCL		2TCLCL		2TCLCL		ns

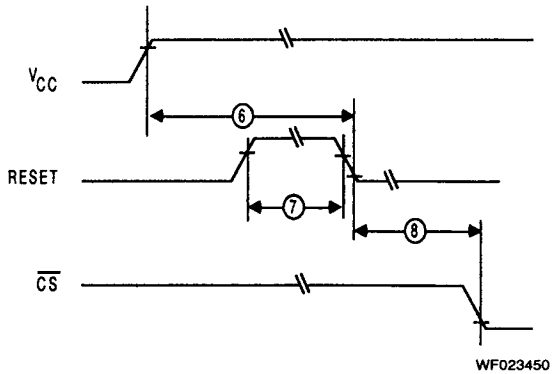
Switching characteristics are targetted numbers and are subject to change without notice.

Note:
1. HLDA is an asynchronous input. If THAHCH or THALCH are violated that only means that HLDA might be recognized one clock cycle later.
2. The rising edge of READY is synchronous with the falling edge of CLK. The delay from CLK is 65 ns max.
3. Min. can be computed from ③ – ㉝ max.
4. Maximum is :
   60 x TCLCL when CEP Busy (not tested)
   16 x TCLCL when CEP Not Busy (not tested)
5. If $\overline{CS}$ is HIGH, READY does not return to LOW after the slave access is completed.
6. The slave access is started when both $\overline{RD}$/$\overline{WR}$ and $\overline{CS}$ are asserted LOW. Thus, $\overline{CS}$ could actually be asserted later than $\overline{RD}$ or $\overline{WR}$. Under this condition, parameter ⑬ applies to $\overline{CS}$.

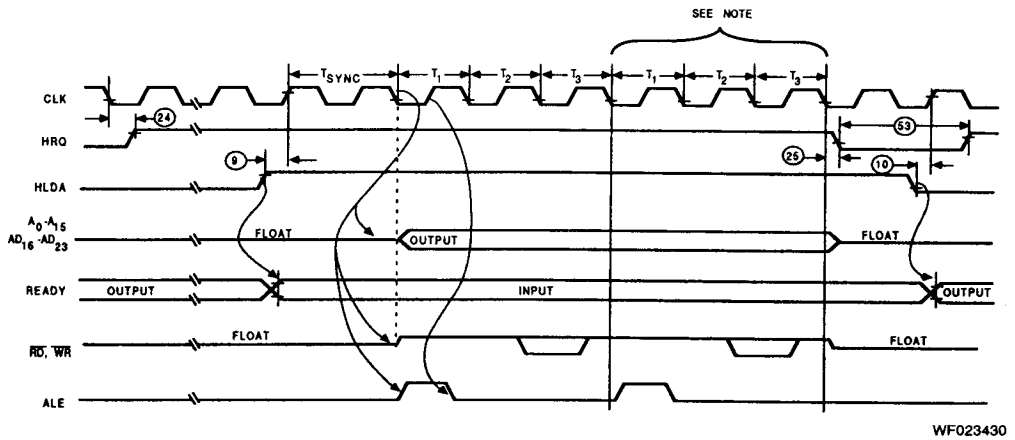
5



Timing Diagram 1. Clock Timing

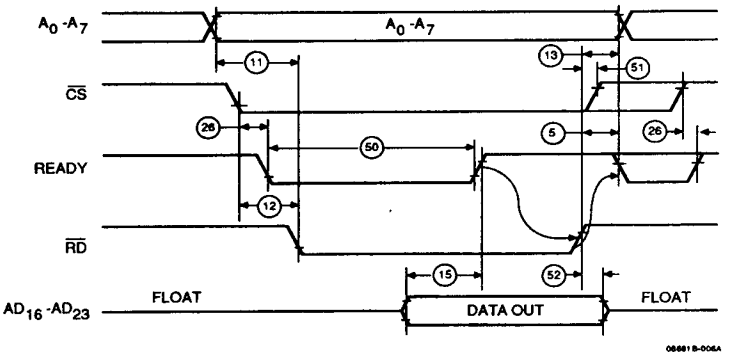


Timing Diagram 2. RESET Timing



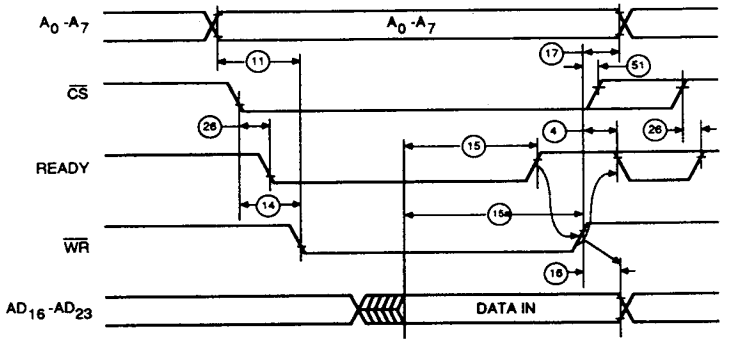
Timing Diagram 3. Bus Exchange Timing (System Interface)

Note: The second transfer cycle occurs only on Am7971A during expansion with Bit offset ≠ 0 with the destination buffer located on the system Bus at the end and beginning of an image line (READ/MODIFY/WRITE).



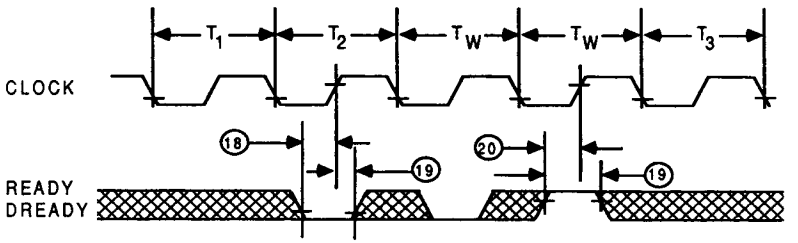
WF026440

Timing Diagram 4. CPU Program Read Timing



WF026450

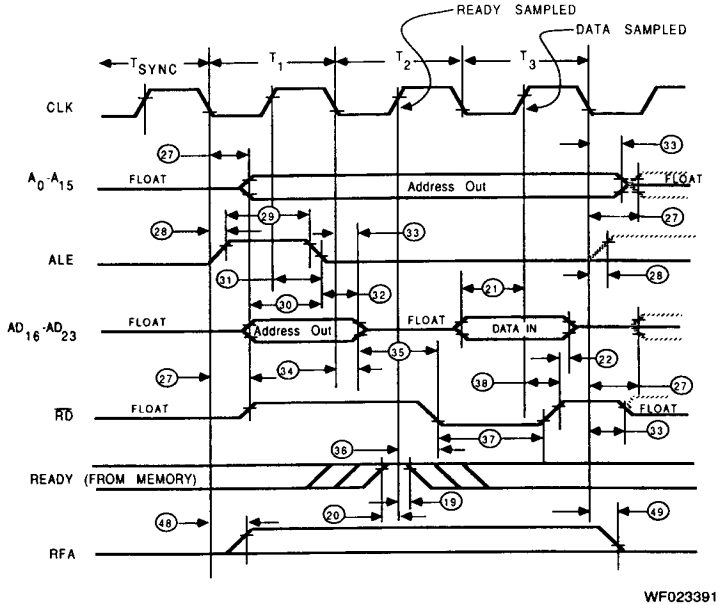
Timing Diagram 5. CPU Program Write Timing



WF023400

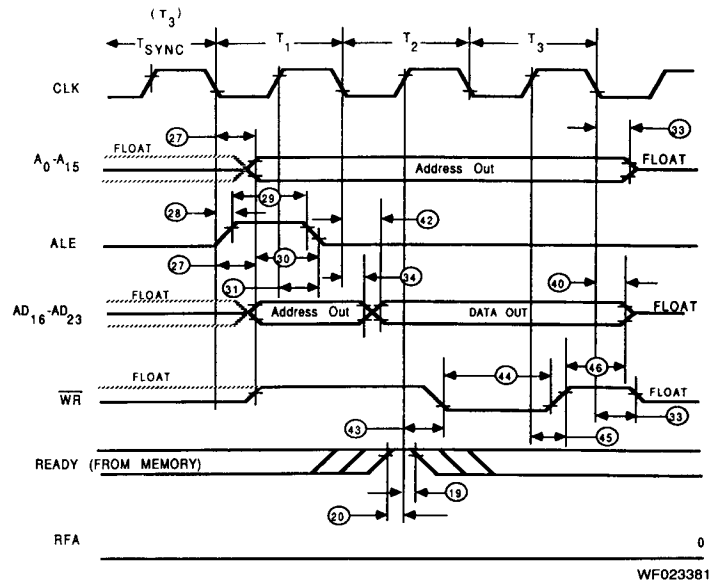
Timing Diagram 6. READY, DREADY Input Timing Master Mode, System and Document Interface

5



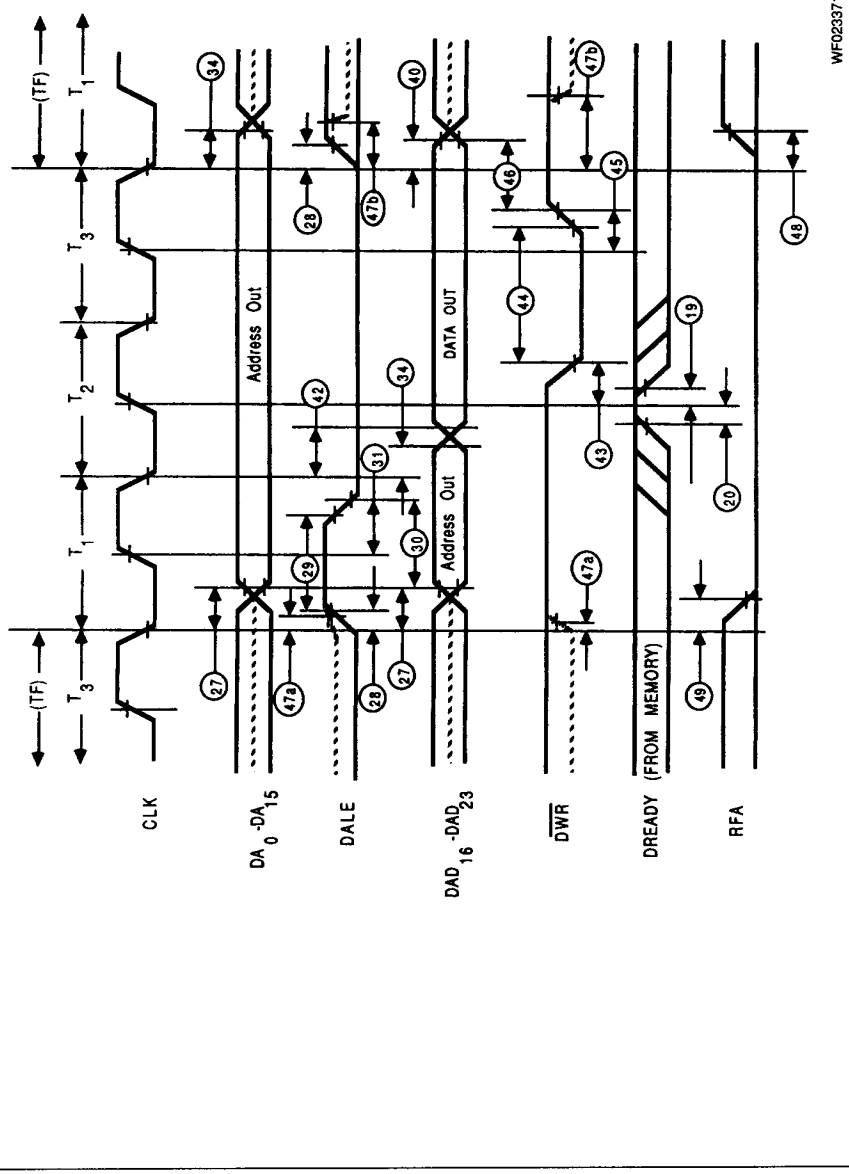
Timing Diagram 7. DMA Read Operation

Indicates additional access during Read-Modify-Write cycle if EBOCR ≠ 0



Timing Diagram 8. DMA Write Operation

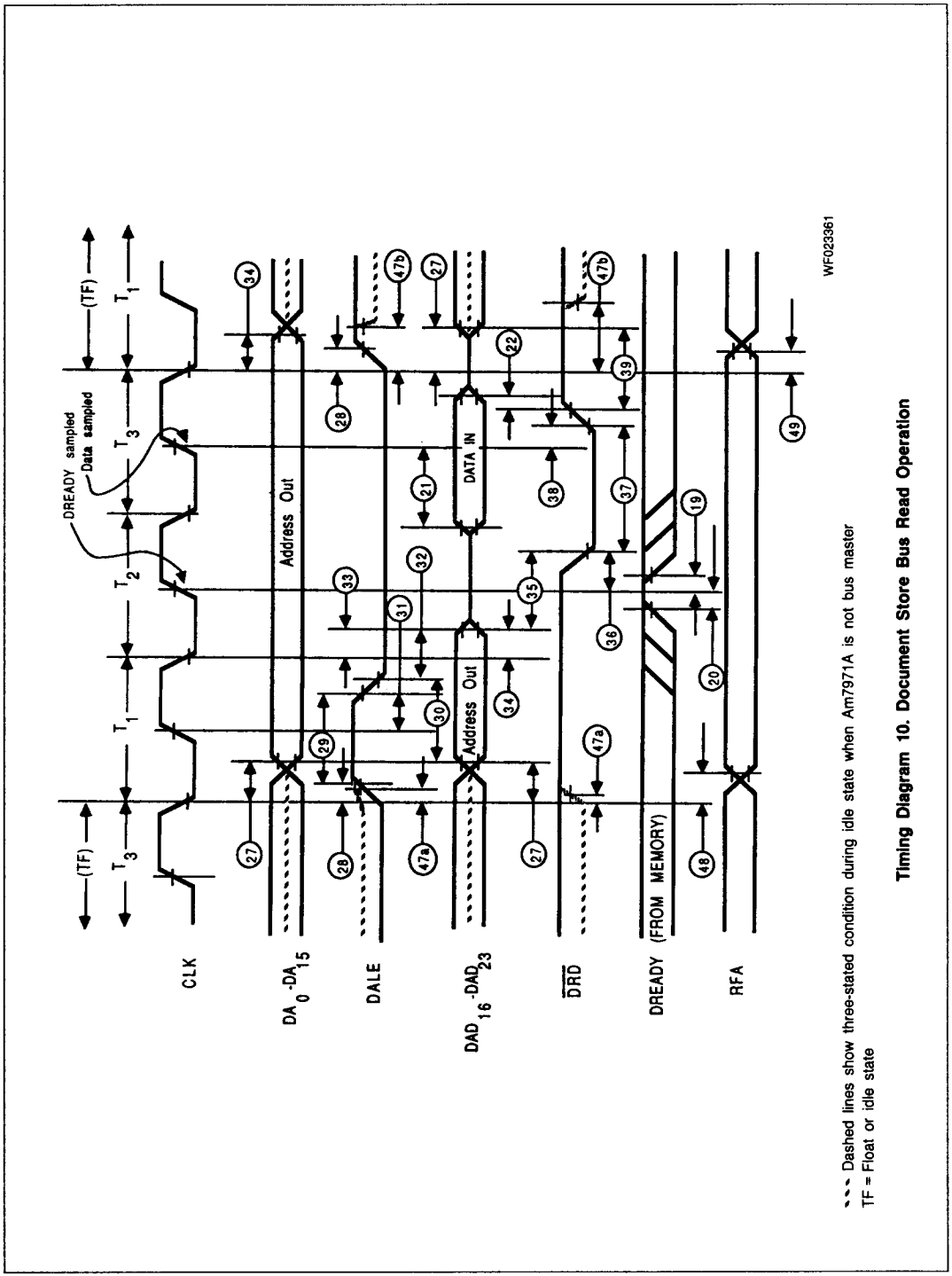
Indicates additional access during Read-Modify-Write cycle if EBOCR ≠ 0



---- Dashed lines show three-stated condition during idle state when Am7971A is not bus master
TF = Float or idle state

Timing Diagram 9. Document Store Bus Write Operation

5



~~~ Dashed lines show three-stated condition during idle state when Am7971A is not bus master
TF = Float or idle state

Timing Diagram 10. Document Store Bus Read Operation
~~~

Am7971A CEP ACRONYM LIST

Acronym	Name	Register
ALE	Address Latch Enable	
BBC	Byte Boundary Control bit	ERCR
CBOCR	Compressor Bit Offset Control Register	
CCITT	Int'l Telegraph and Telephone Consultative Committee	
CDAHR	Compressor Destination Address Holding Register	
CDCAR	Compressor Destination Current Address Register	
CDCHR	Compressor Destination Count Holding Register	
CDLSR	Compressor Destination Line Start Address Register	
CDO	Compressor Destination Overflow bit	CSR
CDWCR	Compressor Destination Working Count Register	
CER	Compressor Express Register	
CFWR	Compressor Frame Width Register	
CIC	Compressor Illegal Command bit	CSR
CIE	Compressor Interrupt Enable bit	CMCR
CKPR	Compressor K Parameter Register	
CMCR	Compressor Master Control Register	
COA	Compressor Busy and New Operation Attempted bit	CSR
CPR	Compressor Parameter Register	
CRCAR	Compressor Reference Current Address Register	
CRCR	Compressor Restart Control Register	
CS	Chip Select	
CSAHR	Compressor Source Address Holding Register	
CSCAR	Compressor Source Current Address Register	
CSCHR	Compressor Source Count Holding Register	
CSLSR	Compressor Source Line Start Address Register	
CSO	Compressor Source Overflow bit	CSR
CSR	Compressor Status Register	
CSWCR	Compressor Source Working Count Register	
CWR	Compressor Wraparound Register	
DAC	Destination Address Control bit	CRCR, ERCR
DALE	Destination Address Latch Enable	
DC	Destination Control bit	CMCR, EMCR
DCC	Destination Count Control bit	CRCR, ERCR
DER	Data Error bit	ESR
DFC	Data Format Control bits	CPR
DLS	Destination Line Start Address Control bit	CRCR, ERCR
DMA	Direct Memory Access	
DRD	Document Store Read	
DREADY	Document Store Ready	
DWR	Document Store Write	
EBOCR	Expander Bit Offset Control Register	
EBY	Expander Busy bit	MSR, ESR
ECD	Extension Code Detected bit	MSR
EDAHR	Expander Destination Address Holding Register	
EDCAR	Expander Desintation Current Address Register	
EDCHR	Expander Destination Count Holding Register	
EDLSR	Expander Destination Line Start Address Register	
EDO	Expander Destination Overflow bit	ESR
EDWCR	Expander Destination Working Count Register	
EFWR	Expander Frame Width Register	
EIC	Expander Illegal Command bit	ESR
EIE	Expander Interrupt Enable bit	EMCR
EKPR	Expander K-Parameter Register	
EMCR	Expander Master Control Register	
EOA	Expander Busy and New Operation Attempted bit	ESR
EOL	End-of-line bit	CPR, EPR

EOP	End-of-page (Group IV)	MSR
EPR	Expander Parameter Register	
EPWR	Expander Page Width Register	
ERCAR	Expander Reference Current Address Register	
ERCR	Expander Restart Control Register	
ESA	Expander Source Address bit	EPR
ESAHR	Expander Source Address Holding Register	
ESCAR	Expander Source Current Address Register	
ESCHR	Expander Source Count Holding Register	
ESLSR	Expander Source Line Start Register	
ESO	Expander Source Overflow bit	ESR
ESR	Expander Status Register	
ESWCR	Expander Source Working Register	
EWR	Expander Wraparound Register	
EXT	Extention bits	MSR
GO	Go	CMCR, EMCR
GP	G-Parameter bits	EPR
HLDA	Hold Acknowledge	
HRQ	Hold Request	
ID	Identification bit (Am7970A = 1; Am7971A = 0)	MSR
INTR	Interrupt Request	
LBO	Left Bit Offset Control bits	CBOCR, EBOCR
LMGR	Left Margin Register	
LPI	Line Processing Incomplete bit	CSR, ESR
MC	Mode Control bits	CMCR, EMCR
MH	Modified Huffman (coding)	
MMR	Modified MR (Group IV coding)	
MR	Modified READ (Group III coding)	
MSR	Master Status Register	
NGC	Negative Compression bit	CSR
OC	Operation Control bits	CMCR, EMCR
PEL	Picture Element	
PIXEL	Picture Element	
RBO	Right Bit Offset Control bits	CBOCR, EBOCR
RD	Read	
READ	Relative Element Address (coding)	
READY	Ready	
RESET	Reset	
RFA	Reference Line Access	
RMGR	Right Margin Register	
RTC	Return-to-Control code (six EOLs)	
SA	Source Attribution bit	CPR, EPR
SAC	Source Address Control bit	CRCR, ERCR
SC	Source Control bit	CMCR, EMCR
SCC	Source Count Control bit	CRCR, ERCR
SLS	Source Line Start Address Control bit	CRCR, ERCR
TFLR	Time Fill Register	
TMGR	Top Margin Register	
WPI	Wrap-around Incomplete bit	CSR, ESR
WR	Write	