intel®

APPLICATION NOTE

AP-259

November 1985

The 82786 CHMOS Graphics Coprocessor Architectural Overview

Order Number: 122711-001

Intel Corporation makes no warranty for the use of its products and assumes no responsibility for any errors which may appear in this document nor does it make a commitment to update the information contained herein.

Intel retains the right to make changes to these specifications at any time, without notice.

Contact your local sales office to obtain the latest specifications before placing your order.

The following are trademarks of Intel Corporation and may only be used to identify Intel Products:

Above, BITBUS, COMMputer, CREDIT, Data Pipeline, GENIUS, i, $\hat{i}$, ICE, iCEL, iCS, iDBP, iDIS, I^2ICE, iLBX, i_m, iMDDX, iMMX, Insite, Intel, int_el, int_eIBOS, Intelevision, $int_eligent$ Identifier, $int_eligent$ Programming, Intellec, Intellink, iOSP, iPDS, iPSC, iRMX, iSBC, iSBX, iSDM, iSXM, KEPROM, Library Manager, MCS, Megachassis, MICROMAINFRAME, MULTIBUS, MULTICHANNEL, MULTIMODULE, ONCE, OpenNET, Plug-A-Bubble, PROMPT, Promware, QUEST, QueX, Ripplemode, RMX/80, RUPI, Seamless, SLD, UPI, and VLSiCEL, and the combination of ICE, iCS, iRMX, iSBC, iSBX, iSXM, MCS, or UPI and a numerical suffix.

MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.

*MULTIBUS is a patented Intel bus.

Additional copies of this manual or other Intel literature may be obtained from:

Intel Corporation
Literature Distribution
Mail Stop SC6-714
3065 Bowers Avenue
Santa Clara, CA 95051

PREFACE

82786 FEATURES AND PERFORMANCE

The 82786 is a powerful, yet flexible component which will be a candidate as a standard for microcomputer graphics applications including personal computers, engineering workstations, terminals, and laser printers. Its advanced software interface contrasts sharply with existing products by making applications and systems level programming efficient and straight-forward. Its performance and high-integration make it a cost-effective component while improving the performance of nearly any design.

The following list is a summary of the 82786's capabilities (assuming 10 MHz system clock and 25 MHz video clock):

Windows:	Practically unlimited support
Colors:	Up to 1024 displayable simutaneously with support for 4 external color palettes
Lines, Polylines, Polygons:	2.5 Million pixels per second
Circles, Arcs:	2.0 Million pixels per second
Fills:	Supported via horizontal line command (30 Million bits per second)
Bit Block Transfer:	24 Million bits per second
Bit-map Memory:	Up to 4 MBytes of directly accessed DRAM
Resolution:	Up to 200 MHz monitors supported; this is equivalent to configurations such as 640 x 480 x 8 or 1024 x 1024 x 2 @ 60 Hz (non-interlaced); up to 4096 x 4096 x 1 or 2048 x 1536 x 8 with video DRAMs.
Zoom:	1 to 64 times vertical and horizontal
Character Drawing:	25 thousand per second with colors, path, and rotation attributes
Character Fonts:	Unlimited number from bit-map or system memory
Character Size:	16 x 16 maximum hardward size; unlimited with bit-block transfer
Scroll, Pan:	Instantaneous in any direction with no external logic

The performance of the 82786 is of little value without applications and system-level software to use it. Customers can write their own software following the suggestions of the 82786 Software Interface Applications Note or the appropriate third-party vendors' software packages. Intel has evaluated several major products and presently recommends Microsoft Windows™, Digital Research GEM™, Novagraphics Nova CGI and GKS™, and Graphic Software Systems CGI and GKS™, Window Manager™, and GKS™. These packages appear to be easily adapted to 82786-based systems, are likely to emerge as de facto industry standards, and would permit a wide array of applications to run with little or no modification on 82786-based products.

For more information on these products, please contact these vendors directly:

Digital Research, Inc.
P. O. Box DRI
Monterey, CA 93942
(408) 649-3896

Graphic Software Systems
P. O. Box 673
Wilsonville, OR 97070
(503) 682-1606

Microsoft Corporation
Box 97200
Bellevue, WA 98009
(206) 828-8080

Novagraphics International Corporation
1015 Bee Cave Woods
Austin, TX 78746
(512) 327-9300

The 82786 was designed to permit compatibility with de facto hardware standards. Use of the 82786 with appropriate Intel microprocessors permits the design of systems which can emulate the family of IBM™ personal computer products. The 82786's support of the IBM Color Graphics Adapter-compatible bit-map eases the task of running existing applications software on new video hardware.

For details please refer to the 82786 PC Compatibility Applications Note. Additional documentation available for the 82786 includes the Data Sheet, the User Manual and Application Notes.

For all questions, clarifications, or requests for additional documentation please contact your local Intel sales office or authorized distributor.

THE 82786 CHMOS GRAPHICS COPROCESSOR ARCHITECTURAL OVERVIEW

CONTENTS

PAGE

CHAPTER
CHAPTER 1 1
INTRODUCTION 1
1.1 Overview 1
1.2 Architectural Model 1
1.3 Bit Maps and Windows 2
1.4 Functional Overview 3

CHAPTER 2 5
GRAPHICS PROCESSOR 5
2.1 Overview 5
2.2 Bit Maps 5
2.3 Graphics Processor Instruction Set 5
2.3.1 Non-Drawing Commands 5
2.3.2 Drawing Control Commands 5
2.3.3 Geometric Commands 6
2.3.4 Bit Block Transfer Commands 6
2.3.5 Character Command 6
2.4 Drawing Attributes 6
2.5 Clipping 7
2.6 Logical Operation 7
2.7 Character Fonts 7
2.8 Graphics Processor Command List Format 8

CHAPTER 3 11
DISPLAY PROCESSOR 11
3.1 Overview 11
3.2 Windows 11
3.3 Cursor 13
3.4 Zoom 13
3.5 Video Interface 13

CHAPTER 4 15
82786 SYSTEMS 15
4.1 Typical System Configurations 15
4.2 DRAM Control 15
4.3 Bus Interface 16

CHAPTER 5 17
PIN DESCRIPTION AND PACKAGE 17
5.1 Overview 17

CHAPTER 1
INTRODUCTION

1.1 OVERVIEW

This document provides the reader with an introduction to the architecture and key features of the Intel 82786 Graphics Coprocessor from Intel. The 82786 serves such applications as graphics terminals and work stations, personal computers, printers, and other products requiring the capability to create, store, and output bit-map graphics.

The 82786 works with all Intel microprocessors, and is a high-performance replacement for sub-systems and boards which have traditionally used discrete components and/or software for graphics functions. The 82786 requires minimal support circuitry for most system configurations, and thus reduces the cost and board space requirements of many applications. The 82786 is based on Intel's advanced CHMOS process.

The advanced performance and ease-of-use of the 82786 make it a candidate for an industry standard for applications in microcomputer graphics markets. Some of the leading features of the 82786 are:

- Fast polygon and line drawing
- Hardware windows
- High speed character drawing
- Interface designed for device independent software standards
  - — Virtual Device Interface
  - — Graphics Kernal System
  - — NAPLPS
- Advanced DRAM controller for graphics memory up to 4 Mbytes
- Fast bit-block copies between system and bit-map memories
- Supports up to 200 MHz CRTs or higher
- Up to 1024 simultaneous colors per frame
- Programmable video timing
- High Integration
- Third-party software support
- 88 pin leaded chip carrier and pin grid array
- Provides support for rapid filling with patterns
- IBM Personal Computer Color Graphics Adapter-compatible bit-map
- International character support
- Advanced CHMOS technology
- Integral video DRAM support

1.2 ARCHITECTURAL MODEL

The 82786 architecture fits with traditional computer graphics models. A typical subdivision of the tasks is:

- Graphics task partitioned into:

  Drawing (line, polygons, characters, block image copies)

  Windowing (concurrent windows on the screen)

  Refresh (CRT timing, video data output)
- Typical integrated solutions to these functions have been:

  First generation IC: 6845, 8275 - refresh

  Second generation LSI: 82720 - drawing + refresh

  Third generation VLSI: 82786 - drawing + windowing + refresh

The 82786 is a co-processor with two separate on-chip processing units, the graphics processor and display processor, which operate concurrently with the system CPU. Instructions to the display and graphics processors are placed in memory by the CPU. Registers on the 82786 are dedicated to pointing to the starting addresses of the first memory blocks of instructions controlling the on-chip processors, and each memory block points to subsequent blocks in a linked-list architecture. Access by the CPU to these registers may be I/O- or memory-mapped, and portions of memory may be shared between the 82786 and the CPU.

1.3 BIT MAPS AND WINDOWS

The 82786 concepts of "bit maps" and "windows" are based upon definitions from the ANSI work on windows.

The 82786 can create and maintain multiple sets of graphics images in memory. These sets of images in memory are called "bit maps". 82786 can combine subsets of these bit-maps into a viewable, multi-region display screen. Each of these separate areas on the screen are called "windows".

Most graphics systems today use software to generate a bit-map representation of the full contents of the display called a "frame buffer". The 82786 uses a high-level window descriptor list and specialized hardware to generate the screen contents using portions from separate bit maps of memory (Figure 1-1). This permits the display to be instantaneously altered, eliminating the time required to update a similar frame buffer image using software alone.

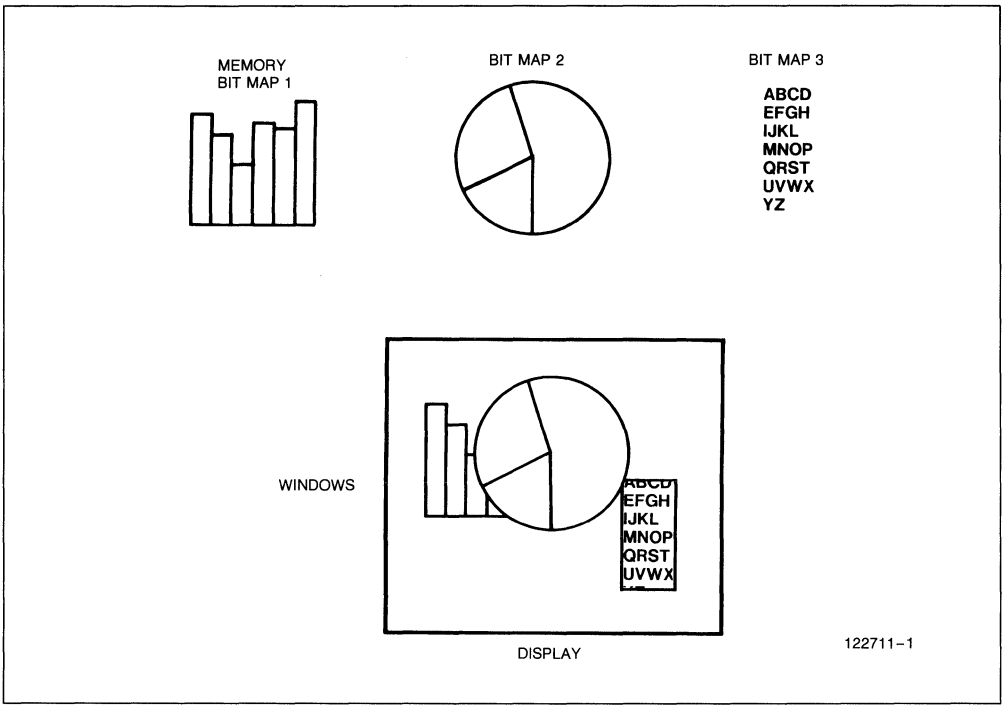

Figure 1-1. Bit Maps and Windows

1.4 FUNCTIONAL OVERVIEW

The 82786 performs many functions within a single integrated circuit. Figure 1-2 identifies a block diagram of the component and explanations of each function module.

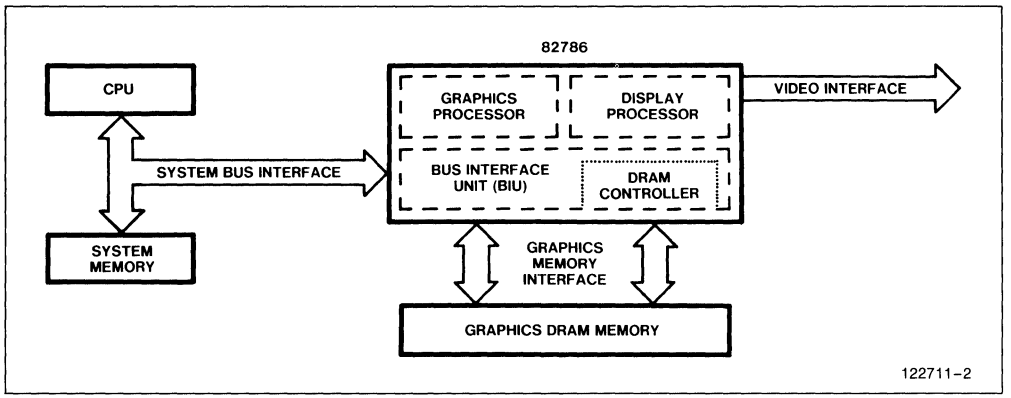

Figure 1-2. 82786 Block Diagram

The major functions of each block are:

- Graphics Processor (GP): — draws lines, circles, polygons, and other primitives
  — draws characters
  — executes block image manipulation instructions
- Display Processor (DP): — manages windows including zoom
  — provides cursor
  — refreshes screen (up to 200 MHz dot rate)
  — loads shift register of video DRAMs
- DRAM Controller: — controls up to 4 Mbytes of interleaved graphics memory including page-, static column-, and fast page-mode DRAMs (interleaved or non-interleaved banks)
- BIU: — allows the CPU to access the graphics memory and the 82786 to access the system memory

CHAPTER 2
GRAPHICS PROCESSOR

2.1 OVERVIEW

The graphics processor creates and updates all of the graphics and text in each of the bit maps within graphics memory. It is responsible for all of the geometric drawing, character drawing and image movement within and between the bit maps. Some features of the graphics processor are:

- permits bit maps to begin at any word in system or graphics memory; only one bit map is active for GP drawing at one time although many bit maps may reside in memory simultaneously.
- permits bit maps to be any size (up to 32K x 32K pixels) and use 2, 4, 16, or 256 colors (i.e. 1, 2, 4, or 8 bits per pixel);
- draws geometric shapes with attributes such as texture and color, into bit maps;
- draws characters with attributes such as color, path, rotation, and proportional spacing using user-defined fonts into bit maps;
- combines one rectangular portion of an image with another area, within the same bit map or into another bit map. (BIT Block Transfer or Bit-Blt);
- all drawing allows logical operations between source and destination (for example Exclusive-Or of the Complement of Source with Destination);
- all drawing can be clipped to a rectangular region;
- supports picking, a mechanism for advanced user interfaces which allows the issuing commands via the selection of "graphic menus" (called icons) by manipulating pointing devices.

The Graphics Processor fetches its instructions directly from a linked list in memory which is created and updated by the CPU. The initial address of the list is contained in a dedicated register in the 82786 and the addresses of subsequent instructions are pointed to by the contents of previous instructions. Each instruction contains a bit which indicates to the graphics processor that it should stop (if set) and await new instructions. More detail on the command format is given in section 2.8 "Graphics Processor Command List Format."

2.2 BIT MAPS

All graphics and text creation is written into bit maps. Bit maps are rectangular drawing area composed of bits of pixel-oriented memory. The bit maps may be up to 32,000 pixels in each direction and contain from one to eight bits of color or gray scale information. Bit maps may be started on any even address in the 4 Mbyte space and the number of bit maps in memory is unlimited (except by the amount of memory available). The variable bits per pixel feature permits the use of several bits per pixel for multicolor graphics while using only a single bit per pixel for efficient text memory.

2.3 GRAPHICS PROCESSOR INSTRUCTION SET

The graphics processor instruction set is divided into five classes:

1. Non-Drawing Commands
2. Drawing Control Commands
3. Geometric Commands
4. Bit Block Transfer (BIT-BLT) Commands
5. Character Block Transfer (CHA-BLT) Commands

2.3.1 Non-Drawing Commands

The first class of commands are used to control the method in which the commands are fetched. Also included in this list are commands to load and dump 82786 internal registers. These commands are:

- NOP - No Operation
- LINK - Link To Next Command (Unconditional Jump)
- ENTER_MACRO - Enter Macro (Subroutine Call)
- EXIT_MACRO - Exit Macro (Subroutine Return)
- INTR_GEN - Generate Interrupt
- DUMP_REG - Dump Internal Register
- LOAD_REG - Load Internal Register

2.3.2 Drawing Control Commands

The graphics processor works in only one bit map and with one set of attributes at a time. The graphics processor maintains an imaginary cursor, GCPP (Graphics Current Position Pointer), which points to a particular position (x, y coordinates) within the bit map from which all relative coordinates are calculated. The GCPP is updated at the end of each drawing command.

The following commands are used to define the current bit map and attributes and set the Current Position Pointer:

- DEF_BIT_MAP - Define Bit Map
- DEF_CLIP_RECT - Define Clip Rectangle (see 2.4)
- DEF_COLORS - Define Colors
- DEF_TEXTURE - Define Texture
- DEF_LOGICAL_OP - Define Logical Operation (see 2.6)
- DEF_CHAR_SET - Define Character Set
- DEF_CHAR_ORIENT - Define Character Orientation
- DEF_CHAR_SPACE - Define Inter Character Spacing
- ABS_MOV - Absolute Move GCPP
- REL_MOV - Relative Move GCPP
- ENTER_PICK - Enter Pick Mode
- EXIT_PICK - Exit Pick Mode

2.3.3 Geometric Commands

These commands allow the 82786 to draw points, lines, and arcs in a variety of ways:

- POINT - Draw Point
- INCR_POINT - Draw Incremental Points
- CIRCLE - Draw Circle
- LINE - Draw Line
- RECT - Draw Rectangle
- POLYLINE - Draw Polyline
- POLYGON - Draw Polygon
- ARC - Draw Arc
- SCAN_LINES - Draw Series of Horizontal Lines

2.3.4 Bit Block Transfer (Bit-Blt) Commands

These commands allow rectangular image pieces to be combined from piece of bit-map memory to another. The graphics processor automatically inserts the new data in the correct order in the destination so that each line of pixels remains consecutive for both existing and new data.

- BIT_BLT - Bit Block Transfer within bit map
- BIT_BLT_M - Bit Block Transfer between bit maps

The command specifies the origin of the source rectangle as well as the height and width. The destination origin is the GCPP coordinates. For Bit-Blts between bit maps, the destination is the active bit map and the memory address of the source origin and source bit map size is specified. Bit-Blts between bit maps can only use bit maps with the same number of bits per pixel.

2.3.5 Character Command

This command allows character fonts stored in memory in pixel form to be drawn into the bit map by an application using character codes such as ASCII:

- CHAR - Draw Character String

The CHAR command defines transparency/opaqueness for a character string, the pointer for the character string, and the number of character in the string. The pixel contents of the character to be drawn may be located anywhere in the memory space of the 82786 and accessed with either an 8- or 16-bit reference to the specific character. The string range specifies the 8- or 16-bit references for each character to be drawn. Section 2.7 discusses the use of character fonts.

Standard character fonts can be flexibly drawn because path and rotation are defined with a DEF_CHAR_ORIENT command and inter-character spacing is defined with a DEF_CHAR_SPACE command. This permits the variable spacing of text, direction of text, and rotation of characters to be specified by the application without making alteration of the font necessary. Simple one-bit per pixel character font definitions can be used in color applications because foreground and background colors are specified by the DEF_COLOR command and the necessary bits are written for each pixel during the drawing process.

2.4 DRAWING ATTRIBUTES

A drawing operation refers to the act of modifying pixels within a bit map during the execution of the GP commands. All drawing that the GP performs (including lines, arcs, characters and Bit-Blts) is subject (with exceptions noted) to six attributes which should be defined before any drawing commands are executed. The attributes are:

1. Pixel Plane Mask;
2. Logical Operation;
3. Clipping Rectangle;
4. Foreground and Background color (not applicable to Bit-Blt);
5. Transparent or Opaque mode (not applicable to Bit-Blt);
6. Pattern mask of 16 bits (not applicable to Bit-Blt or characters).

The pixel plane mask is helpful in restricting the graphics primitives to update a subset of the bits per pixel.

This permits one set of drawings to exist in one or more colors and allow other text or graphics information to reside in different color bits of the same bit map. Raster operations can be used to combine existing pixel information in the bit map with the new pixel information generated as a result of the new drawing operation, such as displaying only the overlapping regions of two shapes. The clipping rectangle limits the effects of drawing operations to a subset of the bit map.

Foreground and background colors set the two colors drawn by all drawing operations (if both are needed). The transparent mode draws only the foreground color into the bit map (for dotted lines or characters) and leaves the pixels between the dots or characters unchanged. The opaque mode draws the foreground color and fills in the background color between the dots or characters. The pattern defined in the mask cause a logical operation with drawing commands and permit dotted and dashed lines, arcs, and other shapes. DEF__PATTERN sets transparent/ opaque for drawing operations other than character, which is defined in CHAR.

2.5 CLIPPING

The clipping rectangle is used to prevent drawing outside a specified rectangular region. The clipping rectangle can be any rectangle within a bit map or the entire bit map. Pixels are not drawn beyond the limits of the clipping rectangle and characters which would be partially clipped are not drawn at all.

In a special mode, "pick mode," the clipping rectangle is used to perform a different function. The clipping rectangle may be controlled by software to support the selection of objects on the display with a pointing device. When in pick mode the drawing commands are executed but pixels are not updated in memory. Instead, a flag is set in a register if any of the pixels generated by the command lie within the clipping rectangle. In this way it is easy to set the clipping rectangle to correspond to the location of a graphics pointing device (such as a mouse) and re-process the graphics command list to find which drawing command corresponds to the selected area.

2.6 LOGICAL OPERATION

The logical operation is an attribute that applies to all subsequent pixel update operations (line, arc, character, Bit-Blt etc.). It is an operation which can logically combine the contents of separate bit-map locations to produce new bit-map patterns. All sixteen binary functions are permitted between both the source and destination.

- AND
- OR
- EXCLUSIVE-OR

Six of the combinations provided are special:

- REPLACE destination with source
- REPLACE destination with complement of source
- SET all destination bits to 0
- SET all destination bits to 1
- REPLACE destination with complement of destination
- REPLACE destination with destination (NOP)

2.7 CHARACTER FONTS

The Graphics Processor supports an unlimited number of character fonts, that can reside anywhere in the 4 Megabyte address space. The character string to be written can be defined either as a string of bytes or as a string of words depending upon the type of font used. The active font type and upper and lower memory addresses of the font to be used are set via the DEF__CHAR__SET command.

Each character in the character font has an independently programmable size of up to 16 by 16 pixels, allowing individual characters to have different sizes for proportional spacing. Each character resides in a block containing n + 1 words of memory where n is the pixel height of the character. The first word contains fourteen bits to define the height and width of the character. The remaining two bits specify if the following character should be an overstrike or if the character exceeds sixteen pixels in either dimension to cause a software trap. Overstriking is useful for efficient implementation of underline and accents, and prevents updating the GCCP after the character is drawn.

For larger characters than 16 by 16, the trap bit in the font can cause an interrupt to the CPU so that software can specially process that character such as a Bit-Blt. The perception of larger characters than 16 by 16 can also be created by dividing characters into subsets such as quadrants, and executing multiple character drawing commands. Software use of the DEF__CHAR__SPACE command supports negative inter-character spacing to permit kerning, such as for italic fonts.

The byte or word strings used as parameters for the CHAR command are used in conjunction with the 22-bit pointer defined in a register by the DEF__CHAR__SET command. Use of 16-bit, or word-mode, characters causes an add between the 22-bit pointer and the 16-bit reference value to access the starting address of the specific character. Because maximum character block size is seventeen words of data, approximately four thousand characters may be contained in one 16-bit font (worst case). Supplementary software in the form of a look-up table can be used to access as many as 65,000 characters in a single font. Bit-Blt can move characters of unlimited size.

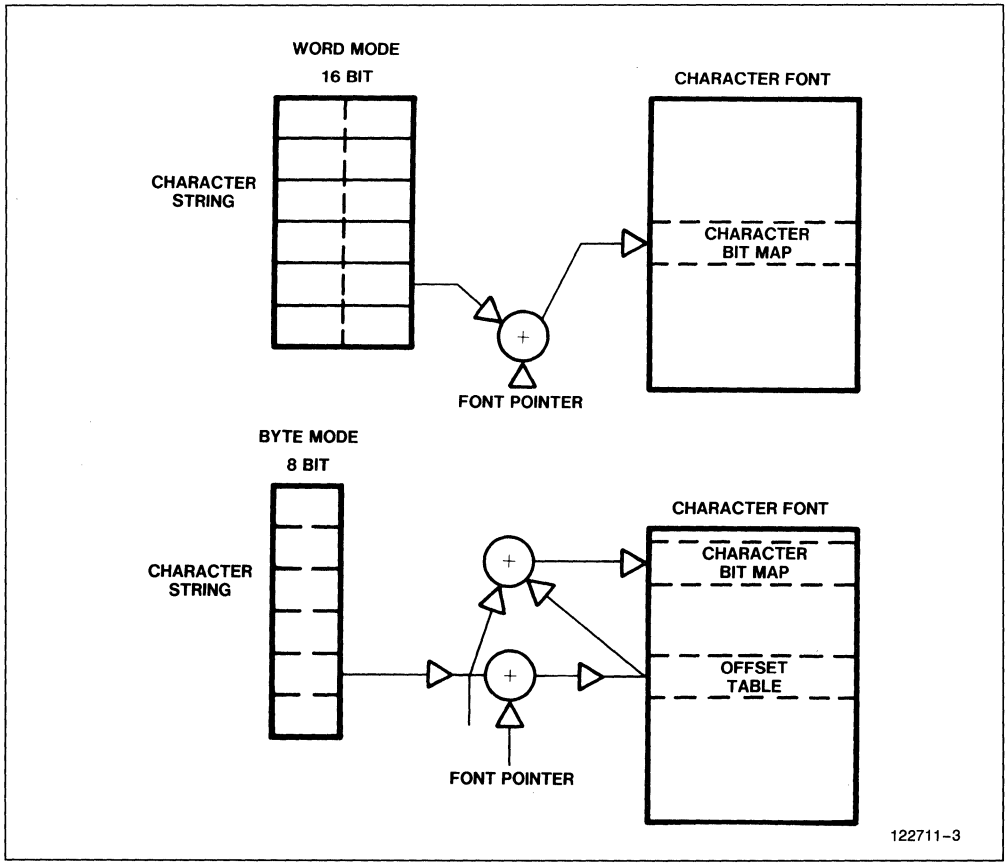

Figure 2-1. Word and Byte Mode

Use of byte-mode permits eight bit references to characters. This is important to permit existing software using ASCII and EBCDIC to be converted to 82786-based systems. 256 words of the font are reserved for a look-up table. Adding the 8-bit string parameter to the font pointer determines the word for the specific character within this table. The word is then added to the pointer to locate the character information in the font. Byte-mode permits only 256 characters in each 8-bit font. Figure 2-1 shows a description of word and byte mode.

2.8 GRAPHICS PROCESSOR COMMAND LIST FORMAT

The graphics processor executes a sequence of instructions resident in memory and runs only when an application needs to change bit-map contents or support some special function such as picking. The general format of an instruction is shown in Figure 2-2.

15 14 13 12 11 10 09 08 07 06 05 04 03 02 01	00
OPCODE	GECLbit
PARAMETER 1	
PARAMETER 2	
PARAMETER N	

Figure 2-2. Instruction Format

Each opcode resides in the high byte of the word with a GECL (Graphics End of Command List) bit in the least significant bit of the low byte and followed by a varying number of parameters in consecutive words. The graphics processor tests the GECL of each instruction and sends the graphics processor into Poll Mode when set to "1" for any opcode. Poll mode halts the graphics processor until a LINK command and upper- and lower-memory values for a link address are loaded into three reserved registers. The graphics processor then begins executing a new linked-list of instructions starting at the specified address when the GECL bit with the LINK instruction in the register is reset to 0.

An example of a graphics command block using linked-lists is shown in Figure 2-3.

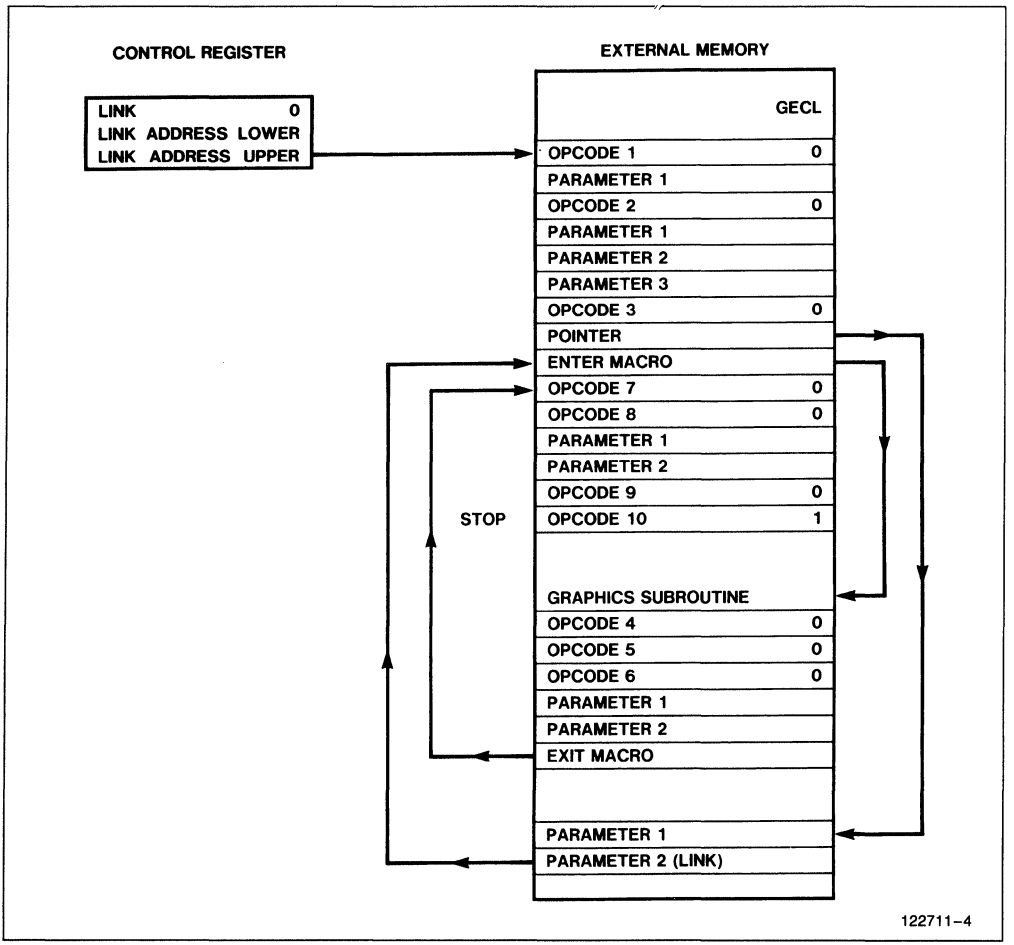

Figure 2-3. Graphics Processor Command Block

CHAPTER 3
DISPLAY PROCESSOR

3.1 OVERVIEW

The display processor has five main functions in generating the display contents for output:

1. To retrieve the memory contents of selected bit maps and output corresponding pixels into separate regions on the display screen (windows);
2. To permit selected portions of bit maps to be magnified on the display (zooming) horizontally and/or vertically via pixel replication;
3. To provide a "pointing symbol" (cursor);
4. To generate control and video data signals to the display hardware;
5. Load the shift registers of video DRAMs.

Control of the display processor is programmed via on-chip registers. Content of the display is dynamically altered by the application (or system software) without causing unacceptable display blinking. Using memory-mapped CPU alteration of parameters, the DP will load the register set with the new parameters during vertical retrace. By altering the registers to point to a new display list, the change of display lists can occur between refresh cycles.

3.2 WINDOWS

Windows are the portions of bit maps which are output by the display processor. Up to 16 window segments or tiles can be displayed on the same scan line of the CRT, while there may be as many windows vertically as the number of scan lines.

The 82786 treats the screen as divided into horizontal strips (Figure 3-1) of arbitrary width, where the horizontal format of window tiles across the strip remains constant for the whole strip. This divides the region into rectilinear areas, which are easy to manage. By combining strips, overlapping windows can easily be obtained.

Windows may essentially be arbitrarily shaped (circular, irregular, etc.) because a new strip may be defined every display line, similar to the format shown in Figure 3-2.

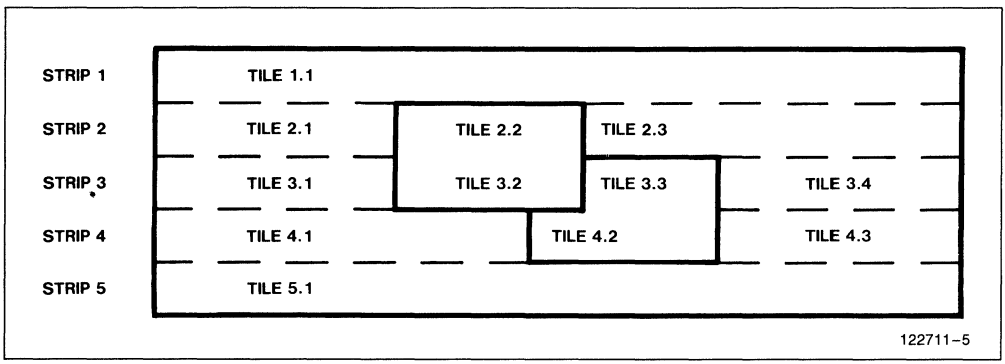

Figure 3-1. Sample Display Implementation of Two Overlapping Windows

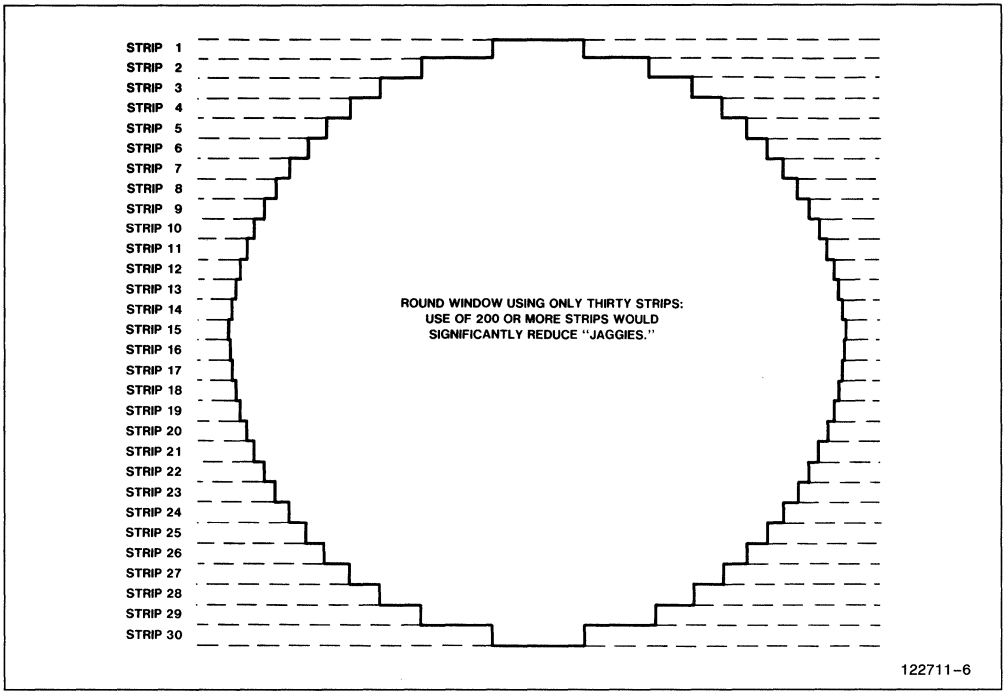

Figure 3-2. Sample Display of Irregular Window

The information needed for the display processor is contained in strip descriptor tables, each made up of a header and one or more tile descriptors. The header contains:

- the number of lines in the strip;
- the number of tiles in the strip;
- upper and lower addresses of the next strip descriptor

Each tile descriptors (which are consecutive in memory) contains:

1. the width of the bit map from which the window is being retrieved (in words);
2. the start address of the bit-map data to be displayed (word in memory and first bit location);
3. the number of words to fetch for the tile;
4. the first and last bit locations of the bit-map data to be displayed;
5. the number of bits per pixel;
6. four bits to indicate border presence for top, bottom, left, and right edges (1 indicates show border, 0 indicates show bit-map for those pixels);
7. window status information which can be used to select color palettes or other attributes (2 bits);
8. two bits to indicate bit-map configuration is byte rather than word-oriented with byte order switched and if bit-map is non-linear (for PC compatibility);
9. bit to indicate if window is to be zoomed by pixel replication of the bit-map data;
10. bit to indicate if tile if field background data.

A one-pixel border can be displayed on any or all sides of each viewport tile. This border color is defined in an 8-bit register and is the same user-definable color for all windows. Borders may be turned on or off for individual tiles.

In the absence of windows, the field background color is displayed. This single color is definable by the user in an 8-bit register. The use of background on the display minimizes system bandwidth because data is only fetched for windows and not for background, and thus saves bit-map memory.

The display processor provides padding bits when bit maps to be displayed have fewer bits/pixel than the hardware display, with no performance decrease. This allows windows of various bits/pixel to be shown simultaneously on the same display. The user programs the desired 8-bit color patterns into three registers, one serving to map each of 1-, 2-, and 4-bits per pixel information into full colors on the display.

All video output from the 82786 can be defined to begin and end at any pixel (except when in accelerated mode using external shift logic). This includes the positioning of every window and the cursor.

The display processor instruction list is controlled by the CPU. The double-word location of the first strip descriptor block is located in a register. The locations of subsequent strip descriptor tables are based upon a linked-list architecture and are provided in the preceding descriptor table. This descriptor linked-list needs only to be updated by the CPU when the window arrangement on the screen changes. New strips and segments are easily inserted into the display list by simply modifying the linked-list pointers of the preceding strips or segments.

The use of redundant lists is possible because the description of a typical display is memory-efficient and requires only about 1,000 bytes. This would permit the CPU to alter the contents of one list while the second is being used to control display processor. When the creation of the new list is complete, the registers pointing to the first strip descriptor table may be switched to the locations for the new list during vertical retrace. This permits the application to alter the display list without causing temporary swimming or blinking of the display.

3.3 CURSOR

The display processor supports a single hardware cursor which may be up to 16 x 16 pixels. This cursor may be positioned by the user anywhere on the screen. The cursor may be defined to be transparent or opaque, and may be either a block cursor or a cross-hair cursor one pixel across stretching the width and height of the screen. The color of the cursor is user-definable, as is the block cursor's pattern. Eight bits of register memory define the color and sixteen 16-bit words of register define the pattern, which is then padded with the cursor color register. Support for a blinking cursor is provided with a register for CURSOR_ON which can be toggled by the CPU as often as necessary to cause an appropriate blink rate. Multiple cursors can be simulated by drawing them in software, especially using bit-blt.

3.4 ZOOM

The display processor allows selected windows to be zoomed (using pixel replication) up to 64 times horizontally and vertically (independently, in steps of one). The setting of the zoom bit in the tile descriptor table causes replication of the pixels in memory according to horizontal and vertical scaling factors contained in registers.

3.5 VIDEO INTERFACE

Eight parallel video data output lines provide video output which may be used as eight bits pixel on the CRT, or externally shifted to boost maximum display resolution. The dot rate output is controlled by an independent video clock which may be up to 25 MHz. Horizontal signals are programmable from 1 to 4096 cycles of the video clock and vertical sync signals from 1 to 4096 scan lines. Use of eight external video data pins allow up to 256 different colors to be directly displayed. Other CRT control lines provided by the display processor are VSYNC, HSYNC, BLANK.

Several 82786s can be used together for higher performance graphics. For multiple 82786 Systems, one 82786 acts as a master generating VSYNC and HSYNC, and the other 82786s act as slaves using the master synch signals for timing through the use of their own VSYNC and HSYNC as inputs. Each 82786 has its own bit-map memory with separate graphics processor lists to form a bit-plane architecture, but use the same display list. The BLANK signal is not used by slave 82786s.

External color palettes are supported, and, by use of the two window status lines, the application may select one of four color combinations for any window. This supports a maximum of 1024 simultaneous colors per frame. The palette may be programmed by latching the default video data when the BLANK pin is high. The display processor can support non-interlaced, and interlaced-synch displays. Selection of the interlacing, control to support external shifting of the video data, default video data contents, and slave/master status for each 82786 are controlled via dedicated registers. The 82786 may be synchronized to an external source ("Gen-Locking").

CHAPTER 4
82786 SYSTEMS

4.1 TYPICAL SYSTEM CONFIGURATIONS

The 82786 can be used in many different configurations, each providing cost and performance appropriate for different applications and markets.

Three typical applications in which the 82786 could be used are:

1. Low-priced personal computer (Figure 4-1);
2. Multi-tasking office workstation (Figure 4-2);
3. High-performance workstation for processing-intensive, high-resolution applications in engineering (Figure 4-3).

4.2 DRAM CONTROL

The DRAM controller on the 82786 supports an array of up to 32 memory chips without extra logic and up to a 4 megabyte address space. DRAMs supported have densities ranging from 8K to 1 megabit and organiza-

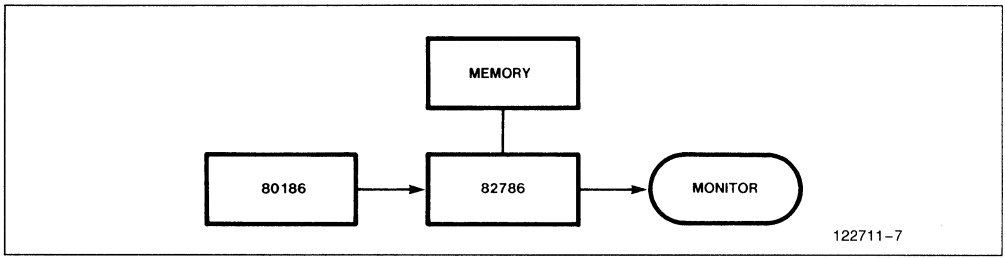

Figure 4-1. Low End Personal Computer

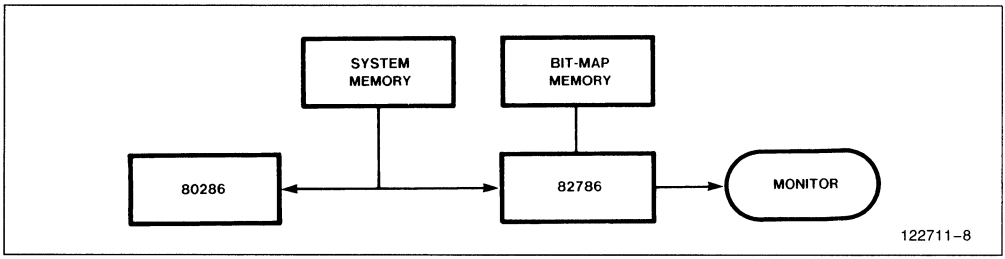

Figure 4-2. Desktop PC/Graphics Terminal

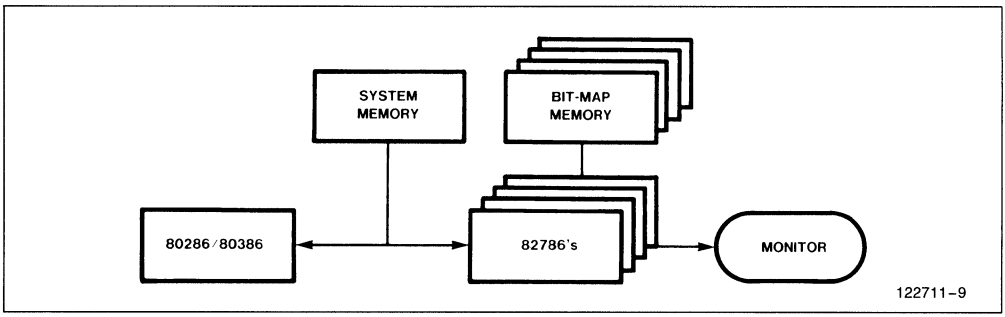

Figure 4-3. High End Workstation

tions of x1, x4, or x8. The bandwidth of the memory system can be increased by interleaving memories and/or using the Ripplemode TM or static-column mode supported by Intel CHMOS DRAMs. Both inter-leaving and Ripplemode TM are completely handled on chip and require no extra external circuits. Use of static-column DRAMs requires one 74X373 latch per bank. Interleaving refers to the use of multiple DRAM banks with one set of memories receiving new CAS signals while the other outputs data. Table 4-1 shows memory burst-bandwidth for the different configurations at 10 MHz.

DRAM refresh is done automatically by the DRAM controller. The memory array can be accessed both by 82786 internal processors (GP, DP) and by external masters (CPUs) through the BIU. The 82786 DRAM controller can be used to control system memory within its 4 megabyte address space, provided the target application can accept the decreased bandwidth of system memory. The portions of the address space dedicated to graphics and system memory are configured at initialization in the DRAM_CONTROL_REGISTER. Graphics memory is assumed to start at OH and continue up to the configuration limit. Memory addresses above this are used for system memory.

4.3 BUS INTERFACE

The Bus Interface Unit of the 82786 is designed to support all 8-, 16-, and 32-bit microprocessors from Intel, with optimization for the 80286. This permits the 82786 to run synchronously with the 80286, increasing throughput by eliminating wait states. A special 8-bit mode allows 82786 to also work with 8-bit data bus microprocessors. The 80386 itself makes interfacing to the 82786 possible. Interfacing to Intel CPUs is detailed in the Hardware Configurations Applications Note.

The bus interface allows slave access by the CPU to the graphics memory controlled through the 82786 DRAM controller. This allows the CPU to update the graphics processor instruction list and the display processor descriptor lists in the graphics memory where maximum throughput can be supported. Low-end systems could use only a single memory shared by both the 82786 and CPU and use the 82786 DRAM controller for this memory.

For performance reasons, many systems will have at least two sections of memory: the 82786 graphics memory (using the on-chip DRAM controller) and the system memory. In this configuration, the 82786 can execute bus cycles on the system bus so the 82786 can access the CPUs own memory. This master mode is designed in accordance with the 80286 definitions. This configuration allows the best of both worlds, the system and graphics memories are split for performance reasons, but the split is transparent to the software for flexibility. Character fonts and graphic objects may be retrieved from disk and placed in system memory locations reserved for access by the 82786 using a virtual mode 80286 or 80386 configuration with appropriate system software.

Table 4-1. 82786 DRAM Bandwidths

	Pagemode DRAM	Ripplemode DRAM
Non Interleaving DRAM banks	10 Megabyte/sec (diagnostics or 640 × 480 × 2)	20 Megabyte/sec (640 × 480 × 4 or 1K × 1K × 1 noninterlaced)
Interleaving DRAM banks	20 Megabyte/sec (640 × 480 × 4 or 1K × 1K × 1 noninterlaced)	40 Megabyte/sec (2K × 2K × 1 interlaced: 1K × 2K × 1. 1K × 1K × 2. 800 × 600 × 4. 640 × 480 × 8 noninterlaced)

CHAPTER 5
PACKAGE AND PIN DESCRIPTION

5.1 OVERVIEW

The 82786 is an eighty-eight pin component due to the large number of functions integrated within the device. It is available in both pin grid array and leaded chip carrier versions. The pinout of a pin grid array is shown in Figure 5-1 and a description of the pins is shown in Table 5-1.

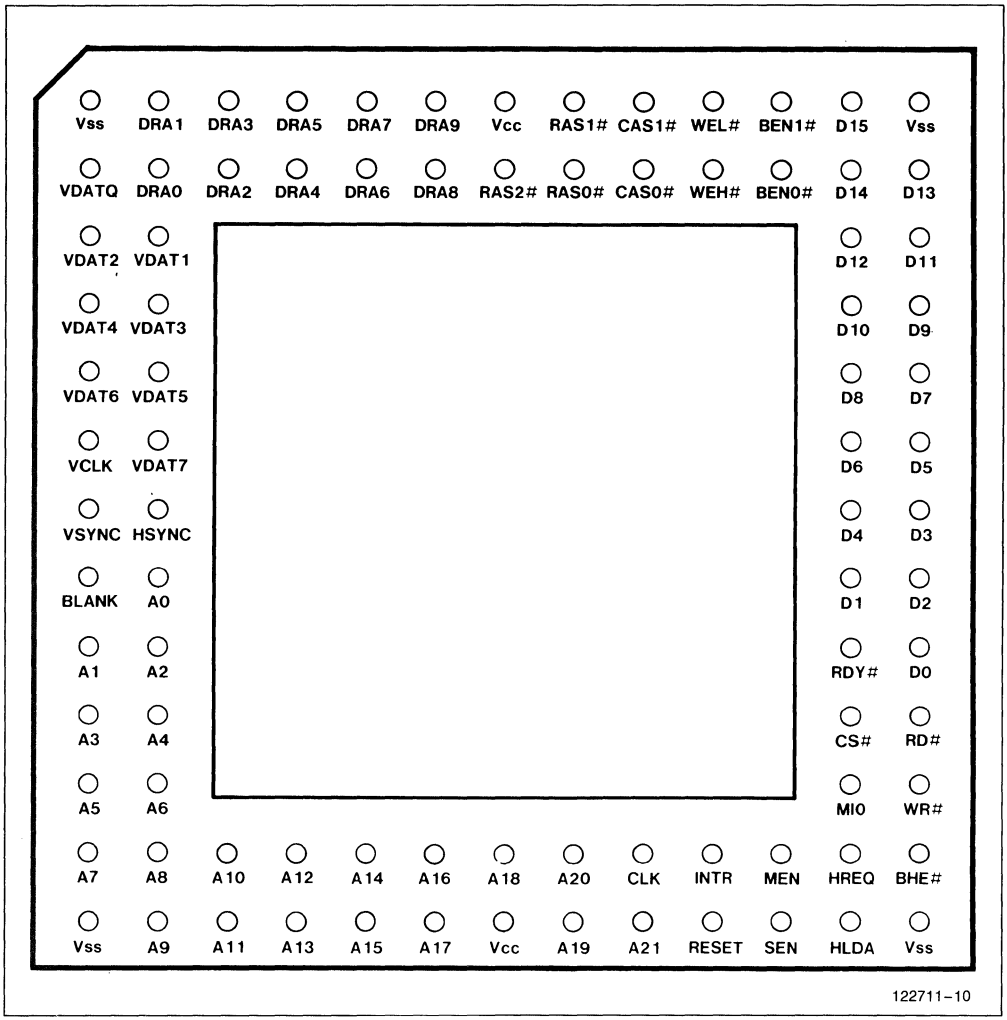

Figure 5-1. PGA Pinout

Table 5-1. 82786 Pin Names and Descriptions

Symbol	Type	Description
A21–0	I/O	**ADDRESS LINES FOR THE LOCAL BUS:** Normally inputs for Slave Mode accesses of the 82786 supported DRAM array or internal memory or I/O mapped registers. Driven by the 82786, when it is the Local Bus Master.
D15–0	I/O	**DATA BUS:** For the 82786 DRAM array and the Local Bus.
BHE#	I/O	**BUS HIGH ENABLE:** An input of the 82786 Slave Interface: driven LOW by the 82786 when it is Local Bus Master. Determines asynchronous vs. synchronous operation for RD#, WR# and HLDA inputs at the falling (trailing) edge of RESET. A HIGH state selects synchronous operation.
RD#	I/O	**READ STROBE:** An input of the 82786 Slave Interface: driven by the 82786 when it is Local Bus Master. Asynchronous vs. synchronous input determined by state of BHE# pin at falling RESET.
WR#	I/O	**WRITE STROBE:** An input of the 82786 Slave Interface: driven by the 82786 when it is Local Bus Master. Asynchronous vs. synchronous input determined by state of BHE# pin at falling RESET.
MIO	I/O	**MEMORY / I/O INDICATION:** An input of the 82786 Slave Interface: driven HIGH by the 82786 when it is the Local Bus Master. Selects 286 Status or Command Mode vs. 8086/186 Status Mode of the 82786 Slave Interface at the falling (trailing) edge of RESET. A LOW state selects the 286 Status or Command Mode.
CS#	I	**CHIP SELECT:** Slave Interface input qualifying the access.
MEN	O	**MASTER ENABLE:** Driven HIGH when the 82786 is in control of the Local Bus, (i.e. HLDA received in response to a 82786 HREQ). Used to steer the data path and select source of bus cycle status commands.
SEN	O	**SLAVE ENABLE:** Driven HIGH when 82786 is executing a Slave bus cycle for an external master on the Local Bus. Used to enable the data path and as a READY indication to the Local Bus Master.
READY#	I	**SYNCHRONOUS INPUT:** To the 82786 when executing Local Bus cycles. Identical to 80286 timing.
HREQ	O	**HOLD REQUEST:** Driven HIGH by the 82786 when an access is being made to the Local Bus by the Display or Graphics Processors. Remains HIGH until the 82786 no longer needs the Local Bus.
HLDA	I	**HOLD ACKNOWLEDGE:** Input in response to a HREQ output. Asynchronous vs. synchronous input determined by state of BHE# pin at falling RESET.
INTR	O	**INTERRUPT:** The logical OR of a Graphics Processor and Display Processor interrupt. Cleared with an access to the BIU Interrupt Register.
CAS0#	O	**COLUMN ADDRESS STROBE 0:** Drives the CAS inputs of the even word DRAM bank if interleaved: identical to CAS1# if non-interleaved DRAM. Capable of driving 16 DRAM CAS inputs.
CAS1#	O	**COLUMN ADDRESS STROBE 1:** Drives the CAS inputs of the odd word DRAM bank if interleaved; identical to CAS0# if non-interleaved DRAM. Capable of driving 16 DRAM CAS inputs.
RAS2–0#	O	**ROW ADDRESS STROBE:** Drives the RAS input pins of up to 16 DRAMs. Drives the first three rows of both banks of DRAM.
DRA9/ RAS3#	O	**MULTIPLEXED MOST SIGNIFICANT DRAM ADDRESS LINE AND RAS3#:** Support of 1Mb DRAMs requires DRA9. When 1Mb DRAMs are used, four rows of DRAMs cannot be supported (RAS3# unnecessary) due to 82786 addressing limit of 4 Mbytes being exceeded.
WEL#	O	**WRITE ENABLE LOW BYTE:** Active LOW strobe to the lower order byte of DRAM.
WEH#	O	**WRITE ENABLE HIGH BYTE:** Active LOW strobe to the higher order byte of DRAM.

Table 5-1. 82786 Pin Names and Descriptions (Continued)

Symbol	Type	Description
DRA8-0	O	**MUTIPLEXED DRAM ADDRESS:** DRAM row and column address are multiplexed on these lines. Capable of driving 32 DRAMs without buffers.
BEN1-0#	O	**BANK ENABLE 1 AND 0:** Enables the output of the DRAM array on to the 82786 data bus (D15-0). BEN1# controls Bank 1. BEN0# controls Bank 0.
BLANK	I/O	**OUTPUT USED TO BLANK THE DISPLAY AT PARTICULAR POSITIONS ON THE SCREEN:** May also be configured as inputs to allow the 82786 to be synchronized with external sources.
VDATA7-0	O	**VIDEO DATA OUTPUT.**
VCLK	I	**VIDEO CLOCK INPUT:** used to drive the display section of the 82786. Its maximum frequency is 25 MHz.
HSYNC/ WST0	I/O	**HORIZONTAL SYNC:** Window status may be multiplexed on this pin. Can also be configured as input to allow the 82786 to be synchronized with external sources. Even as input, window status still output when BLANK is low.
VSYNC/ WST1	I/O	**VERTICAL SYNC:** Window status can be multiplexed on this pin. Can also be configured as input to allow the 82786 to be synchronized with external sources. Even as input, window status still output when BLANK is low.
RESET	I	**RESET INPUT:** internally synchronized. Halts all activity on the 82786 and brings it to defined state. The leading edge of RESET synchronizes the clock to PH1. The trailing edge latches the state of BHE# and MIO to establish the type of Slave Interface. It also latches RD# and WR# to set certain test modes.
CLK	I	**DOUBLE FREQUENCY CLOCK OUTPUT:** Clock input to which pin timings are referenced. 50% duty cycle.
V_{SS}, V_{CC}		**4 V_{SS} AND 2 V_{CC} PINS.**

DOMESTIC SALES OFFICES

ALABAMA

Intel Corp.
5015 Bradford Drive
Suite 2
Huntsville 35805
Tel: (205) 830-4010

ARIZONA

Intel Corp.
11225 N. 28th Drive
Suite 214D
Phoenix 85029
Tel: (602) 869-4980

Intel Corp.
1161 N. El Dorado Place
Suite 301
Tucson 85715
Tel: (602) 299-6815

CALIFORNIA

Intel Corp.
21515 Vanowen Street
Suite 116
Canoga Park 91303
Tel: (818) 704-8500

Intel Corp.
2250 E. Imperial Highway
Suite 218
El Segundo 90245
Tel: (213) 640-6040

Intel Corp.
1510 Arden Way, Suite 101
Sacramento 95815
Tel: (916) 920-8096

Intel Corp.
4350 Executive Drive
Suite 105
San Diego 92121
(619) 452-5880

Intel Corp.*
2000 East 4th Street
Suite 100
Santa Ana 92705
Tel: (714) 835-9642
TWX: 910-595-1114

Intel Corp.*
San Thomas 4
2700 San Thomas Expressway
Santa Clara, CA 95051
Tel: (408) 986-8086
910-338-0255

COLORADO

Intel Corp.
3300 Mitchell Lane, Suite 210
Boulder 80301
Tel: (303) 442-8088

Intel Corp.
4445 Northpark Drive
Suite 100
Colorado Springs 80907
Tel: (303) 594-6622

Intel Corp.*
650 S. Cherry Street
Suite 915
Denver 80222
Tel: (303) 321-8086
TWX: 910-931-2289

CONNECTICUT

Intel Corp.
26 Mill Plain Road
Danbury 06810
Tel: (203) 748-3130
TWX: 710-456-1199

EMC Corp.
222 Summer Street
Stamford 06901
Tel: (203) 327-2934

FLORIDA

Intel Corp.
242 N. Westmonte Drive
Suite 105
Altamonte Springs 32714
Tel: (305) 869-5588

Intel Corp.
6363 N.W. 6th Way, Suite 100
Ft. Lauderdale 33309
Tel: (305) 771-0600
TWX: 510-956-9407

FLORIDA (Cont'd)

Intel Corp.
11300 4th Street North
Suite 170
St. Petersburg 33702
Tel: (813) 577-2413

GEORGIA

Intel Corp.
3280 Pointe Parkway
Suite 200
Norcross 30092
Tel: (404) 449-0541

ILLINOIS

Intel Corp.*
300 N. Martingale Road, Suite 400
Schaumburg 60172
Tel: (312) 310-8031

INDIANA

Intel Corp.
8777 Purdue Road
Suite 125
Indianapolis 46268
Tel: (317) 875-0623

IOWA

Intel Corp.
St. Andrews Building
1930 St. Andrews Drive N.E.
Cedar Rapids 52402
Tel: (319) 393-5510

KANSAS

Intel Corp.
8400 W. 110th Street
Suite 170
Overland Park 66210
Tel: (913) 345-2727

MARYLAND

Intel Corp.*
7321 Parkway Drive South
Suite C
Hanover 21076
Tel: (301) 796-7500
TWX: 710-862-1944

Intel Corp.
7833 Walker Drive
Greenbelt 20770
Tel: (301) 441-1020

MASSACHUSETTS

Intel Corp.*
Westford Corp. Center
3 Carlisle Road
Westford 01886
Tel: (617) 692-3222
TWX: 710-343-6333

MICHIGAN

Intel Corp.
7071 Orchard Lake Road
Suite 100
West Bloomfield 48033
Tel: (313) 851-8096

MINNESOTA

Intel Corp.
3500 W. 80th Street
Suite 360
Bloomington 55431
Tel: (612) 835-6722
TWX: 910-576-2867

MISSOURI

Intel Corp.
4203 Earth City Expressway
Suite 131
Earth City 63045
Tel: (314) 291-1990

NEW JERSEY

Intel Corp.*
Parkway 109 Office Center
328 Newman Springs Road
Red Bank 07701
Tel: (201) 747-2233

Intel Corp.
75 Livingston Avenue
First Floor
Roseland 07068
Tel: (201) 740-0111

NEW MEXICO

Intel Corp.
8500 Menual Boulevard N.E.
Suite B 295
Albuquerque 87112
Tel: (505) 292-8086

NEW YORK

Intel Corp.*
300 Vanderbilt Motor Parkway
Hauppauge 11788
Tel: (516) 231-3300
TWX: 510-227-6236

Intel Corp.
Suite 2B Hollowbrook Park
15 Myers Corners Road
Wappinger Falls 12590
Tel: (914) 297-6161
TWX: 510-248-0060

Intel Corp.*
211 White Spruce Boulevard
Rochester 14623
Tel: (716) 424-1050
TWX: 510-253-7391

NORTH CAROLINA

Intel Corp.
5700 Executive Center Drive
Suite 213
Charlotte 28212
Tel: (704) 568-8966

Intel Corp.
2700 Wycliff Road
Suite 102
Raleigh 27607
Tel: (919) 781-8022

OHIO

Intel Corp.*
6500 Poe Avenue
Dayton 45414
Tel: (513) 890-5350
TWX: 810-450-2528

Intel Corp.*
Chagrin-Brainard Bldg., No 300
28001 Chagrin Boulevard
Cleveland 44122
Tel: (216) 464-2736
TWX: 810-427-9298

OKLAHOMA

Intel Corp.
6801 N. Broadway
Suite 115
Oklahoma City 73116
Tel: (405) 848-8086

OREGON

Intel Corp.
10700 S.W. Beaverton
Hillsdale Highway
Suite 22
Beaverton 97005
Tel. (503) 641-8086
TWX: 910-467-8741

PENNSYLVANIA

Intel Corp.
1513 Cedar Cliff Drive
Camphill 17011
Tel: (717) 737-5035

Intel Corp.*
455 Pennsylvania Avenue
Fort Washington 19034
Tel: (215) 641-1000
TWX: 510-661-2077

Intel Corp.*
400 Penn Center Boulevard
Suite 610
Pittsburgh 15235
Tel: (412) 823-4970

PUERTO RICO

Intel Microprocessor Corp.
South Industrial Park
Las Piedras 00671
Tel: (809) 733-3030

TEXAS

Intel Corp.
313 E. Anderson Lane
Suite 314
Austin 78752
Tel: (512) 454-3628

TEXAS (Cont'd)

Intel Corp.*
12300 Ford Road
Suite 380
Dallas 75234
Tel: (214) 241-8087
TWX: 910-860-5617

Intel Corp.*
7322 S.W. Freeway
Suite 1490
Houston 77074
Tel: (713) 988-8086
TWX: 910-881-2490

Industrial Digital Systems Corp.
5925 Sovereign
Suite 101
Houston 77036
Tel: (713)988-9421

UTAH

Intel Corp.
5201 Green Street
Suite 290
Murray 84123
Tel: (801) 263-8051

VIRGINIA

Intel Corp.
1603 Santa Rosa Road
Suite 109
Richmond 23288
Tel: (804) 282-5668

WASHINGTON

Intel Corp.
110 110th Avenue N.E.
Suite 510
Bellevue 98004
Tel: (206) 453-8086
TWX: 910-443-3002

Intel Corp.
408 N. Mullan Road
Suite 102
Spokane 99206
Tel: (509) 928-8086

WISCONSIN

Intel Corp.
450 N. Sunnyslope Road
Suite 130
Chancellory Park I
Brookfield 53005
Tel: (414) 784-8087

CANADA

BRITISH COLUMBIA

Intel Semiconductor of Canada, Ltd.
301-2245 W. Broadway
Vancouver V6K 2E4
Tel: (604) 738-6522

ONTARIO

Intel Semiconductor of Canada, Ltd.
2650 Queensview Drive
Suite 250
Ottawa K2B 8H6
Tel: (613) 829-9714
TELEX: 053-4115

Intel Semiconductor of Canada, Ltd.
190 Attwell Drive
Suite 500
Rexdale M9W 6H8
Tel: (416) 675-2105
TELEX: 06983574

QUEBEC

Intel Semiconductor of Canada, Ltd.
620 St. Jean Blvd.
Pointe Claire H9R 3K3
Tel: (514) 694-9130
TWX: 514-694-9134

*Field Application Location

INTEL CORPORATION, 3065 Bowers Ave., Santa Clara, CA 95051; Tel. (408) 987-8080

INTEL CORPORATION (U.K.) Ltd., Swindon, United Kingdom; Tel. (0793) 488 388

INTEL JAPAN k.k., Ibaraki-ken; Tel. 029747-8511

Printed in U.S.A./5-86/5K/SCP/PM
Graphic Components Operation