



AP-719

**APPLICATION  
NOTE**

**Intel386™ EX Embedded Microprocessor  
MHT9000 Handheld Terminal**

**Wilfred Martis**

Sr. Applications Engineer

Intel Corporation  
Semiconductor Products Group  
Mail Stop CH6-304  
5000 W. Chandler Blvd.  
Chandler, Arizona 85226

August 1, 1995

Order Number: 272747-001



Information in this document is provided solely to enable use of Intel products. Intel assumes no liability whatsoever, including infringement of any patent or copyright, for sale and use of Intel products except as provided in Intel's Terms and Conditions of Sale for such products.

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 Intel sales office or your distributor to obtain the latest specifications before placing your product order.

MDS is an ordering code only and is not used as a product name or trademark of Intel Corporation.

Intel Corporation and Intel's FASTPATH are not affiliated with Kinetics, a division of Excelan, Inc. or its FASTPATH trademark or products.

\*Other brands and names are the property of their respective owners.

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

Intel Corporation  
Literature Sales  
P.O. Box 7641  
Mt. Prospect, IL 60056-7641

or call 1-800-879-4683

© INTEL CORPORATION 1995



**Application Note:**  
**Intel386™ EX Embedded Microprocessor MHT9000 Handheld Terminal**

1.0	INTRODUCTION.....	1
1.1	MHT9000 Product Description .....	2
2.0	FUNCTIONAL OVERVIEW .....	2
2.1	Digitizer.....	2
2.2	LCD Unit.....	2
2.3	Power Management .....	3
2.4	Special Algorithms.....	3
3.0	THE Intel386 EX EMBEDDED PROCESSOR .....	3
4.0	MHT9000 HARDWARE OVERVIEW.....	3
4.1	Memory and I/O Maps.....	3
4.1.1	RAM .....	4
4.1.2	ROM .....	4
4.1.3	Cartridge Slot.....	4
4.1.4	LCD Controller .....	4
4.2	Digitizer System.....	5
4.2.1	Digitizer Panel.....	5
4.2.2	Digitizer Controller - MTL6560.....	5
4.2.3	Operating Modes .....	5
4.2.4	Command Set for the MTL6560.....	6
4.2.5	CPU Interface .....	6
4.2.6	Digitizer System Operating Theory.....	6
4.3	LCD Display System.....	6
4.3.1	Epson SED1335F Display Controller.....	6
4.3.2	LCD Driver Interface .....	7
4.3.3	SED1335F Command Set .....	7
4.3.4	LCD System Design Considerations.....	7
4.4	Cartridge Slot.....	7
4.5	POWER SUPPLY .....	9
4.6	PC Interface.....	9
5.0	SYSTEM KERNEL .....	10
5.1	Display Function.....	11
5.1.1	Screen Definition .....	11
5.1.2	Text and Graphics Display Windows.....	11
5.1.3	Application-Defined Font .....	11
5.2	Input Function.....	11
5.3	Application Calling.....	12

- 5.4 Cartridge Operation..... 12
  - 5.4.1 Header Structure .....12
- 5.5 Real Mode Operation ..... 13
  - 5.5.1 Memory Bank Switching .....13
  - 5.5.2 Limitations of Real Mode Operation.....13
- 6.0 POWER MANAGEMENT ..... 13
  - 6.1 Digitizer Interface ..... 13
  - 6.2 PC Interface Driver Power Control ..... 13
  - 6.3 Oscillating Frequency ..... 13
  - 6.4 Idle Mode..... 14
- 7.0 SOFTWARE DEVELOPMENT ..... 14
  - 7.1 Development Platform..... 14
  - 7.2 MHT9000 Kernel and Application Development Tools..... 14
- 8.0 RELATED DOCUMENTS ..... 14



**FIGURES**

Figure 1.	MHT9000 Block Diagram .....	1
Figure 2.	Intel386™ EX Embedded Processor Block Diagram .....	3
Figure 3.	MHT9000 Memory Configuration .....	4
Figure 4.	RAM Access Signal.....	4
Figure 5.	Digitizer Driver Circuit .....	5
Figure 6.	MTL6560-to-Intel386™ EX Processor Interface .....	6
Figure 7.	LCD Driver Interface .....	7
Figure 8.	Battery Backup System.....	9
Figure 9.	Interface to Host System.....	10
Figure 10.	Keyboard Interface.....	10
Figure 11.	Memory Map of a Cartridge (in Real Mode).....	13

**TABLES**

Table 1.	Memory Configuration.....	3
Table 2.	Cartridge Chip-Select Signals Usage.....	4
Table 3.	Operating Modes.....	5
Table 4.	LCD Controller Registers.....	7
Table 5.	SED1335F Command Summary.....	8
Table 6.	Cartridge Connector Pin Assignment.....	8
Table 7.	Header Structure and Memory Locations.....	12
Table 8.	P3.5 Memory Mapping.....	13



**1.0 INTRODUCTION**

This application note describes the MHT9000 Hand-held Terminal — a low cost, GUI-based pen-input device that uses the Intel386™ EX embedded processor. With its computing power, high integration and low cost, the Intel386 EX processor is ideal for such hand-held terminals. Contact your Intel Sales Representative for additional information about the OS kernel, the digitizer ASIC and the MHT9000.

The MHT9000, designed by Micom Tech Ltd. of Hong Kong, is ideal for use as a base platform for building hand-held applications for communications, point-of-sale (POS) terminals, digital books and personal digital assistants. The MHT9000 can be used “as is” or as a building block to create a specific solution.

Hand-held terminal applications are typically characterized by palm-sized portable devices for specific uses. Such products are widely used in environments such as retail, service, medical, financial, warehouse, inventory control, shipping, package delivery, industrial, hospitals, law enforcement, and more.

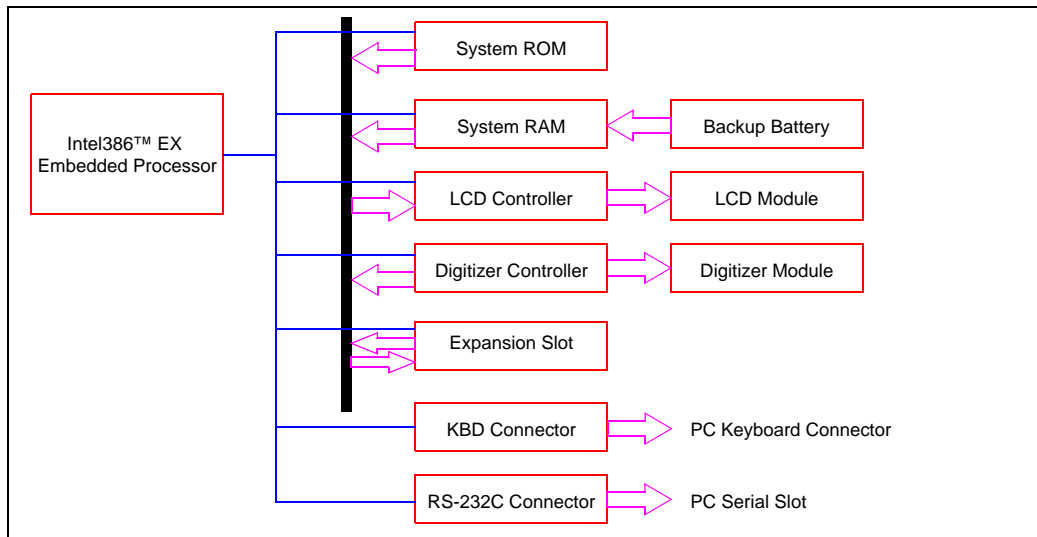
The MHT9000 contains a low-cost proprietary cartridge slot which enables the development of a wide variety of

applications. With appropriate cartridge modules, it becomes:

- An intelligent pen-based input terminal to a DOS-based system
- A hand-held barcode terminal
- A hand-held POS terminal
- An electronic multi-language dictionary
- A digital reference book
- An intelligent voice-based hand-held terminal
- An advanced pen-based PDA with communication capability

The Intel386 EX processor is a single-chip system that incorporates an on-board static Intel386 SX processor core with a host of integrated peripherals, including DMA and interrupt controllers, serial and parallel ports, chip selects, timers/counters and a JTAG unit. Its 26-bit addressing provides a large 64 Mbyte memory address space. Refer to Section 8.0, RELATED DOCUMENTS (pg. 8-14) for a list of documents that contain detailed information about the Intel386 EX processor.

The MHT9000 Hand-held Terminal Reference Design also incorporates a number of additional technologies which may be used as building blocks for many applications.



**Figure 1. MHT9000 Block Diagram**



## 1.1 MHT9000 Product Description

The MHT9000 features include:

- The Intel386™ EX embedded processor
- 320x128 STN or FSTN LCD touch-screen, capable of displaying 40 x 16 Roman characters or 20 x 8 Chinese/Kanji characters
- Resistive digitizer with software-selectable resolution
- Low cost proprietary cartridge slot with optional modules, including: barcode scanner, restaurant POS terminal, multi-language dictionary, electronic book, voice recognition, and more
- RS-232C and PC keyboard port
- Built-in software for Chinese handwriting recognition
- Memo and keyboard operation modes
- 4.5 V design (powered by three AAA batteries)
- Power Management feature - programmable to “sleep” between input strokes
- 32 Kbyte x 16 or 128 Kbyte x 16 SRAM
- 1 Mbyte (512 Kbyte x 16) ROM
- 4.0” x 6.5” x 0.9” (WxHxD) case size
- Less than US\$100 bill of materials (in volumes greater than 10,000)

## 2.0 FUNCTIONAL OVERVIEW

The following list briefly describes the MHT9000’s primary features. The remainder of this application note further describes these features:

- Uses the Intel Architecture — This well-known architecture enables the use of a DOS-based development platform.
- Contains a proprietary OS kernel — To achieve a cost-competitive design, a proprietary OS Kernel was created specifically for the MHT9000. Being a non-DOS-compatible system, memory configuration can be optimized for the application. ROM DOS or BIOS is not required.
- Uses a high-performance processor — The Intel386 EX processor minimizes hardware cost by achieving the intelligence through software instead of using specific hardware components. Spare processing power is reserved for future applications.

- Easy to manufacture — The MHT9000 is designed to take advantage of low-cost manufacturing technology.
- Expandable — The proprietary cartridge slot, a simple memory mapped slot, supports application-specific modules.

## 2.1 Digitizer

The digitizer system, a very important feature of a pen-based hand-held terminal, consists of three subcomponents: touch panel, controller, and driver circuit.

The resistive-type touch panel has four pins for interfacing the panel to the controller through the digitizer driver circuitry:

XH, XL:                resistance R X  
YH, YL:                resistance R Y

The coordinate of any activated point can be calculated as:

$X = (RX/RX_{max}) * X \text{ resolution}$   
 $Y = (RY/RY_{max}) * Y \text{ resolution}$

The digitizer’s controller — an MTL6560 — is a custom-designed LSI device used as a digitizer for analog/resistive pen entry touch panel. It supports digitizer resolution of up to 1024 x 1024. It connects directly to a touch panel through a simple circuitry of transistors and passive components. It connects directly to the Intel386 EX processor through a built-in serial port. X/Y resolution is software programmable. The default X/Y resolution is 640/256.

The driver circuit consists of transistors, diodes and passive components only.

## 2.2 LCD Unit

The LCD unit supports both STN LCD and FSTN LCD types. The FSTN LCD provides superior contrast results when used with a touch panel (it is also more expensive).

LCD resolution is 320 x 128 dots, typically configured as 320 (horizontal) x 128 (vertical). However, the LCD display software drivers also allow a 128 (horizontal) x 320 (vertical) configuration.

Low-power LCD drivers are used to minimize current consumption.



### 2.3 Power Management

As a hand-held device with an attached battery, the power management scheme is significant and essential. The terminal is programmed to “sleep” between input strokes. This significantly reduces the average current consumption.

### 2.4 Special Algorithms

To minimize hardware cost, all required special algorithms are implemented in software — no additional or special hardware is required. Such special algorithms include: handwriting recognition, text compression/decompression, graphic compression/decompression, and voice recognition.

### 3.0 THE Intel386 EX EMBEDDED PROCESSOR

The Intel386 EX processor is a highly integrated, fully static, 32-bit CPU optimized for embedded applications. It features low power and low voltage capabilities, integration of many commonly used embedded peripherals, and a 32-bit programming architecture compatible with DOS-based systems. Figure 2 shows a block diagram of the Intel386 EX processor. Refer to the *Intel386™ EX Embedded Micro-processor User’s Manual* (272485) for additional details.

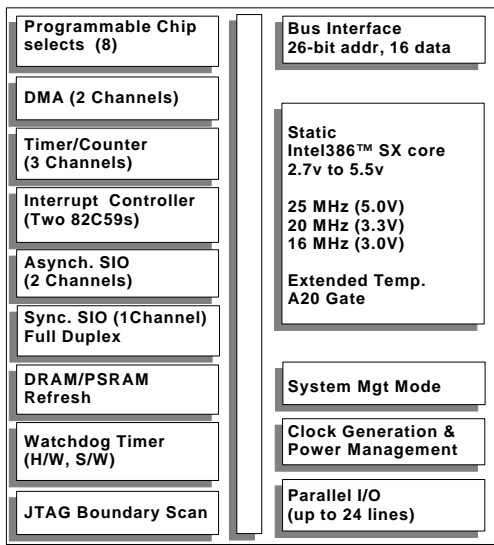


Figure 2. Intel386™ EX Embedded Processor Block Diagram

### 4.0 MHT9000 HARDWARE OVERVIEW

The subsections that follow describe the MHT9000’s major hardware features.

#### 4.1 Memory and I/O Maps

To minimize the external hardware required for memory interfacing, the MHT9000 utilizes the Intel386 EX processor’s Chip-select Unit. Five chip-select signals from the Intel386 EX processor are used to access the memory and I/O devices. However, due to differences in memory allocation for Real Mode and Protected Mode operation, chip select signal configuration is different in each mode. See Table 1. The kernel automatically performs software bank switching in Real Mode operation.

Table 1. Memory Configuration

MODE	SIGNAL	CONFIGURATION
Protected Mode	CS0#	RAM, 000000H - 00FFFFH
	CS1#	Not used
	CS2#	ROM, 3F00000H - 3FFFFFFH
	CS3#	LCD, 1000H - 1002H
	CS4#	Cartridge, 0200000H - 02FFFFFFH
	CS5#	Cartridge, configured by the cartridge application
	CS6#	Not used
Real Mode	UCS#	Used only in reset procedure; not used after memory configuration
	CS0#	RAM,00000H - 0FFFFH
	CS1#	Not used
	CS2#	ROM,C0000H - FFFFFH
	CS3#	LCD, 1000H - 1002H
	CS4#	Cartridge, 40000H - 7FFFFH
	CS5#	Cartridge, 80000H - BFFFFH
CS6#	Not used	
UCS#	Recognition Database, 60000H - 7FFFFH and Chinese Character Generator, 80000H - BFFFFH	



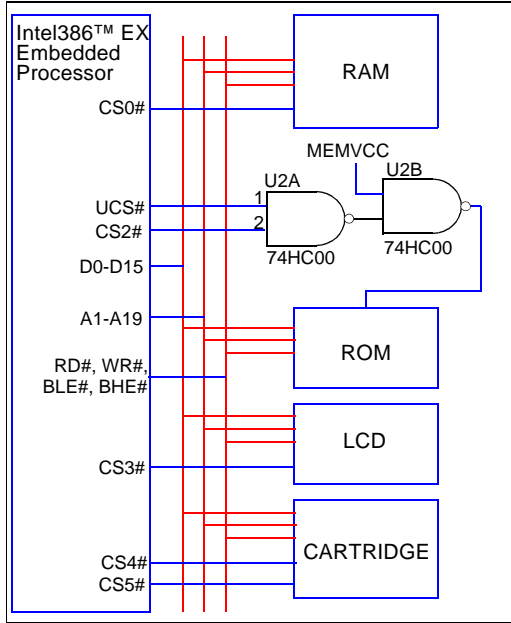


Figure 3. MHT9000 Memory Configuration

4.1.1 RAM

A minimal configuration for a MHT9000 system design requires two 32 K x 8 bit static RAM devices to form the 32 Kbyte word memory. The interface logic is shown in Figure 4. BHE# and BLE# are used to decode the odd and even bytes, respectively. CS0# is used to access this RAM from address 000000H to 00FFFFH at one wait state.

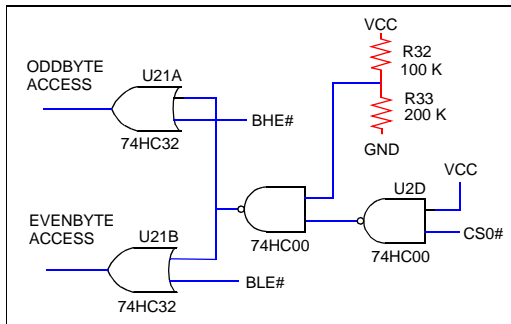


Figure 4. RAM Access Signal

4.1.2 ROM

One 512 K x 16 bit mask-programmed ROM is used in the MHT9000. Two chip select signals, UCS# and CS2# from the Intel386 EX processor, are used to access this ROM, as shown in Figure 3. Upon power-up or reset, UCS# is active which allows the access of ROM for boot-up. After the memory system configuration is completed, CS2# is programmed to access the ROM from address 3F00000H to 3FFFFFFH (1 Mbyte). For Protected Mode operation, UCS# is not used, since CS2# can cover the entire 1 Mbyte range and bank switching is not required. UCS# is only used at initialization in Real Mode operation.

4.1.3 Cartridge Slot

The cartridge is a simple address/data extended slot; as such, only a ROM or Flash module with associated circuitry can be placed in the cartridge slot, to execute a specific application. No decoding and control logic is required. For example, when used as an Electronic Book, the cartridge contains only memory. The MHT9000 provides two chip select signals in the cartridge slot: CS5# and CS4#. These can be configured for the various devices that need to be accessed, as shown in Table 2.

CS4# and CS5# are programmable via the cartridge software. After initialization, in Protected Mode, CS5# is programmed for address 0200000H to 02FFFFFFH at four waitstates and CS4# is disabled. In Real Mode, CS5# is programmed for address 40000H to 7FFFFFFH and CS4# is used for address 80000H to BFFFFFFH.

Table 2. Cartridge Chip-Select Signals Usage

CS4#	CS5#	Example
I/O device	ROM / Flash	Modem Card
RAM	ROM / Flash	PDA
ROM / Flash	ROM / Flash	Electronic Reader, Dictionary, etc.

4.1.4 LCD Controller

The display system is controlled and accessed via the I/O addresses 1000H and 1002H. Two wait states are inserted during accesses to the controller.



## 4.2 Digitizer System

### 4.2.1 Digitizer Panel

The resistive-type digitizer panel is coated with transparent film. Since it is not the same size as the LCD view area, it must be calibrated after each power-up to accurately position the panel. On power-up, the user touches a pen to the corners of the screen to perform the calibration.

### 4.2.2 Digitizer Controller - MTL6560

The MTL6560 custom-designed LSI device is used as a digitizer controller for the resistive pen-entry touch panel. It supports digitizer resolutions up to 1024 x 1024. It is directly connected to the digitizer panel through a simple transistor/resistor driver circuit, as shown in Figure 5.

The MTL6560:

- Employs CMOS process technology
- Requires a single power supply 2.4 V to 5.5 V
- Supports oscillator frequencies up to 1.8432 MHz
- Supports digitizer resolutions of:
  - 256 x 128
  - 320 x 200
  - 320 x 256
  - 640 x 400
  - 640 x 320
  - 640 x 480
  - 640 x 256
  - 1024 x 1024
- Supports two communication modes:
  - Parallel mode (proprietary format)
  - Serial mode with baud rates of 300, 1200, 2400, 4800, 7200 and 9600
- Low operating current: 450  $\mu$ A @ 3.0 V and 1.2 mA @ 5.0 V
- Supports Halt mode
- Contains hardware jumpers for setting the operating speed and default mode

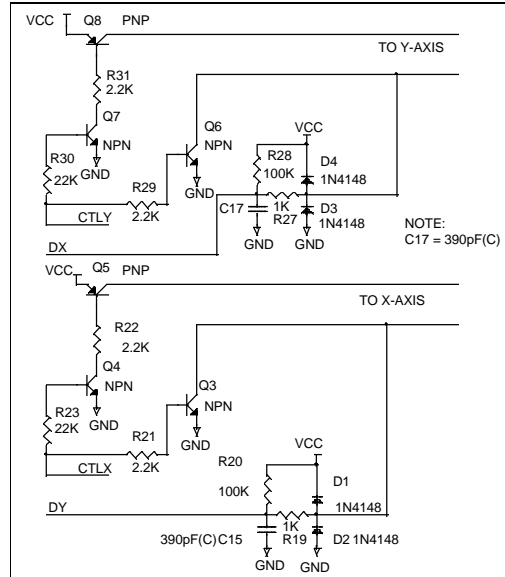


Figure 5. Digitizer Driver Circuit

### 4.2.3 Operating Modes

Table 3 defines the various operating modes of the MTL6560.

Table 3. Operating Modes

MODE	DEFINITION
Sleep	Activated by the Sleep command, the MTL6560 enters Halt mode; the digitizer panel's scanning and data bytes are not sent to the CPU. The controller exits Halt mode by either the activation of DTR or RTS or the appearance of data on the RXDATA pin.
Idle	Waiting for a CPU command.
Command	Activating DTR puts the controller into command mode until the command and respond cycles complete.
Data	While the appropriate serial link is established, the controller scans the digitizer panel and sends data to the CPU through its TXDATA pin (RXD1 on the CPU) until the DTR signal activates.

#### 4.2.4 Command Set for the MTL6560

The command set for the MTL6560 includes:

- Initialize
- Sleep
- Calibrate
- Set X-resolution
- Set Y-resolution
- Select baud rate (serial only; this overrides the default jumper setting)

#### 4.2.5 CPU Interface

To transfer data between the CPU and the MTL6560, the MHT9000 uses a serial link with a baud rate of 9600. The MHT9000 uses the Asynchronous Serial I/O Unit, SIO1, of the CPU to interface to the digitizer (using CPU signals TXD1, RXD1, DTR1#, DSR1#, CTS1# and RTS1#). The MTL6560's RESET pin is connected to the Parallel Port pin P1.1 of the CPU. Upon Reset, the kernel's boot routine uses this pin to reset the MTL6560.

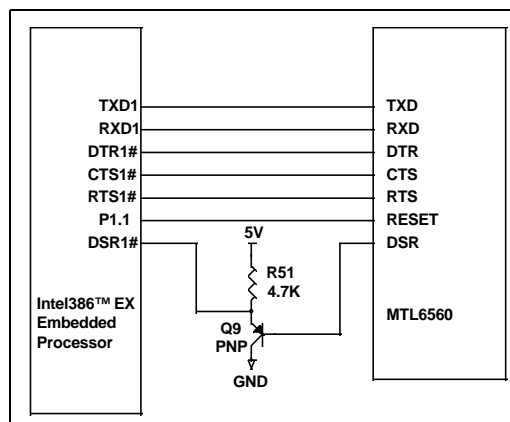


Figure 6. MTL6560-to-Intel386™ EX Processor Interface

#### 4.2.6 Digitizer System Operating Theory

While in data mode, the digitizer controller (MTL6560) continuously scans the digitizer panel to determine when and where the pen is touching the screen. In an X-axis

detection cycle, the digitizer outputs a high level on the CTLX pin and a low on the CTLY pin.

- Q3, Q4 and Q5 are ON
- Q6, Q7 and Q8 are OFF

DC voltage is applied to the X-axis terminals. DX (pin AD1/DX), an analog input signal, is the circuit's feedback point. It is similar to the center terminal of a potentiometer and indicates where the pen is touching along the X-axis. A similar signal, DY (pin AD3/DY), is used during a Y-axis detection cycle. After both axes are detected, the X-Y coordinate data is output to the CPU via the serial link.

To minimize the power supply's effects on the accuracy of X-Y detection, VR and AGND are used to sample the reference voltage each time an acquisition occurs. However, significant ripple on the power line can affect acquisition accuracy.

### 4.3 LCD Display System

#### 4.3.1 Epson SED1335F Display Controller

The Epson\* SED1335F is a LCD display controller that displays text and graphics on a LCD panel. Up to three overlapping layers, with a maximum resolution of 640 x 256 pixels, can be displayed. The SED1335F can display layered text and graphics, scrolling the display in any direction and partitioning the display into multiple screens.

The SED1335F stores text, character codes and bitmapped graphics in external frame buffer memory. Display controller functions include data transfer from the Intel386 EX processor to buffer memory, reading memory data, converting data to display pixels and generating timing signals for the buffer memory and LCD panel.

The features of the SED1335F include:

- Text, graphics and combined text/graphics display modes
- Three overlapping screens in graphics mode
- 640 x 256 pixels LCD panel display resolution
- Programmable cursor control
- Smooth horizontal and vertical scrolling of all or part of the display
- 1/2-duty to 1/256-duty LCD drive



- Supports up to 64 Kbytes of external static RAM frame buffer memory
- Internal character generator
- 160 5 x 7 pixel characters in internal mask-programmed character-generator ROM
- Up to 64 8 x 16 pixel characters in external character-generator RAM
- Up to 256 8 x 16 pixel characters in external character-generator ROM
- Low power consumption: 3.5 mA operating current ( $V_{DD} = 3.5\text{ V}$ ), 0.05  $\mu\text{A}$  standby current
- 2.7 V to 5.5 V

### 4.3.2 LCD Driver Interface

As shown in Figure 7, the MHT9000 uses two types of LCD drivers: the EPSON's SED1600F and SED1630F:

- The SED1600F is a segment driver which can drive up to 80 segments. The MHT9000, with 320 segments in the LCD panel, requires four SED1600F drivers connected in cascade.
- The SED1630F is a common driver which can drive up to 68 common lines. The LCD panel has 128 common lines and requires two SED1630F common drivers connected in cascade.

LCD frame frequency is software programmable in the SED1335F.

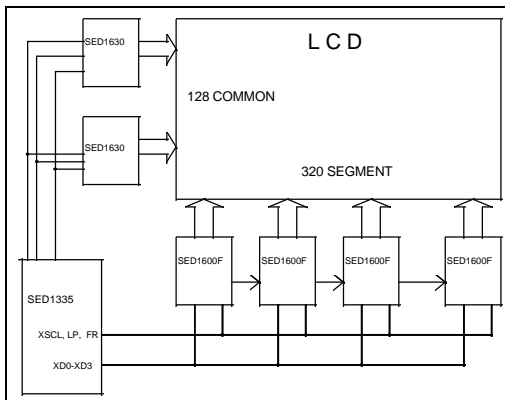


Figure 7. LCD Driver Interface

### 4.3.3 SED1335F Command Set

The SED1335F controls all LCD activity. SED1335F operation and control parameters are programmed via the Intel386 EX processor using the command set, via the I/O ports 1000H and 1002H. Refer to Table 4.

Most commands include their associated parameters. Commands are written to 1002H; associated parameters are written to 1000H. Display memory data is read from 1002H after sending the MEMR command to 1000H. Table 5 shows a listing of the commands.

### 4.3.4 LCD System Design Considerations

The MHT9000 uses a two-layer design: one layer is text, the other is graphics. Due to the requirements that text and graphics must appear simultaneously on-screen, and Chinese characters and multi-size fonts need to be implemented, the MHT9000 displays all text as graphic images. The graphic layer displays the graphic bitmaps and character bitmaps; the text layer displays the cursor. The cursor is software-generated, thus giving the system full control of cursor size and shape.

Unlike in a DOS-based system, display RAM is not directly mapped into the system memory. It is accessed through the SED1335F using the MREAD/MWRITE commands. The memory location pointer is the cursor address, which is set and read by CSRW and CSRR, respectively.

To minimize cost, the relatively small 8 K x 8 static RAM is used as display memory. Only one text layer and one graphic layer can be allocated in these 8 Kbytes.

## 4.4 Cartridge Slot

A proprietary design is used for the cartridge slot connection. The cartridge connector is a 46-pin dual-inline header. Table 6 identifies and defines the cartridge connector signals.

Table 4. LCD Controller Registers

I/O Addr	RD#	WR#	D0-D7
1000H	High	Low	Data/parameters written to SED1335F
1000H	Low	High	Status read from SED1335F
1002H	High	Low	Command write to SED1335F
1002H	Low	High	Data/parameters read from SED1335F

Table 5. SED1335F Command Summary

Class	Command	Code	Command Description	Command Read Parameters
System Control	SYSTEM SET	40	Initialize device and display	8
	SLEEP IN	53	Enter standby mode	0
Display control	DISP ON/OFF	58 59	Enable and disable display and display flashing	1
	SCROLL	44	Set display start address and display regions	10
	CSRFORM	5D	Set cursor type	2
	CGRAM ADR	5C	Set start address of character generator RAM	2
	CSRDIR	4C to 4F	Set direction of cursor movement	0
	HDOT SCR	5A	Set horizontal scroll position	1
	OVLAY	58	Set display overlay format	1
Drawing control	CSRW	46	Set cursor address	2
	CSRR	47	Read cursor address	2
Memory control	MWRITE	42	Write to display memory	-
	MREAD	43	Read from display memory	-

Table 6. Cartridge Connector Pin Assignment

Signal	Description
A1-A23, A25	System address bus. To minimize pin count, A24 is not used.
D0-D7	System Data bus. Cartridge memory must be designed as 8-bit wide memory
RD#	System memory/device read signal
WR#	System memory/device write signal
BLE#	Even-byte enable. Emulates A0 in 8-bit memory system.
PRESET#	Peripheral reset pulse.
CS4#	Extra/complementary chip-select signal for cartridge.
CS5#	Primary chip-select for cartridge.
P3.3	General I/O port. For practical purposes, it functions as an input port (e.g., IR data input or barcode data).
P3.5	General I/O port. Used as an output port. In Real Mode operation, it is used for hardware switching. In Protected Mode, it can be configured as an output port.
P1.6	General I/O port; normally an output. Generally used to control the cartridge's power.
VCC	System power supply.
MEMVCC	System RAM power supply. When power is OFF, backup battery provides 3 V backup voltage at this pin.
GND	Ground.
TMROUT1	Output pin from Timer 1. Intended for use as IR pulse output.
BS8#	Not used in MHT9000. However, when the cartridge is connected to the proprietary-design Flash memory programmer, VPP is present on this pin.

The cartridge connection contains only the bus signals and a few I/O port bits. As such, simple memory can be designed-in to contain data and cartridge software programs. System cost can be further minimized when only memory is located in the cartridge and an 8-bit bus design is used.

- Memory interface: A1-A23, A25, D0-D7, RD#, WR#, CS4#, CS5# and BLE# are used to decode the memory (Flash memory, ROM or even RAM located in the cartridge module) or I/O device (e.g. modem chip).
- I/O ports: Normally P3.5 and P1.6 are programmed as output ports and P3.3 is programmed as an input port to simplify the cartridge's hardware design. P3.5 and P1.6 are set to output LOW before the cartridge program executes. P3.5 is used for memory bank switching in Real Mode execution for overlapping the cartridge memory and system ROM memory. In Protected Mode, it can be used as a general output pin.

In the MHT9000, P1.6 is used to control power to the barcode pen or IR receiver. P3.3 is the data input from the IR receiver or barcode pen. TMROUT1 is the output from the CPU to the IR transmitter (used as a pulse generator).

VCC and GND are the normal power signals to the cartridge. VCC drops to 0 V when the unit is OFF. MEMVCC, which provides VCC directly to system RAM, provides ~3 V when the unit is OFF. Therefore, when the cartridge contains RAM, connect the power pin to MEMVCC.

The MHT9000 does not use BS8#. However, it is connected to the Flash memory's VPP pin. This allows a dedicated programming device to write to the Flash memory through the cartridge connector.

#### 4.5 POWER SUPPLY

System power is provided by two sources: main battery and backup battery. The main battery consists of three AAA batteries (4.5 V); the backup battery is a 3 V button-battery. For high-speed and reliable system performance, a stable power supply is required. The system uses 4.7 V to optimize performance and power consumption. A DC-DC converter (XR-8073) provides the steady supply voltage. The converter circuit consists of a simple circuit composed of an inductor, diode and other supporting components.

Moreover, a low battery detector is incorporated into the XR-8073; the threshold voltage can be adjusted by external

resistors. In the MHT9000, the threshold voltage is ~3.0 V and the detector output is connected to Parallel Port pin P3.2 of the CPU. The kernel detects this signal and informs the user to replace the battery when necessary.

The backup battery is connected to the RAM power supply. When the system is operating, VCC is 4.7 V from the converter and D6 is reverse-biased. This prevents backup battery consumption until the system is turned OFF. When this occurs, VCC drops to VSS. At this time, the backup battery supplies 3 V (MEMVCC) to the RAM devices to retain data. A Powerdown threshold is formed by R32 and R33. When the system is OFF, the NAND gates disable all chip selects to RAM so that no access can be made during power down. VCC of the NAND-gate device (U2) is connected to MEMVCC (the backup battery when power is OFF) to provide power when system is OFF.

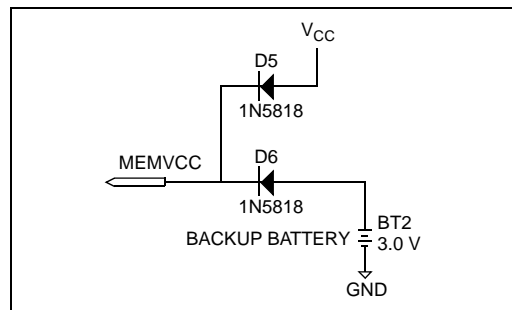


Figure 8. Battery Backup System

#### 4.6 PC Interface

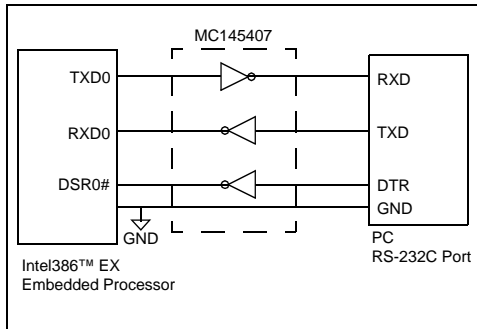
The two PC interface circuits in the MHT9000 are the RS-232 interface and keyboard connector interface.

**RS-232 interface:** As shown in Figure 9, four signals interface the MHT9000 and host system:

- TXD0: data transmit from CPU to the host
- RXD0: data receive from host to the CPU
- GND: common ground reference
- DSR0#: Data Set Ready signal from host (DTR at host end) to indicate that the host is connected

Due to the required ±12 V voltage swing of the RS-232 interface, a line driver IC is required to convert the 0 V-4.7 V to the ±12 V level. The driver is a MC145407 that runs off a single ~5 V supply.



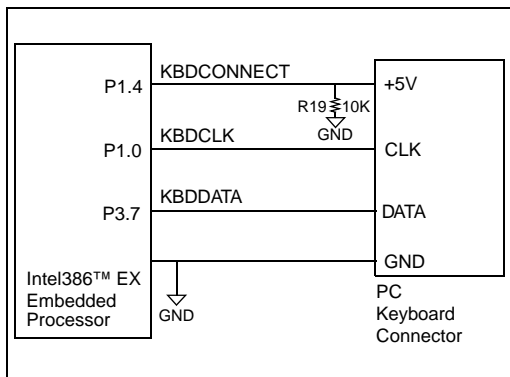


**Figure 9. Interface to Host System**

The RS-232 interface communicates with the host in keyboard mode and keypad mode and is also used for data communication.

**Keyboard interface:** As shown in Figure 10, the four signals between the MHT9000 and the host's keyboard connector are:

- KBDCONNECT: detects when the host is connected
- KBDDATA: This bidirectional signal transmits data between the host and the MHT9000
- KBDCLK: This bidirectional signal acts as the clocking signal when transmitting data
- GND: The common ground reference



**Figure 10. Keyboard Interface**

## 5.0 SYSTEM KERNEL

The MHT9000 is a low-cost hand-held solution using a proprietary OS Kernel. Therefore, no ROM DOS or BIOS is required. The following is a list of primary differences in the hardware design of the MHT9000 versus a standard DOS-based machine:

- The MHT9000 uses a LCD display system; this is radically different from a typical desktop display system. In the MHT9000, the display is command-driven and the display memory is not directly mapped into the system memory. In a DOS-based system's display section, display memory is directly mapped in to the system memory map.
- The MHT9000 uses a pen-interface system and no keyboard. Each application must define the detection zone/area on the pen-based panel to communicate to the system (kernel) when and where the pen is touching the screen. In a DOS-based system, a complete input section is independently implemented such that no initialization is required while waiting for user-input via the keyboard.
- The MHT9000 is targeted as a low-cost product; significantly less RAM is used as compared to a standard DOS-based system. The application must optimize the working memory and data storage requirements.
- The MHT9000 has an "open architecture" in regard to the application interface to the kernel. This yields a flexible — sometimes redundant — and complicated interface format. A DOS-based embedded application cannot use such an "open architecture" when interfacing to a kernel such as DOS. The MHT9000 provides a simple application interface to the kernel; servicing of requests for use of system facilities such as input, display and data storage is done in the kernel.
- In a DOS-based architecture, many system vendors build their own kernel. Such products must conform to DOS specifications that define the interaction between system software and hardware components. The MHT9000 does not have this problem due to the proprietary design of the interface between the application and the system (system software and hardware). As a result, code and interface methodology can be optimized to increase the overall system performance.



Unique features of the MHT9000's interface between the application and system software include:

- Protected mode: the MHT9000's memory model is different from the traditional "8086" processor (80286, 80186 or 8086); it benefits from the Intel386 SX core within the Intel386 EX processor. An advantage of this model can be achieved in mass-storage products — such as a dictionary, electronic book, Chinese language translation/recognition, etc. — due to the elimination of the 1 Mbyte limitation of a Real Mode system. Since a large database is needed in these products, hardware bank-switching technology needs to be implemented. Protected mode eliminates this kind of switching due to the 64 Mbyte physical address range. Programmable chip-select signals simplify the hardware design and tend to make the programming task easier.
- Kernel's C-style procedure calls: The kernel software is developed in C and Intel386 SX assembly. Most system calls for the application/kernel interface — such as input, display and system status — are implemented in C. This provides a much easier, efficient, and flexible design for system calls using C-style procedure calls. These calls are also provided for cartridge-support development. All parameters are stored in the stack rather than in registers. These parameters then generate software interrupts with a function number. Each system function has a unique function number.

## 5.1 Display Function

The MHT9000 incorporates a flexible and powerful LCD display system which supports a mix of text and graphics. On-screen text is actually graphics; this allows the use of multiple fonts on the same screen. Proportional fonts are also supported.

### 5.1.1 Screen Definition

The LCD screen size is 320 dots wide and 128 dots high. This accommodates 40 characters x 16 rows. It can display Chinese/Kanji characters, with a maximum of 8 rows, each containing 20 characters. The screen's coordinates in text mode are from column 0 to 39 and row 0 to 15. In graphic definition, it is from 0 to 319 in x-axis and 0 to 127 in y-axis.

Chinese character size (16x16) is twice that of normal English characters (8x8) in width and height; however, the locating system resolution in the MHT9000 is based on the

English character size. Therefore, the first row of Chinese characters is row 0, the second row is row 2; the first column of the Chinese character is column 0; the second column is column 2.

The MHT9000 uses two different screen locators: one for the cursor and the other for the display location. For normal display functions (except the "absolute" display function), the location of the display character is referred to the "display location" and is updated after each display.

The user-input functions allow the user to configure the hardware cursor as blinking or non-blinking. The LCD display controller can implement a cursor function in hardware; however — to provide additional flexibility — the MHT9000 implements the cursor in software.

## 5.1.2 Text and Graphics Display Windows

Application developers can develop user-interface windows that include text, graphics, or both. Windows can be set up by the application to define all the normal display functions within the window ("absolute" display functions can be done anywhere on the screen). The window can be positioned anywhere within the screen area, as required by the application.

### 5.1.3 Application-Defined Font

Other than the MHT9000's default font, an application can use a unique Character-Generator Table (ASCII only). Proportional fonts are supported. The system supports a total of 10 fonts for use at any one time. Switching from one font to another is a simple process. All screen coordinates are automatically adjusted when fonts are switched.

## 5.2 Input Function

Since the MHT9000 is a pen-based hand-held unit, it does not normally use a keyboard or other input keys. Input is accomplished by detecting the pen's location on the digitizer panel. For example, to enter alphanumeric characters, the application must initialize "zone" information in the system. In this case, the zone is the area that represents a key (typically mapped to emulate a standard keyboard). Initialization specifies zone location, size and the key associated with the zone. When the kernel detects the pen touching anywhere in this zone, the kernel





determines the pen's location and returns the key value that corresponds to the zone.

The detecting area is called the "zone array", an array of locations where the pen can be detected.

### 5.3 Application Calling

Four internal applications can be called from the main menu:

- Memo
- Calculator
- PC Keyboard
- PC Handwriting Pad

The calling convention for all applications is the same. Parameters are passed to control the action taken in the application.

The cartridge application has basically the same characteristics as the internal applications, with one exception: internal applications are called by names in the main menu module, while the cartridge application is called through the entry address in the cartridge's ROM header.

### 5.4 Cartridge Operation

The cartridge connection is a general memory-bus interface. Simple memory or I/O accesses can be performed on this bus. Cartridge memory is allocated inside the system memory map. This is normally done without hardware bank switching with the CPU operating in Protected Mode. However, there is a difference in operation between Real Mode and Protected Mode. In Real Mode, the system can only access 512 Kbytes of memory in the cartridge. If more memory is located in the cartridge (e.g., for Electronic Book or Dictionary applications which may exceed 1 Mbyte in size), hardware switching must be done by the application with the pin P3.5. Actually, cartridge memory (60000 - BFFFF) is multiplexed with the Chinese character-generator and the recognition database. The kernel automatically performs the multiplexing; it is transparent to the cartridge application. In Protected Mode, the memory map does not require multiplexing.

#### 5.4.1 Header Structure

To recognize the cartridge, a fixed header structure must be located at the cartridge ROM. The header system address is:

- 0200000H for Protected Mode
- 40000H for Real Mode

CS4# is activated when accessing the header, so the physical address of the header in ROM must be the first address in the region defined by CS4#. Table 7 defines the header structure.

**Table 7. Header Structure and Memory Locations**

Header Structure	Memory Location
Signature (8 bytes)	0000H - 0007H
Application Signature (8 bytes)	0008H - 000FH
Application ID (2 bytes)	0010H - 0011H
Start up code address (absolute address) <sup>1</sup>	0012H - 0015H
Main code entry address (absolute address, 4 bytes) <sup>1</sup>	0016H - 0019H
Icon bitmap address (absolute address, 5 bytes) <sup>1</sup>	001AH - 001DH

**NOTE:**

<sup>1</sup> For Real Mode, 2 bytes for the segment and 2 bytes for the offset. For Protected Mode, 32-bit absolute memory address.

Care must be taken when assigning the start-up code address, main code entry address and the icon address since only 1 Mbyte range can be accessed by CS4# in Protected Mode, when the kernel starts up. Therefore, all three addresses should be in the range 0200000H - 02FFFFFFH. While in Real Mode, only a 256 Kbyte range can be accessed; therefore, the three addresses should be in the range of 40000H - 7FFFFFFH.

Start-up code in the cartridge is called after the base-system's start-up code (upon Reset) executes. This cartridge start-up code performs stack initialization and — when required — initializes the data area and the necessary hardware, including the Intel386 EX processor's registers. All I/O ports and chip-selects (if necessary) are also initialized in the cartridge start-up code. Control is then returned to the kernel. The main code is called in a manner similar to the one for internal applications. The parameters are also passed similarly.



### 5.5 Real Mode Operation

Real Mode operation is supported on the cartridge. Figure 11 shows the memory map of a cartridge in Real Mode. In the MHT9000 design (Real Mode), the system and application can access a maximum of 1 Mbyte.

00000H - 0FFFFH	RAM	Cartridge Bank 2 00000H - 3FFFFH
10000H - 5FFFFH	Unused Region	
60000H - 7FFFFH	Recognition Database	Cartridge Bank 0 40000H - 7FFFFH
80000H - BFFFFH	Chinese Character-Generator	Cartridge Bank 1 80000H - BFFFFH
C0000H - FFFFFH	Kernel ROM	Cartridge Bank 3 C0000H - FFFFFH

**Figure 11. Memory Map of a Cartridge (in Real Mode)**

#### 5.5.1 Memory Bank Switching

Hardware bank-switching is done in Real Mode, by the cartridge application software, using P3.5. The switching is performed by setting P3.5 high or low. In the MHT9000 cartridge design, P3.5 is connected to A19 in the cartridge module. Table 8 shows the relationship between P3.5 and the memory access range.

**Table 8. P3.5 Memory Mapping**

Memory Access by the Program	P3.5 Status	Physical Address in Cartridge
40000H - 7FFFFH	LOW	40000H - 7FFFFH
40000H - 7FFFFH	HIGH	C0000H - FFFFFH
80000H - BFFFFH	LOW	00000H - 3FFFFH
80000H - BFFFFH	HIGH	80000H - BFFFFH

#### 5.5.2 Limitations of Real Mode Operation

The main limitation of Real Mode operation is the memory size accessible by the application. The whole system can only access 1 Mbyte, ranging from 00000H to 0FFFFH. An application requesting more ROM size must implement the hardware bank-switching technique, making the kernel and hardware more complicated.

### 6.0 POWER MANAGEMENT

Power management requires special attention in a hand-held design. The Intel386 EX processor has a flexible power-management unit that can be used to reduce both CPU and system current consumption. Current consumption can be minimized in several ways:

- reducing the system’s operating frequency
- reducing the system’s operating voltage
- sleeping between ‘keystrokes’
- disabling power to devices not in use
- using devices that have low operating currents

#### 6.1 Digitizer Interface

There is no keyboard on the MHT9000. So “Sleeping between keystrokes” is achieved by halting the CPU when no data is being received from the digitizer controller. The MTL6560 interrupts the CPU when the pen touches the panel, which allows the CPU to “sleep” until:

- an interrupt from the timer or digitizer controller occurs
- data appears from either the host’s keyboard/RS-232 port or the barcode data pin.

When the pen is not touching the LCD panel, it is assumed that no data communication is occurring between the MTL6560 and the Intel386 EX processor. The CPU is placed in Idle mode, reducing current to a minimum.

#### 6.2 PC Interface Driver Power Control

A simple PNP transistor switch controls the power to the MC145407 RS-232 Driver. When the RS-232 interface is not in use, the kernel shuts down the power to this driver, to reduce current consumption.

#### 6.3 Oscillating Frequency

The MHT9000 uses a system frequency of 20 MHz CLK2 (10 MHz processor clock). This frequency reduces current consumption considerably; however, it also impacts system performance. Improving code efficiency can compensate for the loss of performance caused by a lower system frequency.





**6.4 Idle Mode**

When the pen touches the screen, a packet of data (3 bytes) is sent to the Intel386 EX processor's serial port 1 through TXD1. The kernel wakes up due to the serial port interrupt and analyzes the data. After the interrupt is serviced, the Intel386 EX processor goes into idle mode until re-awakened.

**7.0 SOFTWARE DEVELOPMENT**

**7.1 Development Platform**

The kernel is built using the C language. Routines such as start-up, hardware initialization and basic memory control are written in assembly. Code can be developed on a DOS-based system and ported to the Intel386 EX processor with minimum modifications. Because the Intel386 EX processor also integrates DOS-type peripherals, testing of peripheral code can be accomplished on a DOS-based system. Several tools are available for code development on the Intel Architecture, for both Real and Protected Modes.

**7.2 MHT9000 Kernel and Application Development Tools**

When developing applications or the kernel for the MHT9000, you need tools designed for developing and embedding software for either the 80386 Protected Mode or Real Mode architecture. The tools used to develop and test the MHT9000 Protected Mode kernel include the Watcom\* assembler and C-386 compiler with the Systems and Software\* Link/Locate 386 utilities. For the MHT9000 Real Mode kernel the Microsoft\* C/C++ compiler and MASM\* assembler were used with the Systems and Software Link/Locate 386 utilities.

Using other tools may require minor source file modifications. For Protected Mode development the "C" compiler must support flat memory model development. The assembly language source files are designed for Protected Mode operation and require modification for Real Mode development.

An object file containing all kernel function calls is provided for linking to cartridge based programs and to enable third party software development.

In debugging the MHT9000 kernel the Soft-Scope\* debugger from Concurrent Sciences\* and an in-circuit emulator from Kontron\* were used.

Because of all the commonalities of the Intel386 EX processor architecture with the PC platform much of the development and debug can be performed using a standard PC. This includes the development of peripheral drivers again because of the compatibility with standard PC peripherals.

**8.0 RELATED DOCUMENTS**

For additional information on Intel products in the U.S. or Canada, call Intel's Literature Center at (800) 548-4725 or write to:

**Intel Literature**  
**P.O. Box 7641**  
**Mt. Prospect, Illinois 60056-7641**

To order literature outside of the U.S. and Canada contact your local international sales office.

Document Name	Order #
<i>Intel386™ EX Embedded Microprocessor User's Manual</i>	272485
<i>Intel386™ EX Embedded Microprocessor Data Sheet</i>	272420

